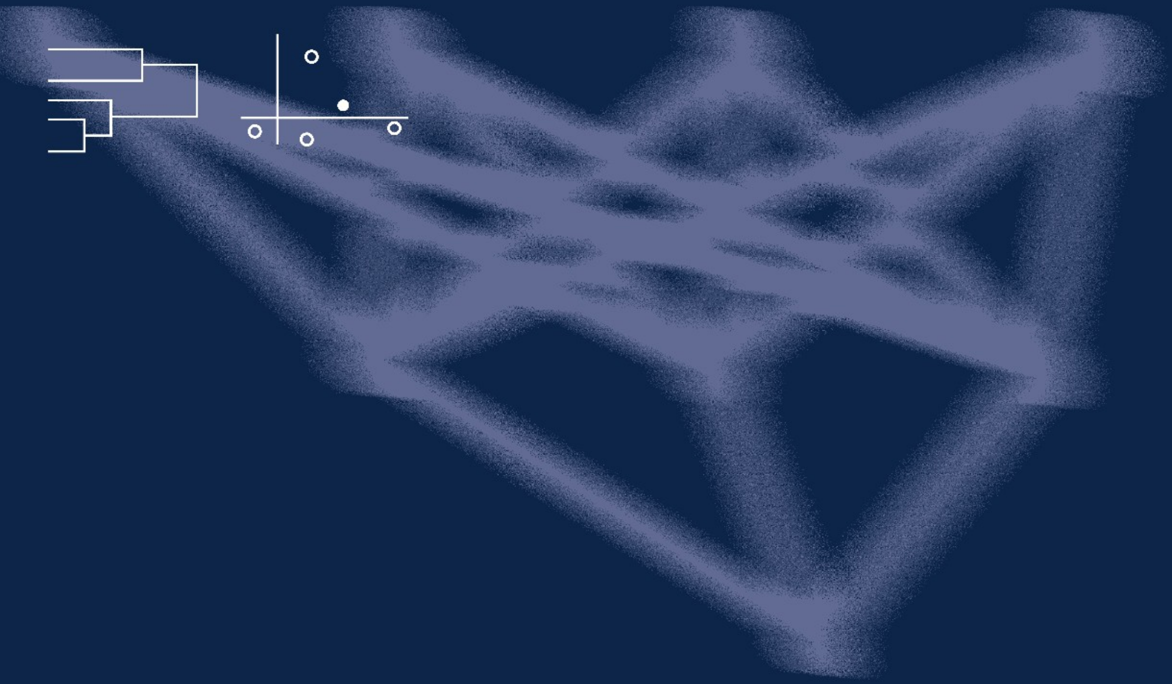
STUDIES IN CLASSIFICATION,
DATA ANALYSIS,
AND KNOWLEDGE ORGANIZATION

P. Brito
P. Bertrand
G. Cucumel
F. de Carvalho
Editors

Selected Contributions in Data Analysis and Classification



Springer

Studies in Classification, Data Analysis, and Knowledge Organization

Managing Editors

H.-H. Bock, Aachen
W. Gaul, Karlsruhe
M. Vichi, Rome

Editorial Board

Ph. Arabie, Newark
D. Baier, Cottbus
F. Critchley, Milton Keynes
R. Decker, Bielefeld
E. Diday, Paris
M. Greenacre, Barcelona
C. Lauro, Naples
J. Meulman, Leiden
P. Monari, Bologna
S. Nishisato, Toronto
N. Ohsumi, Tokyo
O. Opitz, Augsburg
G. Ritter, Passau
M. Schader, Mannheim
C. Weihs, Dortmund

Edwin Diday

Paula Brito · Patrice Bertrand
Guy Cucumel · Francisco de Carvalho
(Editors)

Selected Contributions in Data Analysis and Classification

With a Foreword by Yves Escoufier

With 131 Figures and 78 Tables

Springer

Professor Dr. Paula Brito
Faculty of Economics
University of Porto
Rua Dr. Roberto Frias
4200-464 Porto
Portugal
mpbrito@fep.up.pt

Professor Dr. Guy Cucumel
ESG UQAM
315 East, Sainte-Catherine Street
Montreal (Quebec)
H2X 3X2
Canada
cucumel@guy@uqam.ca

Professor Dr. Patrice Bertrand
Department Lussi
ENST Bretagne
2 rue de la Châtaigneraie, CS 17607
35576 Cesson-Sévigné Cedex
France
Patrice.Bertrand@enst-bretagne.fr

Professor Dr. Francisco de Carvalho
Centre of Computer Science (CIn)
Federal University of
Pernambuco (UFPE)
Av. Prof. Luiz Freire s/n
Cidade Universitária
CEP 50740-540, Recife-PE
Brazil
fatc@cin.ufpe.br

Library of Congress Control Number: 2007932227

ISSN 1431-8814

ISBN 978-3-540-73558-8 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilm or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media

springer.com

© Springer-Verlag Berlin Heidelberg 2007

The use of general descriptive names, registered names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

Production: LE-TEX Jelonek, Schmidt & Vöckler GbR, Leipzig
Cover-design: WMX Design GmbH, Heidelberg

SPIN 12088911 43/3180YL - 5 4 3 2 1 0 Printed on acid-free paper

Foreword

By inviting me to write a preface, the organizers of the event in honour of Edwin Diday, have expressed their affection and I appreciate this very much. This gives me an opportunity to express my friendship and admiration for Edwin Diday, and I wrote this foreword with pleasure. My first few meetings with Edwin Diday date back to 1965 through 1975, days of the development of French statistics. This was a period when access to computers revolutionized the practice of statistics. This does not refer to individual computers or to terminals that have access to powerful networks. This was the era of the first university calculation centres that one accessed over a counter. One would deposit cards on which program and data were punched in and come back a few hours or days later for the results. Like all those who used linear data analysis, the computer enabled me to calculate for each data set the value of mathematical objects (eigenvalues and eigenvectors for example) whose optimality properties had been demonstrated by mathematicians. It was already a big step to be able to do this in concrete experimental situations. With Dynamic Clustering Algorithm, Edwin Diday allowed us to discover that computers could be more than just a way of giving numerical values to known mathematical objects. Besides the efficiency of the solutions he built, he led us to integrate the access to computers differently in the research and practice of data analysis. I think that quite a few works undertaken ever since in France on statistical methods using computers intensively, benefited knowingly or not from the path he had opened.

Thinking about Edwin Diday, I recall his qualities of initiative which greatly benefited the community recognized under the banner of French Data Analysis between 1970 and 1990. He was the founder of club MODULAD, a place of exchange and dissemination of software that each member had created making the methods he had developed accessible to others. It was also a place for valorisation, since these pieces of software put in a coherent package and widely distributed allowed an easy access to recent developments. We should also count Edwin Diday amongst the founders of the Société Francophone de Classification. In this organization which he supported and fought for, we recognize his concern to initiate exchanges between French and Francophone, who were preoccupied with the classification methods. These exchanges which sometimes appeared blunt were ultimately constructive. It is surely this same concern but encompassing a larger field which led him in 1982 to take charge of coordinated research groups (GRECO) recognized by the Centre National de la Recherche Scientifique (CNRS). Under the title Data Analysis and Computer Science, GRECO offered to participate "in the general renewal of the ways of thinking in the field of reduction, description, explanation and synthesis of variations observed on experimental data or observations", which renewal was due to Computer Science. It brought together "researchers from different backgrounds: statistics, numerical analysis, graph theory, combinatorics and for the younger ones, training in the line of Computer Science". The above citation is an extract of a note that I had the pleasure of co-signing with Edwin Diday and Yves Schektman to introduce GRECO to CNRS. In this charge, Edwin Diday knew how to be a unifier who gave everyone the opportunity to express themselves. Here, once again he was a precursor. Due to his reputation and his work for the benefit of our association, he got CNRS to accept the research subject of data analysis and had thus prepared some of the teams to get a seal of approval by the organization.

I would like to recall a fourth initiative of Edwin Diday, that of the "Versailles congress" that he regularly organized between 1977 and 1985. Since 1970, the French statistics has tried to organize itself. The Association des Statisticiens Universitaires (ASU) which later became Association for Statistics and its Uses, maintaining the same acronym, contributed towards the fight against isolation of its teams dispersed all over the territory. The Versailles Congress had the same vision but brought something more to it: an opening to statisticians from abroad. We have had the opportunity, thanks to Edwin Diday, to be able to listen to colleagues from all over Europe, United States and Japan. Many joint initiatives and friendships were developed and have continued by the way of exchange of persons or groups, for example with our English, Italian or Japanese colleagues. Forever a researcher, forever an innovator of ideas for the benefit of the community, Edwin Diday continues even today with the same earnestness and the same success. His personal page on his university website announces 12 articles in journals between 2000 and 2005 and I have not counted the other publications. He explores the field of symbolic data analysis contributing to the extension of data analysis to knowledge analysis. He was a man of the avant-garde in the beginning of his career, and he continues to be so.

Please allow me to conclude this foreword by thinking about Edwin Diday as a man and the enrichment one gets from being close to him. If his concerns of research or service to the community make him sometimes inattentive, they never prevented him from being attentive to other people, thoughtful of all and always available to discuss a new idea, to evaluate a thesis, or to answer favourably an invitation of scientific co-operation. This open-mindedness, this generosity, are deeply rooted in a true humanism from which my relations with Edwin Diday have always benefited. It is a pleasure for me to express it here.

Montpellier, France, August 2007 *Yves Escoufier*

Preface

In the year 1972 Edwin Diday presented a dissertation in which he proposed a new method for clustering objects described by data, a very fashionable topic at that time, often termed as ’automatic classification’. His approach was designed for a multitude of data types and therefore had a great impact on the development of data analysis in France and influenced the professional career of many researchers in data analysis, pattern recognition, and informatics. In particular, a large number of PhD students, naturally from France but also from all over the world (e.g. Vietnam, Portugal, Brazil, Algeria, Turkey) were supervised by Edwin Diday (among them the editors of this volume). Since 35 years, Edwin Diday was active in the field of clustering and data analysis, and by introducing the concept of ’symbolic data’ as early as in 1987, he has also shaped and developed the data-analytic approaches that are known today under the name ’Symbolic Data Analysis’. Other important contributions were the consideration of pyramids as an extension of classical hierarchical classifications, and more recently, spatial clustering embedded in three-dimensional space. During the last few years, he was intensively involved in international projects related to symbolic data analysis.

Edwin Diday was also active in institutions and scientific societies and also the organiser of many conferences and workshops. One of his important activities was the foundation of the Société Française de Classification, later on termed Société Francophone de Classification (SFC), and he was also one of the presidents of this society. Given that the SFC was a founding member of the International Federation of Classification Societies (IFCS), he was among the organisers of the 3rd conference of the IFCS that took place in Paris in the year 1993. Moreover, he was very much involved in establishing the series of conferences ’Data Analysis and Informatics’ from 1977 to 1984, and later on he organised two conferences on his favourite topic “Symbolic-numeric data analysis and learning” (1989 and 1991).

Given such a broad range of activities, colleagues and friends wanted to honour the outstanding work and the scientific career of Edwin Diday by editing a Festschrift in which to collect a series of articles related to his scientific work and its further developments. Many colleagues wanted to contribute to this Festschrift and so we could compile an attractive and actual choice of papers that provides a broad view on the domains were Edwin Diday was active in research or practice, in particular on data analysis, knowledge extraction, and symbolic data analysis. We have clustered these contributions in the following seven chapters:

Analysis of Symbolic Data
Clustering Methods
Conceptual Analysis of Data
Consensus Methods
Data Analysis, Data Mining, and KDD
Dissimilarities: Structures and Indices
Multivariate Statistics

Herewith, all authors, all editors and the members of the organising committee would like to congratulate Edwin Diday for his scientific work and his commitment to the development of data analysis as well as to the education of a large list of young (and meanwhile often established) students and researchers. We all hope that Edwin will be active and healthy for many years and influence the data analysis world also in the future.

Finally, the editors would like to thank all who have contributed to the design and production of this Festschrift, to all authors for their cooperation, as well as to Springer Verlag, in particular Dr. Martina Bihn and Christiane Beisel, for their help concerning all aspects of publication.

Aachen, Montreal, Paris, Porto, Recife, Rennes
September 2007

Patrice Bertrand
Hans-Hermann Bock
Paula Brito
Guy Cucumel
Francisco de Carvalho
Yves Lechevallier
Bruno Leclerc
Gilbert Saporta

Contents

Part I. Analysis of Symbolic Data

Part II. Clustering Methods

Part III. Conceptual Analysis of Data

Part IV. Consensus Methods

Part V. Data Analysis, Data Mining, and KDD

Part VI. Dissimilarities: Structures and Indices

Part VII. Multivariate Statistics

Part I

Analysis of Symbolic Data

Dependencies and Variation Components of Symbolic Interval-Valued Data

Lynne Billard

Department of Statistics, University of Georgia, Athens, GA 30602, USA
lynne@stat.uga.edu

Abstract. In 1987, Diday added a new dimension to data analysis with his fundamental paper introducing the notions of symbolic data and their analyses. He and his colleagues, among others, have developed innumerable techniques to analyse symbolic data; yet even more is waiting to be done. One area that has seen much activity in recent years involves the search for a measure of dependence between two symbolic random variables. This paper presents a covariance function for interval-valued data. It also discusses how the total, between interval, and within interval variations relate; and in particular, this relationship shows that a covariance function based only on interval midpoints does not capture all the variations in the data. While important in its own right, the covariance function plays a central role in many multivariate methods.

1 Introduction

Diday's (1987) seminal paper introduced the concept of symbolic data, bringing to data analysis a new way to think of data, their structures and how to undertake appropriate statistical analyses.

In this paper, the focus is on descriptive statistics for quantitative data. Bertrand and Goupil (2000) introduced expressions for the symbolic sample mean and symbolic sample variance for interval-valued observations. Billard and Diday (2003) extended these to histogram-valued observations. Many examples with and without the presence of logical dependency rules can be found in Billard and Diday (2006).

Finding an expression for the symbolic sample covariance $Cov(Y_{j_1}, Y_{j_2})$ between the random variables Y_{j_1} and Y_{j_2} has been more elusive. While this statistic is important in its own right, it is particularly important through its role in a variety of statistical methodologies for multivariate data such as regression and principal components. For example, if Y_2 is a predictor variable and Y_1 is a dependent variable, a simple linear regression model

$$Y_1 = \beta_1 + \beta_2 Y_2 + e \tag{1}$$

has as its parameter estimators

$$\hat{\beta}_2 = Cov(Y_1, Y_2)/S_{Y_2}, \quad \hat{\beta}_1 = \bar{Y}_1 - \hat{\beta}_2 \bar{Y}_2, \tag{2}$$

where $\bar{Y}_j$ is the sample mean of Y_j, $j = 1, 2$, and $S^2_{Y_2}$ is the sample variance of Y_2. For $(p-1)$ predictor variables, the linear regression model is

$$Y_1 = \beta_1 + \beta_2 Y_2 + \cdots + \beta_p Y_p + e \tag{3}$$

with $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ estimated by

$$\boldsymbol{\beta} = (\boldsymbol{X}'\boldsymbol{X})^{-1}(\boldsymbol{X}'\boldsymbol{Y}). \tag{4}$$

Equivalently, (3) can be written as

$$Y_1 - \bar{Y}_1 = \beta_2(Y_2 - \bar{Y}_2) + \cdots + \beta_p(Y_p - \bar{Y}_p) + e \tag{5}$$

with $\beta_1 = \bar{Y}_1 - \beta_2\bar{Y}_2 - \cdots - \beta_p\bar{Y}_p$. Then, writing $(\boldsymbol{X} - \bar{\boldsymbol{X}}) \equiv (Y_2 - \bar{Y}_2, \ldots, Y_p - \bar{Y}_p)'$ and $(\boldsymbol{Y} - \bar{\boldsymbol{Y}}) \equiv (Y_1 - \bar{Y}_1)$, we have that the parameters $\boldsymbol{\beta} = (\beta_2, \ldots, \beta_p)$ are estimated by

$$\hat{\boldsymbol{\beta}} = [(\boldsymbol{X} - \bar{\boldsymbol{X}})'(\boldsymbol{X} - \bar{\boldsymbol{X}})]^{-1}[(\boldsymbol{X} - \bar{\boldsymbol{X}})'(\boldsymbol{Y} - \bar{\boldsymbol{Y}})] \tag{6}$$

where it is assumed $(\boldsymbol{X} - \bar{\boldsymbol{X}})$ is a nonsingular matrix. Since the theoretical covariance is

$$Cov(Y_{j_1}, Y_{j_2}) = E\{(Y_{j_1} - \bar{Y}_{j_1})(Y_{j_2} - \bar{Y}_{j_2})\},$$

it follows that the (data) terms in (4) or (6) are functions of, or directly involve, the sample estimates of the covariance function.

Studies to date are generally based on the form (3) and hence (4) starting with Billard and Diday (2000, 2002) and most recently with De Carvalho et al. (2004) and Lima Neto et al. (2004, 2005). Published results in effect use a version of the midpoint of the intervals to calculate these covariance related terms. Billard and Diday (2000, 2002) then fit the resulting regression equation to the interval endpoints of the predictor variables $\boldsymbol{X}$ to obtain interval predictions for $Y = Y_1$. De Carvalho et al. (2004) and Lima Neto et al. (2004, 2005) transform each Y_j variable into $Y_j = (Y_{1j}, Y_{2j})$ where Y_{1j} is the interval midpoint and Y_{2j} is the interval length; and then undertake a classical analysis on these $2p$ variables. This is clearly an improvement over the Billard and Diday approach. Unlike Billard and Diday (2002), this range approach has not yet been extended to histogram-valued data. Neither approach however fully accounts for the internal variation of the observed intervals.

More recently, Marino and Palumbo (2003), Lauro and Gioia (2006) and Corsaro and Marino (2006) have used the interval arithmetic results of Moore (1966) to fit a linear regression model to interval-valued data. This produces a set of regression lines, each fitted to specific values inside the observed interval(s). This is computationally intensive. This approach brings in the internal variations indirectly through this set of regressions. The ideas of interval arithmetic unfortunately can only be applied to "short" intervals; nor do they extend to histogram-valued data.

These methods have also been applied to principal components. The De Carvalho and Lima Neto et al. approach, using both the interval midpoints and lengths, was used in a principal component analysis by Palumbo and Lauro (2003); and the interval arithmetic method was used by Gioia and Lauro (2006) and Lauro and Gioia (2006). The same limitations encountered in the respective regression methodologies apply here.

Clearly, a covariance measure that more truly reflects the internal variations of each observation is needed. One such measure is introduced for interval-valued data in Section 2. The proposed covariance function reflects both the variations internal to the observations and those across the observations. They are then compared with those based on the mid-point values only in Section 3.

2 Dependence for interval-valued observations

Let $\boldsymbol{Y} = (Y_1, \ldots, Y_p)$ be a p-dimensional random variable with realizations $\boldsymbol{Y}_u = \boldsymbol{\xi}_u = (\xi_{u1}, \ldots, \xi_{up})$ where $\xi_{uj} = [a_{uj}, b_{uj}]$, $j = 1, \ldots, p$, and $u \in E = \{1, \ldots, m\}$. These $[a_{uj}, b_{uj}]$ intervals can be opened or closed at either end. When $a_{uj} = b_{uj}$, the particular realization is a classical point observation.

Bertrand and Goupil (2000) obtained, under the assumption that possible realizations on $[a, b]$ are uniformly distributed $U \sim (a, b)$, the symbolic sample mean as

$$\bar{Y}_j = \frac{1}{2m} \sum_{u \in E} (b_{uj} + a_{uj}), \tag{7}$$

and the symbolic sample variance as

$$S_j^2 = \frac{1}{3m} \sum_{u \in E} (b_{uj}^2 + b_{uj} a_{uj} + a_{uj}^2) - \frac{1}{4m^2} [\sum_{u \in E} (b_{uj} + a_{uj})]^2. \tag{8}$$

The expression for the sample variance S_j^2 in (8) can be rewritten as

$$S_j^2 = \frac{1}{3m} \sum_{u \in E} [(a_{uj} - \bar{Y}_j)^2 + (a_{uj} - \bar{Y}_j)(b_{uj} - \bar{Y}_j) + (b_{uj} - \bar{Y}_j)^2]. \tag{9}$$

Then, by analogy with (9), for $j = j_1, j_2$, for interval-valued random variables, we let the symbolic sample covariance between Y_{j_1} and Y_{j_2} be

$$Cov(Y_{j_1}, Y_{j_2}) = \frac{1}{3m} \sum_{u \in E} G_{j_1} G_{j_2} [Q_{j_1} Q_{j_2}]^{1/2} \tag{10}$$

with

$$Q_j = (a_{uj} - \bar{Y}_j)^2 + (a_{uj} - \bar{Y}_j)(b_{uj} - \bar{Y}_j) + (b_{uj} - \bar{Y}_j)^2 \tag{11}$$

$$G_j = \begin{cases} -1, \text{ if } \bar{Y}_{uj} \leq \bar{Y}_j, \\ \ \ 1, \text{ if } \bar{Y}_{uj} > \bar{Y}_j, \end{cases} \tag{12}$$

where the overall sample mean $\bar{Y}_j$ is given in (7) and the observation mean is

$$\bar{Y}_{uj} = (a_{uj} + b_{uj})/2. \tag{13}$$

The symbolic sample correlation coefficient is

$$r(Y_{j_1}, Y_{j_2}) = Cov(Y_{j_1}, Y_{j_2})/(S_{j_1} S_{j_2}). \tag{14}$$

Special cases

By comparing (9) and (10), we observe that the special case

$$Cov(Y_j, Y_j) = S_j^2$$

holds. Also, when the data are all classically valued, i.e., when $a_{uj} = b_{uj}$ for all u and j, (10) becomes

$$Cov(Y_{j_1}, Y_{j_2}) = \frac{1}{m} \Sigma (Y_{j_1} - \bar{Y}_{j_1})(Y_{j_2} - \bar{Y}_{j_2}) \tag{15}$$

which is the familiar formula for the covariance function for classical data.

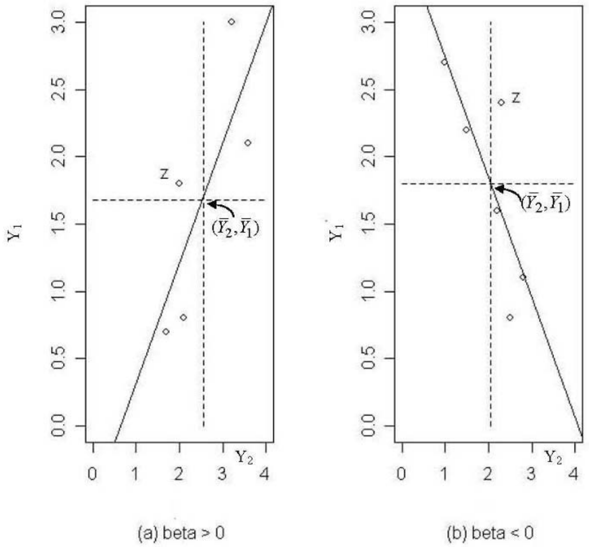
The sign coefficient G_j

It is first noted, by comparing (9) and (11), that Q_j is always positive as it is simply the squared term that enters into the expression for the sample variance S_j^2 for each observation u. However, covariance functions can take positive **or** negative values.

To understand that the G_j of (12) satisfy this property, let us consider the two sets of classical data shown in Fig. 1(a) and Fig. 1(b). Also shown is the fit of the simple regression line (1). For classical data, the estimator of the slope β_2 (as in equation (2)) can be written as

$$\hat{\beta}_2 = \sum_u (Y_{u1} - \bar{Y}_1)(Y_{u2} - \bar{Y}_2) / \sum_u (Y_{u2} - \bar{Y}_2)^2 \tag{16}$$

The sample means $(\bar{Y}_1, \bar{Y}_2)$ fall on the regression line itself. Consider the data of Fig. 1(a) for which the slope is $\beta_2 > 0$. It is easy to see that whenever a particular observation is such that $Y_{u1} < \bar{Y}_1$, the factor $(Y_{u1} - \bar{Y}_1)$ in (16) is negative, and it is positive whenever $Y_{u1} > \bar{Y}_1$; likewise for Y_2. In this case, the tendency is that the $(Y_{u1} - \bar{Y}_1)$ and $(Y_{u2} - \bar{Y}_2)$ terms will be both positive or both negative so that the contribution to the numerator is positive. The reverse tendency holds for the data in Fig. 1(b) where $\beta_2 < 0$. Here, for observations with $Y_{u2} < \bar{Y}_2$, the $(Y_{u1} - \bar{Y}_1)$ terms tend to be positive, to give contributions to the numerator in (16) that are negative; and likewise, for observations with $Y_{u2} > \bar{Y}_2$. (This ”tendency” for the +/- sign value is just that, a tendency and not an absolute; see the observation ”z” in each case where the signs do not take these ”tendency” values). Finally, note that for classical data, $Y_{uj} = \bar{Y}_{uj}$, so that (12) pertains.



(a) beta > 0

(b) beta < 0

Fig. 1. Classical regression slopes.

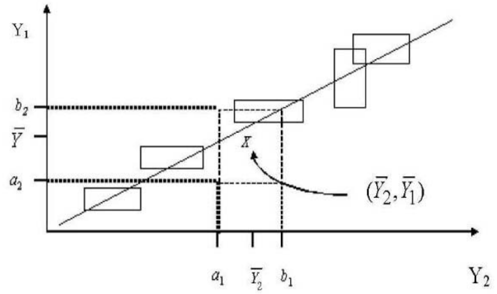


Fig. 2. Interval data regression slope.

Fig. 2 shows a set of interval-valued data (Y_1, Y_2) along with the sample means $(\bar{Y}_1, \bar{Y}_2)$ and the simple linear regression line (1). Note that, unlike for classical data, $(\bar{Y}_1, \bar{Y}_2)$ does not necessarily take a value exactly on this regression line. However, the discussion for classical observations in the previous paragraph carries through analogously where now the observation midpoints $\bar{Y}_{uj}$ replace Y_{uj}, $j = 1, 2$.

An example

The data of Table 1 relate to six sets of performers engaged in a certain dance activity. The random variable Y_2 = Oxygen Intake is a measure of the oxygen capacity of a person and Y_1 = Duration is the time a performer can keep performing (before a prescribed level of exhaustion sets in). Typically the better a person's oxygen capacity (i.e., the less required per unit time), the fitter that person is and so is able to perform longer than those with less capacity.

u	Y_1 Duration $[a_{u1}, b_{u1}]$	Y_2 Oxygen Intake $[a_{u1}, b_{u1}]$
1	[11, 11.2]	[67, 68]
2	[10.3, 11.3]	[62, 64]
3	[11, 11.2]	[57, 59]
4	[11.5, 12.0]	[53, 55]
5	[11.1, 11.6]	[55, 57]
6	[12, 12.1]	[50, 52]

Table 1. Dance Activity.

Substituting into (7), we obtain the sample means

$$\bar{Y}_1 = 11.358, \quad \bar{Y}_2 = 58.250,$$

and substituting into (10) we obtain the sample covariance function as

$$Cov(Y_1, Y_2) = -1.963.$$

The symbolic sample variances are found from (9) as

$$S_1^2 = 0.202, \quad S_2^2 = 30.938.$$

Hence, from (14) the symbolic sample correlation coefficient is

$$r(Y_1, Y_2) = -0.786.$$

An adjustment

A feature that is not uncommon for interval-valued data is that a few of the observations can be such that one or both of the means $\bar{Y}_{j_1}$ and $\bar{Y}_{j_2}$ can fall inside an observation. Fig. 3 plots the actual observations for the data of Table 1, and also shows the respective means $\bar{Y}_1$ and $\bar{Y}_2$. The mean $\bar{Y}_1 = 11.358$ bisects the fifth ($u = 5$) observation with regard to its $Y_1 = (a_{15}, b_{15}) = (11.1, 11.6)$ value; and the mean $\bar{Y}_2 = 58.250$ bisects the third ($u = 3$) observation on its $Y_2 = (a_{23}, b_{23}) = (57, 59)$ value.

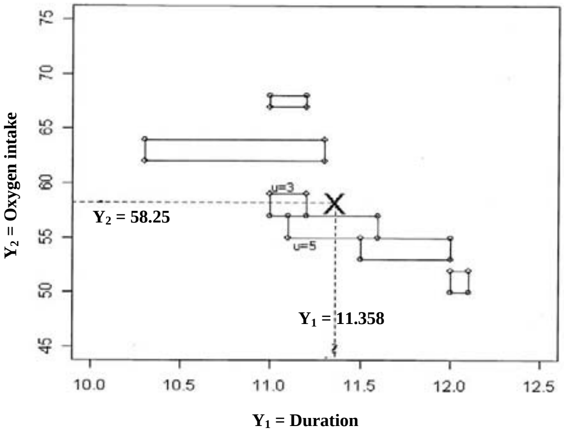


Fig. 3. Dance dataset.

Consider the Y_2 random variable for this $(u = 3)$ observation. Here, $\bar{Y}_2 = 58.250$. From (12), for this observation, $G_2 = -1$; and as written $G_2 = -1$ for all Y_2 values. A refinement is to bisect this observation into two components viz., $u = 31$ (say) with observed values ([11.0, 11.2], [57, 58.25]) and $u = 32$ with observed values ([11.0, 11.2], [58.25, 59]). These components carry weights $w = 0.625$ and $w = 0.375$, respectively. Likewise, the fifth $(u = 5)$ observation is bisected with regard to the Y_1 random variable, to give two "observations" $u = 51$ and $u = 52$ taking values, ([11.1, 11.358], [55, 57]) and ([11.358, 11.6], [55, 57]) with weights 0.516 and 0.484, respectively. The remaining observations take weight $w = 1$. This is summarized as the adjusted data of Table 2.

Also shown in Table 2 are the signs G_j for the unadjusted and adjusted data. In this way, the adjusted data have signs that now more accurately reflect the sign needed for all possible observations in an interval. When, both $\bar{Y}_{j_1}$ and $\bar{Y}_{j_2}$ fall inside an observed rectangle, then bisection relative to both Y_{j_1} and Y_{j_2} occurs, to give four components on an observation, in a completely analogous manner for two observation components. Clearly, for classical observations, this step does not apply.

Substituting the refined values into (10) and using the weight factor w, we obtain the adjusted covariance and hence correlation coefficient as

$$Cov(Y_1, Y_2) = -2.035, \quad r(Y_1, Y_2) = -0.815.$$

The sample means and variances are unchanged.

						Adjustment			
	Duration		Oxygen Intake		Weight	No		Yes	
u	Y_1		Y_2		w	G_1	G_2	G_1	G_2
1	11.0	11.2	67	68	1	-1	1	-1	1
2	10.3	11.3	62	64	1	-1	1	-1	1
31	11.0	11.2	57	58.25	.625	**-1**	**-1**	**-1**	**-1**
32	11.0	11.2	58.25	59	.375			-1	1
4	11.5	!2.0	53	55	1	1	-1	1	-1
51	11.1	11.358	55	57	.516	**-1**	**-1**	**-1**	**-1**
52	11.358	11.6	55	57	.484			**1**	**-1**
6	12.0	12.1	50	52	1	1	-1	1	-1

Table 2. Adjusted Dance Dataset.

3 Within, between and total variations

Let us return to the symbolic sample variance expression (9). We can show that, writing Sum of Squares for Y_j as SS_j,

$$\text{Total}SS_j = \text{Within}SS_j + \text{Between}SS_j \tag{17}$$

where $\text{Total}SS_j = mS_j^2$, with S_j^2 defined as in (9),

$$\text{Within}SS_j = \frac{1}{3}\sum_{u \in E}[(a_{uj} - \bar{Y}_{uj})^2 + (a_{uj} - \bar{Y}_{uj})(b_{uj} - \bar{Y}_{uj}) + (b_{uj} - \bar{Y}_{uj})^2] \tag{18}$$

and

$$\text{Between}SS_j = \sum_{u \in E}[(a_{uj} + b_{uj})/2 - \bar{Y}_j]^2 \tag{19}$$

where $\bar{Y}_{uj}$ and $\bar{Y}_j$ are as defined in (13) and (7), respectively.

Each term of the summation in (18) corresponds to the internal variation of the single observation u. When $a_{uj} = b_{uj} = \bar{Y}_{uj}$ for all u, we have $\text{Within}SS_j = 0$ reflecting that for classical data there is no internal variation. The $\text{Between}SS_j$ of (19) is the sum of squares between the midpoints of all the observations in E. Therefore, methods based on the interval midpoints are using this $\text{Between}SS_j$ to express the variation across the observations, when it is the total variation (i.e., $\text{Total}SS_j$) that should be used. A similar expression to (17) holds for the Sum of Products SP between Y_{j_1} and Y_{j_2}.

An example

This phenomenon is illustrated by the data of Table 1. Table 3 gives the TotalSS, WithinSS and BetweenSS for each of the random variables Y_1 and Y_2, and also the corresponding SPs for the joint (Y_1, Y_2) variable. It is evident that these satisfy the relationship (17). Dividing each by m (= 6 here), we obtain the respective Total, Within and Between variances/covariances. Again, it is clear that use of the interval midpoints in any subsequent analysis does not take into account the internal variations (i.e., the WithinSS_j is neglected).

	Duration Y_1	Oxygen Intake Y_2	Joint (Y_1, Y_2)
Total SS (SP	1.212	185.628	-12.210
Within SS (SP)	0.132	1.752	-0.570
Between SS (SP)	1.080	183.876	-11.640
Total Variation	0.202	30.938	-2.035
Within Variation	0.022	0.292	-0.095
Between Variation	0.180	30.646	-1.940

Table 3. Within, Between and Total Variations.

4 Conclusion

The relationship (17) is one mathematical proof that basing covariance (or functions of covariance) functions on the interval midpoints fails to capture all the variance in the data. Though not discussed herein, it is also a truism that a covariance function based on the BetweenSS does not simplify to the variance (9) for the special case that $Y_{j_1} = Y_{j_2}$, as it should. The covariance function given in (10) has the property that all the variations in the data are utilized and that the special case $Cov(Y_j, Y_j) = S_j^2$ holds.

Therefore, regression and principal component analyses (among others) which depend in some way on a function of the covariance function can now proceed. For example, the Billard and Diday (2000, 2002) regression approach should use the format (10) to estimate the regression parameters. This produces a single prediction equation, but all the data variations have been incorporated into that analysis. Interval predictors can then be found by fitting this prediction equation to the lower and upper interval values for the various predictor variables. The range and midpoint method and the interval arithmetic method can also be suitably adapted.

References

BERTRAND, P. and GOUPIL, F. (2000): Descriptive statistics for symbolic data. In: H.-H. Bock and E. Diday (Eds.): *Analysis of Symbolic Data: Exploratory Methods for Extracting Statistical Information from Complex Data.* Springer-Verlag, Berlin, 103-124.

BILLARD, L. and DIDAY, E. (2000): Regression analysis for Interval-Valued Data. In: H.A.L. Kiers, J.-P. Rasson, P.J.F. Groenen, and M. Schader (Eds.): *Data analysis, Classification, and Related Methods* Berlin: Springer-Verlag, 369-374.

BILLARD, L. and DIDAY, E. (2002): Symbolic regression analysis. In: K. Jajuga, A. Sokolowski, and H.-H. Bock (Eds.): *Classification Clustering, and Data Analysis*. Springer-Verlag, Berlin, 281-288.

BILLARD, L. and DIDAY, E. (2003): From the statistics of data to the statistics of knowledge: Symbolic data analysis. *Journal of the American Statistical Association 98, 470-487.*

BILLARD, L. and DIDAY, E. (2006): *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, Chichester.

BOCK, H.-H. and DIDAY, E. (Eds.) (2000): *Analysis of Symbolic Data: Exploratory Methods for Extracting Statistical Information from Complex Data*. Springer-Verlag, Berlin.

CORSARO, S. and MARINO, M. (2006): Interval linear systems: the state of the art. *Computational Statistics 21, 365-384.*

DE CARVALHO, F.A.T., LIMA NETO, E.A. and TENORIO, C.P. (2004): A new method to fit a linear regression model for interval-valued data. In: *Lecture Notes in Computer Science, KI2004 Advances in Artificial Intelligence*, Springer-Verlag, 295-306.

DIDAY, E. (1987): Introduction à l'approche symbolique en analyse des données. *Premières Journées Symbolique - Numérique*. CEREMADE, Université Paris Dauphine, 21-56.

GIOIA, F. and LAURO, C.N. (2006): Principal component analysis on interval data. *Computational Statistics 21, 343-363.*

LAURO, C. and GIOIA, F. (2006): Dependence and interdependence analysis for interval-valued variables. In: V. Batagelj, H.-H. Bock, A. Ferligoj and A. Ziberna (Eds.): *Data Science and Classification*. Springer-Verlag, 171-183.

LIMA NETO, E.A., DE CARVALHO, F.A.T., and FREIRE, E.S. (2005): Applying constrained linear regression models to predict interval-valued data. In: U. Furbach (Ed.): *Lecture Notes in Computer Science, KI: Advances in Artificial Intelligence*. Springer-Verlag, Berlin, 92-106.

LIMA NETO, E.A., DE CARVALHO, F.A.T. and TENORIO, C.P. (2004): Univariate and multivariate linear regression methods to predict interval-valued features. In: *Lecture Notes in Computer Science, AI 2004 Advances in Artificial Intelligence*. Springer-Verlag, 526-537.

MARINO M. and PALUMBO F. (2003): Interval arithmetic for the evaluation of imprecise data effects in least squares linear regression. *Statistica Applicata, 3.*

MOORE R.E. (1966): *Interval Analysis*. Prentice Hall, Englewood Cliffs, NJ.

PALUMBO F. and LAURO C.N. (2003): A PCA for interval valued data based on midpoints and radii. In: H. Yanai et al (Eds.): *New developments in Psychometrics*. Psychometric Society, Springer-Verlag, Tokyo.

On the Analysis of Symbolic Data

Paula Brito[1,2]

[1] LIAAD/INESC Porto LA
[2] Faculdade de Economia, Universidade do Porto
Rua Dr. Roberto Frias, 4200-464 Porto, Portugal, *mpbrito@fep.up.pt*

Abstract. Symbolic data extend the classical tabular model, where each individual, takes exactly one value for each variable by allowing multiple, possibly weighted, values for each variable. New variable types - interval-valued, categorical multi-valued and modal variables - have been introduced, which allow representing variability and/or uncertainty inherent to the data. But are we still in the same framework when we allow for the variables to take multiple values? Are the definitions of basic notions still so straightforward? What properties remain valid? In this paper we discuss some issues that arise when trying to apply classical data analysis techniques to symbolic data. The central question of the measurement of dispersion, and the consequences of different possible choices in the design of multivariate methods will be addressed.

1 Symbolic data

In the classical tabular model, n individuals $\omega_i, i = 1, \ldots, n$, take exactly one value on each of p variables, $Y_1, \ldots, Y_p$, which may be of quantitative (values are elements of $I\!R$ of some subset of $I\!R$) or qualitative (values are categories of a generally finite set) nature. However, it is often the case that information is too complex to be represented in such a data table. This situation may arise when variables take more than just a single value for one individual. As an example, consider the time used for studying by a given student which varies from day to day, or the means of transportation used in a given year, which may be car, bus, etc. In the first case, the "value" for this variable is an interval (e.g., [20min., 2h]), and in the second case, a frequency distribution (e.g., car 20%, bus 80%). It may also be the case that the described elements are not single individuals but classes of individuals for which internal variability must be taken into account, or that there is some inaccuracy or uncertainty in recording a value which should be considered in the data analysis. In all these cases, the data go beyond the classical paradigm, and we get what has been called *Symbolic Data* (Bock and Diday (2000)).

To represent symbolic data, new kinds of variables have been introduced: *multi-valued* variables, *interval-valued* variables and *modal* variables (Bock and Diday (2000)). A variable is called set-valued if its "values" are nonempty sets of the underlying domain, it is multi-valued if its values are finite subsets of the domain and it is an interval-valued variable if its values are intervals of $I\!R$. A modal variable Y_j with a finite domain $O_j = \{m_1, \ldots, m_{k_j}\}$ is a

multi-state variable where, for each element, we are given a category set and, for each category, a frequency or probability which indicates how frequent or likely that category is for this element. In the case where an empirical distribution is given, the variable is called *histogram* variable (Bock and Diday (2000)).
Let $Y_1, \ldots, Y_p$ be the set of variables, O_j the underlying domain of Y_j and B_j the range of $Y_j, j = 1, \ldots, p$. If Y_j is a classical variable, then $B_j = O_j$; if Y_j is an interval-valued variable, than B_j is the set of intervals contained in O_j; if Y_j is a categorical multi-valued variable B_j is $P(O_j)$, the system of subsets of O_j; and if Y_j is a modal variable then B_j is the set of distributions on O_j. A description of an individual or a class is defined as a p-tuple $(d_1, \ldots, d_p)$ with $d_j \in B_j, j = 1, \ldots, p$.
Let $E = \{\omega_1, \ldots, \omega_n\}$ be the observed entities to be analysed, then $Y_j(w_i) \in B_j$ for $j = 1, \ldots, p, i = 1, \ldots, n$. So, the data array consists in n descriptions, one for each entity $\omega_i \in E : (Y_1(\omega_i), , Y_p(\omega_i)), i = 1, \ldots, n$.

By allowing for new kinds of variables, which take variability or uncertainty explicitly into account, data no longer fit in the classic p-dimensional vector model. The question is then how multivariate data analysis techniques should be extended to the new data types, which properties remain valid, and which notions have to be re-defined. In this paper we address some of these issues, trying to put in evidence the special characteristics of symbolic data.

2 Clustering of symbolic data: dissimilarity versus generalization based methods

Clustering is a multivariate statistical technique that aims at collecting similar individuals in homogeneous classes, on the basis of observed values in a set of variables. The resulting classes may be organized according to different structures. Hierarchical and pyramidal clustering methods produce a structure of nested clusters, in the case of a hierarchy each level corresponds to a partition (i.e. by “cutting” a hierarchy at an appropriate level - according to some given criteria - we get a partition of E); in the case of a pyramid we get, at each level, a family of overlapping clusters (family of non-empty subsets of E which together cover E but are not necessarily disjoint), but all clusters are intervals of a total linear order. Partitional (non-hierarchical) clustering methods produce directly, by means of an iterative process, a partition of E on a generally pre-defined number of disjoint classes, by - most generally locally - optimizing some given criteria.

The consideration of data that go beyond the classical tabular model led to the need of defining, or adapting, clustering methods to the new kinds of data. Moreover, it was intended that the clusters found should be represented within the same formalism as the input data, since symbolic variables allow describing classes, taking into account their internal variability (Diday (1988, 1989)).

Since the initial formalization of Symbolic Data and the first steps in Symbolic Data Analysis (Diday (1988, 1989)), a multitude of methods for clustering symbolic data has been proposed and studied, and applied in different domains. We categorize these methods into two distinct groups:

A Methods that result from adapting classical clustering methods based on dissimilarities to the new kind of data, by properly defining dissimilarity measures for symbolic data. In this case, the clustering methodologies and criteria remain almost unchanged (only necessary adaptations have to be performed) and are applied to the obtained dissimilarity matrices.

B Methods that do not rely on dissimilarities and use the data (i.e., the descriptions of the elements of E) explicitly in the clustering process. The criterion to form classes is to get a “meaningful” class description, and we are in the scope of the so-called *conceptual clustering* methods.

It should be noticed that this categorization is not specific to the clustering of symbolic data, the same applies in the case of clustering classical data arrays. But the purpose here is to put in evidence what is particular to the case of symbolic data, that does not arise when the data follow the classical paradigm.

Clustering methods of type A will tend to cluster together entities with similar descriptions - this similarity being evaluated by one of the proposed measures - irrespective to the intrinsic variability of the underlying descriptions. In other words, however large is the variability inherent to two given descriptions, if they are *alike*, their dissimilarity will have a low value - and the corresponding entities will tend to be clustered together. On the other hand, methods of type B will tend to concentrate on the description of each newly formed cluster, and minimize its inherent variability. This means that this kind of methods may favor the grouping of entities whose descriptions are less alike, if the description of the resulting cluster presents a lower variability.

What we wish to bring forward is that this duality is specific to symbolic data, it does not arise if we are in presence of classical - quantitative or qualitative - data. In the latter case, the closer the values of a given variable, the more specific is their generalization - so both dissimilarity and generalization based methods will tend to elect the same candidate pairs to be aggregated.

Example 1:
Let’s consider the following small illustrative example:
Let Y be a quantitative interval-valued variable, $O = [0, 100]$, B is the set of intervals defined in O.
Let $Y(\omega_1) = I_1 = [10, 20]$, $Y(\omega_2) = I_2 = [30, 40]$, $Y(\omega_3) = I_3 = [10, 100]$, $Y(\omega_4) = I_4 = [9, 99]$.

Which are the more dissimilar pairs, ω_1, ω_2 or ω_3, ω_4 ???
Let Gen be a generalizing operator, that associates to a pair of intervals the smallest interval containing them both. Then, $Gen(I_1, I_2) = [10, 40]$, which covers 30% of O ; $Gen(I_3, I_4) = [9, 100]$, which covers 91% of O. So, generalizing ω_3 and ω_4 leads to a class with a much larger description interval than the one formed by ω_1 and ω_2.
However, if we consider, for instance, the L_2 distance between intervals, then
$L_2(I_1, I_2) = \left[(30-10)^2 + (40-20)^2\right]^{\frac{1}{2}} = \sqrt{800}$ and
$L_2(I_3, I_4) = \left[(100-99)^2 + (10-9)^2\right]^{\frac{1}{2}} = \sqrt{2}$
Analogously, for the Hausdorff distance, $d_H([a, b], [a', b']) =$
$Max\{|a - a'|, |b - b'|\}$, we have:
$d_H(I_1, I_2) =$ Max $\{|30 - 10|, |40 - 20|\} = 20$ and
$d_H(I_3, I_4) =$ Max $\{|10 - 9|, |100 - 99|\} = 1$
So, in a dissimilarity-based clustering algorithm (using for instance L_2 or d_H), ω_3 and ω_4 would be preferred to be clustered together rather than ω_1 and ω_2, and the opposite would happen for generalization-based methods.
Notice, however, that no such dichotomy occurs in the presence of classical quantitative data, for instance, $L_2(10, 30) > L_2(10, 20)$ and also $Gen(10, 30) = [10, 30] \supset Gen(10, 20) = [10, 20]$ (where we identify a real number x with the interval $[x, x]$).

Example 1 deals with interval-valued data, however, the same could easily be illustrated with multi-valued or modal variables. This dichotomy shows that, when clustering a data set described by symbolic variables, it should not be expected that dissimilarity-based methods yield results comparable to those obtained by generalization-based methods. The criteria are different and, in this case, they point in different directions. Therefore it makes no sense to compare results issued by the two kinds of methods: they simply do not have the same objective, since they start from a different concept of "what a cluster is".

3 Dissimilarity based clustering: the standardization problem

When clustering is based on dissimilarities and the underlying variables are quantitative, then the question of comparability of the measurement scales of the different variables is a major issue. In the context of symbolic data analysis, the problem arises when entities are described by interval-valued variables, i.e. $Y_j(\omega_i) = [l_{ij}, u_{ij}], j = 1, \ldots, p, i = 1, \ldots, n$. It is well known, and may often be verified in practical applications, that dissimilarity values and, consequently, clustering results are strongly affected by the variables' scales. So, to make it possible to obtain an 'objective' or 'scale-invariant' result, some standardization must be performed prior to dissimilarity computations in the clustering process. In the symbolic data case, the problem of standardizing interval-valued variables must then be addressed.

It seems reasonable to consider that, since variable values are intervals, the standardization of an interval-valued variable should be performed in such a way that the same transformation is applied to both the lower and the upper bound of all n observed intervals - since they concern one and only variable. In De Carvalho, Brito and Bock (2006), three alternative standardization methods for the case of interval data have been proposed. In all three cases, the variables $Y_j, j = 1, \ldots, p$ are standardized separately, each one in a linear way, with the same transformation for both the lower and the upper bound of all n component intervals $I_{ij} := [l_{ij}, u_{ij}]$, $i = 1, ..., n$. These methods mainly differ in the way dispersion of an interval-valued variable is evaluated.

Standardization 1: Using the dispersion of the interval centers The first method considers the mean and the dispersion of the interval midpoints $(l_{ij} + u_{ij})/2$ and standardizes such that the midpoints of the transformed intervals have zero mean and dispersion 1 in each dimension.
The mean value of all interval midpoints is $m_j := \frac{1}{n}\sum_{i=1}^{n}(l_{ij} + u_{ij})/2$ and their dispersion is evaluated by the empirical variance around this mean: $s_j^2 := \frac{1}{n}\sum_{i=1}^{n}\left((l_{ij} + u_{ij})/2 - m_j\right)^2$. With this notation, the data interval I_{ij} is transformed into the interval $I_{ij}^{'} = [l_{ij}^{'}, u_{ij}^{'}]$ with bounds $l_{ij}^{'} := (l_{ij} - m_j)/s_j$ and $u_{ij}^{'} := (u_{ij} - m_j)/s_j$, $i = 1, ..., n$, where automatically $l_{ij}^{'} \leq u_{ij}^{'}$ for all i, j. As desired, the new intervals $I_{ij}^{'}$ are standardized with $m_j^{'} = 0$ and $s_j^{'2} = 1$.

Standardization 2: Using the dispersion of the interval bounds Another alternative consists in evaluating the dispersion of an interval-valued variable by the dispersion of the interval bounds. This *joint* dispersion of a variable Y_j is defined by $\tilde{s}_j^2 := \frac{1}{n}\sum_{i=1}^{n}((l_{ij} - m_j)^2 + (u_{ij} - m_j)^2)/2$. Consequently, the second standardization method transforms, for each variable j, the intervals $I_{ij} = [l_{ij}, u_{ij}]$ as in the first case, and such that the mean and the joint dispersion of the rescaled interval bounds are 0 and 1, respectively.

Standardization 3: Using the global range A third standardization method transforms, for a given variable, the intervals $I_{ij} = [l_{ij}, u_{ij}]$ $(i = 1, ..., n)$ such that the range of the n rescaled intervals $I'_{ij} = [l'_{ij}, u'_{ij}]$, with $l_{ij}^{'} := \frac{l_{ij} - Min_j}{Max_j - Min_j}$ and $u_{ij}^{'} := \frac{u_{ij} - Min_j}{Max_j - Min_j}$ where $Min_j = Min\{l_{1j}, ..., l_{nj}\}$ and $Max_j = Max\{u_{1j}, ..., u_{nj}\}$ is the unit interval $[0, 1]$.

Simulation studies (De Carvalho, Brito and Bock (2006)) showed that standardization greatly improves the quality of the clustering results in terms of recovery of an imposed structure. Standardization 2 performed slightly better for ill-separated clusters where intervals have large ranges.

In Chavent (2005), an alternative approach for the standardization of interval data is proposed, when a Hausdorff distance is used. In this paper, the author points out that to compute distances between standardized observations is generally equivalent to using a normalized version of the corresponding distance, and determines normalized L_1 and L_∞ Hausdorff distances.

Normalized L_1 Hausdorff distances Let $\hat{\mu}$ be the median of the midpoints of the n intervals $I_{ij}, i = 1, \ldots, n$ and $\hat{\lambda}$ the median of the n intervals' half lengths; in the first case, dispersion of an interval-valued variable is defined by $\sigma_j = \sum_{i=1}^{n} max \left(\left| l_{ij} - \hat{\mu} + \hat{\lambda} \right|, \left| l_{ij} - \hat{\mu} - \hat{\lambda} \right| \right)$.
The Normalized L_1 Hausdorff distance between two individuals ω_i and $\omega_{i'}$ is then defined by $d_1(\omega_i, \omega_{i'}) = \frac{1}{\sigma_j} \sum_{j=1}^{p} d_H(I_{ij}, I_{i'j})$, where $d_H(I_{ij}, I_{i'j}) = Max\{|l_{ij} - l_{i'j}|, |u_{ij} - u_{i'j}|\}$ is the Hausdorff distance between I_{ij} and $I_{i'j}$.

Normalized L_∞ Hausdorff distances Alternatively, dispersion of an interval-valued variable is defined by $\sigma_j = max_{i=1,\ldots,n} \left(\left| l_{ij} - \hat{l}_{ij} \right|, |u_{ij} - \hat{u}_{ij}| \right)$ where $\hat{l}_{ij} = (max_i l_{ij} + min_i l_{ij})/2$ and $\hat{u}_{ij} = (max_i u_{ij} + min_i u_{ij})/2$. The normalized L_∞ Hausdorff distance between two individuals ω_i and $\omega_{i'}$ is then $d_1(\omega_i, \omega_{i'}) = \frac{1}{\sigma_j} max_j d_H(I_{ij}, I_{i'j})$.

The way standardization should be performed depends hence on how dispersion of an interval-valued variable ought to be evaluated and interpreted. Different definitions of dispersion lead to different standardization procedures, and consequently to possibly different results. A question that is not so critical in the analysis of real data.

4 Clustering methods for symbolic data: a generalization based method

A method for "symbolic" hierarchical or pyramidal clustering has been proposed in (Brito (1991, 1994)), allowing clustering for multi-valued data. This method was subsequently developed in order to allow for modal variables (Brito (1998)); later on, Brito and De Carvalho extended this work so as to allow for the existence of hierarchical rules between multi-valued categorical variables (Brito and De Carvalho (1999)) and between modal variables (Brito and De Carvalho (2002)).

The method may be seen within the framework of *conceptual clustering*, since each cluster formed is associated to a conjunction of properties in the input variables, which constitutes a necessary and sufficient condition for cluster membership. Clusters are hence associated to *concepts*, since they are described, both, extensionally by the set of its members, and intentionally by a symbolic description expressing the variability of each variable within the cluster.

The criterion that guides cluster formation is this duality intent-extent: each cluster of the hierarchy or pyramid should correspond to a concept, that is, each cluster is by construction associated with a symbolic description, that provides a generalized description of its members, and no element outside the cluster should fit this description.

An additional criterion must then be considered to choose among the different aggregation possibilities meeting the above condition. The principle is that clusters associated to less general descriptions should be formed first. Since this generality relation is just a partial order relation, a measure has been defined that allows to quantify the generality of a given description: this is the so-called *generality degree*, G. For interval-valued and categorical multi-valued variables, it evaluates the proportion of the underlying domain that is covered by the symbolic description; for modal variables, it evaluates in how much the given distribution is close to the uniform distribution, by computing the affinity between the given distribution and the uniform distribution (see Brito and De Carvalho (2007)). The generality degree is computed variable-wise; the values for each variable are then combined in a multiplicative way to get a measure of the variability of the symbolic description.

Example 2:
Let Y_1 be an interval-valued variable, say percentage of daily time used to study, $O_1 = [0, 100]$, B_1 is the set of intervals defined in O_1, and Y_2 a categorical multi-valued variable, say, spoken languages, B_2 is the power set of $O_2 = \{$Portuguese, Spanish, Italian, French, English, German$\}$, and let $d_1 = (Y_1(\omega_1), Y_2(\omega_1)) = ([10, 25], \{French, English\})$. Then, $G(d_1) = \frac{15}{100} \times \frac{2}{6} = 0,05$, i.e., d_1 covers 5 % of the description space $O_1 \times O_2$.

Let us comment on this definition. First, the fact that generality is evaluated, for interval-valued and categorical multi-valued variables, as the proportion of the description space covered by the given description, corresponds implicitly to assuming that all values within the description space are equally probable: there is hence an underlying uniformity hypothesis in this definition. Secondly, by using a *ratio*, there is no need to standardize quantitative (real or interval-valued) variables. The question remains however whether this ratio should be evaluated with respect to the whole variable domain or to the variability observed in the analysed sample. In either case, since values for different variables are computed separately and may be combined, the method allows easily for clustering data described by variables of mixed types. By combining values for different variables in a multiplicative way, we consider that variables are independent. As concerns modal variables, measuring generality by the affinity with the uniform distribution, corresponds to considering that the more general case arises when all categories of the underlying set are equally probable. The more we deviate from this situation, the more specific is the given description. Finally, such a criterion will favor the formation of clusters with less general descriptions, that is, presenting a global lower variability as respects the underlying variables.

5 Dispersion, association and linear combinations of interval-valued variables

Duarte Silva and Brito (Duarte Silva and Brito (2006)) have addressed the problem of the definition of a linear combination of interval-valued variables, with the aim of establishing conditions under which usual properties hold. Let I be an $n \times p$ matrix representing the values of p interval-valued variables $Y_1, \ldots, Y_p$ on a set $E = \{\omega_i, i = 1, \ldots, n\}$ where each $\omega_i \in E$ is represented by a p-uple of intervals, $I_i = (I_{i1}, \ldots, I_{ip}), i = 1, \ldots, n$, with $I_{ij} = [l_{ij}, u_{ij}], j = 1, \ldots, p$. Let S_I be a covariance matrix of measures of dispersion (s_j^2) and association ($s_{jj'}$) for interval data and $Z = I \bigotimes \beta$ be r appropriately defined linear combinations of the Y's based on $p \times r$ real coefficients $\beta_{j\ell}, j = 1, \ldots, p; \ell = 1, \ldots, r$, stacked in a matrix β.

If we consider a linear combination of interval-valued variables as an extension of the definition of linear combinations of real-valued variables, then it seems natural to stipulate that such a linear combination should satisfy the following basic properties, which are straightforward for the real case:

(P1) $I_i \bigotimes \beta_\ell = \sum_{j=1}^{p} \beta_{j\ell} \times I_{ij}$ where β_ℓ denotes the ℓ-th column of matrix β and $\beta_{j\ell} \times I_{ij} = \{\beta_{j\ell}\ x : x \in I_{ij}\}$; $I_{ij} + I_{i'j} = \{x + y : x \in I_{ij}, y \in I_{i'j}\}$; that is, the resulting interval for individual ω_i is a "linear combination of the intervals" corresponding to each variable, $Y_j(\omega_i)$.

(P2) $S_Z = S_{I \bigotimes \beta} = \beta^t S_I \beta$ that is, the covariance between interval-valued variables should be a symmetric bilinear operator w.r.t. $\bigotimes$.

But do these properties hold in general ? What do we exactly mean by "linear combination of interval-valued variables" ?

One possible, and quite natural definition of linear combination of interval-valued variables is given by

Definition A: $I_i \bigotimes_A \beta_\ell = z_{i\ell A} = [\underline{z}_{i\ell A}, \overline{z}_{i\ell A}], i = 1, \ldots, n$, with

$$\underline{z}_{i\ell A} = \sum_{j=1}^{p} \beta_{j\ell}\ l_{ij} \quad ; \quad \overline{z}_{i\ell A} = \sum_{j=1}^{p} \beta_{j\ell}\ u_{ij} \tag{1}$$

i.e., the resulting interval for individual ω_i is obtained by applying the same linear combination to both the lower and upper bounds of the intervals corresponding to each variable, $I_{ij} = Y_j(\omega_i)$.

Unfortunately, this quite straightforward definition does not satisfy property **(P1)** if at least one element of β_ℓ is negative, since in this case the resulting interval bounds are interchanged (e.g. $(-1)[2,4] = [-4,-2]$).

A definition of a linear combination of interval-valued variables that takes the sign of the elements of β_ℓ into account is given by:

Definition B: $I_i \bigotimes_B \beta_\ell = z_{i\ell B} = [\underline{z}_{i\ell B}, \overline{z}_{i\ell B}], i = 1, \ldots, n$, with

$$\underline{z}_{i\ell B} = \sum_{\beta_{j\ell}>0} \beta_{j\ell}\, l_{ij} + \sum_{\beta_{j\ell}<0} \beta_{j\ell}\, u_{ij} \quad ; \quad \overline{z}_{i\ell B} = \sum_{\beta_{j\ell}>0} \beta_{j\ell}\, u_{ij} + \sum_{\beta_{j\ell}<0} \beta_{j\ell}\, l_{ij} \tag{2}$$

Definition B is the definition we would obtain by applying the rules of Interval Calculus (Moore (1966)) since the resulting intervals include all possible values that are scalar linear combinations of the values within the intervals I_{ij}. However, this definition ignores any connection that may exist between corresponding interval bounds in the original data. The existence (or lack of it) of such connection (and therefore the relevance of property **(P1)**) depends on how a set of interval data ought to be interpreted.

Interval-valued variables generally arise in one of two following situations: either each element $\omega_i \in E$ represents a group of individuals of a set Γ, whose elements are described by real variables y_j, and the interval-valued variables Y_j represent the variability of y_j in each group; or else the interval-valued variable Y_j represents the possible values of an uncertain real variable y_j. In both cases, correlations between underlying real variables may lead to a connection between values within the intervals associated to the corresponding interval-valued variables. When such a connection is present, we say that the variables $Y_j, Y_{j'}$ are *inner correlated.* In the case where two underlying real variables y_j and $y_{j'}$ have a perfect positive ordinal correlation, then the lower bound (resp. upper bound) of Y_j will always be associated with the lower bound (resp. upper bound) of $Y_{j'}$ (and reciprocally for a perfect negative ordinal correlation). Definition A is appropriate when there is *Positive Inner Correlation* in the data which ought to be taken into account; Definition B is appropriate in the absence of inner correlation and it satisfies **(P1)**.

Duarte Silva and Brito (2006) have established that when dispersion s_j^2 and association $s_{jj'}$ measures depend on l_{ij} and u_{ij} symmetrically, then both Definition A and Definition B satisfy **(P2)**. It follows that variances of linear combinations are given by quadratic forms, and ratios are maximized by a traditional eigenanalysis. Classical multivariate methods, such as linear discriminant analysis, may then be adapted in a straightforward way.

6 Conclusions and perspectives

The extension of classical multivariate data analysis methodologies to symbolic data raises new problems: How to evaluate dispersion? How to define linear combinations? Which properties remain valid?

The definition of dispersion is a central one, and the way to evaluate dispersion is not straightforward as in the case of real-valued data. Different alternatives are possible, and the choice of one of these often determines the type of model to be used subsequently.

The important issue remains however the need for statistical models which would allow for estimation and hypothesis testing. This is the real challenge that will surely motivate new research and lead to interesting developments in the analysis of symbolic data in the near future.

References

BOCK, H.-H. and DIDAY, E. (2000): *Analysis of Symbolic Data, Exploratory methods for extracting statistical information from complex data.* Springer-Verlag, Berlin-Heidelberg.

BRITO, P. (1991): *Analyse de Données Symboliques. Pyramides d'Héritage.* PhD Thesis, Mathématiques de la Décision, Univ. Paris-IX Dauphine.

BRITO, P. (1994): Use of pyramids in symbolic data analysis. In: E. Diday, et al. (Eds.): *New Approaches in Classification and Data Analysis.* Springer-Verlag, Berlin-Heidelberg, 378–386.

BRITO, P. (1998): Symbolic clustering of probabilistic data. In: A. Rizzi, M. Vichi and H.-H. Bock (Eds.): *Advances in Data Science and Classification.* Springer, Berlin, 185–190.

BRITO, P. and DE CARVALHO, F.A.T. (1999): Symbolic clustering in the presence of hierarchical rules. In: *Studies and Research, Proceedings of the Conference on Knowledge Extraction and Symbolic Data Analysis (KESDA'98).* Office for Official Publications of the European Communities, Luxembourg, 119–128.

BRITO, P. and DE CARVALHO, F.A.T. (2002): Symbolic clustering of constrained probabilistic data. In: O. Opitz and M. Schwaiger (Eds.): *Exploratory Data Analysis in Empirical Research.* Springer-Verlag, Heidelberg, 12–21.

BRITO, P. and DE CARVALHO, F.A.T. (2007): Hierarchical and pyramidal clustering. In: E. Diday and M. Noirhomme-Fraiture (Eds.): *Symbolic Data Analysis and the SODAS Software.* Wiley, London (in press).

CHAVENT, M. (2005): Normalized k-means clustering of hyper-rectangles. In: *Proceedings of the XIth International Symposium of Applied Stochastic Models and Data Analysis (ASMDA 2005)*, Brest, France, 670-677.

DE CARVALHO, F.A.T., BRITO, P. and BOCK, H.-H. (2006): Dynamic clustering for interval data based on L_2 distance. *Computational Statistics, 21 (2), 231-250.*

DIDAY, E. (1988): The symbolic approach in clustering and related methods of data analysis : the basic choices. In: H.H. Bock (Ed.), *Classification and Related Methods of Data Analysis, Proc. of IFCS'87, Aachen, July 1987.* North Holland, Amsterdam, 673–684.

DIDAY, E. (1989): Introduction à l'analyse des données symboliques. *Revue de Recherche Opérationnelle, 23 (2), 193-236.*

DUARTE SILVA, A.P. and BRITO, P. (2006): Linear discriminant analysis for interval data. *Computational Statistics 21 (2), 289-308.*

MOORE, R.E. (1966): *Interval Analysis.* Prentice Hall, New Jersey.

Symbolic Analysis to Learn Evolving CyberTraffic

Costantina Caruso and Donato Malerba

Dipartimento di Informatica, Universita' degli Studi di Bari
via E. Orabona 4,70126 Bari, Italy,
caruso@di.uniba.it, *malerba@di.uniba.it*

Abstract. Monitoring Internet traffic in order to both dynamically tune network resources and ensure services continuity is a big challenge. Two main research issues characterize the analysis of the huge amount of data generated by Internet traffic: 1) learning a normal adaptive model which must be able to detect anomalies, and 2) computational efficiency of the learning algorithm in order to work properly on-line. In this chapter, we propose a methodology which returns a set of symbolic objects representing an adaptive model of 'normal' daily network traffic. The model can then be used to discover traffic anomalies of interest for the network administrator.

1 Introduction

In the Information Society, it is essential to guarantee the continuity of Internet-based services for both large critical infrastructures and private enterprises. This makes necessary monitoring cybertraffic in order to identify and/or to prevent anomalous behaviors which can be caused by either devices malfunctioning or real intrusion attempts. However, the huge amount of data logged by hosts prevents full human monitoring of network traffic and raises the need of anomaly detection tools.

The realization of effective anomaly detectors requires the consideration of some research issues. First of all, it is important to distinguish between various types of anomalies in order to facilitate their analysis by human operators. In particular, two main types are represented by *outliers* and *change points*. Roughly, outliers are isolated and exceptional points, independently of the temporal dimension, while change points correspond to changing patterns whose semantics is essentially temporal. The difference between outliers and change points is well captured both by Ghoting et al. (2004), who consider two different behaviors when analyzing static or dynamic datasets, and by Takeuchi and Yamanashi (2002), who deal with the issue of detecting outliers and change points from time series. Similarly, also Wang and Stolfo (2004) consider the adaptive property for intrusion detection systems.

In addition, classical supervised learning algorithms are hardly useful for anomaly detection tasks. They are designed for classification tasks and need labeled examples belonging to all the classes the analyzed system can present.

In anomaly detection, we are typically given unlabeled data. In some cases, we can have normal (clean) data, i.e. data we know to be generated from the system in a normal operating condition, and we learn the model from this single class. However, it is difficult to have clean data because they can contain attacks if they are real, while they represent only a partial view of the system if they are simulated.

Works on anomaly detection reported in the literature can be classified by their modeling techniques: distance-based, statistical (probabilistic) approach, and profiling techniques. *Distance-based methods* (Knorr and Ng (1998), Ramaswamy et al. (2000), Breunig et al. (2000), Lazarevic et al. (2003), Eskin et al. (2002)) are widely used for unsupervised data: they can work properly without previous knowledge. The idea in distance based methods is to represent every observation by means of a feature vector and to apply a distance function to measure how much the two observations are far/close. *Statistical approaches* (Shmueli (2005), Yamanishi (2000), Mahoney and Chan (2002), Mahenshkumar et al. (2005)) are used for both multi/single-class data and unlabelled data. Data points are modeled using a stochastic distribution, which is however difficult to estimate for high dimensional mixed-mode data. *Profiling techniques* (Hofmeyr et al.(1998), Tandon and Chan (2003), Wang and Stolfo (2003)) are applied when observation units concerning either human or machine or system behavior are both supervised and arranged in sequences or time series.

Differently from these works, where a single monolithic learning techniques is applied to solve the anomaly detection problem, we propose a two-staged methodology, initially outlined in (Caruso et al. (2007)), which aims to build a normal model of network traffic from real unlabeled observations (i.e. daily network connections). In the first stage, observations are clusterized and sets of rules are generated to describe each cluster. Rules are transformed into symbolic objects (SOs), which represent a *static* daily normal model. To build the *adaptive* normal model of network traffic (second stage), we compute similarities between SOs belonging to subsequent daily normal static models. An anomaly is defined to be a symbolic object which deviates too much from the normal adaptive model; a ranking mechanism is used to differentiate anomalies in order to identify more precisely change points, which express an evolution of the system, and true outliers which can be caused by either malfunctioning or intrusions. The result of the methodology is a longitudinal adaptive normal model of cybertraffic which can be used by a network administrator to identify deviations in network traffic patterns.

In this chapter, after presenting extensively our methodology, we analyze its dependability on the first preprocessing step, i.e. clustering. We show that, by using two different techniques, namely distance-based and probabilistic, the methodology returns comparable results even though the aggregated data are different. This suggests that our methodology could be applied to different data sources by only changing the aggregation technique and leaving the adaptive and anomaly detection phases unchanged.

2 The methodology

2.1 Generating the static model

The first preprocessing step of the methodology has to be based on an Exploratory Data Analysis technique, since we know nothing about our system and we have no labeled examples. This justifies the use of clustering as our starting point. Clusters are an extensional form of knowledge representation, but in our context intensional descriptions are essential since they are both human-interpretable (at least for the network administrator) and computationally light (since a rule synthesizes the properties of many connections). Therefore, we adopt a two-stepped approach. First, we apply a clustering algorithm in order to decrease the size of the problem. Then we generate a set of rules whose consequents represent the cluster membership. The rules provide an intensional description of clusters.

The rules set $\boldsymbol{R(t)}$ generated for the time unit $\boldsymbol{t}$ is a static representation of the network traffic observed in $\boldsymbol{t}$. It should be noted that in this way we can drastically reduce data to treat and to store: the daily network traffic is given by few hundreds of rules vs. thousands of network connections.

A rule $\boldsymbol{R}$ corresponds to homogeneous groups of connections, that is, to second-order objects, or *symbolic objects*, according to the terminology used in symbolic data analysis (Gowda and Diday (1991)). Symbolic objects simplify the change mining process on our data streams because we can easily compare them by means of dissimilarity measures. Therefore, to provide the network administrator with a dynamic representation of network traffic, we propose to transform rules into symbolic objects and then to compute the dissimilarities between SOs of different days. The transformation of a rule $\boldsymbol{R}$ into the corresponding symbolic object is illustrated in (Caruso et al. (2005)) and represents the last step of the pre-processing phase.

The set $\boldsymbol{So(t)}$ of symbolic objects generated for the time unit $\boldsymbol{t}$ is the *static normal model* of network traffic observed at time $\boldsymbol{t}$.

2.2 Detecting anomalies

To build an adaptive model of network traffic $\boldsymbol{M(t)}$, we compare SOs belonging to the subsequent daily normal static models $\boldsymbol{So(t)}$. More precisely, let $\boldsymbol{M(t)}$ be the adaptive normal model at time $\boldsymbol{t}$[1] whose cardinality is $|\boldsymbol{M(t)}|$ and let $\boldsymbol{So(t)}$ be the static normal model at time $\boldsymbol{t}$ whose cardinality is

[1] The time unit we choose in our experiments is the entire day but the approach is general and the most suitable time unit can be used: seconds, minutes, hours or years and so on.

$|\boldsymbol{So(t)}|$. Given a threshold $\boldsymbol{T}$ and a symbolic object $\boldsymbol{So}$ in $\boldsymbol{M(t)}$, all the SOs in $\boldsymbol{So(t)}$ whose dissimilarity from $\boldsymbol{So}$ is less then $\boldsymbol{T}$ are considered similar to $\boldsymbol{So}$, otherwise they are tagged as anomalies.

Let $\boldsymbol{So}_j\boldsymbol{M(t)}$ be the j-th symbolic object in $\boldsymbol{M(t)}$ and $\boldsymbol{So}_k\boldsymbol{(t)}$ the k-th symbolic object in $\boldsymbol{So(t)}$ and let $\boldsymbol{D}$ be a dissimilarity measure between symbolic objects.

Definition 2.1. The symbolic object $\boldsymbol{So}_k\boldsymbol{(t)}$ in $\boldsymbol{So(t)}$ is an *anomaly* if

$$\mathbf{D}_{jk} = \mathbf{D}(\boldsymbol{So}_j\boldsymbol{M\ (t)},\ \boldsymbol{So}_k\boldsymbol{(t)}) > \boldsymbol{T} \text{ for each j=1,.., } |\boldsymbol{M(t)}|$$

Therefore, if $\mathbf{D}_{jk} \leq \boldsymbol{T}$ for some j, the symbolic object $\boldsymbol{So}_k\boldsymbol{(t)}$ is considered a manifestation of a “known” behavior (it is already modeled by $\boldsymbol{M(t)}$), otherwise it is considered an “unknown” behavior, i.e. an anomaly.

2.3 Ranking anomalies

We need to differentiate anomalies on the fly in order to adapt the model only by means of novel events of the system and not by real outliers. In most research works, outlier detection and change point detection have not been related explicitly and the adaptive properties, when considered, are *built-in* in the model. We consider this approach unsound and we propose an explicit ranking mechanism between anomalies.

When analyzing SOs at time $\boldsymbol{t}$, it is significant to know *how much* a SO is similar to all the SOs belonging to the adaptive model $\boldsymbol{M(t)}$ in order to rank its level of dissimilarity. This information about a symbolic object $\boldsymbol{So}_k\boldsymbol{(t)}$ can be obtained by computing its *dissimilarity mean value* defined as follows:

$$D_{mean}(So_k(t)) = \frac{\sum\limits_{j=1}^{|M(t)|} D(So_k(t), So_jM(t))}{|M(t)|}$$

Another interesting parameter is the minimum value of dissimilarity of a SO; indeed, a SO with high mean dissimilarity could be similar to few others but very dissimilar from the remaining ones and the mean value is not able to capture this situation. Therefore we compute the *minimum dissimilarity* between a fixed SO and all the SOs belonging to $\boldsymbol{M(t)}$:

$$D_{\min}(So_k(t)) = \min_{j=1}^{|M(t)|} D(So_k(t), So_jM(t))$$

2.4 The normal adaptive model *M(t)*

The notions of mean and minimum dissimilarity are used to give the complete definition of normal model. Let $\boldsymbol{T}_{mean}$ and $\boldsymbol{T}_{min}$ be two system-defined threshold values.

Definition 2.2. A symbolic object $\boldsymbol{So}_k$ in $\boldsymbol{So(t)}$ is an anomaly of *prevalent type* $\boldsymbol{PA}_k$ if and only if the following condition holds:

$$(\boldsymbol{D}_{mean}(\boldsymbol{So}_k\boldsymbol{(t)})) \leq \boldsymbol{T}_{mean}) \quad \wedge (\boldsymbol{D}_{min}(\boldsymbol{So}_k\boldsymbol{(t)})) \leq \boldsymbol{T}_{min})$$

Definition 2.3. A symbolic object $\boldsymbol{So}_k$ in ***So(t)*** is a anomaly of *secondary type* $\boldsymbol{SA}_k$ if and only if the following condition holds:

$$(\boldsymbol{D}_{mean}(\boldsymbol{So}_k\boldsymbol{(t)}) > \boldsymbol{T}_{mean}) \quad \wedge (\boldsymbol{D}_{min}(\boldsymbol{So}_k\boldsymbol{(t)}) \leq \boldsymbol{T}_{min})$$

Definition 2.4. A symbolic object $\boldsymbol{So}_k$ in ***So(t)*** is an *outlier* $\boldsymbol{O}_k$ if and only if the following condition holds:

$$(\boldsymbol{D}_{mean}(\boldsymbol{So}_k\boldsymbol{(t)}) > \boldsymbol{T}_{mean}) \quad \wedge (\boldsymbol{D}_{min}(\boldsymbol{So}_k\boldsymbol{(t)}) > \boldsymbol{T}_{min})$$

Let ***ChPoints(t)*** = {the set of all $\boldsymbol{PA}_k$ and $\boldsymbol{SA}_k$ found in ***So(t)***}. Then the *normal adaptive model* ***M(t)*** at time ***t*** is defined as follows:

$$\boldsymbol{M(t)} = \boldsymbol{M(t\text{-}1)} \ \cup \ \boldsymbol{ChPoints(t).}$$

that is, it is obtained by adding the set of all change points found in ***So(t)*** to the normal adaptive model at time (***t-1)***.

3 Experiments and results

3.1 Data collection and preprocessing

We tested the proposed methodology on a real dataset obtained from the firewall logs of the Department of Computer Science of our University. Logs refer to twenty-eight days, from May 31^{st} to June 27^{th}, 2004. Starting from a file per day with all logged packets, we reconstruct all connections opened and closed in that day. Indeed, if we did not try to reconstruct connections, it would be impossible to understand what is going on by looking at one packet at a time. In this work, only ingoing connections are analyzed, since we assume that possible attacks to network services come from outside. The total number of ingoing connections reconstructed for the four weeks is 406,773.

Each connection is described by the following attributes:

1. *Proto* (nominal): the protocol used for the connection (udp or tcp);
2. *StartHalfHour* (integer between 0 and 47): the time when a connection begins;
3. *Dst* (integer between 0 and 255): the Internet Protocol (IP) number of public servers of the Department;
4. *SourceIP* (nominal): the IP of external clients;
5. *Service* (nominal): the requested service (http, ftp, smtp and many other ports);
6. *NumPackets* (integer): the number of connection packets;
7. *Length* (integer): the time length of the connection;
8. *NationCode* (nominal): the two digit code of the nation associated to the source IP.

Data carried by each packet (i.e. the *payload*) in the connection is not considered in this work.

In Fig. 2 the statistical profiles of two significant attributes (*Proto* and *Dst*) are represented. Days are reported on the x-axis while the number of connections is reported on the y-axis. The *Proto* graph shows the temporal distribution of the two main protocols considered in this work (udp and tcp), while the *Dst* graph shows the distribution for a subset of possible destination IP values, namely {8, 135, 10, 45, 153}, which identify five public servers of our Department.

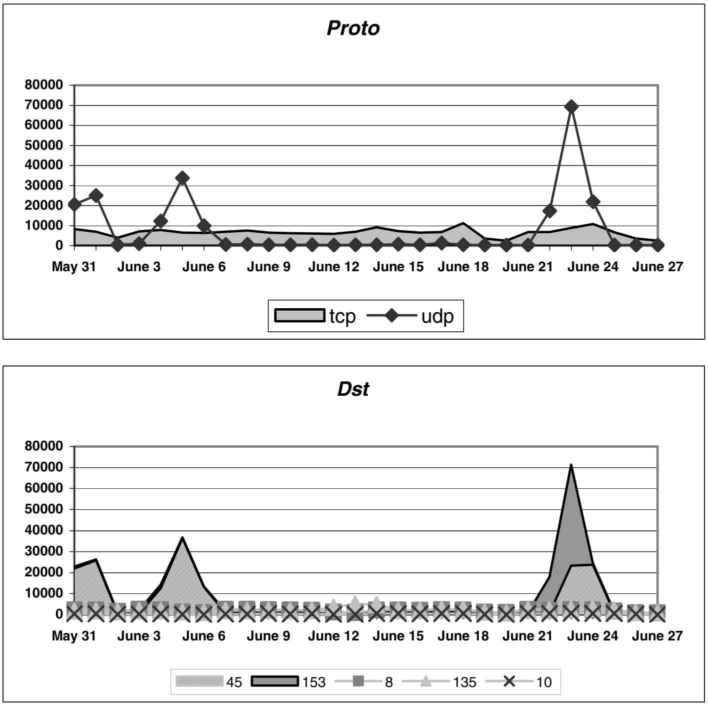


Fig. 1. Distribution of two attributes used for connection description.

3.2 Validation of the normal adaptive model

To demonstrate that the normal adaptive model generated by the proposed methodology is actually able to represent the network traffic and its evolution, we cannot resort to ROC curves as well as to other standard performance measure (e.g., error rate), since we assume to know nothing about network traffic model. We can only consider the statistical profiles of the single attributes to know the prevalent aspects of the network traffic along the four analyzed weeks. For this reason, we aggregated the twenty-eight days in two groups according to their statistical profiles: group A formed by days 31, 1, 4, 5, 6, 22, 23, 24, and group B with all remaining days. The different behavior is ought to the P2P connections explosion we observe for days in group A. We expect that the model retuned by the proposed methodology should be able to differentiate these two groups. More precisely, the normal adaptive model ***M(t)*** is meaningful if new SOs are added to it only when the statistical profile of the analyzed day is different from that one of the previous days. The model should not be modified when we face days whose statistical profile is already known. We expect that the model will be significantly modified in the first week while new SOs are less and less added as days go on. Moreover, the network traffic behavior of the days with an already known profile has to be represented mainly by SOs generated in similar previous days.

3.3 Building and comparing the normal adaptive models

The start-up model ***M(0)*** is initialized to the entire set of symbolic objects generated for the first day (May 31^{st}). Then, for each day, the first ten rules with maximum support and confidence ≥ 0.9, are selected and transformed into SOs. Due to this selection, the SOs we obtain for each day represent the most prevalent aspects in the network traffic and can capture only change points, i.e. points which represent a natural evolution of the network traffic. In a practical application whose final aim is to identify not only the changing points but also true outliers, all rules had to be included in the model since outliers should typically correspond to rules with small support.

In this study, rules are generated by means of the algorithm PART (Witten and Frank (1998)) and provide an intensional description of clusters, while the dissimilarity measure $\boldsymbol{D}$ used to compare SOs is that proposed by Gowda and Diday (1991).

In the experiments, the threshold values for $\boldsymbol{T}$, $\boldsymbol{T}_{mean}$ and $\boldsymbol{T}_{min}$ vary in the following intervals:

- $\boldsymbol{T} = [avgmin,\ avgmean]$,
- $\boldsymbol{T}_{mean} = [avgmean - \mathrm{n},\ avgmean + \mathrm{n}]$,
- $\boldsymbol{T}_{min} = [avgmin - \mathrm{n},\ avgmin + \mathrm{n}]$

where *avgmin* and *avgmean* correspond to the minimum and the mean of all dissimilarity quantities, while the value of n is obtained by a tuning process. In this work the following intervals are considered: $\boldsymbol{T}$ = [2,5], $\boldsymbol{T}_{mean}$ =[3,8], $\boldsymbol{T}_{min}$=[1,4].

For the clustering step, both k-means (Jain et al.(1999)), which is a distance based clustering technique, and EM (McLachlan and Krishan (1997)), which is a probabilistic clustering technique, have been applied to the same data. We observed that the two clustering methods tend to cluster observations differently, which implies the generation of quite different sets of rules and SOs. However, as reported above, we are not interested in the way the network traffic is characterized, but in differentiating the traffic in the two groups of days, i.e. we are interested in checking whether the proposed methodology is able to discover the change points. This implies that network traffic behavior of the days characterized by an already known profile, has to be represented mainly by means of SOs generated in similar previous days.

Fig. 2 shows the models we obtain for K-means and EM; they represent subsequent group A days by means of group A's SOs and subsequent group B days by means of group B's SOs.

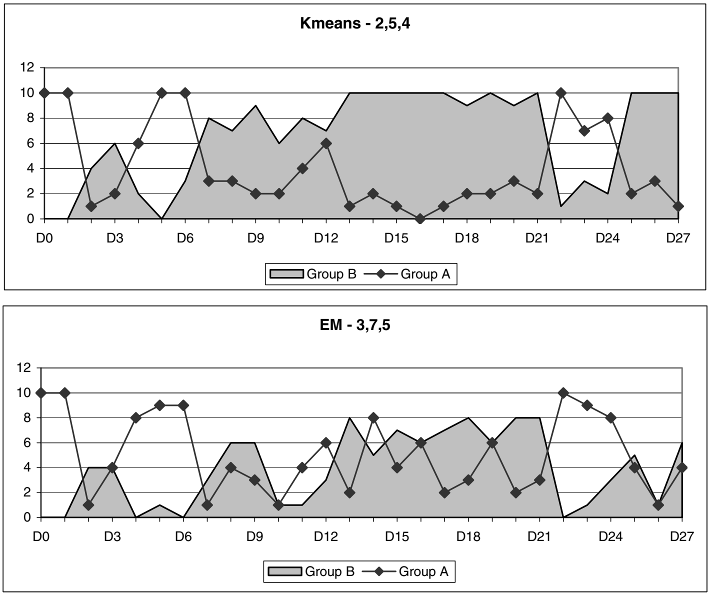


Fig. 2. Number of SOs generated by days in groups A and B for different clustering techniques and triples of the parameters $\boldsymbol{T}$, $\boldsymbol{T}_{mean}$ and $\boldsymbol{T}_{min}$.

As observed before, the model should not be modified when we face days whose statistical profile is already known. We expect that the model will be significantly modified in the first week, while fewer and fewer SOs are added to the model as days go on. This is well shown in the Fig. 3.

4 Conclusions

In this chapter we presented a methodology for network traffic monitoring where symbolic data analysis plays a key role for two main reasons: first, it helps to reduce the size of data to be analyzed from millions of packets per day to a few dozens of SOs, and second, it provides a suitable theoretical framework to deal with similarities between this kind of aggregated data. Symbolic

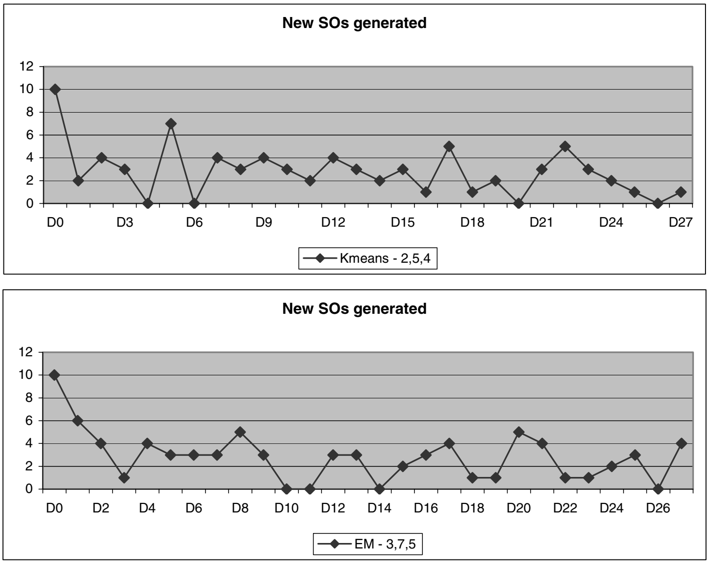


Fig. 3. Number of new SOs generated for different clustering techniques.

objects are obtained from rules representing clusters of connections, and the analysis of similarities between SOs of subsequent days aims to identify a specific class of anomalies, namely change points.

The proposed methodology has been applied to monitor the network traffic of our Department and some promising results confirm its validity. The generality of the features used to describe the network connections in this study also allows us to conclude that the proposed methodology can be applied to monitor the traffic of every network active device. This is an important aspect, since anomaly detectors are often data dependent and hence not portable.

As a future study, we plan to extend our analysis to several dissimilarity measures defined for symbolic data (Esposito et al. (2000)). Moreover, we intend to investigate both the automated selection of the parameters for anomaly detection and the validity of the proposed methodology in identifying outliers.

References

BREUNIG, M., KRIEGEL, H., NG, R., SANDER, J. (2000): LOF: Identifying Density-Based Local Outliers. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, Dallas, Texas, United States.

CARUSO, C. (2007): A Data Mining Methodology for Anomaly Detection in Network Data: Choosing System-Defined Decision Boundaries. *Proceedings of the 15^{th} Italian Symposium on Advanced DataBase Systems*. SEBD2007. To appear.

CARUSO, C., MALERBA, D., PAPAGNI, D. (2005):Learning the daily model of network traffic. *Proceedings of ISMIS 2005, 15th International Symposium*, Saratoga Springs, NY, USA, May 2005. Springer, LNAI 3488; Foundations of Intelligent Systems; pagg. 131-141.

CARUSO, C., MALERBA, D. (2007): A Data Mining Methodology for Anomaly Detection in Network Data. *Proceedings of the 11^{th} International Conference on Knowledge-Based and Intelligent Information & Engineering Systems.* KES2007. To appear.

ESKIN, E., ARNOLD, A., PRERAU, M., PORTNOY, L., STOLFO, S. (2002): A Geometric Framework for Unsupervised Anomaly Detection: Detecting Intrusions in Unlabeled Data. In *Workshop on Data Mining for Security Applications.*

ESPOSITO, F., MALERBA, D., TAMMA V. (2000): Dissimilarity Measures for Symbolic Objects. Chapter 8.3 in H.-H. Bock and E. Diday (Eds.), *Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data*, Series: Studies in Classification, Data Analysis, and Knowledge Organization, vol. 15, Springer-Verlag:Berlin, 165-185.

GHOTING, A., OTEY, M.E., PARTHASARATHY, S. (2004): Loaded: Link-based Outlier and Anomaly detection in Evolving Data Sets. In *Proceeedings of the IEEE International Conference on Data Mining.*

GOWDA, K.C., DIDAY, E. (1991): Symbolic Clustering Using a New Dissimilarity Measure. In *Pattern Recognition, Vol. 24, No. 6, 567-578.*

HOFMEYR, S., FORREST, S., SOMAYAJI, A. (1998): Intrusion Detection using Sequences of System Calls. *Journal of Computer Security 6(1-2), 151-180.*

JAIN, A.K., MURTY, M.N., FLYN, P.J. (1999): Data Clustering: a Review. *ACM Computing Surveys, Vol.31, No.3.*

KNORR, N., NG, P.(1998): Algorithms for Mining Distance-Based Outliers in Large Datasets. *Proceedings of 24th International Conference on Very Large Data Bases, VLDB.*

LAZAREVIC, A., OZGUR, A., ERTOZ, L., SRIVASTAVA, J., KUMAR, V. (2003): A comparative Study of Anomaly Detection Schemes in Network Intrusion Detection. *Proceedings of Third SIAM Conference on Data Mining.*

MAHENSHKUMAR, R.S., NEILL, D.B., MOORE, A.W. (2005): Detecting Anomalous Patterns in Pharmacy Retail Data. *KDD-2005 Workshop on Data Mining Methods for Anomaly Detection..*

MAHONEY, M., CHAN, P. (2002): Learning Nonstationary Models of Normal Network Traffic for Detecting Novel Attacks. *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. ages: 376 - 385.*

McLACHLAN, G.J., KRISHAN, T. (1997): The EM Algorithm and Extensions. John Wiley & Sons.

RAMASWAMY, S., RASTOGI, R., KYUSEOK, S. (2000): Efficient Algorithms for Mining Outliers from Large Data Sets. *Proceedings of the 2000 ACM SIGMOD international conference on Management of data. Pages: 427 - 438 .*

SHMUELI, G. (2005): Current and Potential Statistical Methods for Anomaly Detection in Modern Time Series Data: The Case of Biosurveillance. *KDD-2005 Data Mining Methods for Anomaly Detection.*

TAKEUCHI, J., YAMANASHI, K. (2006): A Unifying Framework for Identifying Changing Points and Outliers. *IEEE Transactions on Knowledge and Data Engineering. Vol.18, No.4.*

TANDON, G., CHAN, P. (2003): Learning Rules from System Call Arguments and Sequences for Anomaly Detection. *Workshop on Data Mining for Computer Security. ICDM 2003.*

WANG, K., STOLFO, S. (2003): One Class Training for Masquerade Detection. *Workshop on Data Mining for Computer Security. ICDM 2003.*

WANG, K., STOLFO, S. (2004): Anomalous Payload-based Network Intrusion Detection. In E. Jonsson,A. Valdes, M. Almgren (Eds.): *Recent Advances in Intrusion Detection*. Springer, Berlin, 203-222.

WITTEN, I., FRANK, E. (1998): Generate Accurate Rule Sets Without Global Optimisation. *Machine Learning: Proceedings of the 15th International Conference, Morgan Kaufmann Publishers, San Francisco, USA.*

YAMANISHI, K. (2000): On-line unsupervised outlier detection using finite mixture with discounting learning algorithms. *Proceedings of the 6th ACM SIGKDD international conference on Knowledge discovery and data mining, 320-324* .

A Clustering Algorithm for Symbolic Interval Data Based on a Single Adaptive Hausdorff Distance

Francisco de A.T. de Carvalho

Centro de Informatica - CIn/UFPE, Av. Prof. Luiz Freire, s/n, Cidade Universitaria, CEP 50740-540, Recife-PE, Brasil, *fatc@cin.ufpe.br*

Abstract. This paper introduces a dynamic clustering method to partitioning symbolic interval data. This method furnishes a partition and a prototype for each cluster by optimizing an adequacy criterion that measures the fitting between the clusters and their representatives. To compare symbolic interval data, the method uses a single adaptive Hausdorff distance that changes at each iteration but is the same for all the clusters. Experiments with real and synthetic symbolic interval data sets showed the usefulness of the proposed method.

1 Introduction

Cluster analysis aims at organizing a set of items into clusters such that items within a given cluster have a high degree of similarity, whereas items belonging to different clusters have high degree of dissimilarity (Jain and Flynn (1999)).

The partitioning dynamic cluster algorithms (Diday and Simon (1976)) are iterative two steps relocation algorithms involving at each iteration the construction of the clusters and the identification of a suitable representative or prototype (mean, factorial axe, probability law, etc.) of each cluster by locally optimizing an adequacy criterion between the clusters and their corresponding prototypes. This optimization process begins from a set of prototypes or an initial partition and interactively applies an *allocation step* (the prototypes are fixed), in order to assign the items to the clusters according to their proximity to the prototypes, and a *representation step* (the partition is fixed), where the prototypes are updated according to the assignment of the patterns in the allocation step, until the convergence of the algorithm is achieved, when the adequacy criterion reaches a stationary value.

The adaptive dynamic clustering algorithm (Diday and Govaert (1977)) also optimizes a criterion based on a measure of fitting between the clusters and their prototypes, but the distances to compare clusters and their prototypes change at each iteration. These distances are not determined once and for all, and moreover, they can be different from one cluster to another. The advantage of these adaptive distances is that the clustering algorithm is able to recognize clusters of different shapes and sizes.

The main difference between these adaptive and non-adaptive algorithms occurs in the representation step which has two stages in the adaptive case: a first stage, where the partition and the distances are fixed and the prototypes are updated, is followed by a second one, where the partition and their corresponding prototypes are fixed and the distances are updated.

Often, objects to be clustered are represented as a vector of quantitative data. However, the recording of interval data has become popular and nowadays this kind of data is often used to describe objects. Symbolic Data Analysis (SDA) is an area related to multivariate analysis, data mining and pattern recognition, which has provided suitable data analysis methods for managing objects described as vectors of intervals (Bock and Diday (2000)).

Concerning dynamical cluster algorithms for symbolic interval data, SDA has provided suitable tools. Verde et al (2001) introduced an algorithm considering context dependent proximity functions and Chavent and Lechevalier (2002) proposed an algorithm using an adequacy criterion based on Hausdorff distances. Souza and De Carvalho (2004) presented a dynamic cluster algorithm for symbolic interval data based on L_1 Minkowsky distances. More recently, De Carvalho et al (2006) proposed an algorithm using an adequacy criterion based on adaptive Hausdorff distances for each cluster.

This paper introduces a new method of dynamic clustering for symbolic interval data based on Hausdorff distances. This method furnishes a partition of the input data and a corresponding prototype (a vector of intervals) for each class by optimizing an adequacy criterion which is based on a single adaptive Hausdorff distance between vectors of intervals. In this method, the prototype of each cluster is represented by a vector of intervals, whose lower bounds, for a given variable, are the difference between the median of midpoints of the intervals computed for the objects belonging to this class and the median of their half-lengths, and whose upper bounds, for a given variable, are the sum of the median of midpoints of the intervals computed for the objects belonging to this class plus the median of their half-lengths. In order to show the usefulness of this method, synthetic interval data sets ranging from different degrees of difficulty to be clustered and an application with a real data set were considered. The evaluation of the clustering results is based on an external validity index.

This paper is organized as follow. Section 2 presents the previous dynamic clustering methods based on Hausdorff distances and introduces the model based on a single adaptive Hausdorff distance. In Section 2 it is presented the evaluation of this method in comparison with previous dynamic clustering methods having adequacy criterion based on Hausdorff (non-adaptive and adaptive for each cluster) distances. The accuracy of the results furnished by these clustering methods is assessed by the corrected Rand index (Hubert and Arabie (1985)) considering synthetic interval data sets in the framework of a Monte Carlo experience and an application with a real data set. Finally, Section 4 presents the conclusions and final remarks.

2 Clustering symbolic interval data based on Hausdorff distances

In this Section we recall the previous dynamic clustering methods based on Hausdorff distances and we introduce the model based on a single adaptive Hausdorff distance.

Let Ω be a set of n objects indexed by i and described by p interval variables indexed by j. An *interval variable* X (Bock and Diday (2000)) is a correspondence defined from Ω in $\Re$ such that for each $i \in \Omega, X(i) = [a,b] \in \Im$, where $\Im$ is the set of closed intervals defined in $\Re$. Each object i is represented as a vector of intervals $\mathbf{x}_i = (x_i^1, \cdots, x_i^p)$, where $x_i^j = [a_i^j, b_i^j] \in \Im = \{[a,b] : a, b \in \Re,\ a \leq b\}$. A prototype $\mathbf{y}_k$ of cluster P_k is also represented as a vector of intervals $\mathbf{y}_k = (y_k^1, \cdots, y_k^p)$, where $y_k^j = [\alpha_k^j, \beta_k^j] \in \Im$.

Here, the distances chosen to compare two intervals are the Hausdorff distances. The Hausdorff distance is defined to compare two sets of objects A and B. In this work, A and B are two intervals $x_i^j = [a_i^j, b_i^j]$ and $x_{i'}^j = [a_{i'}^j, b_{i'}^j]$ and in that case the Hausdorff distance is (Chavent and Lechevallier (2002))

$$d_H(x_i^j, x_{i'}^j) = max\{|a_i^j - a_{i'}^j|, |b_i^j - b_{i'}^j|\} \tag{1}$$

2.1 Clustering of symbolic interval data based on a non-adaptive Hausdorff distance

Here we present a clustering method for symbolic interval data based on a non-adaptive Hausdorff distance (labeled as HNAD). This method has been introduced in Chavent and Lechevallier (2002).

The HNAD method looks for a partition of Ω into K clusters $\{P_1, \ldots, P_K\}$ and a corresponding set of prototypes $\{\mathbf{y}_1, \ldots, \mathbf{y}_K\}$ such that an adequacy criterion $J1$ measuring the fitting between the clusters and their prototypes is locally minimized. This criterion $J1$ is based on a non-adaptive Hausdorff distance and it is defined as:

$$J1 = \sum_{k=1}^{K} \sum_{i \in P_k} \phi(\mathbf{x}_i, \mathbf{y}_k) = \sum_{k=1}^{K} \sum_{i \in P_k} \sum_{j=1}^{p} \Big[max\{|a_i^j - \alpha_k^j|, |b_i^j - \beta_k^j|\} \Big] \tag{2}$$

where

$$\phi(\mathbf{x}_i, \mathbf{y}_k) = \sum_{j=1}^{p} d_H(x_i^j, y_k^j) = \sum_{j=1}^{p} \Big[max\{|a_i^j - \alpha_k^j|, |b_i^j - \beta_k^j|\} \Big] \tag{3}$$

is a (non-adaptive) Hausdorff distance measuring the dissimilarity between an object $\mathbf{x}_i$ $(i = 1, \ldots, n)$ and a cluster prototype $\mathbf{y}_k (k = 1, \ldots, K)$.

The algorithm sets an initial partition and alternates a *representation step* and an *allocation step* until convergence when the criterion $J1$ reaches a stationary value representing a local minimum.

Representation step: definition of the best prototypes. In the representation step, the partition of Ω in K clusters is fixed. Let $\{(a_i^j + b_i^j)\,/\,2 : i \in P_k\}$ be the set of midpoints of the intervals $x_i^j = [a_i^j, b_i^j], i \in P_k$ and let $\{(b_i^j - a_i^j)\,/\,2 : i \in P_k\}$ be the set of half-lengths of the intervals $x_i^j = [a_i^j, b_i^j], i \in P_k$. The prototype $\mathbf{y}_k = (y_k^1, \ldots, y_k^p)$ of cluster P_k $(k = 1, \ldots, K)$, which minimizes the clustering criterion $J1$, has the bounds of the interval $y_k^j = [\alpha_k^j, \beta_k^j]$ updated according to the following: $\alpha_k^j = \mu_j - \gamma_j$ and $\beta_k^j = \mu_j + \gamma_j$, where μ_j is the median of the set $\{(a_i^j + b_i^j)\,/\,2 : i \in P_k\}$ and γ_j is the the median of the set $\{(b_i^j - a_i^j)\,/\,2 : i \in P_k\}$.

Allocation step: definition of the best partition. In the allocation step, the prototypes are fixed and the clusters P_k $(k = 1, \ldots, K)$, which minimizes the criterion $J1$, are updated according to the following allocation rule: $P_k = \{i \in \Omega : \phi(\mathbf{x}_i, \mathbf{y}_k) \leq \phi(\mathbf{x}_i, \mathbf{y}_h),\ \forall h \neq k\,(h = 1, \ldots, K)\}$.

2.2 Clustering symbolic interval data based on a single adaptive Hausdorff distance

This Section presents a clustering method for symbolic interval data based on a single adaptive Hausdorff distance (labeled as SHAD). The main idea of these methods is that there is a distance to compare clusters and their representatives (prototypes) that changes at each iteration but that is the same for all clusters.

This adaptive method looks for a partition of Ω into K clusters $\{P_1, \ldots, P_K\}$ and a corresponding set of prototypes $\{\mathbf{y}_1, \ldots, \mathbf{y}_K\}$ such that an adequacy criterion $J2$ measuring the fitting between the clusters and their prototypes is locally minimized. This criterion $J2$ is based on a single adaptive Hausdorff distance and it is defined as:

$$J2 = \sum_{k=1}^{K} \sum_{i \in P_k} \varphi(\mathbf{x}_i, \mathbf{y}_k) = \sum_{k=1}^{K} \sum_{i \in P_k} \sum_{j=1}^{p} \lambda^j \left[max\{|a_i^j - \alpha_k^j|, |b_i^j - \beta_k^j|\} \right] \quad (4)$$

where

$$\varphi(\mathbf{x}_i, \mathbf{y}_k) = \sum_{j=1}^{p} \lambda^j \left[max\{|a_i^j - \alpha_k^j|, |b_i^j - \beta_k^j|\} \right] \quad (5)$$

is a single adaptive Hausdorff distance measuring the dissimilarity between an object $\mathbf{x}_i$ $(i = 1, \ldots, n)$ and a cluster prototype $\mathbf{y}_k (k = 1, \ldots, K)$, parameterized by the weight vector $\boldsymbol{\lambda} = (\lambda^1, \ldots, \lambda^p)$, which changes at each iteration but is the same for all clusters.

The algorithm sets an initial partition and alternates a *representation step* and an *allocation step* until convergence when the criterion $J2$ reaches a stationary value representing a local minimum.

The representation step has now two stages.

Representation step: definition of the best prototypes. In the first stage, the partition of Ω in K clusters and the weight vector $\boldsymbol{\lambda}$ are fixed.

Proposition 1. *The prototype $\boldsymbol{y}_k = (y_k^1, \dots, y_k^p)$ of cluster $P_k\,(k = 1, \dots, K)$, which minimizes the clustering criterion $J2$, has the bounds of the interval $y_k^j = [\alpha_k^j, \beta_k^j]\,(j = 1, \dots, p)$ updated according to: $\alpha_k^j = \mu_j - \gamma_j$ and $\beta_k^j = \mu_j + \gamma_j$ where μ_j is the median of the set $\{(a_i^j + b_i^j)\,/\,2 : i \in P_k\}$ and γ_j is the the median of the set $\{(b_i^j - a_i^j)\,/\,2 : i \in P_k\}$.*

Representation step: definition of the best distance. In the second stage, the partition of Ω in K clusters and the prototypes are fixed.

Proposition 2. *The vector of weights $\boldsymbol{\lambda}= (\lambda^1, \dots, \lambda^p)$, which minimizes the clustering criterion $J2$ under $\lambda^j > 0$ and $\prod_{j=1}^p \lambda^j = 1$, is updated according to the following expression:*

$$\lambda^j = \frac{\left\{\prod_{h=1}^p \left(\sum_{k=1}^K \left[\sum_{i\in P_k} \left(max\{|a_i^h - \alpha_k^h|, |b_i^h - \beta_k^h|\}\right)\right]\right)\right\}^{\frac{1}{p}}}{\sum_{k=1}^K \left[\sum_{i\in P_k} \left(max\{|a_i^j - \alpha_k^j|, |b_i^j - \beta_k^j|\}\right)\right]}, \; j = 1, \dots, p \tag{6}$$

Allocation step: definition of the best partition. In the allocation step, the prototypes and the weight vector $\boldsymbol{\lambda}$ are fixed.

Proposition 3. *The clusters $P_k\,(k = 1, \dots, K)$, which minimize the criterion $J2$, are updated according to the following allocation rule:*

$$P_k = \{i \in \Omega : \varphi(\boldsymbol{x}_i, \boldsymbol{y}_k) \le \varphi(\boldsymbol{x}_i, \boldsymbol{y}_h), \, \forall h \neq k\,(h = 1, \dots, K)\} \tag{7}$$

2.3 Clustering symbolic interval data based on an adaptive Hausdorff distance for each cluster

Here we present a clustering method for symbolic interval data based on an adaptive Hausdorff distance for each cluster (labelled as HADC). This method has been introduced in De Carvalho et al (2006). The main idea of these methods is that there is a different distance associated to each cluster to compare clusters and their representatives (prototypes) that changes at each iteration, i.e., the distance is not determined once for all, furthermore it is different from one cluster to another. Again, the advantage of these adaptive distances is that the clustering algorithm is able to find clusters of different shapes and sizes.

The HADC adaptive method looks for a partition of Ω into K clusters $\{P_1, \ldots, P_K\}$ and a corresponding set of prototypes $\{\mathbf{y}_1, \ldots, \mathbf{y}_K\}$ such that an adequacy criterion $J3$ measuring the fitting between the clusters and their prototypes is locally minimized. This criterion $J3$ is based on an adaptive Hausdorff distance for each cluster and it is defined as:

$$J3 = \sum_{k=1}^{K} \sum_{i \in P_k} \psi(\mathbf{x}_i, \mathbf{y}_k) = \sum_{k=1}^{K} \sum_{i \in P_k} \sum_{j=1}^{p} \lambda_k^j \left[max\{|a_i^j - \alpha_k^j|, |b_i^j - \beta_k^j|\} \right] \quad (8)$$

where

$$\psi(\mathbf{x}_i, \mathbf{y}_k) = \sum_{j=1}^{p} \lambda_k^j \left[max\{|a_i^j - \alpha_k^j|, |b_i^j - \beta_k^j|\} \right] \quad (9)$$

is an adaptive Hausdorff distance measuring the dissimilarity between an object $\mathbf{x}_i$ $(i = 1, \ldots, n)$ and a cluster prototype $\mathbf{y}_k (k = 1, \ldots, K)$, defined for each class and parameterized by the vectors of weights $\boldsymbol{\lambda}_k = (\lambda_k^1, \ldots, \lambda_k^p)$ $(k = 1, \ldots, K)$, which change at each iteration.

The algorithm sets an initial partition and alternates a *representation step* and an *allocation step* until convergence when the criterion $J3$ reaches a stationary value representing a local minimum.

The representation step has also two stages.

Representation step: definition of the best prototypes. In the first stage, the partition of Ω in K clusters and the vectors of weights $\boldsymbol{\lambda}_k = (\lambda_k^1, \ldots, \lambda_k^p)$ $(k = 1, \ldots, K)$, are fixed.

The prototype $\mathbf{y}_k = (y_k^1, \ldots, y_k^p)$ of cluster P_k $(k = 1, \ldots, K)$, which minimizes the clustering criterion $J3$, has the bounds of the interval $y_k^j = [\alpha_k^j, \beta_k^j]$ updated according to the following: $\alpha_k^j = \mu_j - \gamma_j$ and $\beta_k^j = \mu_j + \gamma_j$, where μ_j is the median of the set $\{(a_i^j + b_i^j) \,/\, 2 : i \in P_k\}$ and γ_j is the the median of the set $\{(b_i^j - a_i^j) \,/\, 2 : i \in P_k\}$.

Representation step: definition of the best distances. In the second stage, the partition of Ω in K clusters and the prototypes are fixed. The vectors of weights $\boldsymbol{\lambda}_k = (\lambda_k^1, \ldots, \lambda_k^p)$ $(k = 1, \ldots, K)$, which minimizes the clustering $J3$ under $\lambda_k^j > 0$ and $\prod_{j=1}^{p} \lambda_k^j = 1$, is updated according to the following expression:

$$\lambda_k^j = \frac{\left\{ \prod_{h=1}^{p} \left[\sum_{i \in P_k} \left(max\{|a_i^h - \alpha_k^h|, |b_i^h - \beta_k^h|\} \right) \right] \right\}^{\frac{1}{p}}}{\sum_{i \in P_k} \left(max\{|a_i^j - \alpha_k^j|, |b_i^j - \beta_k^j|\} \right)}, \; j = 1, \ldots, p$$

Allocation step: definition of the best partition. In the allocation step, the prototypes and the vectors of weights $\boldsymbol{\lambda}_k = (\lambda_k^1, \ldots, \lambda_k^p)\,(k = 1, \ldots, K)$ are fixed. The clusters $P_k\,(k = 1, \ldots, K)$, which minimize the criterion $J3$, are updated according to the minimum distance allocation rule: $P_k = \{i \in \Omega : \psi(\mathbf{x}_i, \mathbf{y}_k) \leq \psi(\mathbf{x}_i, \mathbf{y}_h),\ \forall h \neq k\,(h = 1, \ldots, K)\}$.

3 Experimental results

To show the usefulness of these methods, experiments with synthetic symbolic interval data sets with different degrees of clustering difficulty (clusters of different shapes and sizes, linearly non-separable clusters, etc) and an application with a real data set are considered.

3.1 Synthetic data sets

In each experiment, we considered two standard quantitative data sets in $\Re^2$. Each data set has 450 points scattered among four classes of unequal sizes and elliptical shapes: two classes of size 150 each and two classes of sizes 50 and 100. Each class in these quantitative data sets were drawn according to a bi-variate normal distribution.

We consider two different configurations for the standard quantitative data sets: 1) data drawn according to a bi-variate normal distribution where the class covariance matrices are unequal and 2) data drawn according to a bi-variate normal distribution where the class covariance matrices are almost the same.

Each data point (z_1, z_2) of each one of these synthetic quantitative data sets is a seed of a vector of intervals (rectangle): $([z_1 - \gamma_1/2, z_1 + \gamma_1/2], [z_2 - \gamma_2/2, z_2 + \gamma_2/2])$. These parameters γ_1, γ_2 are randomly selected from the same predefined interval. The intervals considered in this paper are: $[1, 10]$, $[1, 20]$, $[1, 30]$ and $[1, 40]$.

Symbolic interval data set 1 (Figure 1, left side) were constructed from standard data drawn according to the following parameters (configuration 1):

a) Class 1: $\mu_1 = 28$, $\mu_2 = 23$, $\sigma_1^2 = 144$, $\sigma_2^2 = 16$ and $\rho_{12} = 0.8$;
b) Class 2: $\mu_1 = 62$, $\mu_2 = 30$, $\sigma_1^2 = 81$, $\sigma_2^2 = 49$ and $\rho_{12} = 0.7$;
c) Class 3: $\mu_1 = 50$, $\mu_2 = 15$, $\sigma_1^2 = 49$, $\sigma_2^2 = 81$ and $\rho_{12} = 0.6$;
d) Class 4: $\mu_1 = 57$, $\mu_2 = 48$, $\sigma_1^2 = 16$, $\sigma_2^2 = 144$ and $\rho_{12} = 0.9$;

Symbolic interval data set 2 (Figure 1, right side) were constructed from standard data drawn according to the following parameters (configuration 2):

a) Class 1: $\mu_1 = 28$, $\mu_2 = 23$, $\sigma_1^2 = 100$, $\sigma_2^2 = 9$ and $\rho_{12} = 0.7$;
b) Class 2: $\mu_1 = 62$, $\mu_2 = 30$, $\sigma_1^2 = 81$, $\sigma_2^2 = 16$ and $\rho_{12} = 0.8$;

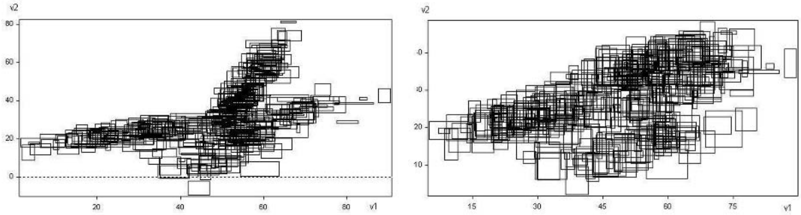


Fig. 1. Symbolic interval data: config. 1 (left side) and config. 2 (right side).

c) Class 3: $\mu_1 = 50$, $\mu_2 = 15$, $\sigma_1^2 = 100$, $\sigma_2^2 = 16$ and $\rho_{12} = 0.7$;
d) Class 4: $\mu_1 = 57$, $\mu_2 = 37$, $\sigma_1^2 = 81$, $\sigma_2^2 = 9$ and $\rho_\rho 12 = 0.8$;

It is expected, for example, that the SHAD clustering method performs well if the data are drawn considering configuration 2.

The evaluation of these clustering methods was performed in the framework of a Monte Carlo experience: 100 replications are considered for each interval data set, as well as for each predefined interval. In each replication a clustering method is run (until the convergence to a stationary value of the adequacy criterion) 50 times and the best result, according to the corresponding criterion, is selected.

The average of the corrected Rand (CR) index (Hubert and Arabie (1985)) among these 100 replications is calculated. The CR index assesses the degree of agreement (similarity) between a *a priori* partition (in our case, the partition defined by the seed points) and a partition furnished by the clustering algorithm. CR can take values in the interval [-1,1], where the value 1 indicates a perfect agreement between the partitions, whereas values near 0 (or negative) correspond to cluster agreements found by chance.

Table 1 shows the values of the average and standard deviation of CR index according to the different methods and data configurations.

Table 1. Comparison between the clustering methods for interval data sets 1 and 2.

Range of values	Interval Data Set 1			Interval Data Set 2		
of γ_i $i = 1, 2$	HNAD	SHAD	HADC	HNAD	SHAD	HADC
$\gamma_i \in [1, 10]$	0.478 (0.002)	0.473 (0.002)	0.542 (0.002)	0.312 (0.002)	0.410 (0.013)	0.375 (0.006)
$\gamma_i \in [1, 20]$	0.480 (0.002)	0.479 (0.002)	0.524 (0.002)	0.301 (0.001)	0.362 (0.010)	0.350 (0.005)
$\gamma_i \in [1, 30]$	0.475 (0.002)	0.473 (0.002)	0.518 (0.002)	0.323 (0.002)	0.324 (0.004)	0.344 (0.003)
$\gamma_i \in [1, 40]$	0.475 (0.002)	0.467 (0.002)	0.511 (0.002)	0.329 (0.001)	0.328 (0.002)	0.328 (0.002)

As expected, in data configuration 1 (the class covariance matrices are unequal) the method based on an adaptive distance for each cluster (HADC) outperforms the method based on a single adaptive distance (SHAD). For this configuration, the method based on a non-adaptive distance (HNAD) presented a similar performance to SHAD method.

Data configuration 2 presents class covariance matrices that are almost the same. In this case, the method based on an adptive distance for each cluster (HADC) outperforms the method based on a single adaptive distance (SHAD) only for $\gamma_i \in [1, 30]$. The method based on a non-adaptive distance (HNAD) has the worst performance.

In conclusion, for these data configurations, the methods based on adaptive distances outperform the HNAD method. Concerning the adaptive methods, their performance depend on the intra-cluster structure: the method based on a single adaptive distance performs well when the *a priori* classes have similar dispersions whereas the method based on an adaptive distance for each cluster performs well when the *a priori* classes have dissimilar dispersions.

3.2 Application to a real data set

A data set with 33 car models described by 8 interval variables is used in this application. These car models are grouped in four *a priori* classes of unequal sizes: *Utilitarian* (size 10), *Berlina* (size 8), *Sporting* (size 7) and *Luxury* (size 8). The symbolic interval variables are: *Price, Engine Capacity, Top Speed, Acceleration, Step, Length, Width* and *Height.*

Concerning this symbolic interval data set, each clustering method is run (until the convergence to a stationary value of the adequacy criterion) 60 times and the best result, according to the adequacy criterion, is selected.

HNAD, SHAD and HADC clustering algorithms have been applied to this data set. The 4-cluster partitions obtained with these clustering methods were compared with the 4-cluster partition known *a priori.* The comparison index used is the corrected Rand index CR which is calculated for the best result. The CR indices were 0.385, 0.558 and 0.558, respectively, for these clustering methods. In conclusion, for this interval data set, the adaptive methods (SHAD and HADC) present the best performance.

4 Conclusions

In this paper, a dynamic clustering method for symbolic interval data is introduced. This method furnishes a partition of the input data and a corresponding prototype for each class by optimizing an adequacy criterion which is based on a single adaptive Haudorff distance between vectors of intervals. Moreover, the prototype of each cluster is represented by a vector of intervals, whose lower bounds, for a given variable, are the difference between the median of midpoints of the intervals computed for the objects belonging to this class and the median of their half-lengths, and whose upper bounds, for a given variable, are the sum of the median of midpoints of the intervals computed for the objects belonging to this class plus the median of their half-lengths.

The evaluation of this method in comparison with dynamic clustering methods having adequacy criterion based on (non-adaptive and adaptive for each cluster) Hausdorff distances have been carried out. The accuracy of the results furnished by these clustering methods was assessed by the corrected Rand index considering synthetic interval data sets in the framework of a Monte Carlo experience and an application with a real data set. Concerning the average CR index for synthetic and real symbolic interval data sets, the methods with adaptive distances outperform the method with non-adaptive distance. Regarding the adaptive methods, their performance depend on the intra-cluster structure: the method based on a single adaptive distance performs well when the *a priori* classes have a similar dispersion whereas the method based on an adaptive distance for each cluster performs well when the *a priori* classes have a dissimilar dispersion.

Acknowledgments: The author would like to thank CNPq, CAPES and FACEPE (Brazilian Agencies) for its financial support.

References

BOCK, H.-H. and DIDAY, E. (2000): *Analysis of Symbolic Data: Exploratory Methods for Extracting Statistical Information from Complex Data.* Springer, Berlin.

CHAVENT, M. and LECHEVALLIER, Y. (2002): Dynamical Clustering Algorithm of Interval Data: Optimization of an Adequacy Criterion Based on Hausdorff Distance. In: A. Sokolowsky, H.-H. Bock and K. Jajuga (Eds): *Classification, Clustering and Data Analysis (IFCS2002).* Springer, Berlin, 53–59

DE CARVALHO, F.A.T, SOUZA, R.M.C.R., CHAVENT, M. and LECHEVALLIER, Y. (2006): Adaptive Hausdorff distances and dynamic clustering of symbolic data. *Pattern Recognition Letters, 27 (3), 167–179.*

DIDAY, E. and GOVAERT, G. (1977): Classification Automatique avec Distances Adaptatives. *R.A.I.R.O. Informatique Computer Science, 11 (4), 329–349.*

DIDAY, E. and SIMON, J.C. (1976): Clustering analysis. In: K.S. Fu (Eds): *Digital Pattern Classification. Springer, Berlin et al, 47–94.*

HUBERT, L. and ARABIE, P. (1985): Comparing Partitions. *Journal of Classification, 2, 193–218.*

JAIN, A.K., MURTY, M.N. and FLYNN, P.J. (1999): Data Clustering: A Review. *ACM Computing Surveys, 31, (3), 264–323.*

SOUZA, R.M.C.R. and DE CARVALHO, F.A.T. (2004): Clustering of interval data based on city-block distances. *Pattern Recognition Letters, 25 (3), 353–365.*

VERDE, R., DE CARVALHO, F.A.T. and LECHEVALLIER, Y. (2001): A dynamical clustering algorithm for symbolic data. In: E. Diday, Y. Lechevallier (Eds): *Tutorial on Symbolic Data Analysis (Gfkl2001), 59–72.*

An Agglomerative Hierarchical Clustering Algorithm for Improving Symbolic Object Retrieval

Floriana Esposito and Claudia d'Amato

Dipartimento di Informatica, Università degli Studi di Bari
Campus Universitario, Via Orabona 4, 70125 Bari, Italy
{*esposito—claudia.damato*}*@di.uniba.it*

Abstract. One of the main novelties of the Symbolic data analysis is the introduction of symbolic objects (SOs): "aggregated data" that synthesize information concerning a group of individuals of a population. SOs are particularly suitable for representing (and managing) census data that require the availability of aggregated information. This paper proposes a new (conceptual) hierarchical agglomerative clustering algorithm whose output is a "tree" of progressively general SO descriptions. Such a tree can be effectively used to outperform the resource retrieval task, specifically for finding the SO to which an individual belongs to and/or to determine a more general representation of a given SO. (i.e. finding a more general segment of information which a SO belongs to).

1 Introduction

The aim of symbolic data analysis (SDA) Bock (2000) is to provide a better representation of the variation and imprecision contained in real data. The unit for the SDA concerns homogeneous classes or groups of individuals (second-order objects) described by symbolic variables that are set-valued (interval, multi-valued, taxonomic or dependent) or modal variables. Data aggregations by census tracts or by enumeration districts are examples of second-order objects. SOs are particularly suitable for representing census data that generally raise privacy issues in all governmental agencies that distribute them.

To face this new formalism and the resulting semantic extension that SDA offers, new approaches for processing and interpreting data are necessary. In this context, several data analysis methods (exploratory, graphical representations, clustering, classification) have been extended for coping with symbolic data. Some examples have been presented in Appice et al. (2004) where a classification method for SOs has been defined, or in Appice et al. (2006), where new visualization techniques for SOs are proposed. Also clustering methods for SOs have been developed. In Brito (1994a), Brito (1994b) a pyramidal clustering algorithm is presented. In Meneses and Rogriguez-Rojas (2006): a hard partitional clustering algorithm is proposed.

Ravi and Gowda (2004) set up a nonhierarchical clustering procedure to improve the generation of SOs. A survey of different techniques for handling symbolic data can be found in Diday (1988).

Several clustering approaches have been formalized in the literature. One of these is the hierarchical clustering approach that is characterized by producing a nested series of partitions based on similarity criteria for merging or splitting clusters. The obtained clustering structure is called *dendrogram.* The basic algorithms for this approach are the *single-link* Sneath and Sokal (1973) and the *complete-link* King (1967) algorithms that are suitable for performing clustering on data sets containing non-isotropic clusters (including well-separated clusters), chain-like clusters, and concentric clusters Jain et al. (1999). Hierarchical algorithms are also appealing for the dendrogram produced at the end of the clustering process, as it can be effectively exploited to outperform the resource retrieval task.

Generally, given a resource to find within a set of available resources, it is searched by matching every resource with the requested one. With the increasing number of the resources, such an approach could clearly be inefficient. To solve this problem, the retrieval task can be subdivided in two different steps: 1) resources are clustered and a dendrogram is returned; 2) the requested resource is found by following the paths of the dendrogram that satisfy the matching condition, while the others are discarded. In this way the search space is drastically cut, thus improving the effectiveness of the retrieval process. Anyway, to make feasible such an approach, intensional cluster descriptions have to be defined at every step of the clustering process.

In this paper a (conceptual) hierarchical agglomerative algorithm for clustering SOs is presented. It is based on a modified version of the *single-link* and *complete-link* algorithms. At each step of the process, an intensional cluster definition is built and it is used for the further execution of the task. In this way, a cluster is always made by a single object rather than a collection of objects, making possible to overcome the drawbacks of the single-link and complete-link algorithms, namely the chaining effect in presence of noisy data (see Jain et al. (1999) for more details).

Hence, the result of the algorithm can be exploited to improve the effectiveness of the retrieval of a SO, namely in order to find the SO that better represents a new element and/or a new SO. Particularly, such an approach could be very useful for managing census data. Indeed, quite common issues in this context are: 1) determine the SO to which a new individual belongs to; 2) give a more general description of a (new) given SO. Both these problems can be regarded as a resource retrieval task.

In the next section the clustering algorithm will be presented. In Section 3 the usage of the clustering algorithm for improving the resource retrieval process will be detailed, while conclusions and future work proposals will be discussed in Section 4.

2 A hierarchical agglomerative algorithm for clustering symbolic objects

Given a data collection, clustering algorithms return a set of meaningful clusters obtained by the use of a similarity criterion. Hence, a clustering algorithm generally requires the availability of (dis-)similarity measures able to cope with and capture the semantics of the objects to cluster. General distance functions for SOs have been defined by Esposito et al. (2000) and Ichino (1988), Ichino and Yaguchih (1994).
Gowda and Diday (1991), Gowda and Diday (1992) have proposed new similarity and dissimilarity measures jointly with an agglomerative clustering method applied to SO representation.

In this paper a hierarchical agglomerative algorithm for clustering SOs is presented. It is a modified version of the single-link and complete-link algorithms, returning as output a dendrogram made in the following way: the actual SOs are the leaves of the structure. Each intermediate node represents an intensionally described superset of children nodes. The root node is the intensional description of all SOs.

The main difference of the proposed algorithm w.r.t. single and complete-link is given by the fact that the latter cannot exploit intensional cluster descriptions. Indeed the used criterion for merging clusters is based on distances among elements belonging to the clusters. Such criterion could sometimes cause drawbacks (chaining effect) in presence of noisy data (see Jain et al. (1999) for more details). Furthermore, even if intensional cluster descriptions are built, they can be only used for further applications (i.e. resource retrieval). In order to overcome these limitations, the realized clustering algorithm generates intensional cluster descriptions at each step of the process, so that clusters are made by a single object that is its description (rather than a collection of objects).

Moreover, differently from the agglomerative hierarchical-pyramidal clustering for SOs Brito (1994a). where cluster aggregations is based on the *completeness condition* and coarse partition criteria, the algorithm proposed here is strongly based on a (dis-)similarity clusters criterion which would ensure clusters homogeneity. The algorithm is detailed in the following.

Let $S = \{S_1, \dots, S_n\}$ a set of available SOs.

```
1. Let $\mathcal{C} = \{C_1, \dots, C_n\}$ the set of initial clusters obtained by considering
   each SO as a single cluster
2. $n := |\mathcal{C}|$
3. For $i := 1$ to $n - 1$ consider cluster $C_i$
   (a) For $j := i + 1$ to $n$ consider cluster $C_j$
       i. compute the dissimilarity values $d_{ij}(C_i, C_j)$
4. compute $\min_{hk} = \min_{i,j=1,\dots,n} d_{ij}$ where $h$ and $k$ are the clusters with
   minimum distance
5. create the intensional cluster description $C_m = gen(C_h, C_k)$
```

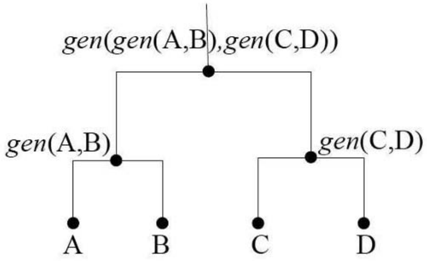


Fig. 1. Dendrogram returned by the algorithm (Section 2) applied to the SOs A, B, C, and D. *gen* is the generalization procedure that given two SOs returns a new general one.

```
6. link $C_m$ to $C_h$ and $C_k$
7. insert $C_m$ in $\mathcal{C}$ and remove $C_h$ and $C_k$ from $\mathcal{C}$
8. if $|\mathcal{C}| \neq 1$ go to 2
```

The algorithm starts considering each SO in a single cluster, hence the couple of SOs having the lowest dissimilarity value[1] is found, and a new SO, generalizing the chosen ones, is created by means of a generalization procedure *gen* Bock (2000). Specifically, given two symbolic objects s_1 and s_2, *gen* procedure returns a new object s that is computed as the union of s_1 and s_2, namely as the union of each couple of variable values of s_1 and s_2 ($s = \bigcup_{i=1}^{p}(s_{i_1}, s_{i_2})$ where p is the number of symbolic variables of a SO). Since a SO can be made by several kinds of symbolic variables (single value, multi-value, numerical, interval, taxonomic, probabilistic, dependent-hierarchical rule), a generalization procedure for every kind of variable is used. The new object, obtained by generalization, is first linked to the objects that it generalizes and then it is inserted in the list of the available clusters, while the selected ones are removed from such a list. The generated description is considered as a cluster made by a single element (SO) and the objects that it describes are represented as its children in the dendrogram under construction. Such a process is repeated iteratively until a unique cluster (describing all SOs) is available.

An example of dendrogram returned by the presented algorithm is shown in Fig.1. Looking at the figure, it is straightforward to note that the dendrogram is a binary tree. This is because, at every step, only two existing clusters are merged into a single one. The clustering process could be speeded up by finding a way for merging more than two clusters at every step. An important result, in this direction, has been shown in Ding and He (2005), where it is proved that, if the measure used for performing the clustering process satisfies

[1] The dissimilarity measure is chosen depending on the the type of SO (boolean or probabilistic). For a survey on the available measures for SO in literature and their behavior w.r.t. different kinds of data-sets see Malerba et al. (2001), Malerba et al. (2002).

the *cluster aggregate inequality property* (namely $d_{A,B+C} \geq \min(d_{A,C}, d_{A,B})$), then more than two clusters can be merged at the same level. Moreover, it has also been proved that such speed up preserves the equivalence of the obtained clusters w.r.t. those obtained without using any speed up. Since most of the measures for SOs (see Bock (2000)) are based on the amount of the information that they share, it can be easily concluded that the amount of information shared by a SO, let us say A, and a generalized SO (let us say $B + C$) is equal or greater than the amount of information shared by A and B or by A and C.

3 Clustering symbolic objects for improving resource retrieval

The dendrogram obtained as result of the clustering algorithm (see Section 2) can be effectively used to outperform the resource retrieval task. This can be particularly useful when resources are represented by SOs synthesizing census data. Indeed, given a set of available SOs representing a sample of a population, a new element can occur. The problem, in this case, is to find the SO to which it belongs to. In the same way, considered a set of available SOs, a new SO object can occur. In this case the problem is to find the most general information that represents it. These two problems can be regarded as a resource retrieval task w.r.t. the set of all available SOs, namely as finding the (most specific) SO that generalizes a new object that occurs .

The most intuitive way to accomplish this task is by checking if each available SO "covers" the new occurred element or SO. The "coverage" test can consist in checking if the value of each symbolic variable of the new object belongs to the set of values of the symbolic variable of the considered SO (see **coverageTest** procedure below). Such an approach could be clearly inefficient with the increasing number of available SOs, as it requires a linear complexity in the number of all available SOs. The resource retrieval could be performed more effectively as detailed in the following, namely by exploiting the dendrogram obtained by clustering the available SOs.

Let Q be a new symbolic object (or equivalently let q a new element) and let C the root of the dendrogram $\mathcal{C}$, output of the clustering process

discoveryProcedure(Q, C)

```
1. returnedSO := null
2. if coverageTest$(Q, C)$ = false then
   (a) return returnedSO
3. else
   (a) returnedSO = $C$
   (b) if $C$ has left child node $C_l$ then
         i. returnedLeftSO = discoveryProcedure$(Q, C_l)$
```

```
    (c) if $C$ has right child node $C_r$ then
        i. returnedRightSO = **discoveryProcedure**$(Q, C_r)$
    (d) if (returnedLeftSO != null) and (returnedRightSO != null) then
        i. compute the **Genrality Degrees**:
           genDegLeft = $G(returnedLeftSO, Q)$,
           genDegRight = $G(returnedRightSO, Q)$
        ii. if genDegLeft $\leq$ genDegRight then
            A. returnedSO = returnedLeftSO
        iii. else
            A. returnedSO = returnedRightSO
    (e) else
        i. if returnedLeftSO != null then
            A. returnedSO = returnedLeftSO
        ii. else if returnedRightSO != null then
            A. returnedSO = returnedRightSO
    (f) return returnedSO
```

```
**coverageTest**$(Q, C)$
Let $Q = \{q_1, \ldots, q_n\}$ and $C = \{c_1, \ldots, c_n\}$
boolean covered := true
i := 1
While $(i \leq n)$ and (covered)

1. if $q_i$ single value variable then
    (a) if valueOf($q_i$) != valueOf($c_i$) then
        i. covered := false
2. if ($q_i$ multi-valued variable) *or* ($q_i$ ordinal variable) or ($q_i$ interval variable) *or* ($q_i$ taxonomic variable) then
    (a) if valueOf($q_i$) *not in* valueOf($c_i$) then
        i. covered := false
3. if $q_i$ modal variable then
    (a) for $j$ := 1 to numValueOf($q_i$)
        i. if prob($value_j(q_i)$) $>$ prob($value_j(c_i)$) then
            A. covered := false
4. i++

return covered
```

In the following the **discoveryProcedure** is analyzed. Given a new symbolic object Q (or equivalently a new element q) it is compared w.r.t. the root of the dendrogram in order to test if Q is covered by the available SOs or not. If the test is verified, the same coverage test is performed for each child node of the root (remember that the dendrogram is a binary tree). If the coverage condition is not satisfied by a child node, the branch rooted in this child node is discarded, otherwise all the children nodes are recursively explored, until a leaf node is reached or until there are no children nodes that satisfy

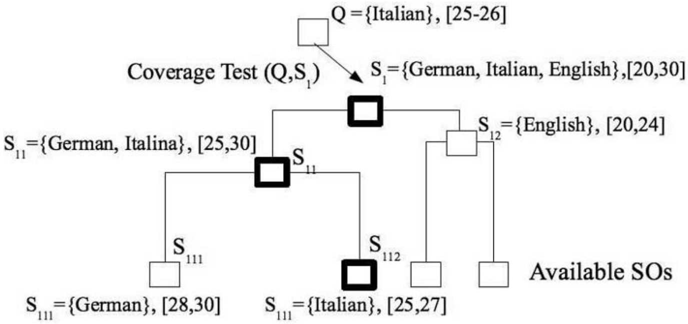


Fig. 2. Retrieval of the query SO Q. Bold boxes represent nodes satisfying the *"coverage" test.*

the coverage condition. In this case, the last node satisfying the coverage condition represents the searched SO. An example of application of *discoveryProcedure* is reported in Fig. 2 where SOs are described by the symbolic variables: *knownLanguage* that is a multi-valued variable representing the known languages by a set of persons and *Age* that is an interval variable.

It is important to note that, at the same level, more than one node could satisfy the coverage condition. Once that all SOs covering the request have been found, the most specific one is selected (and returned). It is determined by recurring to the computation of the *Generalization Degree* of the node w.r.t. the request, introduced in Bock (2000). As an alternative to the exhaustive search in the tree, an heuristic could be adopted for choosing the path to follow. In this case the dissimilarity value between the requested object and the nodes (at the same level) covering the request can be computed. Hence the most similar node can be chosen as the path to follow.

The proposed approach for resource retrieval drastically reduces the dimension of the search space. Indeed the search is restricted only to the branches of the dendrogram whose nodes satisfy the coverage conditions w.r.t. Q. Specifically, a good clustering of n available SOs may reduce the number of necessary comparisons for finding the right SO from $O(n)$ to $O(log\ n)$, thus strongly improving the retrieval time.

4 Conclusions and future work

A new hierarchical agglomerative clustering algorithm has been presented for homogeneously grouping a set of available SOs. It is a modified version of the well known single-link and complete-link algorithm. Its main novelties are given by its application to SO representation (by exploiting proper (dis-)similarity measures) and by defining intensional cluster descriptions during the clustering process, thus obtaining a conceptual clustering algorithm.

The algorithm has been mainly proposed in order to improve the resource retrieval task. Specifically, it has been analytically showed that, by exploiting the dendrogram returned by the algorithm, the complexity of the retrieval task can decrease from $O(n)$ to $O(log\ n)$ in the best case. Such an application of the clustering process is particularly useful in the case of census data, where Symbolic Objects are used to sample group of elements of a given population. Indeed, in this context, a quite often problem is given by finding the SO that represents a new element or finding the most general information to which a new symbolic object refers to.

A possible extension of the algorithm can be given by allowing a phase of incremental clustering. Specifically, once that the SO satisfying the request Q has been found, its description can be updated including also the description of Q. This is in order to enrich the amount of available information and so in order to better accomplish the further requests.

References

APPICE, A., d'AMATO, C., ESPOSITO, F. and MALERBA, D. (2004): Classification of symbolic objects: a lazy learning approach. In: P. Brito and M. Noirhomme-Fraiture (Eds.): *Proceeding of Workshop on Symbolic and Spatial Data Analysis: Mining Complex Data Structures, at ECML/PKDD 2004*, 19–30.

APPICE, A., d'AMATO, C., ESPOSITO, F., MALERBA, D. (2006): Classification of symbolic objects: a lazy learning approach. In: P. Brito and M. Noirhomme-Fraiture (Eds.): *Journal of Intelligent Data Analysis 10, 301-324.*

BOCK, H.-H. and DIDAY, E. (2000): *Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data.* Springer-Verlag, Berlin, Heidelberg.

BRITO, P. (1994a): Use of pyramids in symbolic data analysis. In: E. Diday, Y. Lechevallier, M. Schader et al. (Eds.), *New Approaches in Classification and Data Analysis*, *Proceeding of IFCS-93.* Springer-Verlag, Berlin-Heidelberg, 378–386.

BRITO, P. (1994b): Order structure of symbolic assertion objects. *IEEE Transaction on Knowledge and Data Engineering, 6 (5), 830-835.*

CHRIS, H., DING, Q. and HE, X. (2005): Cluster aggregate inequality and multilevel hierarchical clustering. In: *Proceedings of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases, PKDD.* Springer, LNCS, 3721, 71-83.

DIDAY, E. (1988): The symbolic approach in clustering and related methods of data analysis : the basic choices. In: H.-H. Bock (Ed.), *Classification and Related Methods of Data Analysis, Proc. of IFCS'87, Aachen, July 1987.* North Holland, Amsterdam, 673–684.

ESPOSITO, F., MALERBA, D. and TAMMA, V. (2000): Dissimilarity measures for symbolic objects. In: H.-H. Bock and E. Diday (Eds.), *Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data.* Studies in Classification, Data Analysis, and Knowledge Organization. Springer, Berlin, Heidelberg, 165-185.

ESPOSITO, F., MALERBA, D., GIOVIALE, V. and TAMMA, V. (2001): Comparing dissimilarity measures in Symbolic Data Analysis. In: *Proceedings of the Joint Conferences on New Techniques and Technologies for Statistics and Exchange of Technology and Know-how (ETK-NTTS'01)*, 473-481.

GOWDA, K.C. and DIDAY, E. (1991): Symbolic clustering using a new dissimilarity measure. *Pattern Recognition, 24 (6), 567-578.*

GOWDA, K.C. and DIDAY, E. (1992): Symbolic clustering using a new similarity measure. *IEEE transactions on Systems, Man and Cybernetics, 22 (2), 68-378.*

ICHINO, M. (1988): General metrics for mixed features – The cartesian space theory for pattern recognition. In: *Proc. IEEE Conf. Systems, Man and Cybernetics, Atlanta, GA*, 14–17.

ICHINO, M. and YAGUCHI, H. (1994): General Minkowsky metric for mixed feature type. *IEEE transactions on Systems, Man and Cybernetics, 24, 698-708.*

JAIN, A.K., MURTY, M.N. and FLYNN, P.J. (1999): Data clustering: a review. *ACM Computing Surveys, 31 (3), 264-323.*

KING, B. (1967): Step-wise clustering procedures. *J. Am. Stat. Assoc., 69, 86–101.*

MALERBA, D., ESPOSITO, F. and MONOPOLI, M. (2002): Comparing dissimilarity measures for probabilistic symbolic objects. In: A. Zanasi, C.A. Brebbia, N.F.F. Ebecken and P. Melli (Eds.) *Data Mining III.* WIT Press, Southampton, UK - Management Information Systems 6, 31–40.

MENESES, E. and RODRIGUEZ-ROJAS, O. (2006): Using symbolic objects to cluster web documents. In: *Proceedings of the 15th International Conference on World Wide Web (WWW 2006).* ACM Press, New York, 967-968.

MICHALSKI, R.S. and STEPP, R.E. (1983): Automated construction of classifications: conceptual clustering versus numerical taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 5, 219-243.*

RAVI, T.V. and GOWDA, K.C. (2004): A new non-hierarchical clustering procedure for symbolic objects. In: *Intelligent Data Engineering and Automated Learning – IDEAL 2000: Data Mining, Financial Engineering, and Intelligent Agents* Springer, LNCS.

SNEATH, P.H.A. and SOKAL, R.R. (1973): *Numerical Taxonomy.* Freeman, London, UK.

3WaySym-Scal: Three-Way Symbolic Multidimensional Scaling

Patrick J.F. Groenen[1] and Suzanne Winsberg[2]

[1] Econometric Institute, Erasmus University Rotterdam,
P.O. Box 1738, 3000 DR Rotterdam, The Netherlands
groenen@few.eur.nl

[2] Predisoft, San Pedro, Costa Rica
SuzanneWinsberg@predisoft.com

Abstract. Multidimensional scaling aims at reconstructing dissimilarities between pairs of objects by distances in a low dimensional space. However, in some cases the dissimilarity itself is not known, but the range, or a histogram of the dissimilarities is given. This type of data fall in the wider class of symbolic data (see Bock and Diday (2000)). We model three-way two-mode data consisting of an interval of dissimilarities for each object pair from each of K sources by a set of intervals of the distances defined as the minimum and maximum distance between two sets of embedded rectangles representing the objects. In this paper, we provide a new algorithm called 3WaySym-Scal using iterative majorization, that is based on an algorithm, I-Scal developed for the two-way case where the dissimilarities are given by a range of values ie an interval (see Groenen et al. (2006)). The advantage of iterative majorization is that each iteration is guaranteed to improve the solution until no improvement is possible. We present the results on an empirical data set on synthetic musical tones.

1 Introduction

Classical multidimensional scaling (MDS) models the dissimilarities among a set of objects as distances between points in a low dimensional space. The aim of MDS is to represent and recover the relationships among the objects and to reveal the dimensions giving rise to the space. To illustrate: the goal in many MDS studies, for example, in psychoacoustics or marketing is to visualize the objects and the distances among them and to discover and reveal the dimensions underlying the dissimilarity ratings, that is, the most important perceptual attributes of the objects.

Often, the proximity data available for the n objects consist of a single numerical value for the dissimilarity δ_{ij} between each object pair. Then, the data may be presented in a single dissimilarity matrix with the entry for the i-th row and the j-th column being a single numerical value representing the dissimilarity between the i-th and j-th object (with $i = 1, \ldots, n$ and $j = 1, \ldots, n$). Techniques for analyzing this two-way, one-mode data have been developed (see, e.g., Kruskal (1964), Winsberg and Carroll (1989), or Borg and Groenen (2005)). Sometimes proximity data are collected from K

sources, for example, a panel of K judges or under K different conditions, yielding three-way two-mode data and an $n \times n \times K$ array of single numerical values. Techniques have been developed to deal with this form of data permitting the study of individual or group differences underlying the dissimilarity ratings (see, e.g., Carroll and Chang (1972), Winsberg and DeSoete (1993)).

All of these above mentioned MDS techniques require that each entry of the dissimilarity matrix, or matrices be a single numerical value. However, the objects in the set under consideration may be of such a complex nature that the dissimilarity between each pair of them is better represented by a range, that is, an interval of values, or a histogram of values rather than a single value. For example, if the number of objects under study becomes very large, it may be unreasonable to collect pairwise dissimilarities from each judge and one may wish to aggregate the ratings from many judges where each judge has rated the dissimilarities from a subset of all the pairs. Then, rather than using an average value of dissimilarity for each object pair one would wish to retain the information contained in the interval or histogram of dissimilarities obtained for each pair of objects. Or, it might be useful to collect data reflecting the imprecision or fuzziness of the dissimilarity between each object pair. Then, the ij-th entry in the $n \times n$ data matrix, that is, the dissimilarity between objects i and j, is either an interval or an empirical distribution of values (a histogram). In these cases, the data matrix consists of symbolic data.

By now, MDS of symbolic data can be analyzed by several techniques. The case where the dissimilarity between each object pair is represented by a range or interval of values has been treated by Denœux and Masson (2000) and Masson and Denœux (2002). They model each object as alternatively a hyperbox (hypercube) or a hypersphere in a low dimensional space and use a gradient descent algorithm. Groenen et al. (2006) have developed an MDS technique for interval data which yields a representation of the objects as hyperboxes in a low-dimensional Euclidean space rather than hyperspheres because the hyperbox representation is reflected as a conjunction of p properties where p is the dimensionality of the space. We shall follow this latter approach here.

The hyperbox representation is interesting for two reasons. First a hyperbox is more appealing because it allows a strict separation between the units of the dimensions it uses. For example, the top speed of a certain type of car might be between 170 and 190 km/h and its fuel consumption between 8 and 10 liters per 100 km. These aspects can be easily described alternatively as an average top speed of 180 km/h plus or minus 10 km/h and an average fuel consumption of 9 liters per 100 km plus or minus 1. Both formulations are in line with the hyperbox approach. However, the hypersphere interpretation would be to state that the car is centered around a top speed of 180 km/h and a fuel consumption of 9 liters per 100 km and give a radius. The units of this radius cannot be easily expressed anymore. A second reason for using hyperboxes is that we would like to discover relationships in terms of the underlying dimensions. The use of hyperboxes leads to unique dimensions, whereas the the use of hyperspheres introduces the freedom of rotation so that dimensions are not unique anymore.

Groenen and Winsberg (2006) have extended the method developed by Groenen et al. (2006) to deal with the case in which the dissimilarity between object i and object j is an empirical distribution of values or, equivalently, a histogram.

All of the methods described above for MDS of symbolic data treat the two-way one-mode case. That is, they deal with a single data matrix. Here, we extend that approach to deal with the two-mode three-way case. We consider the case where each of K judges denote the dissimilarity between the i-th and j-th object pair as an interval, or a histogram thereby giving a range of values or a fuzzy dissimilarity. So, the accent here will be on individual differences. Of course, the method also applies to the case where data is collected for K conditions, where for each condition the dissimilarity between the i-th and j-th pair is an interval, or a histogram.

In the next section, we review briefly the I-Scal algorithm developed by Groenen et al. (2006) for MDS of interval dissimilarities based on iterative majorization. Then, we present an extension of the method to the three-way two-mode case and analyze an empirical data sets dealing with dissimilarities of sounds. The paper ends with some conclusions and suggestions for continued research.

2 MDS of interval dissimilarities

We now review briefly the case of two-way one-mode MDS of interval dissimilarities. In this case, an interval of a dissimilarity will be represented by a range of distances between the two hyperboxes of objects i and j. This objective is achieved by representing the objects by rectangles and approximate the upper bound of the dissimilarity by the maximum distance between the rectangles and the lower bound by the minimum distance between the rectangles. An example of rectangle representation is shown in Figure 1. It also indicates how the minimum and maximum distance between two rectangles is defined.

By using hyperboxes, both the distances and the coordinates are ranges. Let the coordinates of the centers of the rectangles be given by the rows of the $n \times p$ matrix $\mathbf{X}$, where n is the number of objects and p the dimensionality. The distance from the center of rectangle i along axis s, denoted by the spread, is represented by r_{is} which is by definition nonnegative. The maximum Euclidean distance between rectangles i and j is given by

$$d_{ij}^{(U)}(\mathbf{X},\mathbf{R}) = \left(\sum_{s=1}^{p} [|x_{is} - x_{js}| + (r_{is} + r_{js})]^2 \right)^{1/2} \tag{1}$$

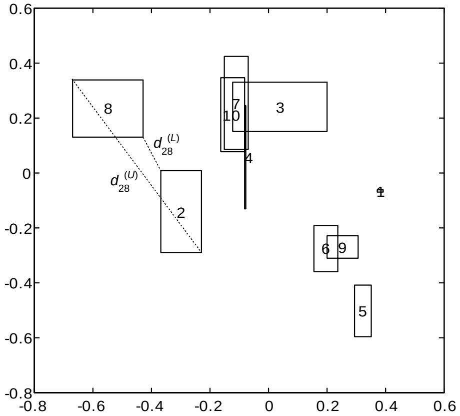


Fig. 1. Example of distances in MDS for interval dissimilarities where the objects are represented by rectangles.

and the minimum Euclidean distance by

$$d_{ij}^{(L)}(\mathbf{X},\mathbf{R}) = \left(\sum_{s=1}^{p} \max[0, |x_{is} - x_{js}| - (r_{is} + r_{js})]^2\right)^{1/2} . \tag{2}$$

This definition implies that rotation of the axes changes the distances between the hyperboxes because they are always parallel to the rotated axes. This sensitivity for rotation can be seen as an asset because it makes a solution rotational unique, which is not true for ordinary MDS. In the special case of $\mathbf{R} = \mathbf{0}$, the hyperboxes become points and the rotational uniqueness disappears as in ordinary MDS.

Symbolic MDS for interval dissimilarities aims at approximating the lower and upper bounds of the dissimilarities by minimum and maximum distances between rectangles. This objective is formalized by the I-Stress loss function

$$\sigma_{\mathrm{I}}^2(\mathbf{X},\mathbf{R}) = \sum_{i<j}^{n} w_{ij} \left[\delta_{ij}^{(U)} - d_{ij}^{(U)}(\mathbf{X},\mathbf{R})\right]^2 + \sum_{i<j}^{n} w_{ij} \left[\delta_{ij}^{(L)} - d_{ij}^{(L)}(\mathbf{X},\mathbf{R})\right]^2 ,$$

where $\delta_{ij}^{(U)}$ is the upper bound of the dissimilarity of objects i and j, $\delta_{ij}^{(L)}$ is the lower bound , and w_{ij} is a given nonnegative weight. $\sigma_{\mathrm{I}}^2(\mathbf{X},\mathbf{R})$ can be minimized by iterative majorization (see Groenen et al. (2006)).

Iterative majorization has the advantage that I-Stress is guaranteed to reduce in each iteration from any starting configuration until a stationary point is obtained. In practice, the algorithm stops at a stationary point that is a local minimum. Another important property for the purpose of this paper is that, in each iteration, the algorithm operates on a quadratic function in $\mathbf{X}$ and $\mathbf{R}$. Groenen et al. (2006) have derived the quadratic majorizing function for $\sigma_{\mathrm{I}}^2(\mathbf{X},\mathbf{R})$ as the one at the right hand side of

$$\begin{aligned}\sigma_{\mathrm{I}}^2(\mathbf{X},\mathbf{R}) \leq & \sum_{s=1}^{p}(\mathbf{x}_s'\mathbf{A}_s^{(1)}\mathbf{x}_s - 2\mathbf{x}_s'\mathbf{B}_s^{(1)}\mathbf{y}_s) \\ & + \sum_{s=1}^{p}(\mathbf{r}_s'\mathbf{A}_s^{(2)}\mathbf{r}_s - 2\mathbf{r}_s'\mathbf{b}_s^{(2)}) + \sum_{s=1}^{p}\sum_{i<j}(\gamma_{ijs}^{(1)} + \gamma_{ijs}^{(2)}), \end{aligned} \tag{3}$$

where $\mathbf{x}_s$ is column s of $\mathbf{X}$, $\mathbf{r}_s$ is column s of $\mathbf{R}$, $\mathbf{y}_s$ is column s of $\mathbf{Y}$ (the previous estimate of $\mathbf{X}$). The matrices $\mathbf{A}_s^{(1)}, \mathbf{B}_s^{(1)}, \mathbf{A}_s^{(2)}$, vectors $\mathbf{b}_s^{(2)}$, and scalars $\gamma_{ijs}^{(1)}, \gamma_{ijs}^{(2)}$ all depend dependent on previous estimates of $\mathbf{X}$ and $\mathbf{R}$, hence they are known at the present iteration. Their exact definition can be found in Groenen et al. (2006). For our purposes, it is important to realize that the majorizing function at the right of (3) is quadratic in $\mathbf{X}$ and $\mathbf{R}$, so that an update can be readily derived by setting the derivatives equal to zero.

Another important feature of the majorizing function being quadratic is that it becomes easy to impose the constraints that we will need for the extension to two-mode three-way symbolic MDS proposed in this paper. For more details on iterative majorization and its use in three-way MDS, see, for example, De Leeuw and Heiser (1980) and Borg and Groenen (2005).

3 Two-mode three-way MDS of interval data

The I-Scal algorithm developed by Groenen et al. (2006) can be extended quite easily to two-mode three-way interval data. In this case, we have an interval available of the dissimilarities available for replication $\ell = 1, \ldots, L$. Then, $\delta_{ij\ell}^{(L)}$ and $\delta_{ij\ell}^{(U)}$ are the lower and upper boundary of the interval of δ_{ij} for replication ℓ. Of course, a normal I-Scal solution could be computed for every replication separately. However, here we impose restrictions of the weighted Euclidean model similar to the Indscal approach of Carroll and Chang (1972).

The main idea is to have a single common space of hyperboxes and allow each replication ℓ to stretch or shrink the dimensions to fit its ranges of dissimilarities as good as possible. Let $\mathbf{X}$ and $\mathbf{R}$ denote here the centers and spreads of the hyperboxes in the common space. Then, the weighted Euclidean model restrictions imply that the hyperboxes for the individual replication ℓ are modelled as

$$\mathbf{X}_\ell = \mathbf{X}\mathbf{V}_\ell \tag{4}$$

$$\mathbf{R}_\ell = \mathbf{R}\mathbf{V}_\ell, \tag{5}$$

where $\mathbf{V}_\ell$ is a $p \times p$ diagonal matrix with dimension weights for replication ℓ. This objective can be obtained by minimizing the 3Way-IStress loss function

$$\sigma^2_{\text{3Way}}(\mathbf{X}, \mathbf{R}, \mathbf{V}_1, \ldots, \mathbf{V}_L) = \sum_\ell \sum_{i<j}^n w_{ij} \left[\delta^{(U)}_{ij\ell} - d^{(U)}_{ij}(\mathbf{X}\mathbf{V}_\ell, \mathbf{R}\mathbf{V}_\ell)\right]^2 + \sum_\ell \sum_{i<j}^n w_{ij} \left[\delta^{(L)}_{ij\ell} - d^{(L)}_{ij}(\mathbf{X}\mathbf{V}_\ell, \mathbf{R}\mathbf{V}_\ell)\right]^2 . \quad (6)$$

Note that without loss of generality, we may require that all diagonal weights in $\mathbf{V}_\ell$ are nonnegative. The reason is that a negative element only reflects an individual axis, but it does not change the distances between the hyperboxes. As the $\mathbf{X}$ and $\mathbf{R}$ are both multiplied by $\mathbf{V}_\ell$, there is nonuniqueness between the scale of the s-th column of $\mathbf{X}$ and $\mathbf{R}$ and the s-th diagonal value of the $\mathbf{V}_\ell$s denoted by $v_{ss\ell}$. To identify them, we impose the restriction $\sum_\ell v^2_{ss\ell} = L$ to (6), although it is sufficient to impose these restrictions after the algorithm has converged.

To find an algorithm for minimizing 3Way-IStress, we use the majorization results obtained for I-Stress. The first step is to apply the majorizing inequality of (3) to (6). Let $\mathbf{Y}_\ell$ and $\mathbf{Y}$ be the previous estimates of $\mathbf{X}_\ell$ and $\mathbf{X}$. Then,

$$\sigma^2_{\text{3Way}}(\mathbf{X}, \mathbf{R}, \mathbf{V}_1, \ldots, \mathbf{V}_L) \leq \sum_{s=1}^p \left(\sum_\ell \mathbf{x}'_{s\ell}\mathbf{A}^{(1)}_{s\ell}\mathbf{x}_{s\ell} - 2\sum_\ell \mathbf{x}'_{s\ell}\mathbf{B}^{(1)}_{s\ell}\mathbf{y}_{s\ell} \right) + \sum_\ell \sum_{s=1}^p \left(\mathbf{r}'_{s\ell}\mathbf{A}^{(2)}_{s\ell}\mathbf{r}_{s\ell} - 2\mathbf{r}'_{s\ell}\mathbf{b}^{(2)}_{s\ell} \right) + \sum_\ell \sum_{s=1}^p \sum_{i<j} (\gamma^{(1)}_{ijs\ell} + \gamma^{(2)}_{ijs\ell}). \quad (7)$$

To find updates it is convenient to substitute $\mathbf{X}_\ell = \mathbf{X}\mathbf{V}_\ell$, $\mathbf{R}_\ell = \mathbf{R}\mathbf{V}_\ell$, and $\gamma = \sum_\ell \sum_{s=1}^p \sum_{i<j} (\gamma^{(1)}_{ijs\ell} + \gamma^{(2)}_{ijs\ell})$ in (7), that is,

$$\sigma^2_{\text{3Way}}(\mathbf{X}, \mathbf{R}, \mathbf{V}_1, \ldots, \mathbf{V}_\ell) \leq \sum_{s=1}^p \left(\sum_\ell v^2_{ss\ell}\mathbf{x}'_s\mathbf{A}^{(1)}_{s\ell}\mathbf{x}_s - 2\sum_\ell v_{ss\ell}\mathbf{x}'_s\mathbf{B}^{(1)}_{s\ell}\mathbf{y}_{s\ell} \right) + \sum_\ell \sum_{s=1}^p \left(v^2_{ss\ell}\mathbf{r}'_s\mathbf{A}^{(2)}_{s\ell}\mathbf{r}_s - 2v_{ss\ell}\mathbf{r}'_s\mathbf{b}^{(2)}_{s\ell} \right) + \gamma. \quad (8)$$

The latter majorizing inequality shows that for fixed $\mathbf{V}_\ell$, the updates of $\mathbf{X}$ and $\mathbf{R}$ are independent because there is no cross product of elements of $\mathbf{X}$ and $\mathbf{R}$ in the quadratic majorizing function (8). The 3WaySym-Scal algorithm defined later updates $\mathbf{X}$ and $\mathbf{R}$ for fixed $\mathbf{V}_\ell$ followed by updating $\mathbf{V}_\ell$ for fixed $\mathbf{X}$ and $\mathbf{R}$ both using the majorizing function at the right of (8).

We start with deriving the update for $\mathbf{X}$. Rewriting the terms of (8) that are dependent on $\mathbf{X}$ gives

$$\sum_{s=1}^{p}\left(\mathbf{x}_s'\left[\sum_{\ell} v_{ss\ell}^2\mathbf{A}_{s\ell}^{(1)}\right]\mathbf{x}_s - 2\mathbf{x}_s'\left[\sum_{\ell} v_{ss\ell}\mathbf{B}_{s\ell}^{(1)}\right]\mathbf{y}_s\right). \tag{9}$$

Setting the derivatives equal to zero yields the linear system

$$\begin{gathered}\left[\sum_{\ell} v_{ss\ell}^2\mathbf{A}_{s\ell}^{(1)}\right]\mathbf{x}_s = \left[\sum_{\ell} v_{ss\ell}\mathbf{B}_{s\ell}^{(1)}\right]\mathbf{y}_s \\ \mathbf{A}\mathbf{x}_s = \mathbf{b}\end{gathered}$$

for all s where the second line is used for notational simplicity. As each matrix $\mathbf{A}_{s\ell}^{(1)}$ (and $\mathbf{B}_{s\ell}^{(1)}$) has the matrix $\mathbf{11}'$ in its null-space, it follows that $\mathbf{A}$ is not of full rank and $\mathbf{b}$ is column centered. Therefore, solving $\mathbf{A}\mathbf{x}_s = \mathbf{b}$ is the same as solving

$$(\mathbf{A}+\mathbf{11}')\mathbf{x}_s^{+} = \mathbf{b} \text{ or } \mathbf{x}_s^{+} = (\mathbf{A}+\mathbf{11}')^{-1}\mathbf{b}, \tag{10}$$

for each dimension s, where $\mathbf{x}_s^{+}$ denotes the update.

For the update for $\mathbf{R}$, we rewrite the terms of (8) that are dependent on $\mathbf{R}$ as

$$\sum_{s=1}^{p}\left(\mathbf{r}_s'\left[\sum_{\ell} v_{ss\ell}^2\mathbf{A}_{s\ell}^{(2)}\right]\mathbf{r}_s - 2\mathbf{r}_s'\left[\sum_{\ell} v_{ss\ell}\mathbf{b}_{s\ell}^{(2)}\right]\right). \tag{11}$$

Setting the derivatives of (11) equal to zero yields the update

$$\mathbf{r}_s^{+} = \left[\sum_{\ell} v_{ss\ell}^2\mathbf{A}_{s\ell}^{(2)}\right]^{-1}\left[\sum_{\ell} v_{ss\ell}\mathbf{b}_{s\ell}^{(2)}\right] \tag{12}$$

for each dimension s that is easily computed as each $\mathbf{A}_{s\ell}^{(2)}$ is diagonal.

For an update of $\mathbf{V}_\ell$ for fixed $\mathbf{X}$ and $\mathbf{R}$, consider rewriting the terms of (8) as

$$\sum_{s=1}^{p}\sum_{\ell}\left(v_{ss\ell}^2\left[\mathbf{x}_s'\mathbf{A}_{s\ell}^{(1)}\mathbf{x}_s + \mathbf{r}_s'\mathbf{A}_{s\ell}^{(2)}\mathbf{r}_s\right] - 2v_{ss\ell}\left[\mathbf{x}_s'\mathbf{B}_{s\ell}^{(1)}\mathbf{y}_{s\ell} + \mathbf{r}_s'\mathbf{b}_{s\ell}^{(2)}\right]\right)$$

for which the update becomes

$$v_{ss\ell}^{+} = \left[\mathbf{x}_s'\mathbf{A}_{s\ell}^{(1)}\mathbf{x}_s + \mathbf{r}_s'\mathbf{A}_{s\ell}^{(2)}\mathbf{r}_s\right]^{-1}\left[\mathbf{x}_s'\mathbf{B}_{s\ell}^{(1)}\mathbf{y}_{s\ell} + \mathbf{r}_s'\mathbf{b}_{s\ell}^{(2)}\right] \tag{13}$$

for all ℓ and s.

The 3WaySym-Scal algorithm for minimizing $\sigma_{\text{3Way}}^2(\mathbf{X}, \mathbf{R}, \mathbf{V}_1, \ldots, \mathbf{V}_L)$ using iterative majorization is shown in Figure 2.

```
Initialize $\mathbf{X}^{(0)}$, $\mathbf{R}^{(0)}$, and $\mathbf{V}_\ell^{(0)} = \mathbf{I}$ for all $\ell$.
Set $k := 0$, $\mathbf{X}^{(-1)} := \mathbf{X}^{(0)}$, $\mathbf{R}^{(-1)} := \mathbf{R}^{(0)}$, and $\mathbf{V}_\ell^{(-1)} = \mathbf{V}_\ell^{(0)}$ for all $\ell$.
Set $\epsilon$ to a small positive value.
While $k = 0$ or $\sigma^2_{\text{3Way}}(k-1) - \sigma^2_{\text{3Way}}(k) \leq \epsilon$
    $k := k + 1$.
    For $s = 1$ to $p$
        Compute the update of $\mathbf{x}_s$ by (10).
        Compute the update of $\mathbf{r}_s$ by (12).
        For $\ell = 1$ to $L$
            Compute the update of $v_{ss\ell}$ by (13).
        End for
    End for
    Set $\mathbf{X}^{(k)} := \mathbf{X}$, $\mathbf{R}^{(k)} := \mathbf{R}$, and $\mathbf{V}_\ell^{(k)} = \mathbf{V}_\ell$.
End
```

Fig. 2. The 3WaySym-Scal algorithm.

Instead of reporting σ^2_{3Way}, we shall report $\sigma^2_{\text{3Way}}/\eta^2$ with

$$\eta^2 = \sum_\ell \sum_{i<j} w_{ij}([\delta^{(U)}_{ij\ell}]^2 + [\delta^{(L)}_{ij\ell}]^2)$$

because $\sigma^2_{\text{3Way}}/\eta^2$ will be between 0 and 1 at a local minimum and is independent of the number of objects, the size of the dissimilarities, or the weights.

4 Synthesized musical instruments

To illustrate our method, we consider an empirical data set where the entries in each of two dissimilarity matrices are an interval of values. These two dissimilarity matrices represent dissimilarities among the same set of objects, given by the same expert on two different occasions; thus combined these two dissimilarity matrices to form a three-way two-mode array. The objects in the study are ten sounds differing with respect to only two physical parameters: the spectral center of gravity and the log attack time. Many previous studies of musical timbre have demonstrated that these two physical parameters are highly correlated with the perceptual axes uncovered when dissimilarity judgments are collected for sounds from different musical instruments playing the same note with the same loudness for the same duration of time.

Until some 35 years ago timbre was considered to be a perceptual parameter of sound that was complex and multidimensional, defined primarily by what it was not, that is what distinguishes two sounds presented in the same manner equal in pitch, subjective duration and loudness (see Plomp, 1970). MDS studies have shown that these two attributes of sound, namely spectral center of gravity and log attack time explain the factors we use to distinguish, say, middle C on the piano from middle C on some other instrument when they are played at the same loudness and the same duration of time (see, for example, McAdams, Winsberg, Donnadieu, DeSoete & Krimphoff, 1995; McAdams & Winsberg, 1999). So, when middle C is played on the piano the sound has some unidimensional attributes such as pitch, corresponding to the fequency of the fundamental, loudness, and duration. In addition, it is characterized by its timbre, that is, a note played by a piano and not some other musical instrument. This last attribute, timbre, is perceptually multidimensional with two important underlying perceptual dimensions relating to spectral center of gravity and log attack time. The spectral center of gravity is the weighted average of the harmonics generated when a note is sounded averaged over the duration of the tone with a running time window of, say, 12ms, and is higher for the harpsichord than for the piano, for example. The log attack time is the logarithm of the rise time measured from the time the amplitude envelope reaches a threshold of 2% of the maximum amplitude to the time it takes to reach the maximum amplitude, and is longer for a wind instrument like the trumpet than for a string instrument like the harp. The ten sounds in this study were generated artificially to represent the range of values found in natural instruments according to the design in Figure 3. The data represents dissimilarity judgments from the same expert listener taken on two occasions. The data are given in Table 2 of Groenen et al. (2006). On each occasion the expert listened to each pair of sounds and indicated a range of dissimilarity for each pair on a calibrated slider scale going from very similar to very different.

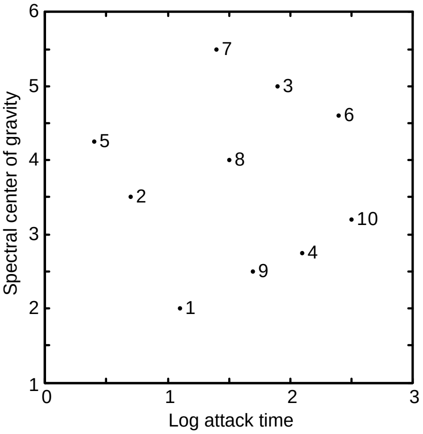
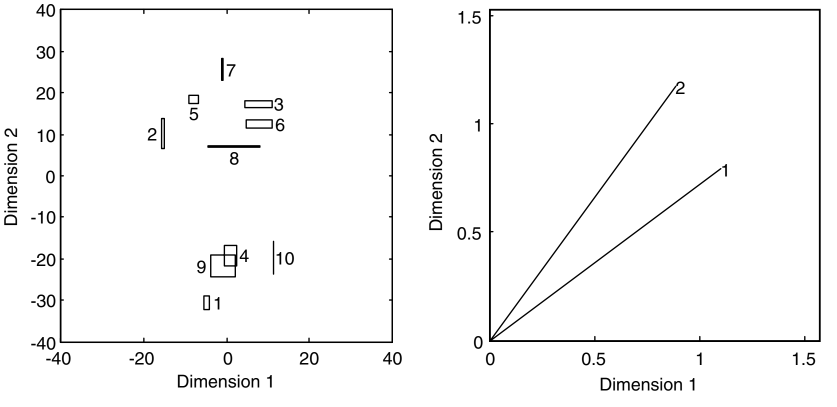


Fig. 3. Design of the ten sounds according to spectral center of gravity (vertical axis) and log attack time (horizontal axes).

The data were analyzed by 3WaySym-Scal for both occasions simultaneously. To reduce the probability of a bad local minimum we have used 100 random starts and chose the best one. The resulting solution with Stress 0.05194421 is shown in Figure 4. Here, the common space with **X** and **R** is shown in the left panel. The right panel shows the weights for the two occasions. We see that at Occasion 1, the first dimension is emphasized more than the second, whereas this situation is reversed at Occasion 2. Another representation of this very same solution can be obtained by showing the individual spaces for each of the occasions, thus using the $\mathbf{XV}_1$ and $\mathbf{RV}_1$ for the first occasion and $\mathbf{XV}_2$ and $\mathbf{RV}_2$ for the second occasion. This represention of the individual spaces is shown in Figure 5. We also present the results obtained analyzing the data from occasion one and occasion two separately using the I-Scal algorithm that is two separate two-way analyses in Figure 6.



a. Common space

b. Dimension weights $\mathbf{V}_l$ for individual spaces

Fig. 4. Common space and dimension weights for the 3WaySym-Scal solution for judgements on synthesized musical instruments for judgements on two occasions by a single professional judge.

It is informative to examine and compare the three-way solution treating the two data matrices simultaneously obtained with 3WaySym-Scal with the solutions obtained for each occasion separately using the two-way I-Scal algorithm. In each case, the horizontal axis represents log attack time and the vertical axes the spectral center of gravity. Without imposing any restrictions, each version of SymScal seems to be able to reconstruct the physical space. The results for the 3WaySym-Scal in Figure 4 indeed reflect the physical space. Notice the groupings 10, 9, 4, 1 and 2, 5, 7 and 3, 6, 8 reflect how these stimuli are grouped in the physical space. Moreover, the relation of these groups to one another approximates their disposition in the physical space reasonably well. The results for the second occasion analyzed alone

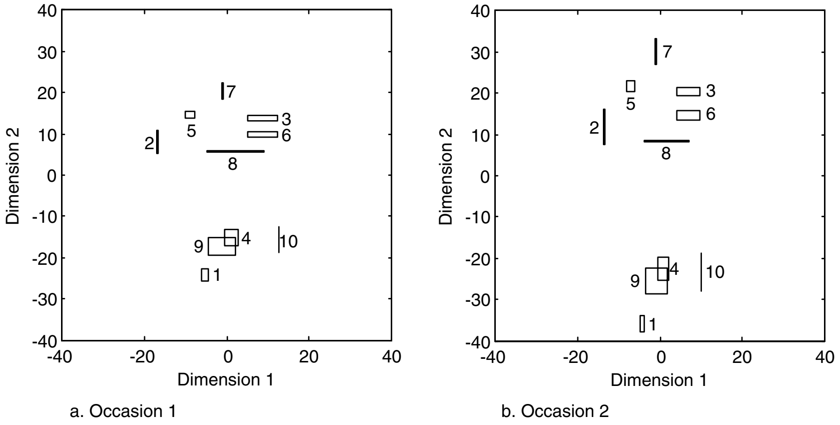


Fig. 5. Three-way interval MDS solution for judgements on synthesized musical instruments for judgements on two occasions by a single professional judge.

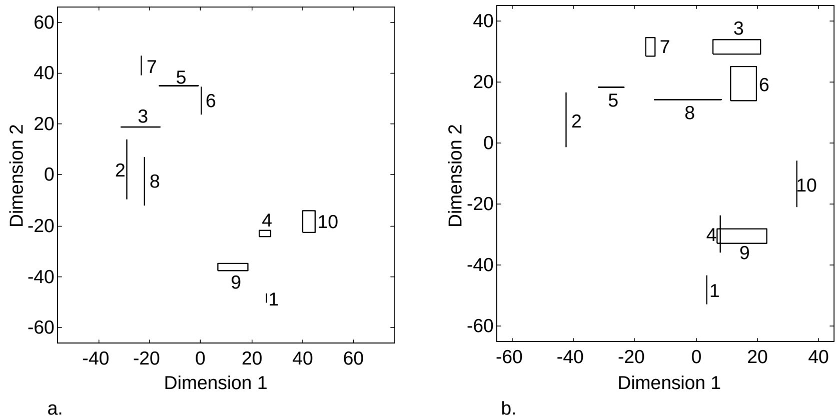


Fig. 6. Unconstrained I-Scal solutions for the sound data obtained by Groenen et al. (2006). Panel (a) gives the results for Occasion 1 with I-Stress .02861128 and Panel (b) for Occasion 2 with I-Stress .04893295.

reflect the physical space the best, and the solution from the first occasion alone shows the most deviations from the physical space: 8, 3, 6 are too far to the left, 3 is too low, 7 is too far to the left, and 1 is too far to the right. Note that these differences from one occasion to another are greater than the range of uncertainty reflected in the solutions. Analyzed alone without looking at the three-way solution one might want to conclude that the improved results on the second occasion indicate that the task is better performed with some practice and with greater familiarity with the group of sounds. However, the expert spent much time familiarizing himself with the sounds before undertaking the task. The results of the three-way analysis combined with the two-way solutions point to the much more interesting idea that greater attention to the spectral center of gravity was necessary to better reproduce the physical space. This additional most interesting information about sound perception could only be teased out by examining all the results. Of course, it also appears from the figures that sounds with long attack times are more difficult to localize, than those with short attack times (with exception to sound number 10).

5 Discussion and conclusions

We have presented an MDS technique for symbolic data that deals with three-way two-mode fuzzy dissimilarities consisting of a interval of values observed for each pair of objects, for each source. In this technique, each object is represented as a series of hyperboxes in a p dimensional space. By representing the objects as hypercubes, we are able to convey information contained when the dissimilarity between the objects or for any object pair needs to be expressed as a interval of values not a single value, and when one has data from more than one source. It may be so, moreover, that the precision inherent in the dissimilarities is such that the precision in one recovered dimension is worse than that for the other dimensions. Our technique is able to tease out and highlight this kind of information.

The 3WaySym-Scal algorithm for MDS of interval dissimilarities is based on iterative majorization, and the I-Scal algorithm created to deal with the case when dissimilarities are two-way, one-mode data and are given by a range or interval of values. The advantage is that each iteration yields better 3Way-IStress until no improvement is possible. Simulation studies have shown that I-Scal and Hist-Scal upon which this algorithm is based, combined with multiple random start and a rational start yields good quality solutions.

Denœux and Masson (2000) discuss an extension for interval data that allows the upper and lower bounds to be transformed. Although it is technically feasible to do so in our case, we do not believe that transformations are useful for symbolic MDS with interval or histogram data. The reason is that by having the available information of a given interval for each dissimilarity, it seems unnatural to destroy this information. Therefore, we recommend applying symbolic MDS without any transformation.

The present model can be extended along at least two lines. First, one could allow for individual rotations of the common space. It remains to be studied how this could be implemented. For example, one could only rotate $\mathbf{X}$ and not $\mathbf{R}$ or one could do both. A second line of extensions could study the use of intervals for $\mathbf{V}_\ell$ as well. The consequences also require further study.

References

BOCK, H.-H. and DIDAY, E. (2000): *Analysis of Symbolic Data* Springer, Berlin.

BORG, I. and GROENEN, P.J.F. (2005): *Modern Multidimensional Scaling: Theory and Applications, Second Edition* Springer, New York.

CARROLL, J.D. and CHANG, J.J. (1972): Analysis of individual differences in multidimensional scaling via an N-way generalization of Eckart-Young decomposition. *Psyhometika, 35, 283-319.*

DE LEEUW, J. and HEISER, W.J. (1980): Multidimensional scaling with restrictions on the configuration. In: P. R. Krishnaiah (Ed.), *Multivariate analysis (Vol. V)* (pp. 501-522). Amsterdam, The Netherlands: North-Holland.

DENŒUX, T. and MASSON, M. (2000): Multidimensional scaling of interval-valued dissimilarity data. *Pattern Recognition Letters, 21, 83-92.*

GROENEN, P.J.F., WINSBERG, S., RODRIGUEZ, O. and DIDAY, E. (2006): I-Scal: Multidimensional scaling of interval dissimilarities. *Computational Statistics and Data Analysis, 51, 360-378.*

GROENEN, P.J.F., and WINSBERG, S. (2006): Multidimensional scaling of histogram dissimilarities. In: V. Batagelj, H.-H. Bock, A. Ferligoj, and A. Ziberna (Eds.), *Proceedings of the ICFS Llubjana, Slovenia* (pp. 161-170). Springer, Berlin.

KRUSKAL, J.B. (1964): Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika, 29, 1-27.*

MASSON, M. and DENŒUX, T. (2002): Multidimensional scaling of fuzzy dissimilarity data; *Fuzzy Sets and Systems, 128, 339-352.*

MCADAMS,S. and WINSBERG,S. [1999]: Multidimensional scaling of musical timbre constrained by physical parameters. *The Journal of the Acoustical Society of America, 105, 1273.*

MCADAMS, S., WINSBERG, S., DONNADIEU, S., DESOETE, G., and KRIMPHOFF,J. (1995) Perceptual scaling of synthesized musical timbres: Common dimensions, specificities, and latent subject classes. *Psychological Research, 58, 177-192.*

PLOMP, R. (1970) : Timbre as a multidimensional attribute of complex tones. In: R. Plomp & G.F. Smoorenburg (Eds.) *Frequency analysis and periodicity detection in hearing* (pp 397-414. Leiden: Sijthoff

WINSBERG, S. and CARROLL, J.D. (1989): A quasi-nonmetric method for multidimensional scaling via an extended Euclidean model. *Psychomtrika, 54, 217-229.*

WINSBERG, S. and DESOETE, G. (1993): A latent class approach to fitting the weighted Euclidean model, CLASCAL. *Psychometika, 58, 31-330.*

Clustering and Validation of Interval Data

André Hardy and Joffray Baune

Department of Mathematics, University of Namur
8 Rempart de la Vierge, 5000 Namur, Belgium
andre.hardy@fundp.ac.be, joffray.baune@fundp.ac.be

Abstract. The paper addresses the problem of assessing the validity of the clusters found by a clustering algorithm. The determination of the "true" number of "natural" clusters has often been considered as the central problem of cluster validation. Many different stopping rules have been proposed in the research literature but most of them are applicable only to classical data (qualitative or quantitative). In this paper we investigate the problem of the determination of the number of clusters for symbolic objects described by interval variables. We consider five classical methods and two hypothesis tests based on the Poisson point process. We extend these methods to interval data. We apply them to the meteorological stations data set.

1 Introduction

The aim of cluster analysis is to identify a structure within a data set and to validate that structure. When hierarchical algorithms are used, an important problem is then to choose one solution in the nested sequence of partitions of the hierarchy. On the other hand, optimization methods for cluster analysis usually require the a priori specification of the number of classes. So most clustering procedures demand the user to fix the number of clusters, or to determine it in the final solution.

In this paper we describe two hypothesis tests for the number of clusters based on the Hypervolumes clustering criterion: the Hypervolumes test (Hardy (1996)) and the Gap test (Kubushishi (1996)). These statistical methods are based on the homogeneous Poisson process (Karr (1991)). We also consider the five best stopping rules for the number of clusters analysed by (Milligan and Cooper (1985)). We show how these methods and tests can be extended to interval data (Bock and Diday (2000)).

In order to generate partitions, we use symbolic clustering procedures: the module SHICLUST (Hardy (2004)) of the SODAS 2 software and the dynamical clustering method SCLUST (Verde, de Carvalho and Lechevallier (2000)).

2 Statistical models based on the Poisson processes

The clustering problem we are interested in is the following.
$E = \{x_1,\ x_2,\ ...,\ x_n\}$ is a set of n objects. On each of the objects we

measure the value of p interval variables $Y_1,\ Y_2,\ ...,\ Y_p$. The objective is to find a "natural" partition $P = \{C_1,\ C_2,\ ...,\ C_k\}$ of the set E into k clusters.

2.1 The Hypervolumes clustering criterion

The Hypervolumes clustering method (Hardy and Rasson (1982)) assumes that the n p-dimensional observation points $x_1,\ x_2,\ ...,\ x_n$ are generated by a homogeneous Poisson process in a set $\mathcal{D}$ included in the Euclidean space R^p where the set $\mathcal{D}$ is supposed to be the union of k disjoint convex domains $\mathcal{D}_1,\ \mathcal{D}_2,\ ...,\ \mathcal{D}_k$. We denote by $D_i \subset \{x_1,\ x_2,\ ..,\ x_n\}$ the subset of the points belonging to $\mathcal{D}_i$ $(1 \leq i \leq k)$. The Hypervolumes clustering criterion is deduced from that statistical model, using maximum likelihood estimation. It is defined by

$$W_k \equiv \sum_{i\,=\,1}^{k} m(H(D_i)) = \sum_{i\,=\,1}^{k} \int_{H(D_i)} m(dx)$$

where $H(D_i)$ is the convex hull of the points belonging to D_i and $m(H(D_i))$ is the multidimensional Lebesgue measure of that convex hull. That clustering criterion has to be minimized over the set of all the partitions of the observed sample into k clusters.

2.2 The generalized Hypervolumes clustering criterion

The generalized Hypervolumes clustering method (Rasson and Granville (1996)) assumes that the n p-dimensional points $x_1,\ x_2,\ ...,\ x_n$ are generated by a nonhomogeneous Poisson process in a set $\mathcal{D}$. $\mathcal{D}$ is the union of k disjoint convex domains $\mathcal{D}_1,\ \mathcal{D}_2,\ ...,\ \mathcal{D}_k$. The generalized Hypervolumes clustering criterion is deduced from that statistical model, using maximum likelihood estimation. It is defined by

$$WG_k \equiv \sum_{i\,=\,1}^{k} \varrho(H(D_i)) = \sum_{i\,=\,1}^{k} \int_{H(D_i)} q(x) m(dx)$$

where q(x) is the intensity of the nonhomogeneous Poisson process and $\varrho(H(C_i))$ the integrated intensity of the process on the convex hull of the points belonging to D_i.

3 Statistical tests for the number of clusters based on the homogeneous Poisson point process

The Hypervolumes test and the Gap test are presented in the case of classical quantitative data. In Section 6 we'll extend these tests to interval data.

3.1 The Hypervolumes test

The statistical model based on the homogeneous Poisson process allows us to define a likelihood ratio test for the number of clusters (Hardy (1996)). Let us denote by $D = \{D_1, D_2, ..., D_\ell\}$ the optimal partition of the sample into ℓ clusters and $C = \{C_1, C_2, ..., C_{\ell-1}\}$ the optimal partition into $\ell - 1$ clusters. We test the hypothesis H_0: $t = \ell$ against the alternative H_A: $t = \ell - 1$, where t denotes the number of "natural" clusters ($\ell \geq 2$). The test statistic (Hardy (1996)) is defined by

$$S(x) = \frac{W_\ell}{W_{\ell-1}}$$

where W_ℓ is the value of the Hypervolumes clustering criterion associated to the best partition into ℓ clusters.

Unfortunately the sampling distribution of the statistic S is not known. But $S(x)$ belongs to $[0, 1[$. Consequently, for practical purposes, we can use the following decision rule: reject H_0 if S is close to 1. We apply the test in a sequential way: if ℓ_0 is the smallest value of $\ell \geq 2$ for which we reject H_0, we choose $\ell_0 - 1$ as the best number of "natural" clusters. Let us mention that in order to compute p-values associated to the Hypervolumes statistic, we use permutation tests (Hardy (2006)).

3.2 The Gap test

The Gap test (Kubushishi (1996)) is based on the same statistical model. We test H_0 : the $n = n_1 + n_2$ observed points are a realization of a homogeneous Poisson process in $\mathcal{D}$ against H_A: n_1 points are a realization of a homogeneous Poisson process in $\mathcal{D}_1$ and n_2 points in $\mathcal{D}_2$ where $\mathcal{D}_1 \cap \mathcal{D}_2 = \emptyset$. The sets $\mathcal{D}, \mathcal{D}_1, \mathcal{D}_2$ are unknown. D (respectively D_1 , D_2) is the set of points belonging to $\mathcal{D}$ (respectively $\mathcal{D}_1$, $\mathcal{D}_2$). The test statistic can be written as

$$Q(x) = \left(1 - \frac{m(\triangle)}{m(H(D))}\right)^n$$

where $\triangle = H(D) \backslash (H(D_1) \cup H(D_2))$ is defined as the "gap space" between the clusters. The test statistic is the Lebesgue measure of the gap space between the clusters.

The decision rule is the result of an asymptotic distribution (Kubushishi (1996)). We reject H_0, at level α, if

$$\frac{nm(\triangle)}{m(H(D))} - \log n - (p-1) \log\log n \geq -\log(-\log(1-\alpha)).$$

4 Classical stopping rules for the number of clusters

Many different methods for the determination of the number of clusters have been published in the scientific literature. The most detailed and complete comparative study has been undertaken by Milligan and Cooper (1985). They analysed and classified thirty indices for the determination of the number of clusters, and they investigated the extent to which these indices were able to detect the correct number of clusters in a series of simulated data sets containing a known structure. The five rules investigated in this study are defined below in the case of classical quantitative data, in the order in which they were ranked in the Milligan and Cooper's investigation. In Section 6 we'll extend these methods to interval data.

The Calinski and Harabasz method The (Calinski and Harabasz (1974)) index is given by

$$CH = \frac{\frac{B}{c-1}}{\frac{W}{n-c}}$$

where n is the total number of objects, and c the number of clusters in the partition. W and B denote, respectively, the total within-cluster sum of squared distances (about the centroids), and the total between-clusters sum of squared distances.

The maximum value of the index is used to indicate the true number of clusters in the data set.

The *J*-index Duda and Hart (1973) proposed a hypothesis test for deciding if a cluster should be subdivided into two sub-clusters. The test statistic is based on the comparison between W_1, the within-cluster sum of squared distances, and W_2, the sum of within-cluster sum of squared distances when the cluster is optimally partitioned into two clusters. The null hypothesis of a single cluster is rejected if

$$J \equiv \frac{-\dfrac{W_2}{W_1} + 1 - \dfrac{2}{\pi p}}{\sqrt{\dfrac{2(1 - \frac{8}{\pi^2 p})}{np}}} > z_{1-\alpha}$$

where p denotes the number of variables, n the number of objects in the cluster being investigated, and $z_{1-\alpha}$ a standard normal quantile specified by the significance level of the test.

The *C*-index That index needs the computation of the sum of all within-cluster pairwise dissimilarities. If the partition has r such dissimilarities, we denote by V_{min} (respectively, V_{max}) the sum of the r smallest (respectively, largest) pairwise dissimilarities. The C-index (Hubert and Levin (1976)) is then defined by

$$C = \frac{V - V_{min}}{V_{max} - V_{min}}$$

where V is the sum of the within cluster dissimilarities.

The minimum value of that index across the partitions into ℓ clusters ($\ell = 1, ..., K$) is used to indicate the optimal number of clusters. The best minimal value is 0 corresponding to $V = V_{min}$.

The *Γ*-index Here comparisons are made between all within-cluster pairwise dissimilarities and all between-cluster pairwise dissimilarities. A comparison is defined as consistent (respectively, inconsistent) if a within-cluster dissimilarity is strictly less (respectively, greater) than a between-cluster dissimilarity. The Γ-index (Baker and Hubert (1975)) is computed as

$$\Gamma = \frac{\Gamma_+ - \Gamma_-}{\Gamma_+ + \Gamma_-}$$

where Γ_+ (respectively Γ_-) represents the number of consistent (respectively, inconsistent) comparisons.

The maximum value of the Γ-index indicates the correct number of clusters. Let us remark that the absolute maximum of that index is 1, when $\Gamma_- = 0$.

The Beale test Beale (1969) proposed a hypothesis test in order to decide if a cluster, at a level of a hierarchy, should be divided into two clusters. The test involves the statistic

$$\frac{\dfrac{W_1 - W_2}{W_2}}{\left(\dfrac{n-1}{n-2}\right) 2^{\frac{2}{p}} - 1}$$

where W_1, W_2, n and p are defined as in the J-index above.

Under the null hypothesis that the cluster should not be subdivided, that statistic has a Fisher and Snedecor distribution with p and $(m-2)p$ degrees of freedom.

A comparative analysis of these five stopping rules for the determination of the best number of natural clusters has been undertaken for classical quantitative data by Hardy and André (1998).

5 Dissimilarity measures for interval data

Clustering algorithms and methods for the determination of the number of clusters are usually based on a dissimilarity matrix D which reflects the similarity structure of the n objects. Such a dissimilarity matrix can be defined for interval data.

Let $E = \{x_1, \ldots, x_n\}$ be a set of n objects described by p interval variables $Y_1, \ldots, Y_p$ with domains $\mathcal{Y}_1, \ldots, \mathcal{Y}_p$ respectively.

p dissimilarity indices δ_1, ..., δ_p are defined on the ranges $\mathcal{B}_j$

$$\delta_j : \mathcal{B}_j \times \mathcal{B}_j \to R^+ : (x_{kj}, x_{\ell j}) \longmapsto \delta_j(x_{kj}, x_{\ell j})$$

where $\mathcal{B}_j$ is the set of intervals of R.

If $x_{kj} = [\alpha_{kj}, \beta_{kj}]$ and $x_{\ell j} = [\alpha_{\ell j}, \beta_{\ell j}]$, we consider in this paper the three following distances for interval data.

- The Hausdorff distance:

$$\delta_j\ (x_{kj}, x_{\ell j}) = \max\{ \mid \alpha_{kj} - \alpha_{\ell j} \mid, \mid \beta_{kj} - \beta_{\ell j} \mid \}$$

- The L_1 distance :

$$\delta_j\ (x_{kj}, x_{\ell j}) = \mid \alpha_{kj} - \alpha_{\ell j} \mid + \mid \beta_{kj} - \beta_{\ell j} \mid$$

- The L_2 distance :

$$\delta_j\ (x_{kj}, x_{\ell j}) = \sqrt{(\alpha_{kj} - \alpha_{\ell j})^2 + (\beta_{kj} - \beta_{\ell j})^2}.$$

The dissimilarity indices δ_1, ..., δ_p are combined in order to obtain a global dissimilarity measure on E

$$\begin{aligned} d :\ & E \times E \longrightarrow R^+ \\ & (x_k, x_\ell) \longmapsto d(x_k, x_\ell) = \left(\sum_{j=1}^{p} \delta_j^2(x_{kj}, x_{\ell j}) \right)^{1/2} \end{aligned}$$

where δ_j is one of the dissimilarity measures defined above.

In order to generate hierarchies or sets of partitions, we apply symbolic clustering methods to interval data: SHICLUST (Hardy (2004)) and SCLUST (Verde, de Carvalho and Lechevallier (2000)). SHICLUST is a module of the SODAS 2 software containing the symbolic extensions of four well-known classical clustering methods: the single link, complete link, centroid and Ward procedures. The symbolic clustering method SCLUST is a generalization of the Dynamic clustering method (Diday (1972)).

6 Determination of the number of clusters for interval data

6.1 Methods based on a dissimilarity matrix

The five best methods for the determination of the number of clusters from the (Milligan and Cooper (1985)) study are based on a dissimilarity matrix. Such a dissimilarity matrix can be computed for interval data (Section 5).

Consequently, the five corresponding stopping rules can be used for interval data, in order to determine the best number of natural clusters. These five methods have been included in a module of the SODAS 2 software named NBCLUST. The indices are computed at each level of the four hierarchies of SHICLUST.

Concerning SCLUST, we select the best partition into ℓ clusters, for each value of ℓ ($\ell = 1, \cdots, K$) (K is a reasonably large integer fixed by the user) and we apply the three stopping rules available for nonhierarchical classification (the Calinski and Harabasz method, the C-index and the Γ-index) (Hardy, Lallemand and Lechevallier (2002)).

The analysis of these indices should provide the "best" number of clusters.

6.2 Statistical tests based on the Poisson point processes

The Hypervolumes test and the Gap test are not based on the existence of a dissimilarity matrix. They need the computation of convex hulls of points. In order to extend these tests to interval data, we use the following modelling: an interval is summarized by two numbers: its center and its half length ((C, L) modelling). So each object can be represented as a point in a $2p$-dimensional space where p is the number of interval variables measured on each object.

For practical purposes, the Hypervolumes test and the Gap test are applied to the points obtained thanks to the (C, L) modelling, in the p two-dimensional spaces associated to each of the p interval variables.

When the Hypervolumes test is applied to the hierarchies of partitions generated by each of the four hierarchical methods included in SHICLUST, it computes, in each of the p (C, L) representations, the areas of the convex hulls corresponding to the generated partitions, at the corresponding level of the hierarchy. Consequently the number of clusters obtained with one interval variable can be different from the number of clusters obtained with another interval variable.

In order to solve that problem we select from the set of all the variables the most discriminant one and we apply the Hypervolumes test and the Gap test in the two-dimensional (C, L) space associated to that variable.

The total inertia (Celeux et al. (1989)) of the set E of objects is defined by

$$T = \sum_{i=1}^{n} (x_i - g)' \ (x_i - g)$$

where g is the centroid of E.

It is possible to compute the contribution of the class C_ℓ or of the j-th variable to the total inertia T.

$$T = \sum_{i=1}^{n} (x_i - g)' \ (x_i - g) = \sum_{\ell=1}^{k} \sum_{j=1}^{p} \sum_{x_i \in C_\ell} (x_{ij} - g_j)^2 = \sum_{\ell=1}^{k} \sum_{j=1}^{p} T_j^{(\ell)} = \sum_{\ell=1}^{k} T^{(\ell)}$$

where $\quad T^{(\ell)} = \sum\limits_{j=1}^{p} T_j^{(\ell)}$.

We also have

$$T = \sum_{i=1}^{n} (x_i - g)' \ (x_i - g) = \sum_{\ell=1}^{k} \sum_{j=1}^{p} \sum_{x_i \in C_\ell} (x_{ij} - g_j)^2 = \sum_{\ell=1}^{k} \sum_{j=1}^{p} T_j^{(\ell)} = \sum_{j=1}^{p} T_j$$

where $T_j = \sum\limits_{\ell=1}^{k} T_j^{(\ell)}$.

So $T^{(\ell)}$ is the contribution of class C_ℓ to the total inertia T. T_j is the contribution of variable j to the total inertia T.

We have a similar decomposition for the inter-class inertia B.

$$B = \sum_{\ell=1}^{k} n_\ell \ (g^{(\ell)} - g)' \ (g^{(\ell)} - g) = \sum_{\ell=1}^{k} \sum_{j=1}^{p} n_\ell \ (g_j^{(\ell)} - g_j)^2 = \sum_{\ell=1}^{k} \sum_{j=1}^{p} B_j^{(\ell)} =$$

$$= \sum_{\ell=1}^{k} B^{(\ell)} \text{ where } B^{(\ell)} = \sum_{j=1}^{p} B_j^{(\ell)} \ \text{ and } \ g^{(\ell)} \text{ is the centroid of } C_\ell.$$

We also have

$$B = \sum_{\ell=1}^{k} n_\ell \ (g^{(\ell)} - g)' \ (g^{(\ell)} - g) = \sum_{\ell=1}^{k} \sum_{j=1}^{p} n_\ell \ (g_j^{(\ell)} - g_j)^2 = \sum_{\ell=1}^{k} \sum_{j=1}^{p} B_j^{(\ell)} = \sum_{j=1}^{p} B_j$$

where $B_j = \sum\limits_{\ell=1}^{k} B_j^{(\ell)}$.

So $B^{(\ell)}$ is the contribution of class C_ℓ to the inter-class inertia B and B_j is the contribution of the j-th variable to the inter-class inertia B.

Thanks to these decompositions, we can determine the most discriminant variable by the following indices:

$$cor(j) = 100.\tfrac{B_j}{T_j} \qquad\qquad ctr(j) = 100.\tfrac{B_j}{B}$$

where B_j is the contribution of the jth variable to the inter-class inertia B, and T_j the contribution of the jth variable to the total inertia T.

We proceed in the following way: the symbolic clustering method SCLUST is applied to the original interval data. We consider the successive partitions of E into ℓ clusters ($\ell = 1, ..., K$ where K is a reasonably large integer fixed by the user). We transform the symbolic objects into classical data using the (C, L) modelling. We apply the Hypervolumes test and the Gap test in the two-dimensional space associated to the most discriminant interval variable.

7 Example: The meteorological stations data set

That data set is extracted from the Long-Term Instrumental Climatic Database of the People's Republic of China. It contains, among other variables, the temperatures observed in 60 meteorological stations. We will consider 12 interval variables: the monthly temperatures observed during the year 1988. Each observation is coded as the interval of the minima and maxima temperatures for each month.

7.1 Application of clustering algorithms

We apply the SCLUST procedure to the interval data with the Hausdorff distance; we consider the best partitions into ℓ clusters (ℓ = 1, ..., 7). The results given by the application of the three stopping rules of NBCLUST available for nonhierachical clustering methods are presented in Tables 1 and 2.

Number of clusters	Calinski and Harabasz	C-index	Γ-index
k=7	91.71558	0.00905	0.93729
k=6	90.40053	0.01282	0.91742
k=5	88.03642	0.02192	0.86976
k=4	84.52790	0.02980	0.87722
k=3	77.88730	0.05797	0.80540
k=2	$\boxed{95.14143}$	0.07597	0.83494
k=1	-	-	-

Table 1. SCLUST and NBCLUST.

The application of the Calinski and Harabasz method lead to the conclusion that the natural structure contains two clusters. The values of the two other indices are not relevantly interpretable.

We've also applied the SHICLUST and NBCLUST modules with the Hausdorff distance. The results are given in Table 3 for the centroid clustering method.

SCLUST	$C.\&H.$	C	Γ
Number of clusters	2	-	-

Table 2. SCLUST and NBCLUST.

Centroid	Calinski and	J-index	C-index	Γ-index	Beale
clustering	Harabasz				test
k=7	56.55797	3.10151	0.01634	0.91143	2.44697
k=6	56.31483	3.77347	0.01907	0.90563	3.64673
k=5	62.09626	2.04255	0.02406	0.89001	1.66512
k=4	66.62522	3.70170	0.03645	0.84857	2.92233
k=3	59.45751	6.19180	0.05254	0.84709	5.22437
k=2	$\boxed{94.37150}$	$\boxed{3.96272}$	0.05598	0.83883	$\boxed{3.58061}$
k=1	-	11.12716	-	-	11.47337

Table 3. Centroid and NBCLUST.

The Calinski and Harabasz index, the Duda and Hart test and the Beale test, applied on the hierarchies of partitions given by the centroid method, lead to the conclusion that there are two clusters in the data set.

If we consider the other hierarchical clustering methods included in the SHICLUST module, we get the following results (Table 4).

SHICLUST	$C.\&H.$	$D.\&H.$	C	Γ	Beale
Single link	6	6	6	6	6
Complete link	2	2	-	-	2
Centroid	2	2	-	-	2
Ward	2	2	-	-	2

Table 4. SHICLUST and NBCLUST.

Summarizing all these results, we conclude that the most interesting partitions contain respectively two and six clusters, the partition into two clusters appearing more often than the one into six clusters. For the partition into two clusters, the first one is composed of seaside or low-lying stations, and the second one to landlocked stations or at high altitude.

7.2 The zoom-stars

This graphical tool (Noirhomme and Rouard (2000)), available in the SODAS 2 software, is relevant for the visualization and the interpretation of the clusters. For each variable, the limits (min/max) of the interval of its values is represented. These limits are both linked and the whole surface is filled. If we examine the partition into two clusters (Figure 1), we can see that the

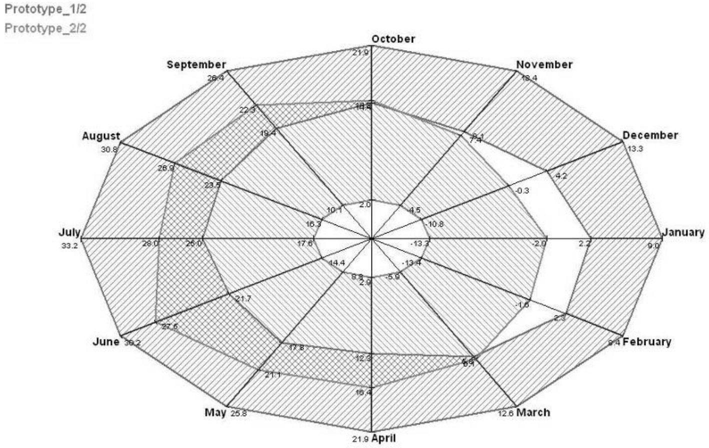


Fig. 1. Visualisation of the interval data.

two clusters of meteorological stations have different temperatures, and that the stations in cluster 1 have higher temperatures than those in cluster 2. The zoom-star associated to the partition into 6 clusters is more difficult to interpret, but further investigations prove the interest of that classification (Baune (2006)).

7.3 Stability measure and selection of the number of clusters

Some indices have been developed by (Bertrand and Bel Mufti (2006)) for measuring the stability (isolation and cohesion) of a partition. The module "STATCLUST" of the SODAS 2 software computes these stability indices in the case of interval variables. For the meteorological stations data set, the results are presented in Table 5.

k	$\boxed{2}$	3	4	5	$\boxed{6}$	7	8
Stability measure of partition validity	$\boxed{0.998}$	0.787	0.804	0.971	$\boxed{0.991}$	0.949	0.885

Table 5. Stability indices.

The partition into two clusters is characterized by the highest value of the stability indice.

8 Conclusion

In this paper we were interested in the determination of the number of clusters for symbolic objects described by interval variables. In order to generate partitions we have applied the four hierarchical procedures included in the module SHICLUST, and the dynamical clustering method SCLUST. The determination of the best number of natural clusters has been undertaken, by proposing and using a symbolic extension to interval data of two hypothesis tests based on the homogeneous Poisson process and five classical stopping rules well-known in the scientific literature. The stopping rules were applied to a real data set: the meteorological stations data set.

References

BAKER, F.B. and HUBERT, L.J. (1975): Measuring the power of hierarchical cluster analysis. *Journal of the American Statistical Association, 70, 31-38.*

BAUNE, J. (2006): *SYKSOM, Méthode de Représentation et de Classification de Données Symboliques basée sur les Cartes de Kohonen.* Mémoire, FUNDP - University of Namur, Namur, Belgium.

BEALE, E.M.L. (1969): Euclidean cluster analysis. *Bulletin of the International Statistical Institute 43 (2), 92-94.*

BERTRAND, P. and BEL MUFTI, G. (2006): Loevinger's measures of rule quality for assessing cluster stability. *Computational Statistics and Data Analysis 50, 992-1015.*

BOCK, H.-H. and DIDAY, E. (2000): *Analysis of Symbolic Data.* Springer, Berlin.

CALINSKI, T. and HARABASZ J. (1974): A dendrite method for cluster analysis. *Communications in Statistics 3 (2), 1-27.*

CELEUX, G., DIDAY, E., GOVAERT, G. and LECHEVALLIER, Y. (1989): *Classification Automatique des Données.* Bordas.

DIDAY, E. (1972): *Nouveaux Concepts et Nouvelles Méthodes en Classification Automatique.* Thèse d'Etat, Université Paris VI.

DUDA R.O. and HART, P.E. (1973): *Classification and Scene Analysis.* Wiley.

GORDON, A.D. (1996): How Many Clusters? An investigation of five procedures for detecting nested cluster structure. In: C. Hayashi et al. (Eds): *Data Science, Classification, and Related Methods.* Springer, Berlin, 109-116.

HARDY, A. and RASSON, J.P. (1982): Une nouvelle approche des problèmes de classification automatique. *Statistique et Analyse des Données 23, 41-56.*

HARDY, A.(1996): On the number of clusters. *Computational Statistics and Data Analysis 23, 83-96.*

HARDY, A. and ANDRE, P. (1998): An investigation of nine procedures for detecting the structure in a data set. In: A. Rizzi, M. Vichi and H.-H. Bock (Eds.): *Advances in Data Science and Classification.* Springer, Berlin, 29-36.

HARDY, A., LALLEMAND, P. and LECHEVALLIER, Y. (2002): La détermination du nombre de classes pour la méthode de classification symbolique SCLUST. In: *Actes des Huitièmes Rencontres de la Socité Francophone de Classification*, 27-31.

HARDY, A. (2004): Les méthodes de classification et de détermination du nombre de classes: du classique au symbolique. In: M. Chavent et al. (Eds.): *Comptes rendus des 11èmes Rencontres de la Société Francophone de Classification*, 48-55.

HARDY, A. (2006): *Application of permutation tests to clustering.* Technical Report, Department of Mathematics, FUNDP - University of Namur, Namur, Belgium.

HUBER, L.J. and LEVIN, J.R. (1976): A general statistical framework for assessing categorical clustering in free recall. *Psychological Bulletin 83, 1076-1080.*

KARR, A.F. (1991): *Point Processes and their Statistical Inference*, Marcel Dekker.

KUBUSHISHI, T. (1996): *On some Applications of the Point Process Theory in Cluster Analysis and Pattern Recognition.* PhD Thesis, FUNDP - University of Namur, Namur, Belgium.

MILLIGAN, G.W. and COOPER, M.C. (1985): An examination of procedures for determining the number of clusters in a data set. *Psychometrika 50, 159-179.*

NOIRHOMME-FRAITURE, M. and ROUARD, M. (2000): Visualizing and editing symbolic objects. In H.-H. Bock and Diday, E. (Eds.): *Analysis of Symbolic Data.* Springer, Berlin, 125-138.

PIRÇON, J.Y. (2004): *Le clustering et les processus de Poisson pour de nouvelles méthodes monothétiques.* PhD. thesis, FUNDP - University of Namur, Namur, Belgium.

RASSON, J.P. and GRANVILLE, V. (1996): Geometrical tools in classification. *Computational Statistics and Data Analysis 23, 105-123.*

VERDE, R., DE CARVALHO, F. and LECHEVALLIER, Y. (2000): A dynamical clustering algorithm for multi-nominal data. In: H.A.L. Kiers et al. (Eds.): *Data Analysis, Classification, and Related Methods.* Springer, Heidelberg, 387-393.

Building Symbolic Objects from Data Streams

Georges Hébrail[1] and Yves Lechevallier[2]

[1] Laboratoire LTCI - UMR 5141 CNRS
Ecole nationale supérieure des télécommunications
46, Rue Barrault, 75013 Paris, France
Georges.Hebrail@enst.fr

[2] INRIA-Rocquencourt,
78153 Le Chesnay Cedex, France
Yves.Lechevallier@inria.fr

Abstract. With the increase of computer use in all sectors of activity, more and more data are available as streams of structured records so that it is not possible to store all data before analyzing them in a data mining perspective. New data management systems have been studied to handle such data streams and new algorithms have been developed to perform stream mining. In this paper, we propose approaches to extend the construction of symbolic objects to data streams: symbolic objects are built and maintained as a representation of a complete stream or a sliding window on the stream.

1 Introduction

More and more data are available today with the development of computerized applications to support human activity. In this context the volume of data submitted to data mining or statistical data analysis methods becomes so large that sometimes all the data necessary for an analysis cannot be stored in a file or in a data base beforehand. Moreover more and more systems (for instance remote sensors) produce data as streams of structured records (i.e. tuples) at a rate which prevents from storing them in a file before analyzing them. This suggested the development of data stream management systems and 'on the fly' data mining algorithms (see Babcock et al. (2002) and Golab and Oszu (2003)).

Similarly to the data mining approach, stream data are not collected for a statistical analysis purpose but are available just because of the computerized nature of the management of human activities. So, goals of statistical analyses are usually not defined before the streams to begin or before the maximum storage of data to be reached. Consequently, there is a strong need for summarizing data streams in order to enable future rich analyses without storing all the available data. Several approaches have been proposed to summarize data streams, for instance: micro-clustering techniques are described in Aggarwal et al. (2003), sampling techniques are described in Csernel et al. (2006).

One important characteristic of data streams is that the distributions of variables evolve with time: this modifies the underlying assumptions of several statistical methods, i.e. samples are issued from the same distribution. In order to solve this problem, summaries are split into separate pieces figuring different periods of time. The constraint of storing the summaries in limited storage has motivated the development of storage approaches with less details for older data than for recent data.

Another way of getting over the problem of data stream volume is to restrict the scope of analyses by defining sliding windows on the streams, for instance applying the analysis on the last 24 hours data or on the last 1000 records. This approach also eliminates the problem of distribution drift. When applying analyses to a sliding window, there are two possibilities: (1) there is enough available storage to memorize all detailed data of the current window ; (2) the size of detailed data belonging to the current window is too large and cannot be stored. In case (1) standard algorithms can be applied but in case (2), algorithms must be incremental and support deletions in order to forget older detailed data getting out of the window as time goes on.

In this paper, we propose approaches to maintain symbolic objects describing the contents of a stream or a sliding window defined on a stream. Symbolic objects are good summaries of detailed data because they can be analysed by methods developed in the context of symbolic data analysis (see Bock and Diday (1999)). To do so, we adapt the algorithms developed in Stephan et al. (1999) to the case of data available in the form of data streams instead of data bases.

In Section 2 we recall the principles of generation of symbolic objects from relational databases. Recent work on data streams suggest some extensions both on existing generalization operators and on the symbolic object structure itself: these extensions are described in Section 3. Section 4 proposes approaches to build and maintain symbolic objects describing a data stream or a sliding window on a data stream. Section 5 is conclusion and suggests new directions of further work.

2 Generation of symbolic objects from relational databases

We use the *two-level-paradigm* where symbolic objects are created quite naturally when aggregating single individuals (described by classical single-valued variables) into classes, and describing the more or less complex properties of these classes. Here, we focus on the generalization process from a classical data set extracted from a relational database.

In this two-level paradigm the DB2SO software (see Stephan et al. (1999)) generates Symbolic Objects (called SO's) from the contents of a database. As described in Bock and Diday (1999), a SO is defined by a triple (a, R, d) where d is a *description*, R is a *comparison operator* between descriptions, and a is a *mapping* which defines the extension of the SO. We focus here on the construction of the description part of SO's. Each description of SO generated by DB2SO is the representation of a group of individuals by some symbolic variables. Symbolic variables describe variations among each group of individuals by figuring:

- The interval of observed values on individuals in the group for numerical variables,
- The list of observed values on individuals in the group for nominal variables,
- The probability distribution of observed values on individuals in the groups for nominal variables, when the user asks for a modal SO.

2.1 Standard basic process

Input of the basic process is a table describing individuals from a population. Individuals are described either by numerical or nominal variables. The user writes an SQL query which returns such a table with the following expected structure: the first column describes the individual ID, the second one the group ID the individual belongs to, and the other columns represent characteristics of individuals.

As an example, we consider here a click stream data set featuring log files of web sites. Each log file contains all requests over a continuous day period and is formatted in the CLF format (Common Log Format, see Mobasher (2000)). Some data preprocessing is done in four steps: *data fusion, data cleaning, data structuration, data summarization.* Generally, in the data fusion step, log files from different web servers are merged into a single log file. During data cleaning, non-relevant resources (e.g. jpg, js, gif files) and robots are eliminated (Arnoux et al., 2003) (Tanasa and Trousse, 2004). Then, preprocessed data are stored in a relational database. Tab.1 shows an example of one table of this relational data base: the REQUEST table. Each row of this table is a web site request from one user: the IDrequest column identifies each request, IPaddress contains the IPaddress of the user issuing the request, Protocol is the type of request, Date and Time are the ones of the request, Code indicates the return code of the request, Size is the number of bytes transmitted.

Another table of the data base is the ERRORCODE table (see Tab.2) containing the labels of different error codes.

In this example, the goal is to build one SO per user session where a user session is defined by an IP address. So groups are user sessions and individuals are requests associated with each user session. An SQL query is written to provide data submitted to DB2SO in the right form, for instance here to transform the error code by its label:

IDrequest	IPaddress	Protocol	Time	Date	Code	Size
1	88.121.0.121	HTTP/1.1	11:02:26 +0200	01/Oct/2006	200	9670
2	83.123.93.164	HTTP/1.1	11:02:27 +0200	01/Oct/2006	200	7534
3	80.100.0.101	HTTP/1.1	11:02:28 +0200	10/Oct/2006	200	7534
...	...					
10987	88.121.0.121	HTTP/1.1	17:26:24 +0200	10/Oct/2006	200	7016
10988	88.122.1.141	HTTP/1.1	17:28:26 +0200	01/Oct/2006	304	5950
...	...					

Table 1. Data table REQUEST.

Code	Label
200	OK
...	...
304	Not Modified
305	Use Proxy
...	...
510	Not Extended

Table 2. Data table ERRORCODE.

select IDrequest, IPaddress, Label, Size **from** REQUEST, ERRORCODE **where** REQUEST.Code = ERRORCODE.Code;

The description of each Symbolic Object (SO) is generated using a *generalization operator* to aggregate individuals of each group:

- *Numerical variables* describing individuals lead to *interval variables* describing groups: generally the set of variable values is generalized by the minimum and maximum values (see for instance the *Size* variable in Tab.3),
- *Nominal variables* describing individuals lead either to *boolean* or *modal* multi-valued variables describing groups. If the user chooses to generate a boolean multi-valued variable, the generalization operator is simply building the list of observed values within the group. If the user chooses to generate a modal multi-valued variable, the generalization operator builds the discrete probability distribution of the nominal variable among individuals of the group (see for instance the *Code* variable in Tab.3).

sessionID	OS	Code	Size
1	88.121.0.121	"OK" (12), "Not Modified" (2)	[547,23781]
2	83.123.93.164	"OK" (6)	[5240,9320]
	...	...	

Table 3. Symbolic Data Table.

As a summary of the approach, all data necessary to build symbolic objects are assumed to be stored in the database. The user writes an SQL query in order to gather in a unique relational structure all the necessary information to build SO's even if information is scattered among different tables.

2.2 Generalization operator

$\Omega = \{1, \ldots, n\}$ is defined as the set of individuals on which SO's are built. The properties of each individual are characterized by p classical single-value variables $Y_1, \ldots, Y_p$. So, each $i \in \Omega$ corresponds to one vector whose description is $(Y_1(i), \ldots, Y_p(i))$. We associate with Ω a partitioning structure into K classes: $C_1, \ldots, C_K$.

We define a *generalization operator* g in a formal way, thereby using a coordinate-wise construction. This operator provides a description for each class C_k in the form of a new symbolic object. Let us define $S = \{s_1, \ldots, s_K\}$ the set of symbolic objects where each s_k is the result of generalization applied on C_k.

Generalization operator g, based on $Y_1, \ldots, Y_p$ is defined as $g = (g_1, ..., g_p)$ with coordinate-wise generalization operators which express common properties of variable values in class C_k:

$$d_k = g(C_k) \text{ where } d_k = (d_{k1}, \ldots, d_{k_p}) \text{ with } d_{kj} = g_j(C_k)$$

We define g_j as a union operator such as:

$$d_{kj} = \begin{cases} [\min_{i \in C_k}(Y_j(i)),\ \max_{i \in C_k}(Y_j(i))] & Y_j \text{ quantitative} \\ \{v \in Y_j \mid \exists i, i' \in C_k, Y_j(i) \leq v \leq Y_j(i')\} & Y_j \text{ ordinal} \\ \{v \in Y_j \mid \exists i \in C_k, Y_j(i) = v\} & Y_j \text{ nominal} \end{cases}$$

where $\mathcal{Y}_j$ is the domain of variable Y_j.

An important property of the generalization operator is the *distributive property*: $g(C_k \cup C_h)$ depends only on $g(C_k)$ and $g(C_h)$, i.e. the generalized value of the union of two disjoint subsets of individuals can be computed from the generalized values of the two subsets.

3 New generalization operator and symbolic object structure

In this section, we describe two extensions of SO structure and generalization operators which are suggested by recent work on data streams. These extensions are not specific to data streams and can be applied as well to the standard construction of symbolic objects described in the previous section.

3.1 New generalization operator

The first extension is a new generalization operator which applies to nominal variables. For each class this operator only selects the values with a frequency within the class above a fixed threshold defined a priori, instead of keeping all values appearing among the individuals belonging to the class. Recalling the term used in the data stream community (see Cormode and Muthukrishnan (2004)) this operator will be called the *Heavy Hitters (HH)* generalization operator. It is defined formally below.

For each $v \in Y_j$, let $f_{v,k}$ be the frequency of value v in class C_k:

$$f_{v,k} = Card(\{i \in C_k \mid Y_j(i) = v\})/Card(C_k)$$

Let ϕ be a positive real ($\phi \in]0,1]$) then the Heavy Hitters HH_{kj} of class k on variable Y_j are defined as follows:

$$HH_{kj} = \{v \in Y_j \mid f_{v,k} \geq \phi\}$$

HH_{kj} is a set of values from the domain of Y_j. It can be shown easily that its cardinality is less than $1/\phi$. So this new operator has the property to limit the number of values appearing in a boolean or modal SO, even if the underlying data set of individuals is very large.

HH_{kj} can be easily computed if the descriptions of all individuals are available in a file or in a database: the computation has to be done for each of the p variables. Pointed as the *Hierarchical Heavy Hitters problem* in Cormode et al. (2003), it is not possible to compute exactly Heavy Hitters of the union of two disjoint subsets of individuals from the Heavy Hitters of the two subsets (this generalization operator is not distributive). However an approximate solution can be given by using a *Count-Min Sketch* with parameters (ϵ, δ) (see Cormode and Muthukrishnan (2004)). A Count-Min Sketch with parameters (ϵ, δ) is represented by a two-dimensional array of counts C with $w = \lceil \frac{e}{\epsilon} \rceil$ columns and $d = \lceil ln(\frac{1}{\delta}) \rceil$ rows with d hash functions chosen uniformly at random from a pairwise-independent family. The Count-Min Sketch data structure gives an estimation of $f_{v,k}$ by $\hat{f}_{v,k} = min_{m=1,\dots,d} C[m, h_m(v)]/Card(C_k)$ with the following properties:

$$f_{v,k} \leq \hat{f}_{v,k}$$

$$P[\hat{f}_{v,k} \leq f_{v,k} + \epsilon] \geq 1 - \delta$$

The estimate $\hat{f}_{v,k}$ verifies $f_{v,k} \leq \hat{f}_{v,k} \leq f_{v,k} + \epsilon$ with a guarantee probability larger than $1 - \delta$.

Let HH^u_{kj} and HH^s_{kj} the Heavy Hitters of two disjoint subsets u and s of individuals. Then approximate Heavy Hitters $HH^{u \cup s}_{kj}$ of the union of subsets u and s can be constructed using the following three rules :

For each $v \in Y_j$ we have:

- 1 - if $v \in HH^u_{kj}$ and $v \in HH^s_{kj}$ then $v \in HH^{u \cup s}_{kj}$ and the frequency of v in $u \cup s$ is obtained by combining the frequencies in u and s.
- 2 - if $v \notin HH^u_{kj}$ and $v \notin HH^s_{kj}$ then $v \notin HH^{u \cup s}_{kj}$
- 3 - otherwise the missing frequency is replaced by its estimate: $v \in HH^{u \cup s}_{kj}$ if its estimated frequency is higher than ϕ.

This new generalization operator can thus be used for the construction of symbolic objects from databases, either for building boolean or modal symbolic descriptions.

3.2 New symbolic object structure

A new symbolic object structure is suggested by work describing two clustering algorithms: BIRCH (see Zhang et al. (1996)) which is an algorithm for clustering very large files, and CLUSTREAM (see Aggarwal et al. (2003)) which is an algorithm for clustering stream data. Both algorithms use a two-step approach. The first step is to build a summary of the file (resp. the stream) in the form of a large number of micro-clusters (typically 1000 to 10000) which are updated by an on-line algorithm. The second step is to perform a clustering using the micro-clusters instead of the original data.

The central structure to describe and maintain micro-clusters is the Cluster Feature Vector (CFV) first introduced by Zhang et al. (1996). A CFV is a description of a set of individuals described by numerical variables: it includes the cardinality of the set and for each variable the sum of values and the sum of squares of values within the set. More formally, if C is a subset of Ω, we have:

$$CFV(C) = (n, CF1(Y_1), CF2(Y_1), ..., CF1(Y_p), CF2(Y_p))$$

where $CF1(Y_j) = \sum_{i \in C} Y_j(i)$ and $CF2(Y_j) = \sum_{i \in C} Y_j^2(i)$.

CFV's have interesting properties: (1) they support easily union operations (by just adding up field by field the CFV values), (2) they represent the set as independent Gaussian distributions.

We propose to extend the CFV structure to represent a multivariate Gaussian distribution. This can be done by adding, for each couple of variables $(Y_j, Y_{j'})$ $(j < j')$, the following sum to the CFV structure: $\sum_{i \in C} Y_j(i) * Y_{j'}(i)$. This extended CFV structure also supports easily union operations.

This suggests to add a new generalization operator for building symbolic object from individuals described by quantitative variables. First d_{kj} is defined as $[m_j(C_k), v_j(C_k)]$ where $m_j(C_k) = \sum_{i \in C_k} Y_j(i)/Card(C_k)$ and $v_j(C_k) = \sum_{i \in C_k} (Y_j(i) - m_j(C_k))^2/Card(C_k)$. Secondly d_k is defined as $[M_k, V_k]$ where M_k is a mean vector $M_k = \sum_{i \in C_k} Y(i)/Card(C_k)$ and V_k is a covariance matrix.

This enables to build a set of symbolic objects modeled by multivariate Gaussian distributions. It is easy to show that this generalization operator is distributive. We will see in the next section that the property of supporting union operations is important in the context of data streams.

4 Building symbolic objects from data streams

Data streams are often referred as infinite sequences of time ordered data, structured in the form of tuples which arrive in a continuous way. We assume here that the result of the query used in DB2SO to build symbolic objects from a database (see Section 2) is only available in the form of a stream, thus not stored nor storable in a database. If the expected query is not directly available in a stream, a DSMS (Data Stream Management System) can be used to transform available streams into new streams, applying SQL-like queries possibly joining raw stream data to some static standard tables. See Golab and Oszu (2003) for more information on DSMS's.

4.1 Incremental computation of generalization operators

Since stream data cannot be stored in a database, the generalization process to build symbolic objects must be done 'on-the-fly', i.e. incrementally. It can be easily proved that all generalization operators described in Section 2 and Section 3 can be computed incrementally, due to the fact that the result of a generalization operator on a set can be computed from the result of the same operator applied to any two disjoint subsets forming a partition of the set. In some cases, some intermediate results need to be maintained instead of the result of the generalization operators, as for instance the sum of values and number of values must be maintained to compute incrementally the mean of values.

Incremental computation of generalization operators can be done either whenever a new tuple appears in the stream or at some refreshment points where all tuples between two refreshment points are processed together. The definition of refreshment points is related to the definition of windows on a data stream. A window is a portion of the stream either expressed by a number of tuples (for instance 1000 tuples) or by a time period (for instance one hour). Windows defined by a number of tuples are called *logical* windows while those defined by a time period are called *physical* windows.

Another characteristic of incremental computation of generalization operators is the amount of memory needed to compute them. For instance, computing the minimum and maximum values for building an interval symbolic variable requires only the storage of these two values, whatever the number of tuples read from the stream. On the contrary, building a boolean or modal symbolic variable requires the storage of all possible values appearing in the stream for the attribute: this may become very large and intractable. That is the reason why we suggested in Section 3.1 to extend existing generalization operators to build boolean and modal objects based on Heavy Hitters (HH) instead of on all values appearing in the stream. We have shown in Section 3.1 that the HH generalization operator is distributive if we accept an approximate result. HH's can be computed incrementally on a stream with limited fixed storage (the size is smaller than $[1/\phi]$) and in an approximated way: the precision of the approximation increases with the available storage.

4.2 Building symbolic objects from a sliding window

In Section 4.1, it has been shown how to build symbolic objects incrementally from a data stream. At each refreshment point current symbolic objects reflect the contents of the stream since its beginning. As mentioned in the introduction, the distribution of data may evolve with time introducing concept drift. A simple solution to solve this problem is to build and maintain symbolic objects over a sliding window on the stream, for instance the contents of the stream over the last 24 hours.

Sliding windows on data streams can be either defined in a *logical* way (for example the last 1000 tuples of the stream) or in a *physical* way (for instance the tuples produced during the last hour). Algorithms for computing the generalization operators will differ depending on the type of sliding window.

Case of a physical sliding window

In this case, the sliding window is defined by a time period F and a refreshment period f. Both F and f are expressed as durations in seconds, hours or days. For instance, if $F = \textit{24 hours}$ and $f = \textit{1 hour}$, this means that the symbolic objects should represent the contents of the stream during the last 24 hours and be updated every hour. For simplicity, we assume that F is a multiple of f, i.e. the sliding window can be exactly divided into F/f slices of time duration f.

A simple algorithm for maintaining symbolic objects over such a sliding window is to maintain $(F/f)+1$ results of generalization operators covering $(F/f)+1$ consecutive sliding periods of length f. Within each time period of length f, the generalization operators are computed incrementally to avoid the storage of the tuples appearing during the period. Then, at every refreshment point, the oldest result is deleted and a new one is created. Thanks to the distributive property of generalization operator computation, symbolic objects of the complete sliding window of size F can be computed from the last complete F/f results. This is illustrated in Fig.1 with $F = \textit{5 hours}$ and $f = \textit{1 hour}$.

As for the new generalization operator introduced in Section 3.1 computing Heavy Hitters for a variable, it would be interesting to study an extension of the algorithm to assess the quality of the approximation when F/f partial results are aggregated to compute Heavy Hitters on the whole sliding window.

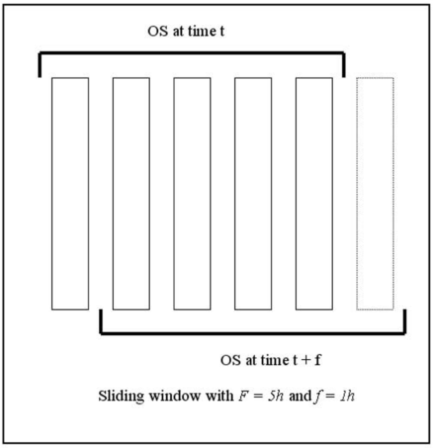


Fig. 1. Maintaining SO over a physical sliding window.

Case of a logical sliding window

In this case, the sliding window is defined by a number l of tuples and a refreshment number m of tuples. For simplicity we assume as before that l is a multiple of m. The sliding window is being defined by the l last tuples and the result of generalization operators are produced every m tuples.

As before, a simple algorithm is to maintain $(l/m)+1$ results representing m tuples each and on which the generalization operators have been applied incrementally. Then, every m tuples, it is easy to produce the result of the generalization operators for the sliding window as the union of the already computed partial results.

If the stream produces tuples at a very high rate, we may need to choose a very large value for l: the number of partial results will become as well very large since its size is also in $O(l)$. The consequence is that the proposed algorithm may become very inefficient. A good solution is to apply the algorithms developed in Datar et al. (2002) and Babcock et al. (2003) which compute mean, variance, k-median on logical sliding windows of size l using only $O(log(l))$ space. Another approach would be to apply the generalization operators to a sample maintained incrementally on the sliding window, using the algorithms described in Babcock, Datar and Motwani (2002).

More generally, it would be interesting to study how to compute all generalization operators on a logical sliding window of size l using storage less than $O(l)$.

5 Conclusion and perspectives

Research on data stream management and mining has been very active these last 5 years. Recent results first suggest two extensions to symbolic data analysis:

- a new generalization operator computing Heavy Hitters of a nominal variable,
- a new symbolic object structure describing a multivariate Gaussian distribution when all variables are numeric.

Since more and more data will be either only available as data streams or too voluminous to be stored in databases before being analyzed, it is necessary to extend the standard generation of symbolic objects from databases to data streams. We have shown that the generalization operators used to build symbolic objects can be applied to streams because they can be computed incrementally (distributivity property). Symbolic objects are built either on the whole stream since the beginning of its observation or on a sliding window representing the recent past of the stream in order to capture drift in data distributions.

This paper is just an overview of possible extensions of symbolic data analysis to data available as streams. This constitutes only a preliminary study on this subject and much work remains to be done, for instance:

- study carefully and formally all suggestions made in this paper to extend the construction of symbolic objects to data available as streams,
- study if it is possible to transpose the concept of *tilted time windows* (proposed in Aggarwal et al. (2003)) to allow the construction of symbolic objects from any portion of the past of the stream with limited storage.

References

AGGARWAL, C.C., HAN, J., WANG, J. and YU, P.S. (2003): A framework for clustering evolving data streams. In: *Proceedings of VLDB 2003*, Berlin.

ARNOUX, M., LECHEVALLIER, Y., TANASA, D., TROUSSE, B. and VERDE, R. (2003): Automatic clustering for the web usage mining. In: *Proceedings of SYNASC03*, Timisoara, 54–66.

BABCOCK, B., BABU, S., DATAR, M., MOTWANI, R. and WIDOM, J. (2002): Models and issues in data stream systems. In: *Proceedings of PODS2002*.

BABCOCK, B., DATAR, M. and MOTWANI, R. (2002): Sampling from a moving window over streaming data. In: *Proceedings of 13th Annual ACM-SIAM Symposium on Discrete Algorithms.*

BABCOCK, B., DATAR, M., MOTWANI, R. and O'CALLAGHAN, L. (2003) Maintaining variance and k-medians over data stream windows. In: *Proceedings of the 22nd Symposium on Principles of Database Systems.*

BOCK, H.-H., DIDAY, E. (2000): *Analysis of Symbolic Data, Exploratory methods for extracting statistical information from complex data.* Studies in Classification, Data Analysis and Knowledge Organisation, Springer-Verlag, Berlin.

CHARIKAR, M., CHEN, K. and FARACH-COLTON, M. (2002): Finding frequent items in data streams. In: *Proceedings of ICALP*, 693–703.

CORMODE, G., KORN, F., MUTHUKRISHNAN, S. and SRIVASTAVA, D. (2003): Finding hierarchical heavy hitters in data streams. In: *International Conference on Very Large Databases*, 464-475.

CORMODE, G. and MUTHUKRISHNAN, S. (2004): An improved data stream summary: the count-min sketch and its applications. In: *Proceedings of Latin American Theoretical Informatics*, 29-38.

CSERNEL, B., CLEROT, F., HEBRAIL, G. (2006): StreamSamp: DataStream clustering over tilted windows through sampling. In: *ECML PKDD 2006 Workshop on Knowledge Discovery from Data Streams.*

DATAR, M., GIONIS, A., INDYK, P. and MOTVANI, R. (2002): Maintaining stream statistics over sliding windows. In: *Proceedings of 13th Annual ACM-SIAM Symposium on Discrete Algorithms.*

DIDAY, E. (1988): The symbolic approach in clustering and related methods of data analysis : the basic choices. In: H.-H. Bock (Ed.), *Classification and Related Methods of Data Analysis, Proc. of IFCS'87, Aachen, July 1987.* North Holland, Amsterdam, 673–684.

GAROFALAKIS, M., GEHRKE, J. and RASTOGI, R. (2002): *Querying and Mining Data Streams: You only get one look. A tutorial.* Tutorial SIGMOD'02, June 2002.

GOLAB, L. and OZSU, M.T. (2003): Issues in data stream management. *SIGMOD Record, 32, 2.*

MOBASHER, B. (2000): Mining web usage data for automatic site personalization. In: *Proceedings of the 24th Annual Conference of the Gesellschaft für Klassifikation*, University of Passau, 15–17.

STEPHAN, V., HEBRAIL, G., LECHEVALLIER, Y. (1999): Generation of symbolic objects from relational databases. In: H.-H. Bock and E. Diday (Eds.): *Analysis of Symbolic Data, Exploratory methods for extracting statistical information from complex data.* Springer-Verlag, Berlin, 78–105.

TANASA, D. and TROUSSE, B. (2004): Advanced data preprocessing for intersites web usage mining. *IEEE Intelligent Systems, 19, 2, 59–65.*

VITTER, J.S. (1985): Random sampling with a reservoir. *ACM Transactions on Mathematical Software, 11, 37–570.*

ZHANG, T., RAMAKRISHNAN, R., LIVNY, M. (1996): BIRCH: An efficient data clustering method for very large databases. *ACM SIGMOD Conference.*

Feature Clustering Method to Detect Monotonic Chain Structures in Symbolic Data

Manabu Ichino

Department of Information and Arts, Tokyo Denki University
Hatoyama, Saitama 350-0394, Japan, *ichino@ia.dendai.ac.jp*

Abstract. Finding a linear structure in multidimensional data is a main purpose of the principal component analysis (PCA). This paper describes a feature clustering method to detect *monotonic chain structures* embedded in symbolic data tables based on the *Cartesian system model* (*CSM*) which is a mathematical model to manipulate symbolic objects.

1 Introduction

Symbolic data analysis (SDA) is a new direction to generalize standard statistical methods (Bock and Diday(2000). For example, the generalization of classical PCA is an interesting research theme. A main purpose in classical PCA is to find a linear structure in multidimensional data. In this paper, we present a feature clustering method to find *monotonic chain structures* embedded in symbolic data tables. We assume a finite set U of objects described in the *Cartesian system model* (*CSM*) which is a mathematical model to manipulate symbolic objects (Ichino and Yaguchi (1994,1998)). We briefly describe the *CSM* in Section 2. In Section 3, we define relative neighborhood sets for each object in U under a selected set of features. Then, based on the relative neighborhood sets, we present a formulation and interpretation of chain connected covering for the set U. As special classes of chain connected covering, we study monotonic chain structures in the relation to the nested coverings of U. In Section 4, we define the similarity between features as the average *Marczewski-Steinhaus* distances for relative neighborhood sets. A simple measure is also defined to evaluate monotonicity of chain connected structures. Section 5 describes a feature clustering method to detect monotonic chain structures in symbolic data based on our similarity and monotonicity measures. We illustrate our approach based on the Fats and Oils data (Ichino and Yaguchi (1994)). Section 6 is a summary.

2 Cartesian system model

Let U be a finite set of K objects as:

$$U = \{\omega_1, \omega_2, ..., \omega_K\}. \tag{1}$$

Let each of K objects be described by d features (attributes). Let D_i be the domain of feature F_i, $i = 1, 2, ..., d$. Then, the feature space is defined by the product set

$$\mathbf{D}^{(d)} = D^1 \times D^2 \times \cdots \times D^d \tag{2}$$

Since we permit the simultaneous use of various feature types, we use the notation $\mathbf{D}^{(d)}$ for the feature space in order to distinguish it from usual d-dimensional Euclidean space $\mathbf{D}^d$.
Each object ω_i in the set U is represented in the feature space $\mathbf{D}^{(d)}$ as:

$$\mathbf{E}_i = E_{i1} \times E_{i2} \cdots \times E_{id} \; or \; \mathbf{E}_i = (E_{i1}, E_{i2}, ..., E_{id}), \tag{3}$$

where $E_{ij}, j = 1, 2, ..., d$, are feature values taken by d features.

The CSM is able to manipulate the following feature types.
1) Continuous quantitative feature: The height and the weight for a person are examples of this feature type.
2) Discrete quantitative feature: The number of cities in a state and the number of family members of a person are examples of this feature type.
3) Ordinal qualitative feature: One's academic background {junior high school, high school, college or university, graduate school} and military rank are examples of this feature type. We assume an appropriate numerical coding.
4)Nominal qualitative feature: The distinction of sex {male, female} and blood type of a person $\{A, B, AB, O\}$ are examples of this feature type.
We permit interval values for feature types 1) - 3), and finite set values for feature type 4). (In the CSM, some tree-type features are also manageable (Ichino and Yaguchi, (1994, 1998)). The Cartesian product of the form (3) described in terms feature types 1) - 4) is called an *event.*

The Cartesian join, $\mathbf{A} \sqcup \mathbf{B}$, of a pair of events $\mathbf{A} = (A_1, A_2, ..., A_d)$ and $\mathbf{B} = (B_1, B_2, ..., B_d)$ in the feature space $\mathbf{D}^{(d)}$, is defined by

$$\mathbf{A} \sqcup \mathbf{B} = [A_1 \sqcup B_1] \times [A_2 \sqcup B_2] \times \cdots \times [A_d \sqcup B_d] \tag{4}$$

where $[A_i \sqcup B_i]$ is the Cartesian join of feature values A_i and B_i for feature F_i and is defined as follows.
When F_i is a quantitative or an ordinal qualitative feature, $A_i \sqcup B_i$ is a closed interval given by

$$[A_i \sqcup B_i] = [min(A_{iL}, B_{iL}), max(A_{iU}, B_{iU})], \tag{5}$$

where A_{iL} and A_{iU}, respectively, are the minimum and the maximum values of the interval A_i, and $min(A_{iL}, B_{iL})$ and $max(A_{iU}, B_{iU})$ are the operators which take the minimum and the maximum values, respectively, among sets $\{A_{iL}, B_{iL}\}$ and $\{A_{iU}, B_{iU}\}$.
When F_i is a nominal feature, $[A_i \sqcup B_i]$ is the union:

$$[A_i \sqcup B_i] = A_i \cup B_i. \tag{6}$$

The Cartesian meet, $\mathbf{A} \sqcap \mathbf{B}$, of a pair of events $\mathbf{A} = (A_1, A_2, ..., A_d)$ and $\mathbf{B} = (B_1, B_2, ..., B_d)$ in the feature space $\mathbf{D}^{(d)}$, is defined by

$$\mathbf{A} \sqcap \mathbf{B} = [A_1 \sqcap B_1] \times [A_2 \sqcap B_2] \times \cdots \times [A_d \sqcap B_d] \tag{7}$$

where $[A_i \sqcap B_i]$ is the Cartesian meet of feature values A_i and B_i for feature F_i defined by

$$[A_i \sqcap B_i] = A_i \cap B_i \tag{8}$$

When the intersection (8) takes the empty value ϕ, for at least one feature, the events $\mathbf{A}$ and $\mathbf{B}$ have no common part. We denote this fact by

$$\mathbf{A} \sqcap \mathbf{B} = \varPhi \tag{9}$$

and we say that $\mathbf{A}$ and $\mathbf{B}$ are completely distinguishable.
We call the triplet $(\mathbf{D}^{(d)}, \sqcup, \sqcap)$ the *Cartesian system model* (*CSM*) (Ichino and Yaguchi (1994, 1998)).

3 Relative neighborhood and neighborhood set

In the following discussion, we treat various subsets of the given set of features. To clarify this, let F_0 be the set of feature numbers given by

$$F_0 = \{1, 2, ..., d\}, \tag{10}$$

and be called the feature set. For a feature subset $F = \{p_1, p_2, ..., p_m\}$ of F_0, an object ω_k in the set $U = \{\omega_1, \omega_2, ..., \omega_K\}$ may be given as follows:

$$\mathbf{E}_k = E_{kp1} \times E_{kp2} \times \cdots \times E_{kpm} \; or \; \mathbf{E}_k = (E_{kp1}, E_{kp2}, ..., E_{kpm}). \tag{11}$$

Definition 1 Join region
For a pair of objects $\omega_p, \omega_q \epsilon U$, let $J(\omega_p, \omega_q|F)$ be the Cartesian join region in the feature space spanned by a feature subset F of F_0 i.e.,

$$J(\omega_p, \omega_q|F) = \prod_{r \epsilon F} [E_{pr} \sqcup E_{qr}], \tag{12}$$

where $\prod$ is the operator for the Cartesian product and square brackets [and] mean here that the boundary values of the Cartesian join for feature F_r are included in the join region (i.e., a closed region).

Definition 2 Relative neighborhood
Two objects $\omega_p, \omega_q \epsilon U$ are called the *relative neighbors* under a feature subset F of F_0, if the following condition is satisfied:

$$J(\omega_p, \omega_q|F) \sqcap E_k \neq E_k \; for \; all \; k \; \neq p, q \tag{13}$$

Table 1 shows eight objects under five features $\{F_1, F_2, ..., F_5\}$. In this table, object pairs $(1, 2)$, $(3, 4)$, $(5, 6)$, and $(7, 8)$ are relative neighbors under F_1, F_3, F_5, and F_2, respectively.

A neighborhood set of an object $\omega \epsilon U$ under a feature subset F, denoted by $n(\omega|F)$, is a non-empty subset of U. The operator $n(\cdot|F)$ is a mapping $n : U \rightarrow 2^U$, where 2^U denotes the power set of U.

Definition 3 Neighborhood set

For each $\omega \epsilon U$, let $n(\omega|F)$ be defined by the set of all neighbors of ω. We assume that each $n(\omega|F)$ includes ω as a neighborhood.

Example 1

In Table 1, $n(1|F_3) = \{1, 2, 4\}$, $n(2|F_3) = \{1, 2\}$, $n(3|F_3) = \{3, 4, 5\}$, $n(4|F_3) = \{1, 3, 4\}$, $n(5|F_3) = \{3, 5, 6, 8\}$, $n(6|F_3) = \{5, 6\}$, $n(7|F_3) = \{7, 8\}$, $n(8|F_3) = \{5, 7, 8\}$ are neighborhood sets.

4 Chain, chain connected covering, and monotonic chain

We define several notions concerning chains.

Definition 4

Two objects $\omega_p, \omega_q \epsilon U$ are called chain connected (or simply connected) under F, if

$$\omega_p, \omega_q \epsilon\ n(\omega_p|F) \cap n(\omega_q|F). \tag{14}$$

Definition 5 Chain

A series of objects $\omega_{p1}, \omega_{p2}, ..., \omega_{pm}$, is called a chain under F if the following conditions are satisfied:

$$\omega_{pk}, \omega_{p(k+1)} \epsilon\ n(\omega_{pk}|F) \cap n(\omega_{p(k+1)}|F), k = 1, 2, ..., m-1, \tag{15}$$

where ω_{p1} and ω_{pm} are called the terminal points, and m is the length of the chain.

Definition 6 Chain connected covering (CCC)

A chain $\omega_{p1}, \omega_{p2}, ..., \omega_{pm}$ is called a chain connected covering (CCC) of U under F if

$$U \subseteq \bigcup_{k=1}^{m} n(\omega_{pk}|F). \tag{16}$$

Example 2

In Example 1, the series of $2, 1, 4, 3, 5, 8, 7$ becomes a CCC of the set U of eight objects under feature F_3. In fact, from Example 1, we have $2, 1 \epsilon\ n(2|F_3) \cap n(1|F_3) = \{1, 2\}$, $1, 4 \epsilon\ n(1|F_3) \cap n(4|F_3) = \{1, 4\}$, $4, 3 \epsilon\ n(4|F_3) \cap$

$n(3|F_3) = \{3,4\}$, $3,5\epsilon\ n(3|F_3) \cap n(5|F_3) = \{3,5\}$, $5,8\epsilon\ n(5|F_3) \cap n(8|F_3) = \{5,8\}$, $8,7\epsilon\ n(8|F_3) \cap n(7|F_3) = \{7,8\}$, and $U \subseteq n(2|F_3) \cup n(1|F_3) \cup n(4|F_3) \cup n(3|F_3) \cup n(5|F_3) \cup n(8|F_3) \cup n(7|F_3) = \{1,2,3,4,5,6,7,8\}$. Therefore, this chain is a CCC of U.

Example 3

For a nominal feature F, we are able to illustrate a CCC. Suppose the following five objects composed of nominal values $a_k, k = 1,2,...,7$: $\omega_1 = \{a_1,a_2,a_3\}$, $\omega_2 = \{a_2,a_3,a_4\}$, $\omega_3 = \{a_3,a_4,a_5\}$,$\omega_4 = \{a_4,a_5,a_6\}$, and $\omega_5 = \{a_5,a_6,a_7\}$. The neighborhood sets become: $n(\omega_1|F) = \{\omega_1,\omega_2,\omega_5\}$, $n(\omega_2|F) = \{\omega_1,\omega_2,\omega_3\}$, $n(\omega_3|F) = \{\omega_2,\omega_3,\omega_4\}$, $n(\omega_4|F) = \{\omega_3,\omega_4,\omega_5\}$, $n(\omega_5|F) = \{\omega_1,\omega_4,\omega_5\}$. Then, we see that $\omega_k, \omega_{k+1}\epsilon\ n(\omega_k|F) \cap n(\omega_{k+1}|F), k = 1,2,...,5$, and $U \subseteq n(\omega_1|F) \cup n(\omega_2|F) \cup \cdots \cup n(\omega_5|F) = \{\omega_1,\omega_2,\omega_3,\omega_4,\omega_5\}$. Therefore these five objects yield a CCC under F, where ω_1 and ω_5 are terminal points.

	Specific gravity F_1	Freezing point F_2	Iodine value F_3	Saponification F_4	Major acids F_5
1.Linseed	$0.930 \sim 0.935$	$-27 \sim -18$	$170 \sim 204$	$118 \sim 196$	L,Ln,O,P,M
2.Perilla	$0.930 \sim 0.937$	$-5 \sim -4$	$192 \sim 208$	$188 \sim 197$	L,Ln,O,P,S
3.Cotton	$0.916 \sim 0.918$	$-6 \sim -1$	$99 \sim 113$	$189 \sim 198$	L,O,P,M,S
4.Sesame	$0.920 \sim 0.926$	$-6 \sim -4$	$104 \sim 116$	$187 \sim 193$	L,O,P,S,A
5.Camellia	$0.916 \sim 0.917$	$-21 \sim -15$	$80 \sim 82$	$189 \sim 193$	L,O
6.Olive	$0.914 \sim 0.919$	$0 \sim 6$	$79 \sim 90$	$187 \sim 196$	L,O,P,S
7.Beef	$0.860 \sim 0.870$	$30 \sim 38$	$40 \sim 48$	$190 \sim 199$	O,P,M,S,C
8.Hog	$0.858 \sim 0.864$	$22 \sim 32$	$53 \sim 77$	$190 \sim 202$	L,O,P,M,S,Lu

Table 1. Fats and Oils data.

L: Linoleic acid, Ln: Linolenic acid, O: Oleic acid, P: Palmitic acid, M: Myristic acid, S: Searic acid, A: Arachic acid, C: Capric acid, Lu: Lauric acid

Definition 7 Monotonic chain

A chain $\omega_{p1}, \omega_{p2}, ..., \omega_{pm}$ is called a monotonic chain under F, if the chain satisfies the nesting property:

$$J(\omega_{p1}, \omega_k|F) \subseteq J(\omega_{p1}, \omega_{k+1}|F), k = 1,2,...,m-1. \tag{17}$$

Example 4

Suppose the following five objects composed of nominal values $a_k, k = 1,2,...,8$: $\omega_1 = \{a_1,a_2,a_3,a_4\}$, $\omega_2 = \{a_2,a_3,a_4,a_5\}$, $\omega_3 = \{a_3,a_4,a_5,a_6\}$, $\omega_4 = \{a_4,a_5,a_6,a_7\}$, and $\omega_5 = \{a_5,a_6,a_7,a_8\}$. The neighborhood sets become: $n(\omega_1|F) = \{\omega_1,\omega_2\}$, $n(\omega_2|F) = \{\omega_1,\omega_2,\omega_3\}$, $n(\omega_3|F) = \{\omega_2,\omega_3,\omega_4\}$, $n(\omega_4|F) = \{\omega_3,\omega_4,\omega_5\}$, $n(\omega_5|F) = \{\omega_4,\omega_5\}$. Then, we see that ω_k, ω_{k+1} $\epsilon\ n(\omega_k|F) \cap n(\omega_{k+1}|F), k = 1,2,3,4,$ $U \subseteq n(\omega_1|F) \cup n(\omega|F) \cup \cdots \cup n(\omega_5|F) =$

$\{\omega_1, \omega_2, \omega_3, \omega_4, \omega_5\}$, and $J(\omega_{p1}, \omega_k|F) \subseteq J(\omega_{p1}, \omega_{k+1}|F), k = 1, 2, 3, 4$. Therefore, these five objects compose a *CCC* and a *monotonic chain* under F.

5 Similarity measure and feature clustering

Let $U = \{\omega_1, \omega_2, ..., \omega_K\}$ be the set of objects described by the feature set F_0. For an object $\omega_k \epsilon U$, let $n(\omega_k|F_1)$ and $n(\omega_k|F_2)$ be neighborhood sets for feature subsets $F_1 \subseteq F_0$ and $F_2 \subseteq F_0$, respectively. Then, the similarity between F_1 and F_2 with respect to object ω_k is defined by

$$S(F_1, F_2|\omega_k) = |n(\omega_k|F_1) \cap n(\omega_k|F_2)|/|n(\omega_k|F_1) \cup n(\omega_k|F_2)|, \tag{18}$$

where $| * |$ denotes the cardinality of a set $*$, and $1 - S(F_1, F_2|\omega_k)$ is called the *Marczewski-Steinhaus metric* between two neighborhood sets [4]. Then, we define the similarity between two feature subsets F_1 and F_2 over the set of objects U as follows.

Definition 8 Similarity measure
The similarity between feature subsets F_1 and F_2 is defined by

$$S(F_1, F_2|U) = \frac{1}{K} \sum_{k=1}^{K} |n(\omega_k|F_1) \cap n(\omega_k|F_2)|/|n(\omega_k|F_1 \cup n(\omega_k|F_2)|). \tag{19}$$

This similarity measure satisfies the inequality:

$$1/K \leq S(F_1, F_2|U) \leq 1. \tag{20}$$

Suppose that objects in U compose a monotonic chain $\omega_1, \omega_2, \omega_3, ..., \omega_K$ under a feature set F. Then, two terminal points ω_1 and ω_K have two relative neighbors, and other objects, $\omega_k, k = 2, 3, ..., K-2$, have three relative neighbors, respectively. Therefore, as the total, K objects have $2 \times 2 + 3 \times (K-2) = 3K - 2$ relative neighbors. Based on this fact, we define a monotonicity measure for a set of objects U under F as follows.

Definition 9 Monotonicity measure

$$M(U|F) = \frac{1}{3K-2} \sum_{k=1}^{K} |n(\omega_k|F)|, \tag{21}$$

where $| * |$ denotes the cardinality of a set $*$. This measure satisfies the inequality:

$$1 \leq M(U|F) \leq K^2/(3K-2). \tag{22}$$

The minimum value is achieved when all objects in U compose a complete monotonic chain, while the maximum value is achieved when all objects in

	F_1	F_2	F_3	F_4	F_5
Specific gravity (F_1)	1	0.254	0.696	0.542	0.318
Freezing point (F_2)	0.254	1	0.352	0.287	0.343
Iodine value (F_3)	0.696	0.352	1	0.419	0.352
Saponification (F_4)	0.542	0.287	0.419	1	0.458
Major acids (F_5)	0.318	0.343	0.352	0.458	1

Table 2. Similarity matrix.

U have the same K relative neighbors.

Example 5 Feature clustering of the Fats and Oils data

Table 2 shows the similarity matrix based on our similarity measure.

In this table, $S(F_1, F_3|U) = 0.696$ is the maximum. We can easily verify that we have a CCC of U with length 7 under features F_1 and F_3, and the monotonicity is $M(U|F_1, F_3) = 1.263$.

Then combining features F_1 and F_3, we have Table 3.

	F_1, F_3	F_2	F_4	F_5
F_1, F_3	1	0.235	0.409	0.352
F_2	0.235	1	0.287	0.343
F_4	0.409	0.287	1	0.458
F_5	0.352	0.343	0.458	1

Table 3. Reduced similarity matrix.

In this reduced similarity matrix, $S(F_4, F_5|U) = 0.458$ is the maximum. However, eight objects in U are not able to compose a CCC under these features. We see that features $\{F_1, F_3, F_4\}$ yield CCC of length 7, and their monotonicity is $M(U|F_1, F_3, F_4) = 1.684$. Similarly, features $\{F_1, F_3, F_4, F_5\}$ yield again CCC of length 7, and their monotonicity is $M(U|F_1, F_3, F_4, F_5) = 1.894$. Finally, the overall feature set $\{F_1, F_2, F_3, F_4, F_5\}$ yields CCC of length 8, and the monotonicity becomes $M(U|F_1, F_3, F_4, F_5) = 2.455$.

In the above example, we should point out the following facts:
1) Our measures of similarity and monotonicity work well for different feature types.
2) Monotonic chain structures are detectable by using our simple feature clustering method.

6 Concluding remarks

We presented notions of the chain connected covering and the monotonic chain structures. Then, we defined a similarity measure between feature sets, and a simple monotonicity measure. In order to show the effectiveness of these measures, we presented a feature clustering method based on the fats and oils data. These measures may be useful tools in the generalization of the classical PCA.

References

BOCK, H.-H. and DIDAY, E. (2000): *Analysis of Symbolic Data* , Springer-Verlag, Berlin.

ICHINO, M and YAGUCHI, H. (1994): Generalized Minkowski metrics for mixed feature-type data analysis, *IEEE Trans. on Syst. Man, Cybern., 24 (4), 698-708.*

ICHINO, M. and YAGUCHI, H. (1998): Symbolic pattern classifiers based on the Cartesian system model. In C. Hayashi, et al., (Eds), *Data Science, Classification, and Related Methods*, Springer-Verlag, Tokyo.

LIN, T.Y. and CERCONE, N. (1997): *Rough Sets and Data Mining*, Kluwer Academic Publishers.

Symbolic Markov Chains

Monique Noirhomme-Fraiture and Etienne Cuvelier

Institut d'Informatique, Université de Namur
rue Grandgagnage, 21, B-5000 Namur, Belgium
mno@info.fundp.ac.be, *ecu@info.fundp.ac.be*

Abstract. Stochastic processes have, since a long time, large applications in quite different domains. The standard theory considers discrete or continuous state space. We consider here the concept of Stochastic Process associated to all the cases of symbolic variables: quantitative, categorical single and multiple, interval, modal. More particularly, we adapt the definition of Markov Chain and give the equivalent of the Chapman-Kolmogorov theorem in all cases.

1 Introduction

Frequently, we have to consider systems which develop in time or space in accordance with probabilistic laws. The study of such systems is called the theory of Stochastic Processes. More precisely, a Stochastic Process is a random variable which depends on time or space.

The aim of this paper is to propose theoretical bases for generalisation of Stochastic Processes to Symbolic variables.

Indeed, Stochastic Processes are defined for variables for which the state space (or values) is a countable or finite set or the real line $(-\infty, \infty)$. In the first case, the process is called a Chain. Here, we want to extend this concept to variables which can be multivalued, interval or even modal.

This problem is practically meaningful. For example, let us consider the evolution of the value of stock. Usually, each day, the stock has several values: open, close, mean, maximum, minimum. The stock value can thus be characterised by an interval of values and not by a unique number.

If we consider daily audience of a TV channel, the audience for a family is given by the percentage of time spent at watching different broadcasts and not by a single category. In this case, the variable is modal.

This paper does not deal with the statistical analysis of symbolic data from Stochastic Processes. We try only to modelise the problem from a probabilistic point of view.

We will concentrate our study to a special case of Stochastic Process which is Markov Chains. We will first recall the definition and principal characteristics of Markov Chain in the case of categorical and continuous variables and we will extend them to the case of multivalued categorical, interval and modal variables.

To simplify the presentation, we will speak only about time and not space. This choice is motivated by the fact that it concerns the more frequent applications.

We have also chosen to present here only the case of discrete time. But continuous time could also be considered, in an extended paper.

Numerous books have been written about Stochastic Processes. From others, let us quote Cox and Miller (1965), Bartlett (1978), Prabhu (1965), Karlin (1966), Neveu (1964), Feller (1968), Bailey (1964) and more recently, Stierzaker (2005), Lawler (2006), Beichelt (2006), Meyn & Tweedie (1993), Girkhman and Skorokhod (2004). On the other hand, in Symbolic Data Analysis, very few has been done in Stochastic Processes. Diday et al. (2004) and De Carvalho et al. (2004) have studied linear symbolic regression. Prudencio et al. (2004) have considered time series. We can also mention the work of Soule et al. (2004) in flow classification.

2 Definitions

Let us consider $(\Omega, \mathcal{A}, Pr)$ a probability space and $\{\underline{X}_t, t \in T\}$ a Stochastic Process defined on this space, i.e. a random variable depending upon the parameter t, considered as the time.

We will consider the particular case where the time is discrete, with values represented by the positive integers. In this case, the Stochastic Process is often written $\{\underline{X}_n, n \in \mathbb{N}\}$.

The set of values of $\underline{X}_t$ is the state space. In the standard theory, it can be continuous or discrete. The study of a Stochastic Process is very complex except if we make hypothesis on the behavior of the process.

One common hypothesis is the Markovian one. A Markov process is a process with the property that, given the value $\underline{X}_t$, the values of $\underline{X}_s$, $s > t$, do not depend on the values of $\underline{X}_u$, $u < t$.

In formal terms, a process is said to be Markovian if

$$\begin{aligned} &Pr[a < \underline{X}_t \leq b \mid \underline{X}_{t_1} = x_1, \underline{X}_{t_2} = x_2, \ldots, \underline{X}_{t_n} = x_n] \\ &\qquad\qquad\qquad\qquad = Pr[a < \underline{X}_t \leq b \mid \underline{X}_{t_n} = x_n] \end{aligned}$$

whenever $t_1 < t_2 < \cdots < t_n < t$.

The function

$$Pr[\underline{X}_t \in A \mid \underline{X}_s = x] \,, \qquad t > s$$

is called the transition probability function and is basic to the study of the structure of Markov processes.

A Markov process is said to have **stationary transition probabilities** if the transition probabilities are function only of $t - s$ and not s. We say also “homogeneous in time”. A Stochastic Process $\underline{X}_t$ for t in T is said to be **stationary** if the joint distribution function of the families of random variables $(X_{t_1+h}, X_{t_2+h}, \ldots, X_{t_n+h})$ and $(X_{t_1}, X_{t_2}, \ldots, X_{t_n})$ are the same for all $h > 0$ and arbitrary selections $t_1, t_2, \dots, t_n$ of T. This property means that the particular times at which we examine the process are of no relevance. In particular, the distribution of $\underline{X}_t$ is the same for each t. Let us note that there is no reason to expect that a Markov process with stationary probabilities is a stationary process (Karlin (1966), p 204).

3 Single valued categorical variables

Let us consider the case where $\underline{X}_n$ means belonging to one category among s at time n.

We can modelise this case in writing $\underline{X}_n = k$, $1 \leq k \leq s$. The process $\{\underline{X}_n, n \in T\}$ is thus a classical Stochastic Process whose state space is the finite set $(1, 2, \dots, s)$. We will suppose that the process is Markovian, with stationary transition probabilities.

The stationary transition probabilities are defined by:

$$P_{ij}(n) = Pr[\underline{X}_{m+n} = j \mid \underline{X}_m = i] \ .$$

For such probabilities, it can be shown easily the Chapman-Kolmogorov property:

$$P_{ij}(m+n) = \sum_k P_{ik}(n)\ P_{kj}(m) \ , \qquad \forall\ i, j$$

or

$$P(m+n) = P(m)\ P(n)$$

with $P(n)$ the matrix with element $P_{ij}(n)$ and

$$P \equiv P(1) \ .$$

¿From this, we have

$$P(n) = P^n$$

which allows the computation of the matrix $P(n)$ when n is small.

With some properties on transition probabilities, it is possible to show that the Markov Chain is stationary and to compute easily $\lim_{n \to \infty} P(n)$ (Cox and Miller (1965)), (Prabhu (1965)).

4 Multivalued categorical variables

In this case, the variable $\overrightarrow{X_t}$ indicates belonging to several categories, among s $(C_1, \dots, C_s)$.

We can modelise this case in considering the multidimensional variable $\underline{\overrightarrow{X_t}}$ with state $\overrightarrow{\jmath} = (j_1, \dots, j_s)$ where

$$j_k = \begin{cases} 1 & \text{if the category } C_k \text{ is present,} \\ 0 & \text{elsewhere.} \end{cases}$$

$\overrightarrow{\underline{X}_t}$ is here a s-vector process.

Such a process is a Markov process if

$$Pr[\overrightarrow{\underline{X}_{t_{n+1}}} = \overrightarrow{a_{n+1}} \mid \overrightarrow{\underline{X}_{t_1}} = \overrightarrow{a_1}, \ldots, \overrightarrow{\underline{X}_{t_n}} = \overrightarrow{a_n}] = Pr[\overrightarrow{X}_{t_{n+1}} = \overrightarrow{a_{n+1}} \mid \overrightarrow{\underline{X}_{t_n}} = \overrightarrow{a_n}]$$
$$for\ all\ t_1 < t_2 < \cdots < t_n < t_{n+1}\ .$$

If we suppose that the transition probabilities are stationary, let us define:

$$P_{\overrightarrow{\imath}\ \overrightarrow{\jmath}}(n) = Pr[\overrightarrow{\underline{X}_{t+n}} = \overrightarrow{\jmath} \mid \underline{X}_t = \overrightarrow{\imath}]\ .$$

The Chapman-Kolmogorov property is still valid :

$$P_{\overrightarrow{\imath}\ \overrightarrow{\jmath}}(n+m) = \sum_{\overrightarrow{k}} P_{\overrightarrow{\imath}\ \overrightarrow{k}}(n)\ P_{\overrightarrow{k}\ \overrightarrow{\jmath}}(m)$$

which allows to compute $P_{\overrightarrow{\imath}\ \overrightarrow{\jmath}}(n)$ from $P_{\overrightarrow{\imath}\ \overrightarrow{\jmath}}(1)$.

5 Single quantitative variable

In this case, the state space of the Markov Chain $\underline{X}_n$ is $(-\infty, +\infty)$. As previously, we restrict to chains with stationary transition probabilities.

$$P_n(x; y) = Pr[\underline{X}_{m+n} \leq y \mid \underline{X}_m = x] \tag{1}$$

defines the n-th order transition distribution function.

In particular, let

$$P_1(x; y) \equiv P(x; y) = Pr[\underline{X}_{m+1} \leq y \mid \underline{X}_m = x]\ .$$

The Chapman-Kolmogorov equation can be written:

$$P_{m+n}(x; y) = \int_{-\infty}^{+\infty} d_z\ P[X_m \leq z \mid X_0 = x]\ Pr[X_{m+n} \leq y \mid X_m = z] \tag{2}$$

or

$$P_{m+n}(x; y) = \int_{-\infty}^{+\infty} d_z\ P_m(x; z)\ P_n(z; y)\ . \tag{3}$$

If $p_m(x; y)$ denotes the probability densities, if they exist, this relation can be written

$$P_{m+n}(x; y) = \int_{-\infty}^{y} p_{m+n}(x; u)\ du = \int_{-\infty}^{+\infty} p_m(x; z) \int_{-\infty}^{y} p_n(z; u)\ du\ dz$$

and thus, it can be proven that (Cox and Miller (1965), p 134) :

$$p_{m+n}(x; u) = \int_{-\infty}^{+\infty} p_m(x; z)\ p_n(z; u)\ dz\ . \tag{4}$$

A Markov process is specified by giving the initial distribution and transition probabilities $P(x;y)$.

The use of Kolmogorov equation gives all the other transition probabilities $P_n(x;y)$ and the state distributions.

An alternative approach is given by the use of Copulas (Nelsen (1999)). A Copula function C is a multivariate uniform distribution (a multivariate distribution with uniform margins).

It can be shown, from Sklar's theorem, that if F is a N-dimensional distribution function with continuous margins $F_1,\ldots,F_N$, then F has a unique Copula representation

$$F(x_1,\ldots,x_N) = C(F_1(x_1),\ldots,F_N(x_N)) \ .$$

The product of Copulas is defined by

$$C_1 \star C_2(u,v) = \int_0^1 \tfrac{\partial}{\partial v}\ C_1(u,z)\ \tfrac{\partial}{\partial u}\ C_2(z,v)\ dz \ .$$

Darsow et al. (1992) prove that if $\underline{X}_t$ is a Markov process and let $C_{m,n}$ denote the Copula of the random variables X_m and X_n, then the Chapman-Kolmogorov equation is equivalent to

$$C_{t,t+m+n} = C_{t,t+m} \star C_{t+m,t+m+n} \qquad (5)$$

where $\star$ denotes the product of Copulas.

With this approach, a Markov process is specified by giving all the marginal distributions and a family of 2-Copulas satisfying (5) (Joe (1997)).

6 Interval symbolic variable

Let us suppose that, at each time, the variable is known only by its belonging to an interval of the real line.

It means that we are here interested by the transition probabilities

$$Pr[a_2 \leq \underline{X}_{m+n} \leq b_2 \mid a_1 \leq \underline{X}_m \leq b_1]$$

which we will write

$$Pr[\underline{X}_{m+n} \in A_2 \mid \underline{X}_m \in A_1] = P_n(A_1;A_2) \qquad (6)$$

if A_1 and A_2 are intervals of $]-\infty,+\infty[$ and if this probability does not depend on m.

We will define an Interval Markov Chain, a chain such that

$$\begin{aligned} &Pr[\underline{X}_{t_{n+1}} \in A_{n+1} \mid \underline{X}_{t_1} \in A_1,\ldots,\underline{X}_{t_n} \in A_n] \\ &\qquad\qquad = Pr[\underline{X}_{t_{n+1}} \in A_{n+1} \mid \underline{X}_{t_n} \in A_n] \qquad (7) \end{aligned}$$

where $A_j = [a_j, b_j]$.

Let us note that we have a particular case

$$Pr[\underline{X}_{t_{n+1}} \in A_{n+1} \mid \underline{X}_{t_1} = a_1, \ldots, \underline{X}_{t_n} = a_n] = Pr[\underline{X}_{t_{n+1}} \in A_{n+1} \mid \underline{X}_{t_n} = a_n]$$

when $a_j = b_j$.

Let $A_1 = [a_1, b_1]$ and

$$P_m(A_1; z) = Pr[\underline{X}_{t+m} \leq z \mid \underline{X}_t \in A_1]$$

which is supposed not depending on t.

Then,

$$P_m(A_1; A_2) = P_m(A_1; b_2) - P_m(A_1; a_2)$$

for an interval $A_2 = [a_2, b_2]$ and continuous P_m function.

If the derivative of $P_m(A_1; z)$ exists, we will note

$$p_m(A_1; u) = \tfrac{\partial}{\partial u}\, P_m(A_1; u)\ .$$

Theorem: *For an Interval Markov Chain with stationary transition probabilities, we have the relation*

$$P_{m+n}(A_1; A_2) = \int_{-\infty}^{\infty} d_z\ P_m(A_1; z)\ P_n(z; A_2) \tag{8}$$

and, if the probability density exists,

$$p_{m+n}(A_1; u) = \int_{-\infty}^{+\infty} p_m(A_1; z)\ p_n(z; u)\ dz\ . \tag{9}$$

Proof. From conditional probability property, we know that

$$\begin{aligned} &Pr[\underline{X}_{t+m+n} \in A_2 \mid \underline{X}_t \in A_1] \\ &= \int_{-\infty}^{\infty} d_z\ Pr[\underline{X}_{t+m} \leq z \mid \underline{X}_t \in A_1]\ Pr[\underline{X}_{t+m+n} \in A_2 \mid \underline{X}_t \in A_1, \underline{X}_{t+m} = z]\ . \end{aligned}$$

Using the Markovian property (7) we have

$$\begin{aligned} &Pr[\underline{X}_{t+m+n} \in A_2 \mid \underline{X}_t \in A_1] \\ &\qquad = \int_{-\infty}^{\infty} d_z\ Pr[\underline{X}_{t+m} \leq z \mid \underline{X}_t \in A_1]\ \ \mathrm{Pr}[\underline{X}_{t+m+n} \in A_2 \mid \underline{X}_{t+m} = z] \end{aligned}$$

and using the fact that the transition probabilities do not depend on time and notations (6)

$$P_{m+n}(A_1; A_2) = \int_{-\infty}^{+\infty} d_z\ P_m(A_1; z)\ P_n(z; A_2)\ .$$

If the densities $p_m(A_1; u)$ and $p_n(z; u)$ exist, then

$$P_{m+n}(A_1; A_2) = \int_{a_2}^{b_2} p_{m+n}(A_1; u)\, du = \int_{-\infty}^{+\infty} p_m(A_1; z) \int_{a_2}^{b_2} p_n(z; u)\, du\, dz\ .$$

Thus

$$p_{m+n}(A_1; u) = \int_{-\infty}^{+\infty} p_m(A_1; z)\, p_n(z; u)\, dz\ . \qquad \blacksquare$$

Remark: It is possible to modelise an interval by two values : its center and its half-length. In this case, $\underline{X}_t$ is in fact a two dimensions continuous variable.

7 Modal variable

A Modal variable is known by the belonging probability to classes $C_1, \ldots, C_s$ (Bock and Diday (2000)).

For a Modal Stochastic Process, it means that the variable $\overrightarrow{X_t}$ is defined by $\Pi_1(t), \ldots, \Pi_s(t)$ with

$$\Pi_1(t) + \cdots + \Pi_s(t) = 1\ , \qquad 0 \leq \Pi_j(t) \leq 1\ , \quad \forall\ j\ .$$

$\overrightarrow{\underline{X}_t}$ is thus in fact a multidimensional continuous process whose value will be written $\overrightarrow{\Pi}_t$ and whose state space is the hypercube $[0,1] \times \cdots \times [0,1]$ with constraint $\sum_{j=1}^{s-1} \Pi_j \leq 1$.

The Markov hypothesis is still

$$Pr[\{\overrightarrow{\underline{X}_{t_{n+1}}} \leq \overrightarrow{\Pi}(n+1) \mid \overrightarrow{\underline{X}_{t_1}} = \overrightarrow{\Pi}(1), \overrightarrow{\underline{X}_{t_2}} = \overrightarrow{\Pi}(2), \ldots \overrightarrow{\underline{X}_{t_n}} = \overrightarrow{\Pi}(n)]$$
$$= Pr[\overrightarrow{\underline{X}_{t_{n+1}}} \leq \overrightarrow{\Pi}(n+1) \mid \overrightarrow{\underline{X}_{t_n}} = \overrightarrow{\Pi}(n)]\ .$$

If the process is homogeneous in time (has stationary transition probabilities), using a multidimensional analog of (1) and (2), we have

$$P_n(\overrightarrow{\Pi}; \overrightarrow{y}) = Pr[\overrightarrow{\underline{X}_{m+n}} \leq \overrightarrow{y} \mid \overrightarrow{\underline{X}_m} = \overrightarrow{\Pi}] \qquad \text{with} \quad y_j \leq 1$$

$$P_n(\overrightarrow{\Pi}; \overrightarrow{y}) = 0 \qquad \text{if} \quad \sum_{j=1}^{s-1} \Pi_j > 1$$

Let

$$p_n(\overrightarrow{\Pi}; \overrightarrow{y}) = \frac{d}{d\overrightarrow{y}}\ P_n(\overrightarrow{\Pi}; \overrightarrow{y})\ .$$

It can be proved that

$$p_{m+n}(\overrightarrow{\Pi}; \overrightarrow{y}) = \int p_m(\overrightarrow{\Pi}; \overrightarrow{z})\ p_n(\overrightarrow{z}; \overrightarrow{y})\ d\overrightarrow{z}$$

where the integral is an $s-1$ multiple integral on the space $[0,1] \times [0,1] \times \cdots \times [0,1]$.

Let us notice that for two categories, as $\Pi_1(t) + \Pi_2(t) = 1$, $\underline{X}(t)$ is a one-dimensional process, so that the problem is a particular case of §5 where the state space is $[0,1]$ and not $]-\infty, +\infty[$.

Conclusion

In this paper, we have defined Symbolic Markov Chain for all the cases of Symbolic variables: quantitative, categorical single and multiple, interval, modal. We have also given the equivalent of Chapman-Kolmogorov equations in all cases. This property is the bases of the theoretical study of Markov Chains. We intend to continue this work in giving the more interesting results which give the knowledge of the state probabilities in interval and modal cases.

Let us note that in the case of continuous state space, we get interesting results with continuous time. In particular, the Kolmogorov equations are then known as the Fokker-Planck diffusion equations.

Acknowledgement

Our thanks to Yves Lechevallier and Paula Brito for their useful suggestions.

References

AFONSO, F., BILLARD, L. and DIDAY, E. (2004): Régression linéaire symbolique avec variables taxonomiques. *Revue RNTI.* G. Hébrail et al. Eds., Vol. 1, 205–210, Cépadues.

BAILEY, T.J.(1964) : *The Elements of Stochastic Processes with Applications to the Natural Sciences.* Wiley, New York.

BARTLETT, M.S. (1978): *An Introduction to Stochastic Processes.* Cambridge University Press, 3rd ed., Cambridge.

BEICHELT, F. (2006): *Stochastic Processes in Science, Engineering and Finance.* Chapman & Hall.

BOCK, H.-H., DIDAY, E. (2000): *Analysis of Symbolic Data, Exploratory methods for extracting statistical information from complex data.* Studies in Classification, Data Analysis and Knowledge Organisation, Springer-Verlag, Berlin.

COX, D.R. and MILLER, H.D. (1965): *The Theory of Stochastic Processes.* Methuen & Co., London.

DARSOW, W.F., NGUYEN, B. and OLSEN, E.T. (1992): Copulas and Markov processes. *Illinois J. Math. 36, 600–642.*

De CARVALHO, F.A.T., LIMA NETO, E.A. and TENÓRIO, C.P. (2004): A new method to fit a linear regression model for interval-valued data. In: S. Biundo, T. Frühwirth and G. Palm (Eds.): *Advances in Artificial Intelligence: Proceedings of the Twenty seventh German Conference on Artificial Intelligence.* Springer-Verlag, Berlin, 295–306.

FELLER, W. (1968): *An Introduction to Probability Theory and its Applications* Wiley, New York.

GIKHMAN, I.I. and SKOROKHOD, A.V. (2004): *The Theory of Stochastic Processes.* Classics in Mathematics. Springer-Verlag, Berlin.

JOE, H. (1997): *Multivariate Models and Dependence Concepts.* Chapman & Hall, London.

KARLIN, S. (1966): *A First Course in Stochastic Processes.* Academic Press, New York.

LAWLER, G.F. (2006): *Introduction to Stochastic Processes, 2nd Ed.* Chapman & Hall.

MEYN, S.P. and TWEEDIE, R.L. (1993): *Markov Chains and Stochastic Stability.* Communications & Control. Springer-Verlag, New York.

NELSEN, R.B. (1999): An introduction to Copulas. *Lecture Notes in Statistics.* Springer, New York.

NEVEU, J. (1964): *Bases Mathématiques du Calcul des Probabilités.* Masson, Paris.

PRABHU, N.V. (1965): *Stochastic Processes.* MacMillan, New York.

PRUDENCIO, R.B.C., LUDERMIR, T., and De CARVALHO, F.A.T. (2004): A modal symbolic classifier for selecting time series models. *Pattern Recognition Letters, 25 (8), 911–921.*

SOULE, A., SLAMETIAN, K., TAFT, N. and EMILION, R. (2004): Flow classification by histograms. *ACM Sigmetrics.* New York, http://ps.lip6.fr/s̃oule/SiteWeb/Publication.php.

STIRZAKER, D. (2005): *Stochastic Processes and Models.* Oxford University Press, Oxford.

Quality Issues in Symbolic Data Analysis

Haralambos Papageorgiou[1] and Maria Vardaki[2]

[1] Department of Mathematics, University of Athens,
Panepistemiopolis, 154 84, Athens, Greece, *hpapageo@cc.uoa.gr*

[2] Department of Mathematics, University of Athens,
Panepistemiopolis, 154 84, Athens, Greece, *mvardaki@cc.uoa.gr*

Abstract. Symbolic Data Analysis is an extension of Classical Data Analysis to more complex data types and tables through the application of certain conditions, where underlying concepts are vital for their further processing. Therefore, the assessment of the quality of Symbolic Data depends extensively on the quality of the collected classical data. However, even though various criteria and indicators have been established to assess quality in classsical statistics, the specificities of Symbolic Data construction challenge the efficacy of the classical quality assessment components. In this paper we initially refer to the quality dimensions that can be considered for the classical data and then emphasize on the extent that these can be applied to symbolic data, taking into account the peculiarities of symbolic approach.

1 Introduction

Quality was defined in the ISO 8402 - 1986 as: “the totality of features and characteristics of a product or service that bear on its ability to satisfy stated or implied needs” and slightly changed in ISO updates. However, regarding quality in statistics, “stated or implied needs” are mainly identified by considering several quality dimensions, criteria or components for the collection, processing and dissemination of statistical information to the public (see for example OMB (2002), Eurostat (2002a) and (2002b), OECD (2003), IMF (2002), Statistics Canada (2003), Statistics Finland (2002), Viggo et al. (2003)).

The amount of information collected and processed by National and International Statistical Organisations is constantly growing, as demands of high quality statistics are steadily increasing. The quality of statistics is commonly assessed by Statistical Institutes with the use of quality dimensions/criteria, like for example, relevance (the degree to which statistics meet current and potential users’ needs), accuracy (refers to the closeness between the values provided and the (unknown) true values), timeliness, accessibility of information, comparability of the statistics (over time, across domains and between countries), etc. However Institutes, operating under strict budgets, try to manage the huge sets of collected data together with their underlying concepts keeping at the same time a satisfactory quality level.

An attempt to tackle the problem of huge datasets control has been made by Symbolic Data Analysis (SDA), thus extending the Classical Data Analysis into SDA through the mathematical design of concepts. Symbolic Data serve not only to summarize large sets of information (Billard and Diday (2003)), but they also lead to more complex data tables, thus enabling the manipulation of huge datasets (Bock and Diday (2000)). Using the Symbolic Data technique, data are aggregated into macrodata, forming Symbolic Objects (SO) and Symbolic Data Tables (SDT) (Bock and Diday (2000), Noirhomme (1997)).

In order to assess the quality of these complex data files, a number of quality criteria should be satisfied. Since, as mentioned, symbolic data depend on the classical/original data, their quality assessment is associated with the quality of the original data collected and mainly on their underlying concepts and procedures required for their extension into Symbolic data.

In this paper we refer to the quality criteria/dimensions used to assess classical statistics, mentioning some of the quality indicators frequently used for this purpose. Then, we examine which of these criteria can be applied to Symbolic Data emphasizing on the prerequisites of their eligibility. The main part of the paper stress on preservation of quality in two stages of symbolic data items creation, i) during the construction of a SO and a SDT and ii) during transformations of the already constructed SOs.

2 Classical and Symbolic Data Analysis

In classical data analysis, the statistical population, the sample derived through a sampling method, as well as the individual sampling units examined (called individuals thereafter) and the related (classical) variables, are the key issues to be evaluated when conducting a survey. In SDA, Symbolic Objects are the central items. SOs are triplets (α, R, d), where d is the descriptions of individuals (from a set of descriptions D), R the relation between the descriptions and α is a mapping from the set of individuals Ω in L. For further details see Bock and Diday (2000).

In addition, a classical (micro)data table refers to individuals and to classical variables. A Symbolic Data Table looks like a (micro)data table in the sense that it contains rows that correspond to (groups of) individuals and columns that correspond to (symbolic) variables.

3 The classical quality assessment framework

When carrying out a classical survey, the quality of statistics produced can be evaluated with the use of certain quality dimensions/criteria, like for example in Eurostat (2002a, 2002b):

- Relevance - the degree to which statistics meet current and potential users' needs,
- Accuracy - refers to the closeness between the values provided and the (unknown) true values,

- Timeliness - reflects the length of time between its availability and the event or phenomenon it describes,
- Punctuality - refers to the possible time lag existing between the actual delivery date of data and the target date when it should have been delivered,
- Accessibility of information - refers to the physical conditions in which users can obtain data: where to go, how to order, delivery time, clear pricing policy, convenient marketing conditions (copyright, etc.),
- Clarity - mainly refers to additional information provided together with the statistics (graphs, maps, etc) for their better understanding,
- Coherence of statistics - their adequacy to be reliably combined in different ways and for various uses,
- Comparability of the statistics (over time, across domains and between countries) - aims at measuring the impact of differences in applied statistical concepts and measurement tools/procedures when statistics are compared between geographical areas, non-geographical domains, or over time. We can say it is the extent to which differences between statistics are attributed to differences between the true values of the statistical characteristic.

The breakdown of quality into components is not unique neither invariant over time. Organizations use slightly different sets of quality dimensions. Some examples can be the following (see also Vardaki and Papageorgiou (2006)): the European Statistical Service, (Eurostat), follows six components to assess quality in statistics namely: Relevance, Accuracy, Timeliness and punctuality, Accessibility and clarity, Coherence and finally Comparability (Eurostat (2000b)). The Organization of Economic Cooperation and Development (OECD) proposes eight quality criteria (OECD (2003)) namely: Relevance, Accuracy, Credibility, Timeliness, Punctuality, Assessibility, Interpretability and Coherence; the International Monetary Fund (IMF) has developed its own Data Quality Assessment Framework (IMF (2002)). In addition, National Statistical Institutes (NSIs) and other organizations have also developed their own quality framework mainly taking into account the criteria proposed by the International Organization they are obliged or willing to report their results (see for example, Statistics Finland (2002), Statistics Canada (2003), Viggo et al (2003)).

In order to proceed towards a unified approach, at least among European Union and candidate countries, a list of “Standard Quality Indicators” (Linden and Papageorgiou (2004)) has been proposed. These producer-oriented indicators intend to measure the quality of classical statistics in relation to one or more of the above-mentioned quality criteria. Examples of such indicators can be the following: “Coefficient of Variation”, “The unit non-response rate”, “Average size of revisions”, etc, measuring accuracy, “Length of comparable time-series”, “Punctuality of time schedule of effective publication”, etc, measuring timeliness, “Rate of available statistics” and ”User satisfaction index" for relevance, "Length of comparable time series" for comparability, etc.

Except from these indicators a number of others can be considered when specific requirements should be met. The best way for these indicators to be applied on a dataset, is to incorporate them automatically in the workflow process of the production of classical statistics with the use of a metadata model (Vardaki and Papageorgiou (2004)). For example, in Figure 1 we illustrate part of a metadata model resembling the one in (Papageorgiou and Vardaki (2006)) where a number of indicators have been modeled. Examples include indicators measuring the following:

i) Accuracy - Non-response rate, Missing Values, Seasonal Adjustments, Sampling errors and corrections, Reporting method.
ii) Timeliness - Time elapses between event and processing, Date of availability of publication, Punctuality of time schedule.
iii) Accessibility and Clarity - Clarity of contents.
iv) Completeness - Rate of completeness.

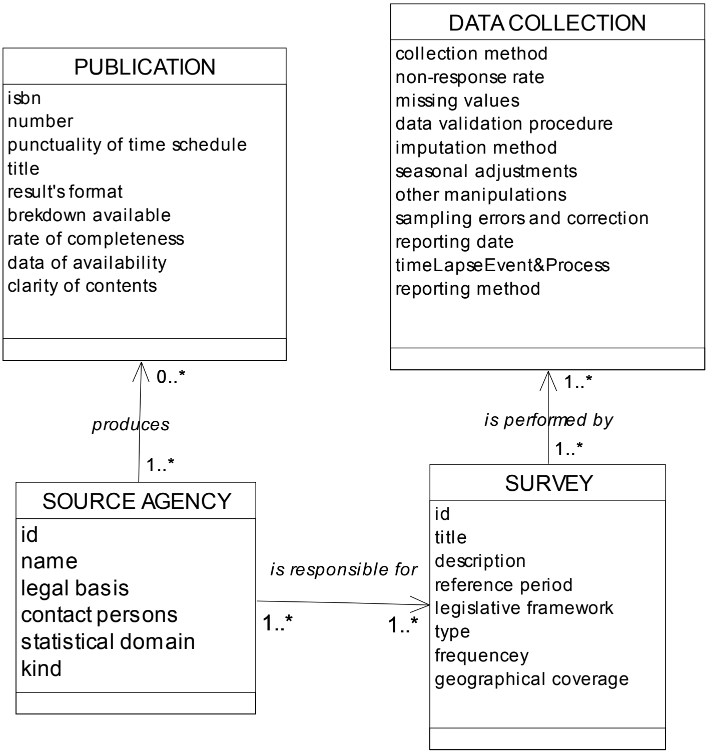


Fig. 1. Modeling quality issues for classical data.

4 Defining the symbolic data analysis quality framework

Since Symbolic Data are created from Classical Data satisfying specific conditions, the quality issues examined in the previous sections are a pre-requisite for any further quality assessment. In order to examine the role of quality in Symbolic Data Analysis we should consider it regarding two dimensions:

a) Quality assessment of the construction of a SO and a SDT.
b) Quality preservation during transformations of the already constructed SOs.

We elected to examine each of the two above cases according to the quality criteria used from Eurostat for the classical data (Eurostat (2002b)): Relevance, Accuracy, Timeliness and punctuality, Accessibility and clarity, Comparability and finally, Coherence.

4.1 Quality issues when modeling the construction of SO and SDT

Figure 2 illustrates the construction of SOs and SDTs after the creation of Symbolic Variables (SVars). It also depicts how Symbolic Data depend on the Individuals, the Statistical Population and other elements of the Original/Classical Data (Papageorgiou and Vardaki (2006)). In this part of the model it is presented that a group of individuals satisfying a set of conditions on the original/classical variables describes a symbolic object.

It is also important to consider that Symbolic data are constructed and/or managed by various sources (indicated by ‘Multiple sources’ class in the model). That enables the model to keep information about each person or institute attempting to use the already stored data and create new symbolic objects. In such cases, quality standards followed by this source (quality framework integrated in the processes, quality criteria, etc) play a crucial role in SOs’ quality assessment. In order to proceed with implementation of the quality criteria on Symbolic Data, we need to examine the specific requirements for the creation of SVars, SOs and SDTs.

Groups of Individuals

In composing groups of individuals the need to describe them and their process of synthesis, provides useful information both for the interpretation of the results and for the handling of the output for further processing. Essential knowledge on the SOs is the number of individuals from the sample that compound the SO, as well as the related number of individuals that correspond to the whole population using the sampling weights.

Symbolic Variables (SVars)

The SVars are produced from the operation of creation of groups of individuals. Each group is associated with a set of values, rather than a single one,

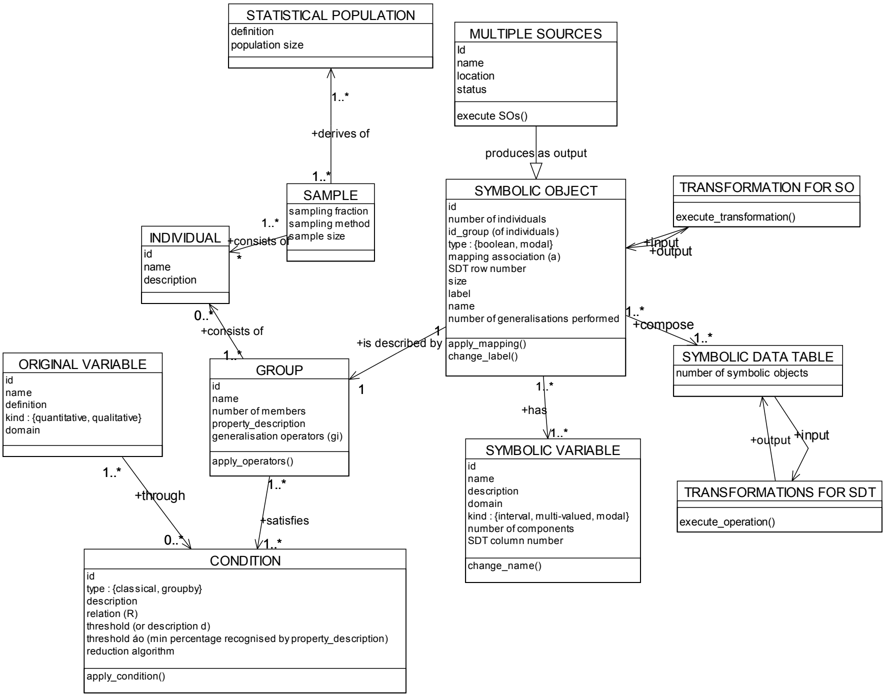


Fig. 2. Modeling SO and SDT construction.

as in classical data. An operation on this set forms the symbolic variable. The underlying concepts which should be evaluated include information on how these variables were created from the original variables, their nature, components and domain (see also Vardaki (2005a)).

Symbolic Objects (SOs)

The quality of the process of the symbolic object creation should be assessed by denoting the class membership variables, the operator applied to those variables (Average, sum etc.), the type of condition, the concept to which the SO is associated and the corresponding values (upper and lower limits, thresholds etc.). One notable difference with the classical setting is that information about individuals (now groups of individuals) is very important in symbolic analysis while in the classical setting information about the individuals themselves is not of primary interest. By composing sets (groups) of individuals the need to describe them and the process of synthesis, provides useful details both for the interpretation of the results and for the handling of the output for further processing.

Symbolic Data Tables (SDT)

A SDT looks like a (micro)data table in the sense that it contains rows that correspond to (groups of) individuals and columns that correspond to (symbolic) variables.

The quality assessment of the constructed SOs and SDTs pre-supposes the assessment of SOs and SVars quality. Regarding the quality components mentioned in classical setting, we can observe the following similarities and differences:

- Since *relevance* is associated with user needs the “Rate of Available SDT and SOs” will always be of interest, as well as the “User Satisfaction Index”. However, these indicators should be mainly considered on the produced SDTs and SOs and not too much on SVars and groups of individuals themselves.
- *Accuracy* plays a more important role on the collection of original statistics and in symbolic analysis it mainly concerns the methods of SO and SDT creation. Therefore, since the SO is described by the ‘Groups of individuals’, then the ‘Condition’ that should be satisfied for their construction should be evaluated (threshold, reduction algorithm, etc) as well as the method for the generalization of operators of each Group. In addition, a number of accuracy indicators of classical data can be applied. The “Over coverage and misclassification rates” may be applied to evaluate if the Group of individuals was formed properly. Also the “Average size of revisions” is always applicable in all kinds of data.
- *Timeliness and Punctuality* mainly affects classical data and seems that there is no need to examine them in symbolic analysis. However, the only possible effect on timeliness is when symbolic statistics are expected from multiple sources and should be delivered on time in order to contribute to a publication.
- *Comparability* can be considered from two perspectives: i) symbolic data constructed by different sources and ii) when a SDT is constructed by time series of SOs. In both cases, a high accuracy level of the constructed SOs is a precondition in order to further examine comparability. In addition, differences in the underlying concepts of individuals will be directly related to lack of comparability. Therefore, the ‘property description’ attribute of the class ‘individual’ of the model (see Figure 2) as well as the ‘description’ of the ‘condition’ should be equivalent to ensure comparability when SOs are created by different sources. In the second case, for SOs constructed by time series, although comparability is influenced mainly by changes in the original surveys, comparability over time may be also endangered by the underlying concepts of symbolic variables, as well as by any structure effects or administrative rules. Indicators like “Length of comparable time-series”, “Number of comparable time series” etc, may also be used for quality assessment of Symbolic Data.
- *Coherence* can be mainly influenced by the various compilation data sources (‘Multiple sources’ in the model) which may apply different standards when they construct a SO and a SDT.

- *Assessibility and Clarity* are critical for symbolic data mainly since users are not yet very familiar with the concept of SDA and the notion of SO and SDT.
- *Regarding completeness*, the indicator "Rate of completeness" refers to available information regarding the specific symbolic statistics including mainly information on related metadata accompanying the symbolic datasets. It is worth noticing to refer to the latest version of the SODAS software developed in the framework of the ASSO Eurostat project (Analysis System of Symbolic Official data, IST-2000-25161) where metadata both for original and symbolic data have been incorporated in its Library and can accompany SDTs and SOs creation.

Finally, it should be also noted that, although not usually identified as a measure of quality, the cost involved in the production and dissemination of statistics as well as the burden of respondents - both of which in the case of Symbolic Setting may be critical to a non-trained analyst - act as a constraint of quality.

4.2 Quality preservation in transformations of SOs and SDTs

Except from the above considerations, quality assessment should be considered also in the case of possible operations (called transformations in the model of Figure 2 and thereafter) that a user can apply on a symbolic object and corresponsing tables. There are transformations for SOs, such as the selection, the addition or deletion of a symbolic variable, etc and for the SDT, such as the addition or deletion of a symbolic object, the selection or projection of symbolic objects, or the sorting of symbolic objects contained in a specific SDT. All these operations maintain the property of closure (Papageorgiou et.al. (2000)), that is, when applied on symbolic data tables, the result is a new SDT.

Considering these transformations, we deduce that mainly relevance, coherence, comparability and clarity may be affected. Relevance can be increased when more symbolic data will be available and thus users can retrieve more information to satisfy their requirements; however, this may have severe implications to clarity if the new SDTs created are not propertly accompanied by explanations and definitions of any new concept that may emerge. Furthermore, in the case of any "deletion transformation" either of of a SVar from a SO or a SO from a SDT (for more information on such transformations see Papageorgiou and Vardaki (2006)), this omission may limit useful information for the understanding of the particular symbolic setting and thus lower the clarity levels. Regarding comparability and coherence, mainly the "addition" transformations in the previously mentioned cases may severely affect these two quality dimensions if the underlying concepts of the new SVar or SO are not fully compatible with the ones of the already existing SO and SDT respectively.

5 Conclusions and suggestions for a standard framework of Symbolic Data

In this paper we discussed that a number of quality criteria evaluating the quality of classical statistics can be implemented also in symbolic data and it appears that quality assessment for SDA has been considered for the first time.

On this basis, we can lead towards a standard quality framework for symbolic statistics defining a set of quality indicators and guidelines for their quality assurance. Some of the indicators, initially proposed and given below, are also part of the Standard Quality Indicators examined in Linden and Papageorgiou (2004):

- User satisfaction index (assessing Relevance)
- Rate of available statistics (assessing Relevance)
- Over-coverage and misclassification rates (assessing Accuracy) - Average size of revisions (assessing Accuracy)
- Length of comparable time-series (assessing Comparability)
- Number of comparable time series (assessing Comparability)

Further steps should include the definition of an entire quality framework, extending any guidelines used for classical statistics. Then the basic quality issues should be included, in the form of metadata, in the model described in previous section to ensure automatic quality assessment in all stages of SDA processes.

References

BILLARD, L. and DIDAY, E. (2003): From the Statistics of Data to the Statistics of knowledge: Symbolic Data Analysis. *Journal of the American Statistical Association (JASA), 98 (462), 470-487.*

BOCK, H.-H. and DIDAY, E. (2000): *Analysis of Symbolic Data*, Springer-Verlang, Berlin.

DIDAY, E. (2000): Symbolic Data Analysis and the SODAS project: Purpose, History, Perspective. In: H.-H.Bock and E.Diday (Eds.): *Analysis of Symbolic Data*, Springer-Verlang, Berlin, 1-22.

DIDAY, E. (2002): An introduction to Symbolic Data Analysis and the Sodas software. *The Electronic Journal of SYMBOLIC Data Analysis (JSDA), 0 (0), 1-25.*

EUROSTAT (2002a): Definition of quality in statistics. Retrieved from http://forum.europa.eu.int/Public/irc/dsis/Home/main.

EUROSTAT (2002b): Standard quality report. Retrieved from http://forum.europa.eu.int/ Public/irc/dsis/Home/main.

IMF (2002): Data Quality Assessment Framework and Data Quality Program. Retrieved from http://www.imf.org/external/np/sta/dsbb/2003/eng/dqaf.htm

LINDEN, H. & PAPAGEORGIOU, H. (2004): *Standard Quality Indicators.* European Conference on Quality and Methodology in Official Statistics (Q2004), Mainz, Germany. Also to appear in *Statistical Research Reference Material*, Japan Statistical Research Institute, Hosei University, Tokyo, Japan.

NOIRHOMME-FRAITURE, M. (1997): *Zoom-Star, a solution to complex statistical objects representation.* In: St. Howard, J.Hammond and G.Lindgaard (Eds.): Proc. INTERACT '97, Sydney, Australia.

OECD (2003): Quality framework and guidelines for OECD statistical activities. Retrieved from http://www.oecd.org/dataoecd/26/42/21688835.pdf

OFFICE OF MANAGEMENT AND BUDGET (OMB) (2002): Information Quality Guidelines. Retrieved from http://www.whitehouse.gov/omb/inforeg/iqg_oct2002.pdf

PAPAGEORGIOU, H., VARDAKI, M. and PENTARIS, F. (2000): Data and Metadata Transformations. *Research in Official Statistics (ROS), 3(2), 27-43.*

PAPAGEORGIOU, H., PENTARIS, F., THEODOROU, E., VARDAKI, M. and PETRAKOS, M. (2001): A statistical metadata model for simultaneous manipulation of data and metadata. *Journal of Intelligent Information Systems (JIIS), 17 (2/3), 169-192.*

PAPAGEORGIOU, H., VARDAKI. M., THEODOROU, E. and PENTARIS, F. (2002): *The use of Statistical Metadata Modelling and related transformations to assess the quality of statistical reports.* Joint UNECE/Eurostat Seminar on Integrated Statistical Information Systems and Related Matters (ISIS 2002), Geneva, Switzerland.

PAPAGEORGIOU, H. and VARDAKI, M. (2006): A Statistical Metadata Model for Symbolic Objects, to appear in the forthcoming E.Diday & M.Noirhomme (Eds.): *Symbolic Data Analysis and the SODAS Software*, Wiley.

STATISTICS CANADA (2003): Quality guidelines. Fourth Edition. Statistics Canada, Ottawa. Retrieved from http://www.statcan.ca/english/freepub/12-539-XIE/ 12-539XIE03001.pdf

STATISTICS FINLAND (2002): *Quality guidelines for Official Statistics.* Statistics Finland, Handbooks 43b. Helsinki, Finland.

VARDAKI, M. (2005a): Metadata for Symbolic Objects. *Electronic Journal of Symbolic Data Analysis (JSDA), 2(1), 1-8.*

VARDAKI, M. (2005b): Statistical Metadata in Data Processing and Interchange. In J. Wang (Ed): *Encyclopedia of Data Warehousing and Mining*, IDEA Group publishing, 2, 1048-1053.

VARDAKI, M. and PAPAGEORGIOU, H. (2004): *An integrated metadata model for statistical data collection and processing.* Proc. of the Sixteenth International Conference on Scientific and Statistical Database Management (SSDBM), Santorini, Greece, 363-372.

VARDAKI, M. and PAPAGEORGIOU, H. (2006): Statistical Data and Metadata Quality Assessment. To appear in the forthcoming M. Khosrow-Pour (Ed,): *Encyclopedia of Public Information Technologies*, IDEA Group Publishing, USA.

VIGGO, S.H., BYFUGLIEN, J. and JOHANNESSEN, R. (2003): Quality Issues at Statistics Norway. *Journal of Official Statistics (JOS), 19(3), 287-303.*

Dynamic Clustering of Histogram Data: Using the Right Metric

Rosanna Verde and Antonio Irpino

Facoltá di studi politici e per l'alta formazione europea e mediterranea
Seconda Universitá degli studi di Napoli, Caserta - Italy
rosanna.verde@unina2.it, irpino@unina.it

Abstract. In this paper we present a review of some metrics to be proposed as allocation functions in the Dynamic Clustering Algorithm (DCA) when data are distribution or histograms of values. The choice of the most suitable distance plays a central role in the DCA because it is related to the criterion function that is optimized. Moreover, it has to be consistent with the *prototype* which represents the cluster. In such a way, for each proposed metric, we identify the corresponding *prototype* according to the minimization of the criterion function and then to the best fitting between the partition and the best representation of the clusters. Finally, we focus our attention on a Wassertein based distance showing its optimality in partitioning a set of histogram data with respect to a representation of the clusters by means of their barycenter expressed in terms of distributions.

1 Introduction

In many real experiences, data are collected and/or represented by frequency distributions. If $\mathbf{Y}$ is a numerical and continuous variable, many distinct values y_i can be observed. In these cases, the values are usually grouped in a smaller number H of consecutive and disjoint bins I_h (groups, classes, intervals, etc.). The frequency distribution of the variable $\mathbf{Y}$ is obtained considering the number of data values n_h falling in each I_h. The histogram is then the typical graphical representation for the variable $\mathbf{Y}$.

The interest in analyzing data expressed by frequency distributions, as well as by histograms, is evident in many fields of research. In particular, we may refer to the treatment of experimental data that are collected in a range of values, whereas the measurement instrument gives only approximated (or rounded) values. An example can be given by sensors for air pollution control located in different zones of an urban area. The different distributions of measured data about the different levels of air pollutants across a day, allow us to compare, and then to group into homogeneous clusters, the different controlled zones.

In a different context of analysis, histograms are the key to understanding digital images. A digital image is basically a mosaic of square tiles or "pixels" of uniform color that are so thin that the composite image appears uniform and smooth. Instead of sorting them by color, they can be sorted into 256 levels of brightness from black (value 0) to white (value 255) with 254 gray levels in between. The height of each vertical "bar" tells you how many pixels there are for that particular brightness level.

In the present paper, we aim to analyze data expressed by distributions represented in form of “histograms”. The clustering of this kind of data can be useful to discover typologies of phenomena on the basis of the similarity of the frequency distributions.

Dynamic Clustering (DC) (Diday (1971), Diday and Simon (1976)) is here proposed as a suitable method to partition a set of frequency distributions data. We recall that DC is based on the definition of a criterion of the best fitting between the partition of a set of elements and the representation of the clusters of such partition. The algorithm simultaneously searches for the best partition into k clusters and their best representation. Thus, the DC needs to define a proximity function, to assign the individuals to the clusters, and a way to represent the clusters that is consistent with the optimized criterion. In the context of Symbolic Data Analysis (SDA), the clustering issued from a DCA on symbolic data (i.e., data characterized by multi-valued attributes) have to be represented by the so-called *prototypes* (Irpino et al. (2006)). The idea was to synthesize in the most suitable way the characteristics of the objects belonging to each cluster. The choice of the *prototype* was done according to the dissimilarity function used in the algorithm to allocate the elements to the cluster, in order to minimize a criterion of internal homogeneity. The consistence between the representation and the allocation function guarantees the convergence of the algorithm to a stationary value of the criterion. Moreover, the choice of a proximity (similarity or distance) function plays a central role in the DC algorithm. Issues that can affect a good proximity measure include their capability to be interpretable for the problem at hand, to have important theoretical properties.

In section 2, we outline the general scheme of DC. In section 3 we give the main definitions for histogram data. In section 4, we present the metrics generally used to compare histograms, their proprieties, and their usefulness in the context of DCA. Among the different proposed measures we emphasize the use of the Wasserstein distance for the DC of histogram data in section 4.5. In the case of DC on barycenters (known as Algorithm of “centres mobiles”, Benzécri (1973)), we prove that it is possible to define an inertia measure among data that satisfies the Huygens theorem of decomposition of inertia, considering the *prototypes* as barycenters. In section 5 we report some concluding remarks.

2 Dynamic clustering algorithm

The proximity measure assumes a great relevance for the interpretability of the problem at hand. Let E be a set of n data characterized by p continuous variables Y_j $(j = 1, \ldots, p)$. The dynamic clustering algorithm looks for the partition $P \in P_K$ of E in K classes among all the possible partitions P_K, and the vector $L \in L_K$ of K prototypes representing the classes in P such that the $\varDelta$ fitting criterion between L and P is minimized:

$$\varDelta(P^*, L^*) = Min\{\varDelta(P, L) \mid P \in P_K, L \in L_K\}. \tag{1}$$

Such a criterion is defined as the sum of dissimilarity or distance measures $\delta(y_i, G_k)$ of fitting between each element y_i belonging to a class $C_k \in P$ and the class representation $G_k \in L$:

$$\varDelta(P, L) = \sum_{k=1}^{K} \sum_{i \in C_k} \delta(y_i, G_k). \tag{2}$$

A prototype G_k associated with a class C_k is an element of the space of description of E, and it can be represented, in this context, as a histogram.

In order to introduce the next sections, we recall the general scheme of the DCA:

a) *Initialization*: Start from a random partition $P = (C_1, \ldots, C_k, \ldots, C_K)$ of the set E in K clusters,

b) *representation step*: for $k = 1$ to K, look for the prototype G_k which minimizes the criterion:

$$f_{C_k}(G) = \sum_{i \in C_k} \delta(y_i, G), \quad G \in L \tag{3}$$

```
c) allocation step
    - $test \longleftarrow 0$
    - for $i = 1$ to $n$ do:
        * Find the cluster $C_m$ to which $i$ belongs
        * Find the index $k$ such that: $k = argmin_{k=1,\ldots,K} D(y_i, G_k)$
        * if $k \neq m$
            · $test \longleftarrow 1$
            · $C_k = C_k \cup \{i\}$ and $C_m = C_m - \{i\}$
d) if $test = 0$ then stop, otherwise go to b)
```

At each iteration of the algorithm, a new couple (P, L) is found and the decrease of the $\varDelta$ criterion can be proven under the following conditions:

- uniqueness of the cluster allocation for each object $i \in E$
- uniqueness of the prototype G_k which minimizes the criterion f_{C_k} in (3) for all the clusters C_k of the partition P of E.

3 Histogram data

Let $\mathbf{Y}$ be a continuous variable defined on a finite support $\mathbf{S} = [\underline{y}; \overline{y}]$, where $\underline{y}$ and $\overline{y}$ are the minimum and maximum values of the domain of $\mathbf{Y}$. The variable $\mathbf{Y}$ is supposed partitioned into a set of contiguous intervals (bins) $\{I_1, \ldots, I_h, \ldots, I_H\}$, where $I_h = [\underline{y}_h; \overline{y}_h)$. Given N observations on the variable $\mathbf{Y}$, a function $\Psi(I_h) = \sum_{u=1}^{N} \Psi_{y_u}(I_h)$, where $\Psi_{y_u}(I_h) = 1$ if $y_u \in I_h$ and 0 otherwise, is associated with each semi-open interval I_h. Thus, it is possible to associate to I_h an empirical distribution $\pi_h = \Psi(I_h)/N$.

A histogram of $\mathbf{Y}$ is then the graphical representation where each pair (I_h, π_h) (for $h = 1, \ldots, H$) is represented by a vertical bar, with base the interval I_h along the horizontal axis and the area proportional to π_h. Having so defined histogram data, we assume E as a set of n empirical distributions $\mathbf{Y(i)}$ $(i = 1, \ldots, n)$.

In the case of a histogram description it is possible to assume that $S(i) = [\underline{y}_i; \overline{y}_i]$, where $y_i \in \Re$. Considering a set of uniformly dense intervals $I_{hi} = \left[\underline{y}_{hi}, \overline{y}_{hi}\right)$, such that:

$$\begin{array}{ll} i. & I_{li} \cap I_{mi} = \emptyset; \;\; l \neq m \;\; ; \\ ii. & \bigcup\limits_{s=1,\ldots,n_i} I_{si} = [\underline{y}_i; \overline{y}_i] \end{array}$$

the support can also be written as $S(i) = \{I_{1i}, ..., I_{ui}, ..., I_{n_i i}\}$. We denote with $\psi_i(y)$ the (empirical) density function associated with the description of i and with $\Psi_i(y)$ its distribution function. It is possible to define the description of $\mathbf{Y(i)}$ as:

$$Y(i) = \{(I_{ui}, \pi_{ui}) \mid \forall I_{ui} \in S(i);\; \pi_{ui} = \int\limits_{I_{ui}} \psi_i(y)dy \geq 0\} \;\text{ where } \int\limits_{S(i)} \psi_i(y)dy = 1.$$

Let $U(a, b)$ be a uniform density defined on the interval $[a, b]$, we may also interpret a histogram description as a particular mixture density distribution, i.e.:

$$Y(i) = \sum_{h=1,\ldots,H} \pi_{hi}\, U(\underline{y}_{hi}, \overline{y}_{hi})$$

4 Metrics for histogram data

Several distances defined between histograms can be proposed as allocation functions in a classical DCA scheme. According to such proposal, it needs to associate suitable *prototypes* to represent the obtained partition such that it is optimized a best fitting criterion between the partition and the representation of the clusters. As usually, in DCA the *prototype* is an element at minimum

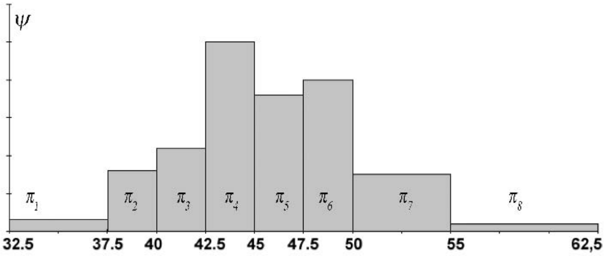


Fig. 1. An example of histogram description: the temperature in Fahrenheit degrees observed in Alabama in January from 1895 to 2004.

distance from all the elements of the cluster. A set of metrics, defined in probability measure spaces, seems particularly interesting to measure the similarity between distributions. So, they can be proposed in the DCA when data are considered as (empirical) distributions. These metrics were born in the framework of convergence theory. In particular we focus our attention on those metrics which respect the usual properties of distance measures. In the following, we present: the f-divergence based measures, the discrepancy metric, the Kolmogorov (or Uniform metric), the Prokhorov-Lévy distance and the Wasserstein-Kantorovich-Monge-Gini distance.
Let Ω be a measurable space with a σ-algebra $\mathcal{B}$, in our case Ω is a convex subset of $\mathbb{R}$ such that $S(i) \subset \mathit{\Omega}$ and then $I_{si} \subset \mathit{\Omega}$. Let $\mathcal{M}$ be the space of all probability measures on $(\Omega, \mathcal{B})$. In the following, we denote with μ and ν two probability measures (like the π_{ih} are) on Ω. Let f and g be the corresponding density functions with respect to a σ-finite dominant measure λ. If $\Omega = \mathbb{R}$, F and G denote the corresponding distribution functions.

4.1 F-divergence based measures

The *f-divergence* indexes (Csiszar (1967)) are based on a family of metrics where for every convex function ϕ one may define:

$$d_{\phi}\left(\mu,\nu\right) = \sum_{\omega} \nu\left(\omega\right) \phi\left(\frac{\mu\left(\omega\right)}{\nu\left(\omega\right)}\right)$$

- $\phi(x) = (x-1)^2$ yields d_{χ^2}, the Chi-square measure that, unfortunately is not symmetric and, thus cannot be considered as a dissimilarity,
- $\phi(x) = x \log x$ yields d_I,the Kullback-Leibler (or Relative entropy) divergence, that is not symmetric and, then, it is not a dissimilarity measure,
- $\phi(x) = |x-1|/2$ yields d_{TV}, the Total variation distance,
- $\phi(x) = (\sqrt{x}-1)^2$ yields d_H^2, the Hellinger distance.

Total variation For any measurable space is defined as:

$$d_{TV}\left(\mu,\nu\right) := \sup_{A \subset \mathit{\Omega}} \left|\mu\left(A\right) - \nu\left(A\right)\right| = \frac{1}{2} \max_{|h| \leq 1} \left| \int h \, d\mu - \int h \, d\nu \right|$$

where $h : \Omega \rightarrow \mathbb{R}$ satisfies $|h(x)| \leq 1$.
For countable spaces it is:

$$d_{TV}(\mu,\nu) := \frac{1}{2} \sum_{x \in \Omega} |\mu(x) - \nu(x)|$$

which is the half L^1 norm between two measures. It assumes values in $[0,1]$. Without loosing in generality, it is possible to define a *prototype* of a set of histograms as an histogram having the same support of the union of the supports of the clustered histogram and weights equal to the median of weights. Naturally there are not guarantees that the sum of weights of the *prototype* is equal to 1. Indeed, being the criterion of DC equal to:

$$\Delta(P,L) = \sum_{k=1}^{K} \sum_{i \in C_k} d_{TV}(y_i, G_k). \tag{4}$$

the definition of the distribution of the *prototype* G_k

$$G_k \sim \sum_{h=1,\ldots,H} \pi_{G_k h} U(\underline{y}_h, \overline{y}_h)$$

where H is the minimum number of the intervals partitioning the support, is done according to the minimization of the within simple variation of the cluster k:

$$\min\{\sum_{i \in C_k} \sum_{h=1,\ldots,H} |\pi_{ih} - \pi_{G_k h}|\}$$

In this case, it is possible to show that

$$\pi_{G_k h} = Med\{\pi_{ih} | i : y_i \in C_k\} \;\; for \; h = 1, \ldots, H$$

it is not assured that

$$\sum_h \pi_{G_k h} = 1,$$

then the *prototype* distribution of a set of histograms belonging to the cluster k is only a linear combination of their distributions but not a convex combination (as a mixture) of them.

Hellinger The distance is attributed to Hellinger (1901) that firstly used the quantity $\left(1 - \frac{1}{2} d_H^2\right)$ known as *Hellinger affinity.* For any measurable space, the distance can be formalized as:

$$d_H(\mu,\nu) := \left[\int_\Omega \left(\sqrt{f} - \sqrt{g}\right)^2 d\lambda\right]^{1/2} = \left[2\left(1 - \int_\Omega \sqrt{fg} d\lambda\right)\right]^{1/2}.$$

For countable space its version is:

$$d_H(\mu,\nu) := \left[\sum_{\omega\in\Omega}\left(\sqrt{\mu(\omega)}-\sqrt{\nu(\omega)}\right)^2\right]^{1/2}$$

It assumes values in $[0,\sqrt{2}]$. Without loosing in generality, it is possible to define the distribution of a *prototype* of a set of histograms as an histogram having the same support of the union of the supports of the clustered histograms and the optimal weights are equal to the squares of the square root averages of the weights. Indeed, the definition of the weights of G_k

$$G_k \sim \sum_{h=1,\ldots,H} \pi_{G_k h} U(\underline{y}_{hi}, \overline{y}_{hi})$$

is done according to the minimization of the within to cluster k sum of distances:

$$\min\{\sum_{i\in C_k}\sum_{h=1,\ldots,H}\left(\sqrt{\pi_{ih}}-\sqrt{\pi_{G_k h}}\right)^2\}$$

In this case it is possible to show that

$$\pi_{G_k h} = \left[\frac{1}{n_k}\sum_{i\in C_k}\sqrt{\pi_{ih}}\right]^2$$

Similarly to d_{TV} it is not assured that $\sum_h \pi_{G_k h} = 1$. For these kind of distances that violate the sum to one of weights, it is possible to introduce the constraint $\sum_h \pi_{G_k h} = 1$ in the minimization formula and, then, solving, when possible, the constrained optimization problem. Otherwise, the normalization of the $\pi_{G_k h}$ to one by the $\sum_h \pi_{G_k h}$, can arise from the lost of the optimality of the solution.

4.2 Discrepancy

It is defined on any metric space as:

$$d_D(\mu,\nu) := \sup_{\text{all closed balls } B} |\mu(B)-\nu(B)|$$

It assumes values in $[0,1]$. Diaconis (1988, p. 34) showed that it can be used to study weak convergence on random walks on groups and show some bounds for particular distributions using Fourier transformations of probability measures on compact sets. For univariate support the definition of the *prototype* is similar to the case of the total variation.

4.3 Kolmogorov (or Uniform) metric

It is defined on any metric space as:

$$d_K\left(F,G\right) := \sup_x \left|F\left(x\right) - G\left(x\right)\right|, x \in \mathbb{R}$$

It assumes values in $[0,1]$. The definition of the *prototype* is done accordingly to the definition of the G_k minimizing the within to cluster k sum of distances:

$$\arg \min_{G_k(x)} \left\{ \sum_{i \in C_k} \sup_x \left|F_i\left(x\right) - G_k\left(x\right)\right| \right\}. \qquad (5)$$

In this case, the solution is not unique. We can show that with a simple example. Let us consider two distributions functions with uniform densities: $U(10,20)$ and $U(30,40)$. According to the Kolmogorov distance formulation, their distance is equal to 1. If we need to find their *prototype*, we have to identify a distribution function satisfying the equation (5). If we limit our search only on the best mixture of the two uniforms it is possible to verify that the equation (5) has this kind of solution:

$$G_k \sim \alpha\, U(10,20) + \beta\, U(30,40)$$

where $\alpha \geq 0$, $\beta \geq 0$ and $\alpha + \beta = 1$. Then, we have infinite solutions according to the pairs (α, β) that are linked by the constraint $\beta = 1 - \alpha$.

4.4 Prokhorov (or Lévy-Prokhorov) metric

It is defined on any metric space as:

$$d_P\left(F,G\right) := \inf\left\{\epsilon > 0 \, : \, \mu\left(B\right) \leq \nu\left(B^{\epsilon}\right) + \epsilon, \text{for all Borel sets} \, B\right\}$$

where $B^{\epsilon} = \{x \, : \, \inf_{y \in B} d(x,y) \leq \epsilon\}$. It can be also rewritten as:

$$d_P\left(\mu,\nu\right) = \inf\left\{\epsilon > 0; \inf P\left[d(X,Y) > \epsilon\right] \leq \epsilon\right\}.$$

It generalizes the Lévi distance that is defined on $\mathbb{R}$

$$d_L\left(F,G\right) := \inf\left\{\epsilon > 0 \, : \, G\left(x - \epsilon\right) - \epsilon \leq F\left(x\right) \leq G\left(x + \epsilon\right) + \epsilon, \forall x \in \mathbb{R}\right\}$$

While not easy to compute, this metric is theoretically important because it permits to compute rate of convergence between two distributions on any separable metric space (Huber (1981)). The Prokorov distance between two random variables can be considered as the minimum distance in probability between the two random variables generated by μ and ν. In order to find a prototype between two distributions The definition of the *prototype* is done accordingly to the definition of the G_k that minimizes the within to cluster k sum of distances:

$$\arg\min_{G_k(x)} \left\{ \sum_{i \in C_k} \inf\left\{\epsilon > 0; \inf P\left[d(Y(i), G_k) > \epsilon\right] \leq \epsilon\right\} \right\}. \tag{6}$$

Until now we have not found a way to represent G_k and assure a single solution for the minimization of (6).

4.5 Wasserstein metric for histogram data

If F and G are the distribution functions of μ and ν respectively, the Kantorovich-Wasserstein metric is defined by

$$d_W(\mu, \nu) := \int_{-\infty}^{+\infty} |F(x) - G(x)|\, dx = \int_0^1 \left|F^{-1}(t) - G^{-1}(t)\right| dt.$$

In particular, we focus our attention on the following distance:

$$d_M(Y(i), Y(j)) := \sqrt{\int_0^1 \left(\Psi_i^{-1}(w) - \Psi_j^{-1}(w)\right)^2 dw} \tag{7}$$

also known as Mallow's (Mallow (1972)) distance in L^2, derived from the Wasserstein metric. Given a histogram description of $\mathbf{Y(i)}$ by means of H_i weighted intervals:

$$Y(i) = \left\{(I_{1i}, \pi_{1i}), ..., (I_{ui}, \pi_{ui}), ..., (I_{H_i i}, \pi_{H_i i})\right\},$$

we define the following quantities w_{li}

$$w_{li} = \begin{cases} 0 & l = 0 \\ \sum\limits_{h=1,\ldots,l} \pi_{hi} & l = 1, \ldots, H_i \end{cases} \quad . \tag{8}$$

in order to represent the cumulative weights associated with the elementary intervals of $Y(i)$. Assuming a uniform density for each I_h, we may write the empirical distribution function as:

$$\Psi_i(y) = w_i + \left(y - \underline{y}_{li}\right) \frac{w_{li} - w_{l-1i}}{\overline{y}_{li} - \underline{y}_{li}} \quad \textit{iff } \underline{y}_{li} \leq y \leq \overline{y}_{li}.$$

Thus, the quantile function is given by the following piecewise function defined as:

$$\Psi_i^{-1}(t) = \underline{y}_{li} + \frac{t - w_{l-1i}}{w_{li} - w_{l-1i}} \left(\overline{y}_{li} - \underline{y}_{li}\right) \qquad w_{l-1i} \leq t < w_{li}.$$

To compute the distance between two histogram descriptions $\mathbf{Y(i)}$ and $\mathbf{Y(j)}$ we need to identify a set of uniformly dense intervals to be compared on the basis of the two quantile functions. Let w be the set of the cumulated weights of the two distributions:

$$w = \left\{w_{0i}, ..., w_{ui}, ..., w_{H_i i}, w_{0j}, ..., w_{vj}, ..., w_{H_j j}\right\}$$

we extract a vector $\mathbf{w}$ of the sorted (without repetition) values of w

$$\mathbf{w} = \left[w_0, ..., w_l, ..., w_m\right]$$

where $w_0 = 0$, $w_m = 1$ and $\max(H_i, H_j) \leq m \leq (H_i + H_j - 1)$. Then, the squared distance between two histogram descriptions is:

$$d_M^2(Y(i), Y(j)) := \sum_{l=1}^{m} \int_{w_{l-1}}^{w_l} \left(\Psi_i^{-1}(t) - \Psi_j^{-1}(t)\right)^2 dt. \tag{9}$$

Each couple (w_{l-1}, w_l) allows us to identify two uniformly dense intervals, one for i and one for j, having respectively the following bounds:

$$I_{li} = [\Psi_i^{-1}(w_{l-1}); \Psi_i^{-1}(w_l)] \quad \text{and} \quad I_{lj} = [\Psi_j^{-1}(w_{l-1}); \Psi_j^{-1}(w_l)].$$

For each interval, the centers and the radii are:

$$c_{li} = (\Psi_i^{-1}(w_l) + \Psi_i^{-1}(w_{l-1}))/2 \; ; \; r_{li} = (\Psi_i^{-1}(w_l) - \Psi_i^{-1}(w_{l-1}))/2.$$

Because intervals are uniformly distributed, we may express them, using the function of the center and of the radius as: $c + r(2t - 1)$ for $0 \leq t \leq 1$. The equation (9) can be rewritten as:

$$d_M^2(Y(i), Y(j)) := \sum_{l=1}^{m} \pi_l \left[(c_{li} - c_{lj})^2 + \frac{1}{3}(r_{li} - r_{lj})^2\right]. \tag{10}$$

Given a set of n histogram data, it is possible to define its “barycenter” as a histogram itself (the so-called *prototype*). The prototypal histogram $\mathbf{Y(b)}$ is computed minimizing the following (sum of distance) function:

$$\begin{aligned} f(\mathbf{Y(b)}|\mathbf{Y(1)}, \ldots, \mathbf{Y(n)}) &= \\ = \sum_{i=1}^{n} d^2(Y(i), Y(b)) &= \sum_{i=1}^{n}\sum_{j=1}^{m} \pi_j \left[(c_{ji} - c_{jb})^2 + \frac{1}{3}(r_{ji} - r_{jb})^2\right]. \end{aligned} \tag{11}$$

It is easy to prove that the function (11) holds a minimum when:

$$c_{jb} = n^{-1}\sum_{i=1}^{n} c_{ji} \quad ; \quad r_{jb} = n^{-1}\sum_{i=1}^{n} r_{ji}.$$

The barycenter (*prototype*) of the n histogram data is expressed by the couples: $([c_{jb} - r_{jb}; c_{jb} + r_{jb}], \pi_j)$ of intervals with associated weighted by π_j. The identification of a barycenter permits us to show a second property of the criterion distance. Being d_M^2 a squared Euclidean distance, the total inertia with respect to the barycenter $\mathbf{Y}(\mathbf{b})$ of a set of n histogram data is given by:

$$TI = \sum_{i=1}^{n} d_M^2(\mathbf{Y}(\mathbf{i}), \mathbf{Y}(\mathbf{b})).$$

The TI can be decomposed into within (WI) and between (BI) clusters inertia, according to the Huygens' theorem:

$$TI = WI + BI = \sum_{k=1}^{K} \sum_{i \in C_k} d_M^2(\mathbf{Y}(\mathbf{i}), \mathbf{Y}(\mathbf{b_k})) + \sum_{k=1}^{K} |C_k| d_M^2(\mathbf{Y}(\mathbf{b_k}), \mathbf{Y}(\mathbf{b})) \tag{12}$$

where $\mathbf{Y}(\mathbf{b_k})$ is the barycenter of the k-th cluster. The decomposition of the inertia allows to use the classical criteria to interpret the quality of the obtained partition (Celeux et al., 1989).

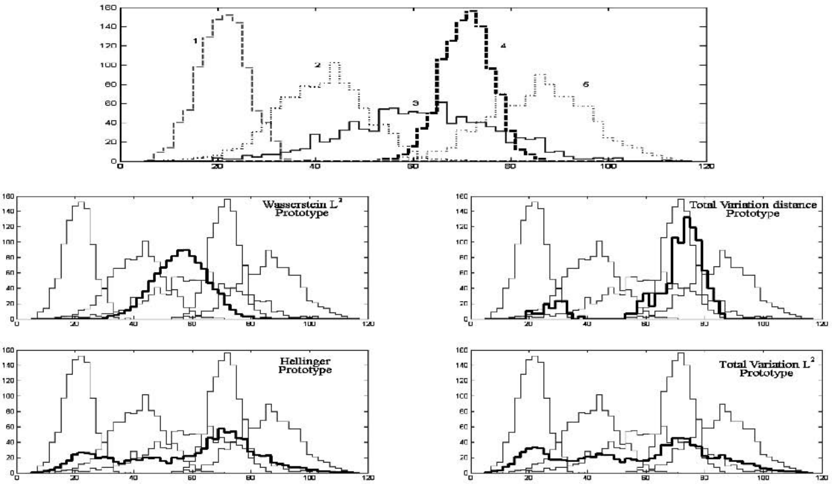


Fig. 2. Five histograms belonging to the same cluster (Top figure). Mixture-based *prototypes* (Center right and bottom figures). Average *prototype* histogram (Center left figure).

5 Conclusions: two different types of *prototypes*

In this paper we have presented a set of distances which can be used to cluster data represented by histograms. We here briefly summarize the kinds of prototypes of the clusters of a partition in DCA that have been identified using the different metrics. In Fig. 2 is presented an example of different *prototypes* computed according to different distances. We started our application considering five histograms that describe five artificial datasets consisting each one of 1,000 observations randomly extracted from five normal distributions ($N(20,5)$,$N(40,9)$,$N(60,15)$,$N(70,5)$,$N(85,10)$). The five histograms are represented at the top of Fig. 2, while the *prototypes* associated with four metrics are represented at the bottom of the figure. All metrics, except for the Wasserstein based one, allow to find *prototypes* that are represented by a combination of density distributions in a mixture-like representation. In this way, the description of a cluster k is done accordingly to a suitable choice of a set of weights to be associated with the intervals I_{ki} that partition the domain of the histograms: I_{ki} are fixed while $\pi_{G_{ki}}$ have to be found. The use of Wasserstein metric in the DC criterion allows to find the *prototype* as a histogram that is *barycentric* with respect the elements of the cluster, or as showed an average histogram (at the center left side of Fig. 2). A possible extension of the proposed distances to the multivariate case can be performed in the sense of Minkowski considering data as described by bivariate (or multivariate) uncorrelated histograms.

References

BENZÉCRI, J.P. (1973): *Théorie de l'information et classification d'après un tableau de contingence.* L'Analyse des données, Tome 1, Dunod.

CELEUX, G., DIDAY, E., GOVAERT G., LECHEVALLIER, Y., RALAMBONDRAINY, H. (1989) : *Classification Automatique des Données*, Environnement Statistique et Informatique. Bordas, Paris.

CHAVENT, M., DE CARVALHO, F.A.T., LECHEVALLIER, Y., and VERDE, R. (2006): New clustering methods for interval data, *Computational statistics*,Phisica–Verlag, 21, 211–229.

CSISZAR, I. (1967): Information type measures of difference of probability distributions and indirect observations, *Studia Sci. Math. Hungar*, 2, 299–318.

DIACONIS, P. (1988). *Group Representations in Probability and Statistics*, Institute of Mathematical Statistics, Harvard University, CA.

DIDAY, E., and SIMON, J.C. (1976): Clustering analysis, *In: K.S. Fu (Eds.),Digital Pattern Recognition*, 47–94, Springer Verlag, Heidelberg.

DIDAY, E. (1971): La méthode des nuées dynamiques, *Revue de Statistique Appliquée*, 19, 2, 19–34.

GIBBS, A.L. and SU, F.E. (2002): On choosing and bounding probability metrics, *International Statistical Review*, 70, 419.

HELLINGER, E. (1907): *Die Orthogonalinvarianten quadratischer Formen von unendlich vielen Variablen*, Dissertation, Göttingen.

HUBER, P.J. (1981): *Robust Statistics*, John Wiley and Sons, New York.

IRPINO, A., VERDE, R., and LECHEVALLIER Y. (2006): Dynamic clustering of histograms using Wasserstein metric, *in COMPSTAT 2006*, (Eds. Rizzi, Vichi), Springer, Berlin,869–876.

MALLOWS, C.L. (1972): A note on asymptotic joint normality. *Annals of Mathematical Statistics*, 43(2),508-515.

Part II

Clustering Methods

Beyond the Pyramids: Rigid Clustering Systems

Jean-Pierre Barthélemy[1,3,5], Gentian Gusho[2,3], Christophe Osswald[4]

[1] Département L.U.S.S.I., ENST Bretagne
CS 83818, 29238 Brest Cedex 3, France
JP.Barthelemy@enst-bretagne.fr

[2] Laboratoire de Statistique, Université de Haute Bretagne
Place du Recteur H. Le Moal, CS 24307, 35043 Rennes Cedex, France
Gentian.Gusho@uhb.fr

[3] TAMCIC, UMR CNRS 2872

[4] Laboratory E^3I^2, EA 3876, ENSIETA, 29806 Brest Cedex 9
Christophe.Osswald@ensieta.fr

[5] CAMS, UMR CNRS 8557
E.H.E.S.S
54 bvd Raspail, 75270 Paris Cedex 6

Abstract. This paper is devoted to, more or less new extensions of the notion of pyramid introduced by Diday (1984, 1986) and Fichet (1984, 1986). It is related to the notion of "rigid clustering system" or "rigid hypergraph" (topics related to combinatorial theory). Pyramids are representations of clusterings systems whose classes are connected subgraphs of a path (or, in other words, intervals of some linear order). More generally, we shall consider clustering systems whose classes are connected components of some graph. After reviewing some classical results, we shall emphasize relations between rigidity and minimal spanning trees.

1 Introduction

Pyramids are representations of clustering systems whose clusters are connected classes of a path (or, equivalently intervals of some linear order). In that framework the notion was introduced independently by Diday (1984, 1986) and Fichet (1984, 1986, under the name of "pseudo-hierarchies"). It has been intensively studied by many researchers like Batbedat (1990), Bertrand (1986, 1992, 1995), Durand and Fichet (1988) and many others ... As far as we know, this notion has several origins and many extensions:

Seriation in archeology (Robinson (1951)). The idea is that a dissimilarity (or similarity) matrix between archaeological objects, built on common (and different) morphological characters can reveal their chronology.

Observations in cluster analysis. The classes of a hierarchy are intervals of some linear order (Batbedat (1990); Bertrand (1992, 1995); Bertrand and Diday (1985, 1990, 1991); Janowitz (1995); Bertrand and Janowitz (2002); Bertrand (2002); Barthélemy *et al.* (2004); Brito (1991); Brossier (1980);

Brucker (2005); Brucker *et al.* (2003); Diday and Bertrand (1996); McMorris and Powers (1996); Diday (1983) and others).

Similarity analysis has been essentially developed in the context of social sciences. The idea is to account for the internal structure of clusters of similar objects. Here the paths are replaced by graphs (Flament (1962, 1976); Flament *et al.* (1971, 1979); Degenne and Verges (1973)). A hypergraph $\mathcal{H}$ is said to be rigid on a graph G whenever every hyperedge of $\mathcal{H}$ is a connected class of G. Within similarity analysis, relevant applications have been developed (Vergès (1970); Degenne (1973); Flament (1967, 1978, 1979, 1981)) as well as significant mathematical tools (Flament (1962, 1976, 1978); Flament *et al.* (1971, 1976, 1979)).

The mathematical approach to rigidity. It corresponds to interval hypergraphs (*i.e.* hypergraphs rigid on a path, like pyramids are): Berge (1972); Duchet (1978, 1979, 1984, 1995); Tucker (1972); Lehel (1983). It has also been extended to the case where G is a tree: Ryser (1969); Duchet (1976); Slater (1978); Leclerc (1984); Lehel (1983, 1985); Brucker (2005). Rigidity on a cycle has been developped by Quillot (1984) and Osswald (2003a, 2003b). Some intractability results and polynomial cases are given in Osswald (2003b), Brucker (2003a), and Brucker *et al.* (2003).

This paper is divided in three parts. The first one is devoted to basic considerations on clustering systems. The second provides some (more or less) known results about rigidity. They are essentially extracted from Osswald's PhD thesis (2003b). The last one gives recent advances from Gusho's PhD thesis (2007) linking rigidity with the minimum spanning tree of a graph.

2 Basic considerations on clustering systems

2.1 Hypergraphs and clustering systems

A *hypergraph* is a pair $\mathcal{H} = (X, \mathcal{E})$ consisting of a set X and set $\mathcal{E}$ of non-empty subsets of X: $\mathcal{E} \subseteq 2^X$. The set X is called the *vertex* set of $\mathcal{H}$ and the set $\mathcal{E}$ its *hyperedge* set. A hypergraph $\mathcal{H}$ is said to be *closed* for intersection, if for any A and B two hyperedges of $\mathcal{H}$ with a non-empty intersection, then $A \cap B$ is a hyperedge of $\mathcal{H}$.

A *clustering system* (CS) is a hypergraph $\mathcal{K}$, admitting the whole set X and each singleton $\{x\}$ as hyperedges. The hyperedges of $\mathcal{K}$ are called its *clusters*. In the following, we shall not distinguish between the CS $\mathcal{K}$ as a hypergraph and the set of its clusters. They will be both denoted by $\mathcal{K}$.

A *hierarchy* is a CS $\mathcal{K}$ such that $A, B \in \mathcal{K}$ implies $A \cap B \in \{A, B, \emptyset\}$.

A *pyramid* (Diday (1984); also called a *pseudo hierarchy* by Fichet (1984)) is a closed CS, whose clusters are intervals of some linear order on X. Following the terminology presented in the introduction, pyramids are *rigid* on a path.

An *indexed CS* is a pair $(\mathcal{K}, f)$, where the CS $\mathcal{K} = (X, \mathcal{E})$ is associated to a real valued mapping f defined on $\mathcal{E}$. The mapping f,called an *index*, verifies

$f(\{x\}) = 0$ for each singleton $\{x\}$ of X and is monotone towards inclusion lattice: if A is strictly included into B, then $f(A) < f(B)$.

2.2 Clustering systems associated with dissimilarity measures

A *dissimilarity* on the set X is a function d from the cartesian product $X \times X$ to the set of real numbers, such that:

- $d(x, y) = d(y, x)$,
- $d(x, y) \geq 0$ and $d(x, x) = 0$.

The dissimilarity d is said to be *proper* whenever $d(x, y) = 0$ implies $x = y$. Hereafter, a dissimilarity d will be assumed to be proper. Recall that d is a *distance*, whenever $d(x, z) \leq d(x, y) + d(y, z)$.

Ultrametrics constitute a particular type of dissimilarities. They fulfill the inequality: $d(x, z) \leq \max\{d(x, y), d(y, z)\}$. Obviously an ultrametric is a distance.

There are many ways to associate a CS with a dissimilarity d. A canonical one is to consider the maximal cliques of the threshold graphs induced by d. The threshold graph $G(d, \sigma)$ induced by d is defined as having X as vertex set and $\{x, y\}$ is an edge if and only if $d(x, y) \leq \sigma$.

As is well known, the maximal cliques of the various graphs $G(d, \sigma)$ constitute a clustering system, written K_d. The *clusters* of K_d are also called the clusters of d. Variants are the *balls* (Benzécri (1973)), the *2-balls* (Diatta and Fichet (1994, 1998)) and the *realizations* which are defined for every two vertices x and y as being the intersection of all the maximal cliques of the threshold graphs containing x and y (Brucker (2003a)).

Bijection theorems make dissimilarities equivalent to class models. The most famous links hierarchical clustering systems and ultrametric distances (Johnson (1967); Jardine *et al.* (1967); Benzécri (1973); *cf.* also Hartigan (1967)). Many others has been designed, in particular the equivalence between pyramids and the so-called strong Robinson dissimilarities (Diday (1984); Fichet (1984)).

Recall that a dissimilarity d on X is a *strong Robinsonian* on X if and only if there exists a linear order $\leq$ on X such that:

(i) $x \leq y \leq z$ implies $max\{d(x, y), d(y, z)\} \leq d(x, z)$;
(ii) $x \leq y \leq z \leq t$ and $d(x, z) = d(y, z)$ implies $d(x, t) = d(y, t)$;
(iii) $x \leq y \leq z \leq t$ and $d(y, t) = d(y, z)$ implies $d(x, z) = d(x, t)$.

3 An overview on rigidity

3.1 Rigid hypergraphs

As stated in the introduction, an hypergraph $\mathcal{H} = (X, \mathcal{E})$ is said to be *rigid* on a graph **G** with X as vertex set whenever each hyperedge of $\mathcal{H}$ is a connected class of G.

We say that a graph $G = (X, E)$ is a minimum rigidity graph, whenever $\mathcal{H}$ is rigid on G and for any graph $G' = (X, E')$, with $|E'| < |E|$, $\mathcal{H}$ is not rigid on G'.

3.2 NP-hardness results

Proposition 1. *(Osswald (2003b))The following problem is NP-complete:*

Name: *Rigidity Graph Decision Problem (DEC-RIG).*
Instance: *A hypergraph $\mathcal{H}$ with X as vertex set, an integer k.*
Question: *Does there exist a graph G, with X as vertex set and at most k edges such that $\mathcal{H}$ is rigid on G?*

It follows that the construction problem MIN-RIG is NP-hard.

Name: *Minimum rigidity graph (MIN-RIG).*
Instance: *A hypergraph $\mathcal{H}$ with X as vertex set.*
Construction: *A graph G, with X as vertex set and a minimum number of edges such that $\mathcal{H}$ is rigid on G.*

Remark that, if we set $n = |X|$ and $H = (X, \mathcal{E})$, for $k < \max\{|A| \mid A \in \mathcal{E}\}$, there is obviously no solution to DEC-RIG. If $k \geq n(n-1)/2$, the complete graph with n vertices is a trivial solution of DEC-RIG.

3.3 A polynomial instance

A hypergraph is said to be *prebinary* (Barthélemy (2003)) whenever, for each two vertices x, y, with $x \neq y$, the set of all hyperedges containing both x and y admits a smallest element for inclusion.

Prebinary hypergraphs are very usual in hierarchical clustering: hierarchies and more generally pyramids and quasi-hierarchies (Bandelt and Dress (1989); Diatta and Fichet (1994, 1998); Diatta (1996)).

Proposition 2. *(Osswald (2003b); Barthélemy et al. (2004)).*

When $\mathcal{H}$ is a prebinary hypergraph, the problems DEC-RIG and MIN-RIG can be solved in polynomial time.

3.4 Squeletons and rigidity on hypergraphs

A hypergraph $\mathcal{H} = (X, \mathcal{E})$ is said to be *connected* if there exists an order on its hyperedges such that we can index them A_1, A_2, …, A_m and have these two conditions :

- $\forall\, 2 \leq i \leq m$, $A_i \cap (\cup\{A_j \mid j < i\}) \neq \emptyset$
- $\cup\{A_i \mid i \leq m\} = X$.

A hypergraph $\mathcal{H} = (X, \mathcal{E})$ is said to be *rigid* on a hypergraph $\mathcal{H}' = (X, \mathcal{E}')$ if, for any hyperedge A of $\mathcal{H}$ and $\mathcal{E}'_A = \{B \in \mathcal{E}' \mid B \subseteq A\}$, the hypergraph $\mathcal{H}'|_A = (A, \mathcal{E}'_A)$ is connected.

Like the problems linked to rigidity on graphs, finding a *minimum rigidity hypergraph* is NP-hard.

Name: *Minimum rigidity hypergraph (HRIG-MIN).*
Instance: *A hypergraph $\mathcal{H}$ with X as vertex set.*
Construction: *A hypergraph $\mathcal{H}'$, with X as vertex set and a minimum number of hyperedges such that $\mathcal{H}$ is rigid on $\mathcal{H}'$.*

However, if we search a minimum rigidity hypergraph of $\mathcal{H} = (X, \mathcal{E})$ among its *partial hypergraphs*, $\mathcal{H}' = (X, \mathcal{E}')$ with $\mathcal{E}' \subseteq \mathcal{E}$, the solution is unique and can be found in polynomial time: $\mathcal{O}(|X|^2|\mathcal{E}|^2)$ operations. This partial hypergraph is called the *squeleton* of $\mathcal{H}$: Sq($\mathcal{H}$) (Flament et al. (1979)).

4 Rigidity and minimal spanning trees, some recent results

The results of this part concern some approximations of a dissimilarity d on a set of objects X. It is well known that an unique ultrametric u_d on X such that $u_d \leq d$, named *subdominant ultrametric*, is associated with d. On the other hand, different strong Robinsonians on X named *lower maximal strong Robinsonians* can be associated with the same dissimilarity (Brucker (2001)). We note r_d each one of them. In general, any lower maximal strong Robinsonian associated with a dissimilarity d provides a better approximation than the respective subdominant ultrametric: $u_d \leq r_d \leq d$.

We study the hierarchy corresponding to the subdominant ultrametric u_d and the pyramids corresponding to the lower maximal strong Robinsonians r_d. The dissimilarity d is represented as it is by its *realizations* (Brucker, 2003a). The classes of the hierarchy and the classes of any pyramid are exactly the realizations of, respectively, the subdominant ultrametric u_d and the corresponding lower maximal strong Robinsonian r_d (Brucker, 2003b). In general, there exist several minimum rigidity graphs associated with the realization of a dissimilarity d (Brucker (2003a), Osswald (2003b), Gusho (2005)).

In the definition of a rigidity graph (Flament *et al.* (1979)), the edges rigidifying the hypergraph are not weighted. In our context, for the hypergraph corresponding to a dissimilarity d, we value the edges of the corresponding rigidity graphs by the dissimilarity d (the weight of the edge xy is $d(x, y)$). The first result concerns the *length* of a minimum rigidity graph which is the sum of all the weights of the edges presented in the graph.

Proposition 3. *(Gusho (2005)).*
Among all the rigidity graphs of the realizations of a dissimilarity d, the minimum rigidity graphs have the same length which is the minimum one.

4.1 Minimum spanning trees (MSTs) associated with a dissimilarity

There are many algorithms for constructing minimum spanning trees associated with a connected weighted graph (Kruskal, Prim, etc.). A dissimilarity d on X is represented by a complete graph on X whose edges are weighted by the corresponding values of d. Then, we can talk about minimum spanning trees associated with the dissimilarity d. Figure 1 shows a dissimlarity which will be used for the illustration of all the results.

d	2	3	4	5	6	7
1	88	88	86	85	87	73
2		69	88	88	89	85
3			85	80	80	88
4				10	15	82
5					16	87
6						86

Fig. 1. *Dissimilarity d on $X = \{1, 2, 3, 4, 5, 6, 7\}$.*

We use the Kruskal algorithm to compute these trees. First, we range the values of d in increasing order and consider every object of X as a connected component. For $x, y \in X$ such that $d(x, y)$ is minimum, we call xy an *edge of Kruskal algorithm* only if it links two different connected components C_x and C_y containing, respectively, x and y. At the end of this step of the algorithm, we add the new connected component $C = C_x \cup C_y$ and erase the old ones, C_x and C_y, from the list of the connected components. The algorithm ends when we obtain X as a connected component. It is easy to see that all the edges of Kruskal algorithm form a minimum spanning tree of d.

A minimum spanning tree associated with d is computed in polynomial time $\mathcal{O}(|X|^2 log|X|)$ and, in general, is not unique. Their number will depend on the different choices we have, in any step of the algorithm, among the equal values of d linking two different connected components.

In Figure 2 are represented the two minimum spanning trees associated with d.

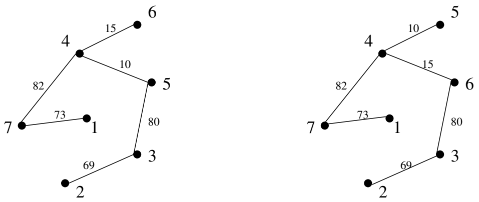


Fig. 2. *Minimum spanning trees associated with the dissimilarity d.*

4.2 Subdominant ultrametric and minimum spanning trees

First Gower and Ross (1969) and after Leclerc (1981, 1996a, 1996b) and many others showed that subdominant ultrametrics are strongly related with the minimal spanning trees associated with a dissimilarity d. For every two objects x and y, the value of the subdominant ultrametric $u_d(x, y)$ associated with the dissimilarity is calculated from each minimum spanning tree as being the greatest weight of the edge in the unique path of the tree linking x with y. In Figure 3, the bold numbers represent the values of the subdominant ultrametric u_d which are also values of d.

u_d	2	3	4	5	6	7
1	82	82	82	82	82	**73**
2		**69**	80	80	80	82
3			80	**80**	**80**	82
4				**10**	**15**	**82**
5					15	82
6						82

Fig. 3. *Sub-dominant ultrametric u_d asssociated with d.*

More than a way for computing the subdominant ultrametric, the minimum spanning trees have another role, this time concerning its clusters. In fact, the clusters of the subdominant ultrametric which form the corresponding hierarchy, are the components of all the threshold graphs of any minimum spanning tree. Thus, subdominant ultrametric clusters are rigid on every minimum spanning tree of d. Moreover, from the minimality of a tree structure for the number of edges and the connectivity, every minimum spanning tree associated with a dissimilarity d is a *minimum rigidity graph* for the hierarchy corresponding to the subdominant ultrametric u_d.

All the minimum spanning trees of d are also minimum spanning trees of the subdominant ultrametric u_d and they all have the same length. Thus, we join the result of Proposition 3 for the realizations of the subdominant ultrametric u_d. On the other hand, these trees are not the only ones rigidifying the clusters of the subdominant ultrametric. Because of the approximation procedure, other minimum spanning trees whose edges are weighted by values proper to the subdominant ultrametric rigidify its classes.

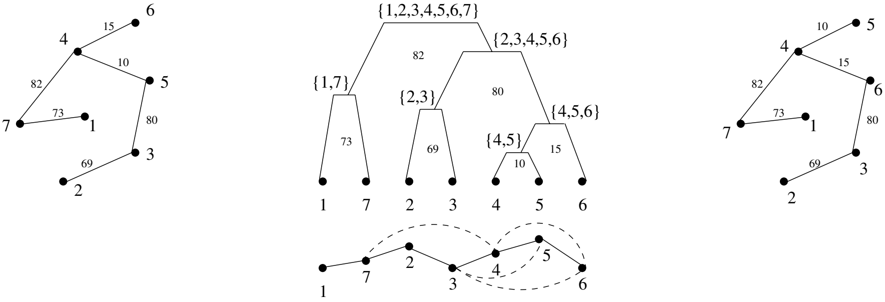


Fig. 4. *Minimum rigidity graphs of the hierarchy associated with u_d.*

In Figure 4 are represented three minimum spanning trees of u_d and the corresponding hierarchy. The MST below the hierarchy corresponds to the path for which the clusters of the hierarchy are intervals of the induced order. In general, this path is not an MST of d. The MSTs on the left and on the right of the hierarchy are at the same time MSTs of d.

As follows, we show that from the path we can obtain all the minimum spanning trees of d by replacing its edges weighted by u_d with edges weighted by d. In our example, if we replace in the path of Figure 4 the edges 72, 34 and 56 by, respectively, the edges in dotted lines 74, 35 and 46 weighted by d, we obtain the MST of d on the left. Identically, if we replace in the same path the edges 72, 34 and 56 by, respectively, the edges in dotted lines 74, 36 and 46 weighted by d, then we obtain the other MST of d on the right.

4.3 Lower maximal strong Robinsonians and minimum spanning trees

Robinsonians are dissimilarities such that all their clusters are intervals of some linear order on the set X. Thus, for a Robinsonian on X, the path formed by the ordered vertices is a minimum rigidity graph for the clusters of the associated pyramid. In general, for any lower maximal strong Robinsonian r_d associated with a dissimilarity d, this path which is a minimum spanning tree of r_d, is not a minimum spanning tree of d.

Figure 5 shows the unique lower maximal strong robinsonian r_d associated with the dissimilarity d and the bold numbers represent the values of d kept unchanged in r_d. As we can see by comparing Figure 3 and Figure 5, there are in r_d other bold numbers in addition to those presented in u_d which justify that lower maximal strong Robinsonians provide better approximations than the subdominant ultrametric.

r_d	2	3	4	5	6	7
1	85	85	82	**85**	82	**73**
2		**69**	85	85	85	**85**
3			80	**80**	**80**	85
4				**10**	**15**	**82**
5					**16**	85
6						82

Fig. 5. *Lower maximal strong Robinsonian r_d asssociated with d.*

In Figure 6 are represented the path for which the clusters of r_d are intervals, the two minimum spanning trees of d which are also minimum spanning trees of r_d and the pyramid associated with r_d. In the path, the weight of the edge 76 is $r_d(7,6) = 82$ which is different from $d(7,6) = 86$. Thus, the path is not a minimum spanning tree of d. In the same way than for the subdominant ultrametric, we show that from the path we can obtain all the MSTs of d. If we replace in the path of the Figure 6 the edge 76 by the edge in dotted line 74 which is weighted by d, we obtain the MST of d on the left. Moreover, if we replace the edge 53 by the edge 36 then we obtain the other MST of d on the right. In both cases, the clusters of r_d remain rigid on the MSTs of d.

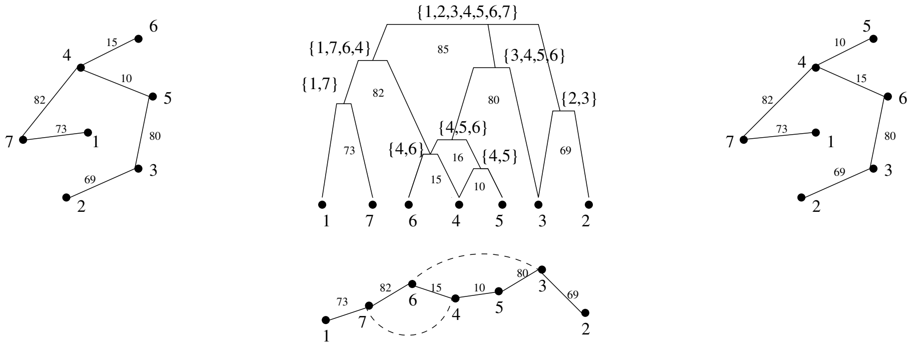


Fig. 6. *Minimum rigidity graphs of the pyramid associated with r_d.*

Proposition 4. *(Gusho (2007)).*

Every MST of d is a minimum rigidity graph of the pyramid associated with any lower maximal strong Robinsonian r_d.

Sketch of Proof. For each lower maximal strong Robinsonian r_d, we study the squeleton of the associated pyramid. For any cluster B of the pyramid squeleton, we note A the union of all the pyramid squeleton clusters strictly included in B. The following property concerns the *height* of B in the pyramid:

$$diam_{r_d}(B) = Min\{d(x,y) : x \in A \;\; and \;\; y \in B - A\}$$

From the last property and the maximality of r_d, we prove that all the pyramid squeleton clusters are rigidified by the edges of Kruskal algorithm which form a minimum spanning tree. In that way, we obtain all the minimum spanning trees of d.

4.4 Realizations, subdominant ultrametric and lower maximal strong Robinsonians associated with a dissimilarity

As follows, Proposition 5 shows the relation between the minimum rigidity graphs of the realizations of a dissimilarity d and the MSTs of d.

Proposition 5. *(Gusho (2005)).*

Every minimum rigidity graph of the realizations of d contains at least one of the MSTs of d.

In general, there are some MSTs of d which are not included in any minimum rigidity graph of the realizations of d. Figure 7 shows the unique minimum rigidity graph of the realizations of d. It contains the MST on the left of Figure 2 but not the MST on the right. The dotted lines represent the edges we have to add to the MST of d in order to obtain the minimum rigidity graph of the realizations of d.

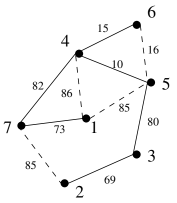


Fig. 7. *Minimum rigidity graphs of the realizations of d.*

Theorem 1 shows one way to obtain the MSTs of d included in a minimum rigidity graph of the realizations of d.

Theorem 1. *(Gusho (2005)).*

Each MST of any minimum rigidity graph of the realizations of d is an MST of d.

Conclusion

Usually, classification theory is devoted to external structures of classes: partitions, hierarchies, weak-hierarchies and so on... Following a very long tradition, the problem of the internal structure of classes has been emphasized in archeology (Robinson (1951)) and similarity analysis (Flament et al. (1976, 1979)). Moreover, the notion of internal structure of a class has been pointed by many botanists and zoologists like Linné, Tournefort, Cuvier, Buffon and others. In this paper, we have tried to make links between old questions and some, more or less, recent ones. The notion of rigidity inheriting from graphs and applying to clustering systems in order to look at the internal structure(s) of classes allows to account for this kind of problem.

References

BARTHELEMY, J.-P. (2003): Classifications binaires et quasi-hiérarchies. In: Y. Dodge, G. Melfi (Eds.): *Méthodes et Perspectives en Classification*, 67–69.

BARTHELEMY, J.-P., BRUCKER, F. and OSSWALD, C. (2004): Combinatorial optimization and hierarchical classifications. *4-OR 2, 179–219.*

BATBEDAT, A. (1990): *Les approches pyramidales dans la classification arborée.* Dunod, Paris.

BENZECRI, J. (1973): *La Taxonomie: L'Analyse des Données, Vol 1.* Dunod, Paris.

BERGE, C. (1972): *Graphes et Hypergraphes.* Dunod, Paris.

BERTRAND, P. (1986): *Étude de la Représentation Pyramidale.* PhD Thesis, Université de Paris-Dauphine.

BERTRAND, P. (1992): Propriétés et caractérisations topologiques d'une représentation pyramidale. *Mathématiques, Informatique et Sciences Humaines 30, 5-28.*

BERTRAND, P. (1995): Structural properties of pyramidal clustering. In: I. Cox, P. Hansen and B. Julesz (Eds:) *Partitionning Data Sets, DIMACS Series in Discrete Mathematics and Theoretical Computer Science* 19, AMS, Providence, R.I., 35–53.

BERTRAND, P.(2002): Set systems for which each set properly intersects at most one other set – application to pyramidal clustering. In: K. Jajuga and A. Sokolowski (Eds.):, *IFCS2002, Classification, Clustering, and Data Analysis*, 38–39.

BERTRAND, P. and DIDAY, E. (1985): A visual representation of the compatibility between an order and a dissimilarity index: the Pyramids CQS, *Computational Statistics Quarterly 2, 31–42.*

BERTRAND, P. and DIDAY, E. (1990): Une généralisation des arbres hiérarchiques: les représentations pyramidales. *Revue de Statistique Appliquée 38, 53–78.*

BERTRAND, P. and DIDAY, E. (1991): Les pyramides classifiantes: une extention de la structure hiérarchique. *Comptes Rendus de l'Académie des Sciences 1, 693–696.*

BERTRAND, P. and JANOWITZ, M.F. (2002): Pyramids and weak hierarchies in the ordinal model for clustering. *Discrete Applied Mathematics 122, 55–81.*

BRITO, P. (1991): *Analyse de Données Symboliques. Pyramides d'Héritage*, PhD thesis, University Paris IX Dauphine.

BROSSIER, G. (1980): Représentation ordonnée des classifications hiérarchiques. *Statistique et Analyse des Données 7, 2, 209–218.*

BRUCKER, F. (2001): *Classification en classes empiétantes.* Thesis, Ecole des Hautes Études en Sciences Sociales et ENST-Bretagne.

BRUCKER, F. (2003a): Réalisations de dissimilarité. In: Y. Dodge, G. Melffi (Eds.): *Méthodes et Perspectives en Classification*, 7–10.

BRUCKER, F. (2003b): communications personnelles

BRUCKER, F. (2005): From hypertrees to arboreal quasi-ultrametrics. *Discrete Applied Mathematics 147 (1), 3–26.*

BRUCKER, F., OSSWALD, C. and BARTHELEMY, J.-P. (2003): Rigid hypergraphs: combinatorial optimization problems in clustering and similarity analysis. *INOC Proceedings* 126–133.

CHEPOI, V. and FICHET, B. (1997): Recognition of Robinsonian dissimilarities. *Journal of Classification 14, 311–325.*

DEGENNE, A. (1973): Migrations intérieures - méthodes d'observation et d'analyse. *Actes du 4ème Colloque National de Démographie.*

DEGENNE, A. and VERGES, P. (1973): Introduction à l'analyse de similitude. *Revue française de sociologie 14, 471–512.*

DIATTA, J. (1996): *Une Extension de la Classification Hiérarchique: Les Quasi-Hiérarchies.* Thesis in Applied Mathematics, Université de Provence-Aix Marseille I, France.

DIATTA, J. and FICHET, B. (1994): From Asprejan hierarchies and Bandelt-Dress weak-hierarchies to quasi-hierarchies. In: E. Diday (Ed.): *New approaches in classification and data analysis.* Springer-Verlag, 111–118.

DIATTA, J. and FICHET, B. (1998): Quasi-ultrametrics and their 2-ball hypergraphs. *Discrete mathematics, 87–102.*

DIDAY, E. (1983): Inversion en classification automatique: application la construction adaptive d'indices d'agrégation. *Revue de Statistique Appliquée 31 (1), 45–62.*

DIDAY, E. (1984): *Une Représentation Visuelle des Classes Empiétantes: Les Pyramides.* Research Report, INRIA 291.

DIDAY, E. (1986): Orders and overlapping clusters in pyramids. In: J. de Leew et al. (Eds.): *Multidimensional data analysis proceedings*, 201–234.

DIDAY, E. and BERTRAND, P. (1996): An extension of hierarchical clustering: the pyramidal representation. In: E.S. Gelsama and L.N. Kanal (Eds.): *Pattern Recognition in Practice.* Elsevier Science Publishers (B.V.), North-Holland, Amsterdam, 411-424.

DUCHET, P. (1978): Propriété de Helly et probèmes de représentation. *Problèmes combinatoires et théorie des graphes, 117–118.*

DUCHET, P. (1979): *Représentation, Noyaux en Théorie des Graphes et Hypergraphes.* Thesis, Université Paris VI.

DUCHET, P. (1984): *Topics in perfect graphs.* Chapter Classical perfect hypergraphs. An introduction with emphasis on triangulalated and interval graphs. Amsterdam, North-Holland, 67–96.

DUCHET, P. (1995): *Hypergraphs*, chapter 7. The handbook of Combinatorics, 381–432.

DURAND, C. and FICHET, B. (1988): One to one correspondences in pyramidal representation: an unified approach. In: *Classification and related methods of data analysis*, 85–90.

FICHET, B. (1984): Sur une extension de la notion de hiérarchie et son équivalence avec quelques matrices de Robinson. In: *Actes des Journées de Statistique de la Grande Motte*, 12–12.

FICHET, B. (1986): Data analysis: geometric and algebraic structures. In: Prohorov et al. (Eds): *First world congress of the Bernoulli society proceedings.* V.N.U. Science Press, 123–132.

FLAMENT, Cl. (1962): *L'Analyse de la Similitude.* Cahier du Centre de la Recherche Opérationnelle 4, 63–97.

FLAMENT, Cl. (1967): Représentation dans une situation conflictuelle: une étude interculturelle. *Psychologie Franaise 12, 297–304.*

FLAMENT, Cl. (1976): *Hypergraphes et Analyse des Données.* Séminaire INRIA.

FLAMENT, Cl. (1978): Hypergraphes arborés. *Discrete Mathematics 21, 223–227.*

FLAMENT, Cl. (1979): Du biais de l'équilibre structural á la représentation de groupe. *Social Representations 4, 63–97.*

FLAMENT, Cl. (1981): L'analyse de la similitude, une technique pour les recherches sur les représentations sociales. *Cahier de psychologie cognitive* 1, 375–395.

FLAMENT, Cl., DEGENNE, A. and VERGES, P. (1971): *The Analysis of Similarity.* Research Program RCP 254, CNRS (Paris).

FLAMENT, Cl., DEGENNE, A. and VERGES, P. (1976): *L'Analyse de Similitude.* Report to the DGRST.

FLAMENT, Cl., DEGENNE, A. and VERGES, P. (1979): Analyse de la similitude ordinale. *Informatique et Sciences Humaines 40-41, 223–231.*

GOWER, J.C., ROSS, G.J.S. (1969): Minimal spanning trees and single linkage cluster analysis. *Applied Statistics 18, 54–64.*

GUSHO, G. (2005): Properties of minimum rigidity graphs associated with a clustering system. In: B.Mirkin and G.Magoulas (Eds.): *UKCI 2005: Computational Intelligence*, London, 220–225.

GUSHO, G. (2007): *Modèles de classification en classes empiétantes: application à la génomique dans le milieu marin.* Thesis, Université de Bretagne Sud et ENST-Bretagne (On-going).

HARTIGAN, J.A. (1967): Representation of dissimilarity matrices by trees. *Journal of the American Mathematical Society 62, 1140–1158.*

JANOWITZ, M.F. (1995): Generalization of pyramids. In: *Proceedings of the OSDA95 Meeting*, Paris.

JARDINE, J.P.J, JARDINE, N. and SIBSON, R. (1967): The structure and construction of taxonomic hierarchies. *Mathematical Biosciences 1, 171–179.*

JARDINE, N. and SIBSON, R. (1971): *Mathematical Taxonomy.* Chichester: Wiley.

JOHNSON, S.C. (1967): Hierarchical clustering schemes. *Psychometrika 32, 241–254.*

LECLERC, B. (1981): Description combinatoire des ultramétriques. *Mathématiques et Sciences Humaines 73, 5-37.*

LECLERC, B. (1984): *Comment reconnaître un hypergraphe arboré.* Cahier du CAMS, Paris.

LECLERC, B. (1996a): Minimum spanning trees and dissimilarity analysis. In: E.Diday et al. (Eds.): *Ordinal and Symbolic Data Analysis.* Berlin, Springer-Verlag, 215–224.

LECLERC, B. (1996b): Minimum spanning trees and types of dissimilarities. *European Journal of Combinatorics 17. 2/3, 255–264.*

LEHEL, J. (1983): Helly hypergraphs and interval abstract stuctures. *Ars Combinatoria 16, 239–253.*

LEHEL, J. (1985): A characterization of totally balanced hypergraphs. *Discrete Mathematics 57, 59–65.*

MC MORRIS, F.R. and POWERS, R.C. (1996): Intersection rules for consensus hierarchies and pyramids. In: E.Diday et al. (Eds.): *Ordinal and Symbolic Data Analysis.* Berlin, Springer-Verlag, 301–308.

OSSWALD, Ch. (2003a): Dissimilarités circulaires et hypercycles. In: Y. Dodge, G. Melfi (Eds.): *Méthodes et Perspectives en Classification*, 165–168.

OSSWALD, Ch. (2003b): Classification, analyse de la similitude et hypergraphes. *Thesis, École des Hautes Études en Sciences Sociales et ENST-Bretagne.*

QUILLOT, A. (1984): Circular representation problems on hypergraphs. *Discrete Mathematics 51, 251–264.*

ROBINSON, W.S. (1951): A method for chronologically ordering archeological deposits. *American Antiquities 16, 295–601.*

RYSER, H.J. (1969): Combinatorial configurations. SIAM *Journal of Applied Mathematics 17, 593–602.*

SLATER, P.J. (1978): A characterization of SOFT hypergraphs. *Canadian Mathematical Bulletin 21, 335–337.*

TUCKER, A. (1972): A structure theorem for the consecutive 1's property. *Journal of Combinatorial Theory 12, 153–162.*

VERGES, P. (1970): *Analyse de la Similitude et Méthode des Graphes Appliqués aux Réalités Socio-Économiques: Cas des Niveaux de Vie et de Développement des Régions Rurales du Liban (1960-1970).* Laboratoire Lebret, Lyon.

Indirect Blockmodeling of 3-Way Networks

Vladimir Batagelj[1], Anuška Ferligoj[2], and Patrick Doreian[3]

[1] University of Ljubljana, FMF, Department of Mathematics,
Jadranska 19, SI-1000 Ljubljana, Slovenia, *vladimir.batagelj@fmf.uni-lj.si*
[2] University of Ljubljana, Faculty of Social Sciences,
Kardeljeva pl. 5, SI-1000 Ljubljana, Slovenia, *anuska.ferligoj@fdv.uni-lj.si*
[3] University of Pittsburgh, Department of Sociology,
Pittsburgh, USA, *pitpat+@pitt.edu*

Abstract. An approach to the indirect blockmodeling of 3-way network data is presented for structural equivalence. This equivalence type is defined formally and expressed in terms of an interchangeability condition that is used to construct a compatible dissimilarity. Using Ward's method, the three dimensional partitioning is obtained via hierarchical clustering and represented diagrammatically. Artificial and real data are used to illustrate these methods.

1 Introduction

One of the tasks specified under 'extending generalized blockmodeling' Doreian et al. (2005) is the blockmodeling of 3-way networks – a 3 dimensional matrix defined on 3 sets of units. If 2 sets are equal we speak about 3-way 2-mode network. Here, we present work on a sub task, the indirect approach to *structural equivalence blockmodeling in 3-way networks*. *Indirect* means embedding the notion of equivalence in a *compatible* dissimilarity and determining a clustering based on that dissimilarity. The idea of blockmodeling 3-way data was proposed in an *ad hoc* fashion in Baker (1986) and in Everett and Borgatti (1992). We present a more systematic and general approach.

Two units are *structurally equivalent* iff they can be interchanged without producing change in the structure – the equivalent units have the **same** connection pattern to the **same** neighbors Batagelj et al. (1992).

In a usual 2-way 1-mode network $N = (\mathbf{U}, R)$, $R \subseteq \mathbf{U} \times \mathbf{U}$, x and y are structurally equivalent iff:

s1. $xRy \Leftrightarrow yRx$ s3. $\forall z \in \mathbf{U} \setminus \{x, y\} : (xRz \Leftrightarrow yRz)$
s2. $xRx \Leftrightarrow yRy$ s4. $\forall z \in \mathbf{U} \setminus \{x, y\} : (zRx \Leftrightarrow zRy)$

The blockmodeling of 2-way 2-mode networks is discussed in Doreian et al. (2004).

2 Structural equivalence in 3-way networks

A *3-way network* N over the basic sets X, Y and Z is determined by a ternary relation $R \subseteq X \times Y \times Z$.

The relation R can be represented by a 3-dimensional binary matrix $R_{X \times Y \times Z}$

$$R[i,p,u] = \begin{cases} 1 & R(i,p,u) \\ 0 & \neg R(i,p,u) \end{cases}$$

We define the following items:

Plane: $R(i,\cdot,\cdot) = \{(i,p,u) : p \in Y \wedge u \in Z \wedge R(i,p,u)\}$

Line: $R(i,\cdot,u) = \{(i,p,u) : p \in Y \wedge R(i,p,u)\}$

Truncated line: $R(i,-T,u) = \{(i,p,u) : p \in Y \setminus T \wedge R(i,p,u)\}$

Representations of these elements by binary vectors will be indicated by replacing braces with brackets.

The subsets $X_1 \subseteq X$, $Y_1 \subseteq Y$, $Z_1 \subseteq Z$ determine a *block* $R(X_1, Y_1, Z_1) = R \cap X_1 \times Y_1 \times Z_1$. If $R(X_1, Y_1, Z_1) = \emptyset$ the block is called a *null block*; if $R(X_1, Y_1, Z_1) = X_1 \times Y_1 \times Z_1$ the block is called a *complete block*.

In the following we shall also need a dissimilarity $D(a,b)$ between vectors a and b defined in R–like notation as

$$D(a,b) = \mathrm{sum}(\mathrm{abs}(a-b))$$

For example, for $a = [0,1,1,0,1]$ and $b = [1,1,0,0,0]$ we have

$$D(a,b) = \mathrm{sum}(\mathrm{abs}([-1,0,1,0,1])) = \mathrm{sum}([1,0,1,0,1]) = 3$$

The notion of structural equivalence depends on which of the sets X, Y and Z are (considered) the same. There are three basic cases: 1) all three sets are different – 3-mode network; 2) two sets are the same – 2-mode network; and 3) all three sets are the same – 1-mode network.

3 Case 1: All three sets are different

In this case we have a structural equivalence on each of the sets X, Y and Z. This is defined on the set X as follows:

The units $i, j \in X$ are *structurally equivalent*, $i \approx j$, iff

$$\forall p \in Y \forall u \in Z : (R(i,p,u) \Leftrightarrow R(j,p,u))$$

This is equivalent to the conditions that the 'planes' corresponding to i and j are equal $R(i,\cdot,\cdot) = R(j,\cdot,\cdot)$. The corresponding dissimilarity

$$d(i,j) = D(R[i,\cdot,\cdot], R[j,\cdot,\cdot])$$

is *compatible* with structural equivalence

$$i \approx j \Leftrightarrow d(i,j) = 0$$

The other two cases can be reduced to this one by permuting dimensions.

The solution consists of three structural equivalences $\approx_X$, $\approx_Y$ and $\approx_Z$, corresponding to three partition functions (π, σ, τ): $i \approx_X j \Leftrightarrow \pi(i) = \pi(j)$, etc.

If R is a 3D structural equivalence the only possible blocks in R with respect to clusters determined by this solution are null and complete blocks.

4 Case 2: Two sets are the same

Assume that $Y = Z$ and X is (considered as) different. The other two cases can be reduced to this one by permuting dimensions. The solution consists of two equivalencies / partitions (π, σ). The first equivalence is defined in the same way as in Case 1.

For the second equivalence the conditions are less trivial. Conceptually two units p and q are structurally equivalent if they are interchangeable – but in our case if we swap units p and q in the set Y we have to swap them also in the set Z.

The *interchangeability condition* – definition of structural equivalence, $p \approx q$, is now

$$\begin{aligned}\forall i \in X : (\ & \forall r \in Y \setminus \{p,q\} : (R(i,p,r) \Leftrightarrow R(i,q,r)) \\ & \wedge \forall r \in Y \setminus \{p,q\} : (R(i,r,p) \Leftrightarrow R(i,r,q)) \\ & \wedge (R(i,p,q) \Leftrightarrow R(i,q,p)) \\ & \wedge (R(i,p,p) \Leftrightarrow R(i,q,q))\ )\end{aligned}$$

The corresponding compatible dissimilarity is

$$\begin{aligned}d(p,q) = & D(R[\cdot,p,-\{p,q\}], R[\cdot,q,-\{p,q\}]) \\ & + D(R[\cdot,-\{p,q\},p], R[\cdot,-\{p,q\},q]) \\ & + D(R[\cdot,p,q], R[\cdot,q,p]) \\ & + D(R[\cdot,p,p], R[\cdot,q,q])\end{aligned}$$

If R is a 3D structural equivalence the blocks in R with respect to the partitions (π, σ) are null and complete blocks, but on $Y \times Z$ diagonals can be also zero diagonal planes in complete blocks and one diagonal planes in null blocks.

5 Case 3: All three sets are the same

In this case $X = Y = Z$. The units i and j are swapped in all three sets. They are structurally equivalent, $i \approx j$, iff

$$\begin{aligned}& \forall u,r \in X \setminus \{i,j\} : (R(i,u,r) \Leftrightarrow R(j,u,r)) \\ \wedge\ & \forall u,r \in X \setminus \{i,j\} : (R(u,i,r) \Leftrightarrow R(u,j,r)) \\ \wedge\ & \forall u,r \in X \setminus \{i,j\} : (R(u,r,i) \Leftrightarrow R(u,r,j)) \\ \wedge\ & \forall r \in X \setminus \{i,j\} : (R(i,j,r) \Leftrightarrow R(j,i,r)) \\ \wedge\ & \forall r \in X \setminus \{i,j\} : (R(i,r,j) \Leftrightarrow R(j,r,i)) \\ \wedge\ & \forall r \in X \setminus \{i,j\} : (R(r,i,j) \Leftrightarrow R(r,j,i)) \\ \wedge\ & \forall r \in X \setminus \{i,j\} : (R(i,i,r) \Leftrightarrow R(j,j,r)) \\ \wedge\ & \forall r \in X \setminus \{i,j\} : (R(i,r,i) \Leftrightarrow R(j,r,j)) \\ \wedge\ & \forall r \in X \setminus \{i,j\} : (R(r,i,i) \Leftrightarrow R(r,j,j)) \\ \wedge\ & (R(i,i,j) \Leftrightarrow R(j,j,i)) \\ \wedge\ & (R(i,j,i) \Leftrightarrow R(j,i,j)) \\ \wedge\ & (R(j,i,i) \Leftrightarrow R(i,j,j)) \\ \wedge\ & (R(i,i,i) \Leftrightarrow R(j,j,j))\end{aligned}$$

The corresponding compatible dissimilarity is

$$\begin{aligned}
d(i,j) = {} & D(R[i,-\{i,j\},-\{i,j\}],R[j,-\{i,j\},-\{i,j\}]) \\
& + D(R[-\{i,j\},i,-\{i,j\}],R[-\{i,j\},j,-\{i,j\}]) \\
& + D(R[-\{i,j\},-\{i,j\},i],R[-\{i,j\},-\{i,j\},j]) \\
& + D(R[i,j,-\{i,j\}],R[j,i,-\{i,j\}]) \\
& + D(R[i,-\{i,j\},j],R[j,-\{i,j\},i]) \\
& + D(R[-\{i,j\},i,j],R[-\{i,j\},j,i]) \\
& + D(R[i,i,-\{i,j\}],R[j,j,-\{i,j\}]) \\
& + D(R[i,-\{i,j\},i],R[j,-\{i,j\},j]) \\
& + D(R[-\{i,j\},i,i],R[-\{i,j\},j,j]) \\
& + D(R[i,i,j],R[j,j,i]) \\
& + D(R[i,j,i],R[j,i,j]) \\
& + D(R[j,i,i],R[i,j,j]) \\
& + D(R[i,i,i],R[j,j,j])
\end{aligned}$$

We illustrate the indirect approach to structural equivalence for 3-way data with two examples. One is an artificial data set (with known properties) and one real example drawn from the social network literature.

To support the indirect approach to 3-way blockmodeling based on structural equivalence the package `ibm3m` was developed in R (see references).

6 Example 1: Artificial dataset

The first 3-mode dataset consists of randomly generated ideal structure (5 clusters in X, 6 clusters in Y, and 4 clusters in Z; each set has 35 units) obtained using the function `rndMat3m(c(5,6,4),c(35,35,35))` from the package `ibm3m`.

Figure 1 shows the generated data with no obvious patterned structure. The right part of Figure 1 and Figure 2 show the dendrograms obtained for each of the three modes using Ward's method and Figure 3 shows the reordered three mode data. The complete three dimensional blocks are shown clearly on the upper left with the remaining three diagrams showing some slices with block structures.

7 Example 2: Krackhardt's dataset

The real example of a social network is taken from Krackhardt (1987) and takes the form of a 21 × 21 × 21 cube. The dimensions X and Y correspond to individuals in the management team of a high-tech company. The X mode consists of choices made by individuals (with regard to advice getting), the Y mode has the received choices for individuals. The Z mode consists of each individual's perception of the advice getting network for the management team.

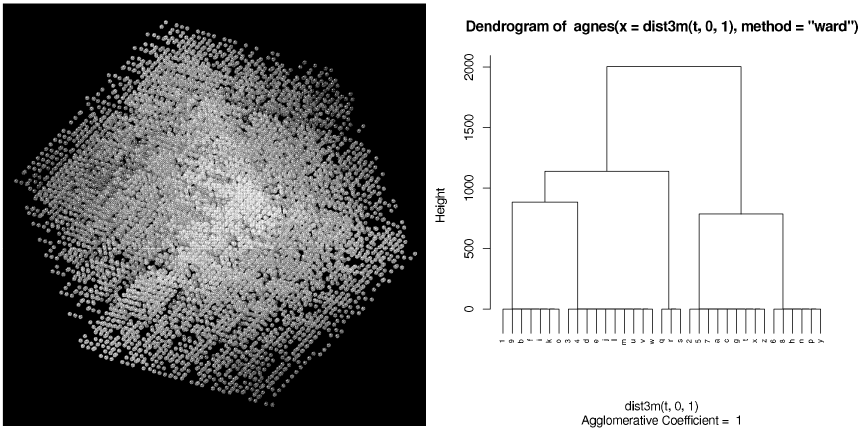


Fig. 1. Artificial dataset – original data and dendrogram on X.

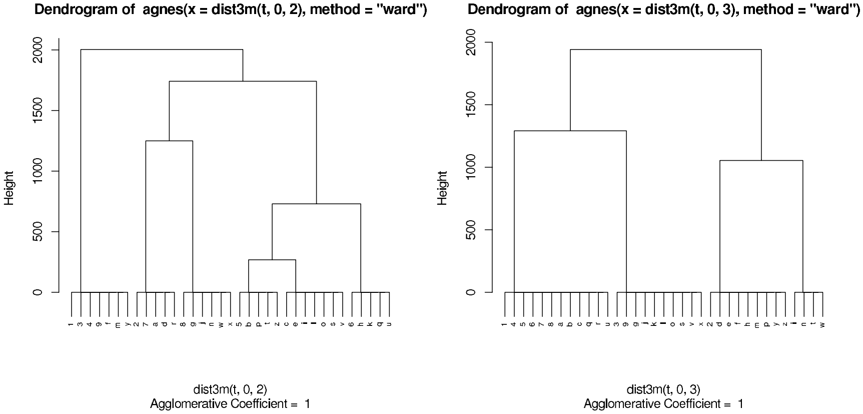


Fig. 2. Artificial dataset – dendrograms on Y and Z.

Figure 4 shows the dendrogram for each of the three dimensions. The dendrogram on the left depicts (approximate) structural equivalence for the sending of help seeking choices and the middle dendrogram depicts structural equivalence for the receipt of these choices. The right hand dendrogram shows structural equivalence for the perceptions of the help seeking relation. The full structural equivalence partition is shown in Figure 5 together with a slice. We can notice that the solution is very far from an ideal structural solution, but some structure can be seen.

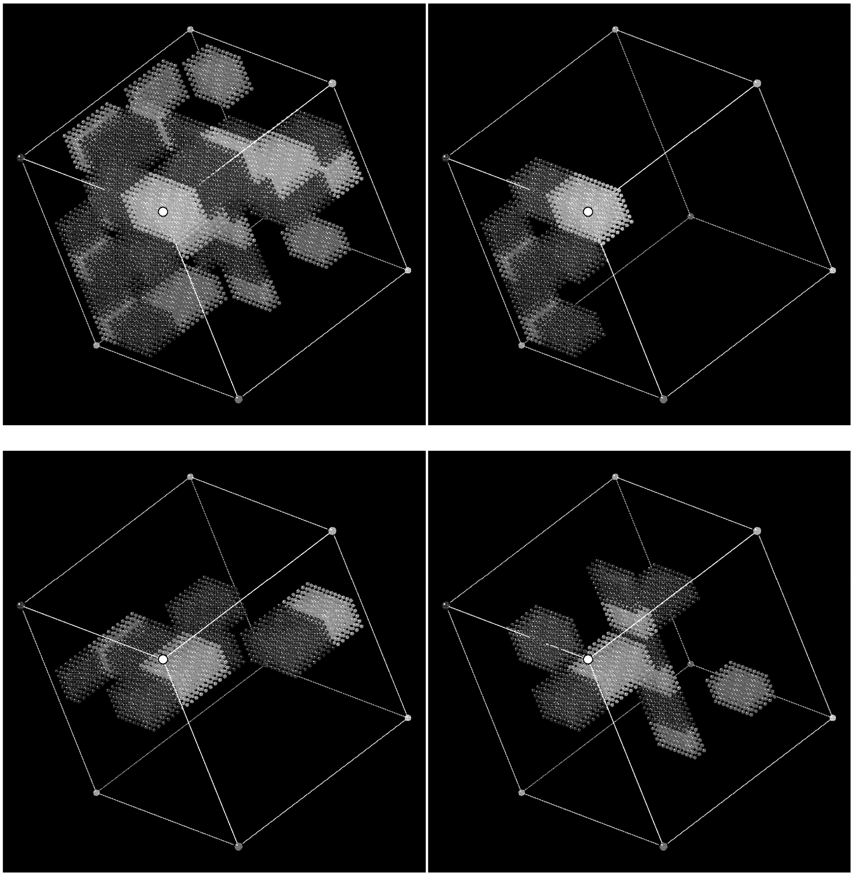


Fig. 3. Artificial dataset – reordered data; complete and some slices.

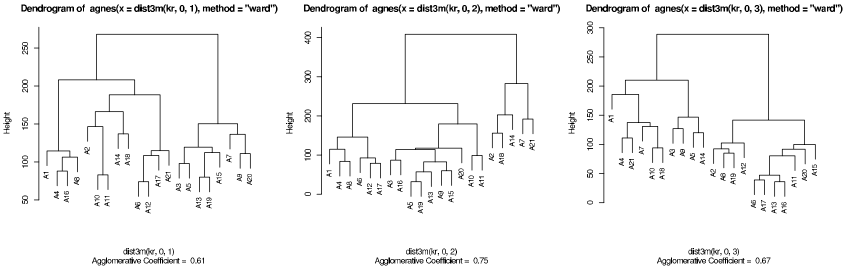


Fig. 4. Krackhardt – dendrograms (X, Y, Z).

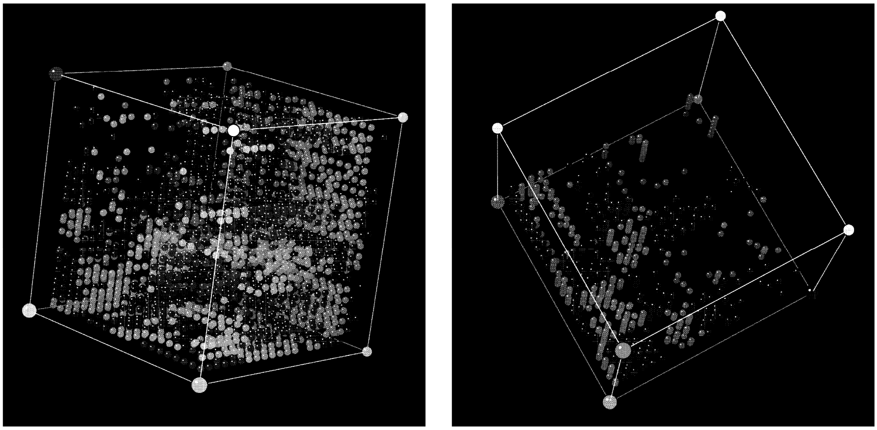


Fig. 5. Krackhardt – Structural Equivalence Partition.

8 R code for the analyses (`ibm3m`)

Here is a test procedure using the package `ibm3m` to generate and cluster random ideal datasets. For hierarchical clustering of the obtained dissimilarities and dendrograms drawing the functions `agnes` and `plot` from the R-package `cluster` (Kaufman and Rousseeuw (1990)) are used. The 3D pictures of the data matrix and its slices are exported into kinimages format (see references).

```
rndTest <- function(m=c(3,3,3),n=c(30,30,30),p=0.35){
  t <- rndMat3m(m,n,p)
  saveTriplets3m('test.tri',t,tit="random test")
  rx <- agnes(dist3m(t,0,1),method='ward')
  ry <- agnes(dist3m(t,0,2),method='ward')
  rz <- agnes(dist3m(t,0,3),method='ward')
  pdf('testXD.pdf')
  plot(rx,which.plots=2,nmax.lab=50,cex=0.6); dev.off()
  pdf('testYD.pdf')
  plot(ry,which.plots=2,nmax.lab=50,cex=0.6); dev.off()
  pdf('testZD.pdf')
  plot(rz,which.plots=2,nmax.lab=50,cex=0.6); dev.off()
  kin3m('testOrg.kin',"test - original",t,
    seq(n[1]),seq(n[2]),seq(n[3]))
  kinBlocks3m('testXYZ.kin',"test - all different",t,rx,ry,rz,m)
  if (n[2]==n[3]){
    rb <- agnes(dist3m(t,1,1),method='ward'); pdf('testBD.pdf')
    plot(rb,which.plots=2,nmax.lab=50,cex=0.6); dev.off()
    kinBlocks3m('testXYY.kin',"test - two equal",t,rx,rb,rb,m)
  }
  if ((n[1]==n[2])&(n[2]==n[3])){
    ra <- agnes(dist3m(t,2,0),method='ward'); pdf('testAD.pdf')
    plot(ra,which.plots=2,nmax.lab=50,cex=0.6); dev.off()
    kinBlocks3m('testXXX.kin',"test - all equal",t,ra,ra,ra,m)
  }
}
```

The procedure for analysis of Krackhardt's data is similar – only we have to read the data files:

```
kr <- readDL3m('krack.dat')
la <- paste('A',1:21,sep=''); dimnames(kr) <- list(la,la,la)
```

The computation of dissimilarities is quite time consuming – it is of order $O(n^4)$. But for small networks as in the above examples it takes only some seconds.

9 Discussion

We have presented an approach to blockmodeling 3-way network data using indirect methods. The indirect approach is feasible only when an appropriate compatible dissimilarity can be defined, as is the case for structural equivalence. When a compatible dissimilarity cannot be defined, direct and graph theoretical methods are appropriate. Both approaches to blockmodeling 3-way data are the focus of ongoing work.

Acknowledgments

This work was partially supported by the Slovenian Research Agency, Project J1-6062-0101 and P5-0168.

The paper is a detailed version of the talk presented at *IFCS 2006* Conference, July 25–29, 2006, Ljubljana, Slovenia.

References

BAKER, W.E. (1986): Three-dimensional blockmodels. *Journal of Mathematical Sociology, 12, 191-223.*

BATAGELJ, V., FERLIGOJ, A. and DOREIAN, P. (1992): Direct and indirect methods for structural equivalence. *Social Networks, 14, 63-90.*

BATAGELJ, V. and MRVAR, A.(1996-): Pajek – Program for analysis and visualization of large network.
`http://vlado.fmf.uni-lj.si/pub/networks/pajek/`.

BATAGELJ, V. (2006): ibm3m – A package for indirect 3D structural blockmodeling.
`http://vlado.fmf.uni-lj.si/pub/networks/progs/R/ibm3m.zip`.

DE NOOY, W., MRVAR, A. and BATAGELJ, V. (2005): *Exploratory Social Network Analysis with Pajek.* Cambridge University Press.

DOREIAN, P., BATAGELJ, V. and FERLIGOJ, A. (2005): *Generalized Blockmodeling.* Cambridge University Press.

DOREIAN, P., BATAGELJ, V. and FERLIGOJ, A. (2004): Generalized blockmodeling of two-mode network data. *Social Networks, 26, 29-53.*

EVERETT, M.G. and BORGATTI, S.P. (1992): Regular blockmodels of multiway, multimode matrices (in the print version it has a wrong title Regular Colouring of Digraphs, Networks and Hypergraphs). *Social Networks, 14, 91-120.*

KAUFMAN, L. and ROUSSEEUW, P.J. (1990): *Finding Groups in Data. An Introduction to Cluster Analysis.* Wiley, New York.

King / Mage: `http://kinemage.biochem.duke.edu/`.

KRACKHARDT, D. (1987): Cognitive social structures. *Social Networks, 9, 109-134.*

R-project: `http://www.r-project.org/`.

Clustering Methods: A History of *k*-Means Algorithms

Hans-Hermann Bock

Institute of Statistics, RWTH Aachen University, D-52056 Aachen, Germany,
bock@stochastik.rwth-aachen.de

Abstract. This paper surveys some historical issues related to the well-known k-means algorithm in cluster analysis. It shows to which authors the different versions of this algorithm can be traced back, and which were the underlying applications. We sketch various generalizations (with references also to Diday's work) and thereby underline the usefulness of the k-means approach in data analysis.

1 Introduction

Cluster analysis was a main topic in the beginning of Edwin Diday's scientific career. In fact, the monograph 'Principles of numerical taxonomy' by Sokal and Sneath (1963) motivated world-wide research on clustering methods and initiated the publication of books such as 'Les bases de la classification automatique' (Lerman (1970)), 'Mathematical taxonomy' (Jardine and Sibson (1971)), 'Cluster analysis for applications' (Anderberg (1973)), 'Cluster analysis' (Bijnen (1973)), 'Automatische Klassifikation' (Bock (1974)), 'Empirische Verfahren zur Klassifikation' (Sodeur (1974)), 'Probleme und Verfahren der numerischen Klassifikation (Vogel (1975)), 'Cluster-Analyse-Algorithmen (Späth (1975, 1985)), and 'Clustering algorithms' (Hartigan (1975)). With the consequence that the basic problems and methods of clustering became well-known in a broad scientific community, in statistics, data analysis, and - in particular - in applications.

One of the major clustering approaches is based on the sum-of-squares criterion and on the algorithm that is today well-known under the name 'k-means'. When tracing back this algorithm to its origins, we see that it has been proposed by several scientists in different forms and under different assumptions. Later on, many researchers investigated theoretical and algorithmic aspects and modifications of the method, e.g., when considering 'continuous' analogues of the SSQ criterion (Cox (1957), Fisher (1958), Bock (1974)), by investigating the asymptotic behaviour under random sampling strategies (Hartigan (1975), Pollard (1982), Bock (1985)), and by extending its domain to new data types and probabilistic models. Certainly, Diday's monograph (Diday et al. 1979), written with 22 co-authors, marks a considerable level of generalization of the basic idea and established its usage for model-based clustering.

This article surveys the origins and some extensions of the k-means algorithm. In Section 2 we formulate the SSQ clustering problem and the *k-means algorithm.* Section 3 describes the most early papers proposing the SSQ criterion and the k-means algorithm. Section 4 concentrates on extensions of the SSQ criterion that lead to *generalized k-means algorithms.* Section 5 deals with one- and two-parameter criteria and shows how a 'convexity-based' clustering criterion can be minimized with a *k-tangent algorithm.*

2 k-means clustering for the SSQ criterion

There are two versions of the well-known SSQ clustering criterion: the 'discrete' and the 'continuous' case.

Discrete SSQ criterion for data clustering: Given n data points $x_1, ..., x_n$ in $\mathbb{R}^p$ and a k-partition $\mathcal{C} = (C_1, ..., C_k)$ of the set $\mathcal{O} = \{1, ..., n\}$ of underlying 'objects' with non-empty classes $C_i \subset \mathcal{O}$, the discrete SSQ criterion (also termed: variance criterion, inertia, or trace criterion) is given by

$$g_n(\mathcal{C}) := \sum_{i=1}^{k} \sum_{\ell \in C_i} ||x_\ell - \overline{x}_{C_i}||^2 \ \rightarrow \ \min_{\mathcal{C}} \tag{1}$$

where $\overline{x}_{C_i}$ denotes the centroid of the data points x_ℓ 'belonging' to class C_i (i.e. with $\ell \in C_i$). We look for a k-partition of $\mathcal{O}$ with minimum criterion value $g_n(\mathcal{C})$. The one-parameter optimization problem (1) is related, and even equivalent, to the two-parameter optimization problem

$$g_n(\mathcal{C}, \mathcal{Z}) := \sum_{i=1}^{k} \sum_{k \in C_i} ||x_\ell - z_i||^2 \ \rightarrow \ \min_{\mathcal{C}, \mathcal{Z}} \tag{2}$$

where minimization is also w.r.t. all systems $\mathcal{Z} = (z_1, ..., z_k)$ of k points $z_1, ..., z_k$ from $\mathbb{R}^p$ (class representatives, class prototypes). This results from part (i) of the following theorem:

Theorem 1:
(i) For any fixed k-partition $\mathcal{C}$ the criterion $g_n(\mathcal{C}, \mathcal{Z})$ is partially minimized w.r.t. $\mathcal{Z}$ by the system of class centroids $\mathcal{Z}^* = (\overline{x}_{C_1}, ..., \overline{x}_{C_k}) =: \mathcal{Z}(\mathcal{C})$:

$$g_n(\mathcal{C}, \mathcal{Z}) \geq g_n(\mathcal{C}, \mathcal{Z}^*) = \sum_{i=1}^{k} \sum_{k \in C_i} ||x_k - \overline{x}_{C_i}||^2 \ = g_n(\mathcal{C}) \quad \textit{for all } \mathcal{Z}. \tag{3}$$

(ii) For any fixed prototype system $\mathcal{Z}$ the criterion $g_n(\mathcal{C}, \mathcal{Z})$ is partially minimized w.r.t. $\mathcal{C}$ by any minimum-distance partition $\mathcal{C}^* := (C_1^*, ..., C_k^*) =: \mathcal{C}(\mathcal{Z})$ induced by $\mathcal{Z}$, i.e. with classes given by $C_i^* := \{\ell \in \mathcal{O} \mid d(x_\ell, z_i) =$

$\min_{j=1,...,k} d(x_\ell, z_j)\}$ *($i = 1, ..., n$) where $d(x,z) = ||x-z||^2$ is the squared Euclidean distance:*

$$g_n(\mathcal{C}, \mathcal{Z}) \geq g_n(\mathcal{C}^*, \mathcal{Z}) = \sum_{\ell=1}^{n} \min_{j=1,...,k} \{ \; ||x_\ell - z_j||^2 \; \} \qquad \textit{for all } \mathcal{C}. \tag{4}$$

A broad range of methods has been designed in order to minimize the discrete criteria (1) and (2), either exactly or approximately. They can be roughly grouped into enumeration methods, mathematical and combinatorial programming for exact minimization (Hansen and Jaumard (1997), Grötschel and Wakabayashi (1989)), integer, linear, and dynamic programming (Jensen (1969), Vinod (1969), Rao (1971)), heuristical and branch & bound methods (see also Anderberg (1973), Mulvey and Crowder (1979)).

The *k-means algorithm* tries to approximate an optimum k-partition by iterating the partial minimization steps (i) and (ii) from Theorem 1, in turn. It proceeds as follows[1]:

$t = 0$: Begin with an arbitrary prototype system $\mathcal{Z}^{(0)} = (z_1^{(0)}, ..., z_k^{(0)})$.

$t \to t+1$:

(i) Minimize the criterion $g_n(\mathcal{C}, \mathcal{Z}^{(t)})$ w.r.t. the k-partition $\mathcal{C}$, i.e., determine a minimum-distance partition $\mathcal{C}^{(t+1)} := \mathcal{C}(\mathcal{Z}^{(t)})$.

(ii) Minimize the criterion $g_n(\mathcal{C}^{(t+1)}, \mathcal{Z})$ w.r.t. the prototype system $\mathcal{Z}$, i.e., calculate the system of class centroids $\mathcal{Z}^{(t+1)} := \mathcal{Z}(\mathcal{C}^{(t+1)})$.

Stopping: Iterate the steps (i) and (ii) until stationarity.

By construction, this algorithm yields a sequence $\mathcal{Z}^{(0)}, \mathcal{C}^{(1)}, \mathcal{Z}^{(1)}, \mathcal{C}^{(2)}, ...$ of prototypes and partitions with decreasing values of the criteria (1) and (2) that converge to a (typically local) minimum value.

Remark 1: In mathematical terms, the k-means algorithm is a *relaxation method* for minimizing a function of several parameters by iterative partial minimization steps (see also Mulvey and Crowder 1979), and also called an *alternating optimization* method.

Continuous SSQ criterion for space dissection: Considering $x_1, ..., x_n$ as realizations of a random vector X with distribution P in $I\!R^p$, we may formulate the following 'continuous' analogues of (1) and (2): We look for a k-partition $\mathcal{B} = (B_1, ..., B_k)$ of $I\!R^p$ with minimum value

$$g(\mathcal{B}) := \sum_{i=1}^{k} \int_{B_i} ||x - E[X|X \in B_i]||^2 \; dP(x) \;\; \to \;\; \min_{\mathcal{B}}. \tag{5}$$

As before we can relate (5) to a two-parameter optimization problem:

$$g(\mathcal{B}, \mathcal{Z}) := \sum_{i=1}^{k} \int_{B_i} ||x - z_i||^2 \; dP(x) \;\; \to \;\; \min_{\mathcal{B}, \mathcal{Z}} \tag{6}$$

[1] This is the *batch version* of the k-means algorithm; see *Remark 2*.

and formulate the analogue of Theorem 1:

Theorem 2:
(i) For any fixed k-partition $\mathcal{B}$ of $I\!R^p$ the criterion $g(\mathcal{B},\mathcal{Z})$ is partially minimized w.r.t. $\mathcal{Z}$ by the prototype system $\mathcal{Z}^* = (z_1^*,...,z_k^*) =: \mathcal{Z}(\mathcal{B})$ given by the conditional expectations $z_i^* := E[X|X \in B_i]$ of B_i:

$$g(\mathcal{B},\mathcal{Z}) \geq g(\mathcal{B},\mathcal{Z}^*) = \sum_{i=1}^{k} \int_{B_i} ||x - E[X|X \in B_i]||^2 \ = g(\mathcal{B}) \quad \textit{for all } \mathcal{Z}. \tag{7}$$

(ii) For any fixed prototype system $\mathcal{Z}$ the criterion $g(\mathcal{B},\mathcal{Z})$ is partially minimized w.r.t. $\mathcal{B}$ by any minimum-distance partition $\mathcal{B}^* = (B_1^*,...,B_k^*) =: \mathcal{B}(\mathcal{Z})$ generated by $\mathcal{Z}$, i.e. with classes given by $B_i^* := \{x \in I\!R^p \mid d(x,z_i) = \min_{j=1,...,k}\{d(x,z_j)\}\ \}$ $(i = 1,...,n)$:

$$g(\mathcal{B},\mathcal{Z}) \geq g(\mathcal{B}^*,\mathcal{Z}) = \int_{\mathcal{X}} \min_{j=1,...,k}\{||x - z_j||^2\}\ dP(x) \quad \textit{for all } \mathcal{B}. \tag{8}$$

It is obvious that Theorem 2 can be used to formulate, and justify, a continuous version of the k-means algorithm. However, in contrast to the discrete case, the calculation of the class centroids might be a computational problem.

3 First instances of SSQ clustering and *k*-means

The first formulation of the SSQ clustering problem I know has been provided by Dalenius (1950) and Dalenius and Gurney (1951) in the framework of optimum 'proportional' stratified sampling: For estimating the expectation $\mu = E[X]$ of a real-valued random variable X with distribution density $f(x)$ (e.g., the income of persons in a city), the domain $(-\infty,+\infty)$ of X is dissected into k contiguous intervals ('strata', 'classes') $B_i = (u_{i-1},u_i]$ $(i = 1,...,k+1$, with $u_0 = -\infty$ and $u_{k+1} = \infty)$ and from each stratum B_i a fixed number n_i of persons is sampled where $n_i = n \cdot P(B_i)$ is proportional to the probability mass of B_i. This yields n real data $x_1,...,x_n$. The persons ℓ with income value x_ℓ in B_i build a class C_i with class average $z_i^* := \overline{x}_{C_i}$ $(i = 1,...,k)$. The linear combination $\hat{\mu} := \sum_{i=1}^{k}(n_i/n) \cdot \overline{x}_{C_i}$ provides an unbiased estimator of μ with variance given by the SSQ criterion: $Var(\hat{\mu}) = g(\mathcal{B})/n$. Dalenius wants to determine a k-partition $\mathcal{B}$ with minimum variance, i.e., maximum accuracy for $\hat{\mu}$ – this means the continuous clustering problem (5).

Dalenius did not use a k-means algorithm for minimizing (5), but a 'shooting' algorithm that is based on the fact that for an optimum partition $\mathcal{B}$ of $I\!R^1$ the class boundaries u_i must necessarily lie midway between the neigbouring class centroids such that $u_i = (z_i^* + z_{i+1}^*)/2$ or $z_{i+1}^* = 2u_i - z_i^*$ must hold for $i = 1,...,k-1$. Basically, he constructs a sequence $z_1 < u_1 < z_2 < u_2 < \cdots$ of centers and boundaries by

– choosing, for $i = 1$, an initial value $z_1 \in I\!R^1$

- determining, for $i = 1$, the upper boundary u_i of $B_i = (u_{i-1}, u_i]$ from the equation $E[X|X \in B_i] = [\int_{u_{i-1}}^{u_i} xf(x)dx]/[\int_{u_{i-1}}^{u_i} f(x)dx] \overset{!}{=} z_i$ (the expectation is an increasing function of u_i)
- then calculating the next centroid by $z_{i+1} = 2u_i - z_i$
- and iterating for $i = 2, 3, ..., k$.

By trial and error, the initial value z_1 is adapted such that the iteration stops with k classes and the k-th upper boundary $u_k = \infty$. A 'data version' of this approach for minimizing (1) has been described, e.g., by Strecker (1957), Stange (1960), and Schneeberger (1967).

Steinhaus (1956) was the first to propose explicitly the k-means algorithm in the multidimensional case. His motivation stems from mechanics (even if he refers also to examples from anthropology and industry): to partition a heterogeneous solid $\mathcal{X} \subset \mathbb{R}^p$ with internal mass distribution $f(x)$ into k subsets $B_1, ..., B_k$ and to minimize (6), i.e., the sum of the partial moments of inertia with respect to k points $z_1, ..., z_k \in \mathbb{R}^p$ by a suitable choice of the partition $\mathcal{B}$ and the z_i's. He does not only describe the (continuous version of the) k-means algorithm, but also discusses the existence of a solution for (6), its uniqueness ('minimum parfait', examples and counterexamples), and the behaviour of the sequence of minimum SSQ values for $k \to \infty$.

The first to propose the discrete k-means algorithm for clustering data, i.e., for solving (1), was Forgy (1965)[2], Jancey (1966a) was the first to mention it explicitly in a publication (see also Jancey (1966b)). The k-means method became a standard procedure in clustering and is known under quite different names such as *nuées dynamiques* (Diday 1971, 1972), *dynamic clusters method* (Diday 1973; Diday and Schroeder 1974a), *iterated minimum-distance partition method* (Bock 1974), *nearest centroid sorting* (Anderberg 1973), etc.

Remark 2: The name 'k-means algorithm' was first used by MacQueen (1967), but not for the 'batch algorithm' from Section 2. Instead he used it for his sequential, 'single-pass' algorithm for (asymptotically) minimizing the continuous SSQ criterion (5) on the basis of a sequence of data points $x_1, x_2, ... \in \mathbb{R}^p$ (sampled from P): The first k data (objects) defined k initial singleton classes $C_i^{(k)} = \{i\}$ with class centroids $z_i^{(k)} := \overline{x}_{C_i^{(k)}} = x_i$ $(i = 1, ..., k)$. Then, for $\ell = k+1, k+2, ...$, the data x_ℓ were sequentially observed and assigned to the class $C_i^{(\ell-1)}$ with closest class centroid $z_i^{(\ell-1)} := \overline{x}_{C_i^{(\ell-1)}}$ and (only) its class centroid was updated: $z_i^{(\ell)} := \overline{x}_{C_i^{(\ell)}} = z_i^{(\ell-1)} + (x_\ell - \overline{x}_{C_i^{(\ell-1)}})/|C_i^{(\ell)}|$. When stopping at some 'time' T, the minimium-distance partition $\mathcal{B}(\mathcal{Z}^{(T)})$ of $\mathbb{R}^p$ induced by the last centroid system $\mathcal{Z}^{(T)} = (\overline{x}_{C_1^{(T)}}, ..., \overline{x}_{C_k^{(T)}})$ approximates a (local) solution of (5) if T is large. This single-pass interpretation of 'k-means

[2] Forgy's abstract of his talk does not mention the k-means algorithm, however, details of his lecture were given by Anderberg (1973), p. 161 and MacQueen (1967) p. 294.

algorithm' is used in many monographs. – In Späth (1975) the batch-version of k-means is called HMEANS, whereas KMEANS denotes an algorithm that exchanges single objects between classes in order to decrease (1). Hartigan (1975) uses the term 'k-means' for various algorithms working with k class centroids, e.g. for Späth's exchange algorithm (on page 85/86), and k-means as described in our Section 2 is one of several options mentioned on page 102 of Hartigan (1975) (see also Hartigan and Wong (1979)).

In computer science and pattern recognition communities the k-means algorithm is often termed *Lloyd's algorithm I.* Lloyd (1957) considers the continuous SSQ clustering criterion (6) in $\mathbb{R}^1$ in the context of pulse-code modulation: 'Quantization' means replacing a random (voltage) signal X by a discretized approximate signal $\hat{X}$ that takes a constant value z_i ('quantum') if X belongs to the i-th class B_i of the partition $\mathcal{B} = (B_1, ..., B_k)$ of $\mathbb{R}^1$ such that $\hat{X} = z_i$ iff $X \in B_i$ $(i = 1, ..., k)$. Optimum quantification means minimization of the criterion (6). Lloyd reports the optimality of the class centroids $z_i^* = E[X|X \in B_i]$ for a fixed partition $\mathcal{B}$ and describes the one-dimensional version of the k-means algorithm as his 'Method I' whereas his 'Method II' is identical to the 'shooting method' of Dalenius.

4 Generalized *k*-means methods

The two-parameter SSQ clustering criteria (2) and (6) have been generalized in many ways in order to comply with special data types or cluster properties. In the discrete case, typical criteria have the two-parameter form

$$g_n(\mathcal{C}, \mathcal{Z}) := \sum_{i=1}^{k} \sum_{\ell \in C_i} d(\ell, z_i) \to \min_{\mathcal{C}, \mathcal{Z}} \tag{9}$$

where $d(\ell, z)$ measures the dissimilarity between an object ℓ and a class prototype z (sometimes written as $d(x_\ell, z)$ or $d_{\ell z}$ etc., depending on the context). There is much flexibility in this approach since

(1) there is almost no constraint on the type of underlying data (quantitative and/or categorical data, shapes, relations, weblogs, DNA strains, images)
(2) there are many ways to specify a family $\mathcal{P}$ of appropriate or admissible 'class prototypes' z to represent specific aspects of the clusters (points, hyperspaces in $\mathbb{R}^p$, subsets of $\mathcal{O}$, order relations),
(3) there exists a wealth of possibilities to choose the dissimilarity measure d, and we may, additionally, introduce weights w_ℓ for the objects $\ell \in \mathcal{O}$.

In all these cases, the following *generalized k-means algorithm* can be applied in order to attain a (locally or globally) optimum configuration $(\mathcal{C}, \mathcal{Z})$:

$t = 0$: Begin with an arbitrary prototype system $\mathcal{Z}^{(0)} = (z_1^{(0)}, ..., z_k^{(0)})$.

$t \to t + 1$:

(i) Minimize the criterion $g_n(\mathcal{C}, \mathcal{Z}^{(t)})$ w.r.t. the k-partition $\mathcal{C}$ from $\mathcal{P}$. Typically, this yields a minimum-distance partition $\mathcal{C}^{(t+1)} = \mathcal{C}(\mathcal{Z}^{(t)})$ with k classes $C_i^{(t+1)} := \{\ell \in \mathcal{O} \mid d(\ell, z_i^{(t)}) = \min_{j=1,...,k} d(\ell, z_j^{(t)})\ \}$.

(ii) Minimize the criterion $g_n(\mathcal{C}^{(t+1)}, \mathcal{Z})$ w.r.t. the prototype system $\mathcal{Z}$. Often, this amounts to determining, for each class $C_i = C_i^{(t+1)}$, a 'most typical configuration' $z_i^{(t+1)}$ in the sense:

$$Q(C_i, z) := \sum_{\ell \in C_i} d(\ell, z) \quad \rightarrow \quad \min_{z \in \mathcal{P}}. \tag{10}$$

Stopping: Iterate the steps (i) and (ii) until stationarity.

The first paper to propose the general criterion (9) and its generalized k-means method is Maranzana (1963): He starts from a $n \times n$ dissimilarity matrix $(d_{\ell t})$ for n factories $\ell = 1, ..., n$ in an industrial network where $d_{\ell t}$ are the minimum road transportation costs between ℓ and t. He wants to partition the set of factories into k classes $C_1, ..., C_k$ and to find a selection $\mathcal{Z} = (z_1, ..., z_k)$ of k factories as 'supply points' such that when supplying all factories of the class C_i from the supply point $z_i \in \mathcal{O}$, the overall transport costs are minimized in the sense of (9) where $d(\ell, z_i) = d_{\ell, z_i}$ means the dissimilarity between the factory (object) ℓ and the factory (supply point) $z_i \in \mathcal{O}$ (where we have omitted object-specific weights from Maranzana's formulation). So the family $\mathcal{P}$ of admissible prototypes consists of all singletons from $\mathcal{O}$ and (ii) means determining the 'most cheapest supply point' in C_i. Kaufman and Rousseeuw (1987, 1990) termed this method 'partitioning around medoids' (the *medoid* or *centrotype* of a class C_i is the most typical object in C_i in the sense of (10)).

Many authors, including Diday (1971, 1972, 1973) and Diday et al. (1979), have followed the generalized clustering approach via (9) in various settings and numerous variations and thereby obtained a plethora of generalized k-means algorithms, e.g., by

- using Mahalanobis or L_q distance in (1) instead of the Euclidean one, eventually including constraints (Diday and Govaert (1974, 1977): *méthode des distances adaptatives*)
- characterizing clusters by prototype hyperplanes, resulting in *principal component clustering* (Bock (1974) chap. 17, Diday and Schroeder (1974a)) and *clusterwise regression* (Bock (1969), Charles (1977), Späth (1979)).
- *projection pursuit clustering* where class centers are located on a low-dimensional hyperplane (Bock (1987, 1996c), Vichi (2005)),
- characterizing a class by the most typical subset (pair, triple,...) of objects from this class (Diday et al. (1979)).

A major step with new insight was provided by Diday and Schroeder (1974a, 1974b, 1976) and Sclove (1977) who detected that under a probabilistic 'fixed-partition' clustering model, maximum-likelihood estimation of an unknown k-partition $\mathcal{C}$ leads to a clustering criterion of the type (9) and can therefore be handled by a k-means algorithm[3]. The *fixed-partition model* considers the data $x_1, ..., x_n$ as realizations of n independent random vectors $X_1, ..., X_n$ with distributions from a density family $f(\cdot; \vartheta)$ (w.r.t. the Lebesgue or counting measure) with parameter ϑ (e.g., a normal, van Mises, loglinear,... distribution). It assumes the existence of a fixed, but unknown k-partition $\mathcal{C} = (C_1, ..., C_k)$ of $\mathcal{O}$ together with a system $\theta = (\vartheta_1, ..., \vartheta_k)$ of class-specific parameters such that the distribution of the data is class-specific in the sense that $X_\ell \sim f(\cdot; \vartheta_i)$ for all $\ell \in C_i$ $(i = 1, ..., n)$. Then maximizing the likelihood of $(x_1, ..., x_n)$ is equivalent to

$$g_n(\mathcal{C}, \theta) \;\; := \;\; \sum_{i=1}^{k} \sum_{\ell \in C_i} [-\log f(x_\ell; \vartheta_i)] \;\; \rightarrow \;\; \min_{\mathcal{C}, \theta}, \tag{11}$$

this is the criterion (9) with $z_i \equiv \vartheta_i, \mathcal{Z} \equiv \theta$, and $d(\ell, z_i) = -\log f(x_\ell; \vartheta_i)$. The minimum-distance assignment of an object ℓ in (i) means maximum-likelihood assignment to a class C_i, and in (ii) optimum class prototypes are given by the maximum-likelihood estimate $\hat{\vartheta}_i$ of $z_i \equiv \vartheta_i$ in C_i. A major advantage of this approach resides in the fact that we can design meaningful clustering criteria also in the case of qualitative or binary data, yielding, *entropy clustering* and *logistic clustering* methods (Bock 1986), or models comprizing random noise or outliers (Gallegos (2002), Gallegos and Ritter (2005)). – A detailed account of these approaches is given, e.g., in Bock (1974, 1996a, 1996b, 1996c) and Diday et al. (1979).

5 Convexity-based criteria and the *k*-tangent method

The derivation of the k-means algorithm in Section 2 shows that it relies on the fact that the intuitive SSQ optimization problem (1) for *one* parameter $\mathcal{C}$ has an equivalent version (2) where optimization is w.r.t. *two* parameters $\mathcal{C}$ and $\mathcal{Z}$. In order to extend the domain of applicability of the k-means algorithm we may ask, more generally, if for an intuitively defined one-parameter clustering criterion we can find a two-parameter version such that both resulting optimization problems are equivalent and a k-means algorithm can be applied. A general investigation of this problem has been given by Windham (1986, 1987) and Bryant (1988). In the following we describe a situation where the answer is affirmative and leads to a new *k-tangent algorithm* (Bock (1983, 1992, 2003), Pötzelberger and Strasser (2001).

We consider the following 'convexity-based' clustering criterion for $x_1, ..., x_n \in \mathbb{R}^p$ that should be maximized w.r.t the k-partition $\mathcal{C}$:

$$k_n(\mathcal{C}) \;\; := \;\; \sum_{i=1}^{k} (|C_i|/n) \cdot \phi(\overline{x}_{C_i}) \;\; \rightarrow \;\; \max_{\mathcal{C}} \tag{12}$$

[3] This fact was already known before, e.g., in the case of SSQ and the normal distribution, but these authors recognized its importance for more general cases.

Here $\phi(\cdot)$ is a smooth convex function, and (12) is a generalization of the SSQ clustering problem (1) since for $\phi(x) := ||x||^2$ (12) reduces to (1). Similarly, the continuous version

$$k(\mathcal{B}) \;:=\; \sum_{i=1}^{k} P(B_i)\cdot\phi(E[X|X\in B_i]) \;\to\; \max_{\mathcal{B}} \tag{13}$$

is equivalent to (5), its generalization

$$K(\mathcal{B}) \;=\; \sum_{i=1}^{k} P_0(B_i)\cdot\phi(E_0[\lambda(X)|X\in B_i])) \;\to\; \max_{\mathcal{B}} \tag{14}$$

looks for an optimum dissection of $\mathbb{R}^p$ such that, for two equivalent alternative distributions P_0, P_1 on $\mathbb{R}^p$ with likelihood ratio $\lambda(x) = (dP_1/dP_0)(x) = f_1(x)/f_0(x)$, the discretized distributions $(P_0(B_1),...,P_0(B_k))$ and $(P_1(B_1),..., P_1(B_k))$ will be as different as possible. (Note that K is Cszizar's ϕ-divergence and reduces, e.g., to Kullback-Leibler and χ^2 distance for $\phi(u) = -\log u$ and $\phi(u) = (u-1)^2$, respectively; for other functions λ see Bock (2003).) Some analysis based on the convexity of ϕ shows that maximizing $K(\mathcal{C})$ is equivalent to the two-parameter minimization problem

$$G(\mathcal{B},\mathcal{Z}) \;:=\; \sum_{i=1}^{k} \int_{B_i} [\phi(\lambda(x)) - t(\lambda(x); z_i)]\; dP_0(x) \;\to\; \min_{\mathcal{B},\mathcal{Z}} \tag{15}$$

where $\mathcal{Z} = (z_1,...,z_k) \in \mathbb{R}_+^k$ and $t(\lambda; z) := \phi(z) + \phi'(z)(\lambda - z)$ is the tangent (support plane) of $y = \phi(\lambda)$ in the support point $z > 0$ ([....] is the weighted 'volume' between the curve and the corresponding segments of the tangents). Therefore we can apply the alternating partial minimization device. The resulting method is termed 'k-tangent algorithm' and comprizes the steps:

(i) For a given support point system $\mathcal{Z}$, determine the 'maximum-tangent partition' $\mathcal{B}$ with classes defined by maximum tangent values:

$$B_i \;:=\; \{\; x\in\mathbb{R}^p \mid t(\lambda(x); z_i) = \max_{j=1,\ldots,k} t(\lambda(x); z_j) \;\} \tag{16}$$

(ii) For a given k-partition $\mathcal{B}$ determine the system $\mathcal{Z}$ of class-specific discrete likelihood ratios:

$$z_i \;:=\; E_0[\;\lambda(X)] \mid X\in B_i\;] \;=\; \frac{P_1(B_i)}{P_0(B_i)} \qquad i = 1,...,k. \tag{17}$$

Iteration of (i) and (ii) yields a sequence of partitions with decreasing values in (14). – Pötzelberger and Strasser (2001) investigate the theoretical properties of the optimum partitions of (12) and (13), Bock (2003) shows, e.g., how the k-tangent method can be applied to the simultaneous classification of the rows and columns of a contingency table.

References

ANDERBERG, M.R. (1973): *Cluster analysis for applications.* Academic Press, New York.

BIJNEN, E.J. (1973): *Cluster analysis.* Tilburg University Press, Tilburg, Netherlands.

BOCK, H.-H. (1969): *The equivalence of two extremal problems and its application to the iterative classification of multivariate data.* Paper presented at the Workshop 'Medizinische Statistik', February 1969, Forschungsinstitut Oberwolfach.

BOCK, H.-H. (1974): *Automatische Klassifikation. Theoretische und praktische Methoden zur Strukturierung von Daten (Clusteranalyse).* Vandenhoeck & Ruprecht, Göttingen.

BOCK, H.-H. (1985): On some significance tests in cluster analysis. *Journal of Classification 2, 77-108.*

BOCK, H.-H. (1983): *A clustering algorithm for choosing optimal classes for the chi-square test.* Bull. 44th Session of the International Statistical institute, Madrid, Contributed Papers, Vol 2, 758-762.

BOCK, H.-H. (1986): Loglinear models and entropy clustering methods for qualitative data. In: W. Gaul, M. Schader (Eds.): *Classification as a tool of research.* North Holland, Amsterdam, 19-26.

BOCK, H.-H. (1987): On the interface between cluster analysis, principal component analysis, and multidimensional scaling. In: H. Bozdogan, A.K. Gupta (Eds.): *Multivariate statistical modeling and data analysis.* Reidel, Dordrecht, 17-34.

BOCK, H.-H. (1992): A clustering technique for maximizing ϕ-divergence, noncentrality and discriminating power. In: M. Schader (Ed.): *Analyzing and modeling data and knowledge.* Springer, Heidelberg, 19-36.

BOCK, H.-H. (1996a): Probability models and hypotheses testing in partitioning cluster analysis. In: P. Arabie, L.J. Hubert, G. De Soete (Eds.): *Clustering and classification.* World Scientific, Singapore, 377-453.

BOCK, H.-H. (1996b): Probabilistic models in partitional cluster analysis. *Computational Statistics and Data Analysis 23, 5-28.*

BOCK, H.-H. (1996c): Probabilistic models in cluster analysis. In: A. Ferligoj, A. Kramberger (Eds.): *Developments in data analysis.* Proc. Intern. Conf. on 'Statistical data collection and analysis', Bled, 1994. FDV, Metodoloski zvezki, 12, Ljubljana, Slovenia, 3-25.

BOCK, H.-H. (2003): Convexity-based clustering criteria: theory, algorithms, and applications in statistics. *Statistical Methods & Applications 12, 293-317.*

BRYANT, P. (1988): On characterizing optimization-based clustering methods. *Journal of Classification 5, 81-84.*

CHARLES, C. (1977): *Regression typologique.* Rapport de Recherche no. 257. IRIA-LABORIA, Le Chesnay.

COX, D.R. (1957) Note on grouping. *J. Amer. Statist. Assoc. 52, 543-547.*

DALENIUS, T. (1950): The problem of optimum stratification I. *Skandinavisk Aktuarietidskrift 1950, 203-213.*

DALENIUS, T., GURNEY, M. (1951): The problem of optimum stratification. II. *Skandinavisk Aktuarietidskrift 1951, 133-148.*

DIDAY, E. (1971): Une nouvelle méthode de classification automatique et reconnaissance des formes: la méthode des nuées dynamiques. *Revue de Statistique Appliquée XIX (2), 1970, 19-33.*

DIDAY, E. (1972): Optimisation en classification automatique et reconnaissance des formes. *Revue Française d'Automatique, Informatique et Recherche Opérationelle (R.A.I.R.O.) VI, 61-96.*

DIDAY, E. (1973): The dynamic clusters method in nonhierarchical clustering. *Intern. Journal of Computer and Information Sciences 2 (1), 61-88.*

DIDAY, E. et al. (1979): *Optimisation en classification automatique. Vol. I, II.* Institut National der Recherche en Informatique et en Automatique (INRIA), Le Chesnay, France.

DIDAY, E., GOVAERT, G. (1974): Classification avec distance adaptative. *Comptes Rendus Acad. Sci. Paris 278 A, 993-995.*

DIDAY, E., GOVAERT, G. (1977): Classification automatique avec distances adaptatives. *R.A.I.R.O. Information/Computer Science 11 (4), 329-349.*

DIDAY, E., SCHROEDER, A. (1974a): The dynamic clusters method in pattern recognition. In: J.L. Rosenfeld (Ed.): *Information Processing 74.* Proc. IFIP Congress, Stockholm, August 1974. North Holland, Amsterdam, 691-697.

DIDAY, E., SCHROEDER, A. (1974b): *A new approach in mixed distribution detection.* Rapport de Recherche no. 52, Janvier 1974. INRIA, Le Chesnay.

DIDAY, E., SCHROEDER, A. (1976): A new approach in mixed distribution detection. *R.A.I.R.O. Recherche Opérationelle 10 (6), 75-1060.*

FISHER, W.D. (1958): On grouping for maximum heterogeneity. *J. Amer. Statist. Assoc. 53, 789-798.*

FORGY, E.W. (1965): Cluster analysis of multivariate data: efficiency versus interpretability of classifications. Biometric Society Meeting, Riverside, California, 1965. Abstract in *Biometrics 21 (1965) 768.*

GALLEGOS, M.T. (2002): Maximum likelihood clustering with outliers. In: K. Jajuga, A. Sokolowski, H.-H. Bock (Eds.): *Classification, clustering, and data analysis.* Springer, Heidelberg, 248-255.

GALLEGOS, M.T., RITTER, G. (2005): A robust method for cluster analysis. *Annals of Statistics 33, 347-380.*

GRÖTSCHEL, M., WAKABAYASHI, Y. (1989): A cutting plane algorithm for a clustering problem. *Mathematical Programming 45, 59-96.*

HANSEN, P., JAUMARD, B. (1997): Cluster analysis and mathematical programming. *Mathematical Programming 79, 191-215.*

HARTIGAN, J.A. (1975): *Clustering algorithms.* Wiley, New York.

HARTIGAN, J.A., WONG, M.A. (1979): A k-means clustering algorithm. *Applied Statistics 28, 100-108.*

JANCEY, R.C. (1966a): Multidimensional group analysis. *Australian J. Botany 14, 127-130.*

JANCEY, R. C. (1966b): The application of numerical methods of data analysis to the genus Phyllota Benth. in New South Wales. *Australian J. Botany 14, 131-149.*

JARDINE, N., SIBSON, R. (1971): *Mathematical taxonomy.* Wiley, New York.

JENSEN, R.E. (1969): A dynamic programming algorithm for cluster analysis. *Operations Research 17, 1034-1057.*

KAUFMAN, L., ROUSSEEUW, P.J. (1987): Clustering by means of medoids. In: Y. Dodge (Ed.): *Statistical data analysis based on the L_1-norm and related methods.* North Holland, Amsterdam, 405-416.

KAUFMAN, L., ROUSSEEUW, P.J. (1990): *Finding groups in data.* Wiley, New York.

LERMAN, I.C. (1970): *Les bases de la classification automatique.* Gauthier-Villars, Paris.

LLOYD, S.P. (1957): Least squares quantization in PCM. Bell Telephone Labs Memorandum, Murray Hill, NJ. Reprinted in: *IEEE Trans. Information Theory IT-28 (1982), vol. 2, 129-137.*

MacQUEEN, J. (1967): Some methods for classification and analysis of multivariate observations. In: L.M. LeCam, J. Neyman (eds.): *Proc. 5th Berkeley Symp. Math. Statist. Probab. 1965/66.* Univ. of California Press, Berkeley, vol. I, 281-297.

MARANZANA, F.E. (1963): On the location of supply points to minimize transportation costs. *IBM Systems Journal 2, 129-135.*

MULVEY, J.M., CROWDER, H.P. (1979): Cluster analysis: an application of Lagrangian relaxation. *Management Science 25, 329-340.*

PÖTZELBERGER, K., STRASSER, H. (2001): Clustering and quantization by MSP partitions. *Statistics and Decision 19, 331-371.*

POLLARD, D. (1982): A central limit theorem for k-means clustering. *Annals of Probability 10, 919-926.*

RAO, M.R. (1971): Cluster analysis and mathematical programming. *J. Amer. Statist. Assoc. 66, 622-626.*

SCHNEEBERGER, H. (1967): Optimale Schichtung bei proportionaler Aufteilung mit Hilfe eines iterativen Analogrechners. *Unternehmensforschung 11, 21-32.*

SCLOVE, S.L. (1977): Population mixture models and clustering algorithms. *Commun. in Statistics, Theory and Methods, A6, 417-434.*

SODEUR, W. (1974): *Empirische Verfahren zur Klassifikation.* Teubner, Stuttgart.

SOKAL, R.R., SNEATH, P. H. (1963): *Principles of numerical taxonomy.* Freeman, San Francisco - London.

SPÄTH, H. (1975): *Cluster-Analyse-Algorithmen zur Objektklassifizierung und Datenreduktion.* Oldenbourg Verlag, München - Wien.

SPÄTH, H. (1979): Algorithm 39: Clusterwise linear regression. *Computing 22, 367-373.* Correction in *Computing 26 (1981), 275.*

SPÄTH, H. (1985): *Cluster dissection and analysis.* Wiley, Chichester.

STANGE, K. (1960): Die zeichnerische Ermittlung der besten Schätzung bei proportionaler Aufteilung der Stichprobe. *Zeitschrift für Unternehmensforschung 4, 156-163.*

STEINHAUS, H. (1956): Sur la division des corps matériels en parties. *Bulletin de l'Académie Polonaise des Sciences, Classe III, vol. IV, no. 12, 801-804.*

STRECKER, H. (1957): *Moderne Methoden in der Agrarstatistik.* Physica, Würzburg, p. 80 etc.

VICHI, M. (2005): Clustering including dimensionality reduction. In: D. Baier, R. Decker, L. Schmidt-Thieme (Eds.): *Data analysis and decision support.* Springer, Heidelberg, 149-156.

VINOD, H.D. (1969): Integer programming and the theory of grouping. *J. Amer. Statist. Assoc. 64, 506-519.*

VOGEL, F. (1975): *Probleme und Verfahren der Numerischen Klassifikation.* Vandenhoeck & Ruprecht, Göttingen.

WINDHAM, M.P. (1986): A unification of optimization-based clustering algorithms. In: W. Gaul, M. Schader (Eds.): *Classification as a tool of research.* North Holland, Amsterdam, 447-451.

WINDHAM, M.P. (1987): Parameter modification for clustering criteria. *Journal of Classification 4, 191-214.*

Overlapping Clustering in a Graph Using *k*-Means and Application to Protein Interactions Networks

Irène Charon[1], Lucile Denœud[1,2], and Olivier Hudry[1]

[1] ENST, 46, rue Barrault, 75634 Paris cedex 13, France
{*Irene.Charon,Olivier.Hudry,Lucile.Denoeud*}*@enst.fr*
[2] CERMSEM Paris I, 106-112, boulevard de l'Hôpital 75013 Paris, France.

Abstract. In this article, we design an overlapping clustering method in a graph in order to deal with a biological issue: the proteins annotation. Given an unweighted and undirected graph G, we search for subgraphs of G that are dense in edges. The method consists in three steps. First we determine some intial kernels of the classes by means of a local density function; then we improve these kernels using a k-means process; last the kernels are extended to overlapping classes. The method is tested on random graphs and finally applied to a protein interactions network.

1 Introduction

This work deals with a tricky problem of overlapping clustering (Arabie (1996), Brossier (2003), Hansen (1997)). We consider an unweighted and undirected graph $G = (X, E)$, each edge representing a similarity between its extremities. Our objective is to detect some areas with high density of edges, corresponding to a set of similar vertices. Since we want to find some natural classes, we relax the too strict partition constraints and we accept a vertex to be in several classes as well as in none.

The resolution of this problem can contribute to solve a current and fundamental biological issue: to understand the cellular mechanisms identifying the involved proteins and their different roles in these processes. The protein interaction networks are unweighted and undirected graphs in which the vertices represent proteins and there is an edge between two vertices if the corresponding proteins are known to interact. Since the proteins involved in a same cellular function interact, it is likely that if two proteins have a lot of common interactors in the protein interaction network, then their functions should be related (Brun et al. (2002, 2004)). The detection of high density areas in such a graph should therefore allow the annotation of some proteins of unknown functions by assigning to them common functions of the proteins of the same class. Since proteins may have several distinct functions, it is justified to look for overlapping classes.

The proposed method is an extension of the work proposed in Colombo et al. (2003) and consists in two steps. The first determines the number of classes and builds disjoint kernels of the classes. Some initial kernels are given by the mean of a local density function (it is precisely explained in Colombo and Guénoche (submitted)). Then we use an adaptation of the method of k-means, developped in France by E. Diday (1971), under the French name of *nuées dynamiques*), in order to improve these kernels. During the second step, the kernels are extended to overlapping classes following some criterion related to the quality of the classes.

In Section 2, we present the method more precisely. Then we prove its efficiency by applying it on random graphs containing some classes in Section 3. Section 4 is dedicated to an application of the method to a real protein interaction network. Finally, we conclude in Section 5.

2 Presentation of the method

The method consists in two main steps. The first one constructs the kernels of the classes, the second extends these kernels to overlapping classes. In this section, we present these two steps in details.

2.1 Creation of the kernels

In order to create the kernels of the classes, we first find some initial kernels by means of a local density function defined on the vertices of the graph. Then this solution is improved by a k-means algorithm, allowing the modification of the number of kernels.

Initial kernels We use a method of classification by density (introduced by Wishart (1976)) in order to build initial kernels of the classes. Let $d(s)$ be the degree of the vertex s, $N_t(s)$ the number of triangles in the neighbourhood of s (number of edges between two neighbours of s) and Δ the maximum degree in the graph. Let $\Gamma(s)$ be the set of the neighbours of s (notice the equality $|\Gamma(s)| = d(s)$). We define the function De on the set X by:

- if $d(s) > 1$,

$$De(s) = \frac{N_t(s)}{\frac{1}{2}\ \Delta\ (d(s) - 1)}$$

- otherwise,

$$De(s) = 0.$$

(Other density functions have been tested in Denœud (2006); the function above is the one which seems to provide the best results.) The function De evaluates the local density of edges in the neighbourhood of any vertex. This function takes into account the degree of the vertex s and the percentage of edges in $\Gamma(s)$. Its values belong to $[0, 1]$. The value 1 corresponds to a vertex of degree Δ and such that all its neighbours are connected with each other: the subgraph of G induced by s and $\Gamma(s)$ is complete.

We use the function De in order to create the initial kernels of the classes. More precisely, we select the vertices s achieving local maxima of De and with a density larger than the average density of the graph, i.e. such that:

$$\forall s' \in \Gamma(s), De(s) \geq De(s') \text{ and } De(s) \geq \overline{De}$$

where $\overline{De}$ is the average density on the graph: $\overline{De} = \frac{1}{|X|}\sum_{s \in X} De(s)$. The initial kernels are the connected components of the subgraph induced by these vertices.

Improvement of the kernels We improve these initial kernels using a k-means method. Such a method permits to build partitions of sets of items and consists in the alternation of a recentring step (computation of the centers of the classes) and an allocation step (creation of a partition of the items by assigning each item to the closest center) (Diday (1971)). We adapt this principle in order to allow the modification of the number of classes and to relax the partitioning constraint: some vertices may not be clustered.

Let consider two vertices s and s' of the graph G. Let S (resp. S') denote the set $s \cup \Gamma(s)$ (resp. $s' \cup \Gamma(s')$). We consider the distance of Dice (1945), denoted $Dice$, defined on X^2 as:

$$Dice(s, s') = \frac{|(S \cup S') \setminus (S \cap S')|}{|S| + |S'|}.$$

(See Denœud (2006) for a study involving another distance.) The center c of a class C is chosen, according to this distance, as the vertex achieving the mimimum of the sum of the distance from the other vertices of the class: $c = \text{argmin}_{s \in C}(\sum_{s' \in C} Dice(s, s'))$. We remark that several vertices can sometimes be centers of a same class.

During the assignment step, we compute, for any vertex s and any class C, the average distance between s and the centers belonging to C. Then we assign s to the class achieving the minimum average distance, but only if it is unique. Otherwise the vertex s remains unclustered for the moment.

The modification of the number of classes is done by means of three processes, applied after the stabilisation of the solution in the k-means method:

- If the average distance between the centers of two classes is lower than or equal to the average distance within any class, the two classes merge.
- If a vertex is at maximum distance of any center (which implies that it has not been clustered during the previous allocation step), this vertex becomes a new center.
- We check the connectivity of the classes; if not connected, each connected component becomes a class.

The method consists then in the repetition of the k-means algorithm followed by these three processes until the stabilisation of the classes, or during a given number of iterations.

This first main step creates k disjoint classes, that will be extended in overlapping classes in the second main step. It is fundamental since it determines the number of classes and the kernels of the classes.

2.2 Extension of the classes

We consider a function evaluating the quality of a class. The principle of the extension is to deal with each class independently, adding iteratively to a class the vertices that are the most connected with it if the quality of the class increases. This quality function must depend on :

- the percentage of edges in the subgraph induced by the class;
- the cardinality of the class.

Indeed the objective of the method is to find high edge-density areas in the graph, but between two classes of equal density, we want to favour the largest class. It seems then immediate to consider the average degree as the quality function:

$$\bar{d}(C) = \frac{\sum_{s \in C} d(s)}{|C|} = \frac{2q}{p}$$

where p denotes the cardinality of C and q the number of edges between vertices of C. According to this criterion, a class C will be extended by the set S of candidate vertices (that is to say the set of vertices the most connected to the class C) if and only if:

$$\bar{d}(C \cup S) \geq \bar{d}(C).$$

Let e be the number of edges between any vertex of S and vertices of C (notice that e is the same for all the vertices of S). If $|S| = 1$, the average degree criterion can be rewritten as: $e \geq \frac{q}{p}$. We generalize the average degree criterion by considering the following extension rules, for any given positive factor α:

$$C \text{ becomes } C \cup S \text{ if and only if } e \geq \frac{\alpha q}{p}.$$

The greater α, the stricter the extension criterion. We obtain then a family of extension rules more or less strict according to the value of α, which could be chosen following the graph and the user requirements.

3 Validation of the method

In this section, we test the proposed method on random graphs that contain initial classes. First we present the method used to generate the random graphs; then we apply the first step of the classification method on these graphs and validate the k-means process. The second step is not tested in this section, but its behaviour will be analysed on a real protein interaction network in Section 4.

3.1 Generation of random graphs

In order to validate the method, we build some random graphs that contain some initial edge-dense classes (our process is close to the one developed in Colombo et al. (2003)). The graphs are generated following four parameters: n (number of vertices), nbc (number of classes), p_i (probability of initial edges) and p_r (probability of recabling an edge). The generation process is the following:

- we split the vertices uniformly the n vertices in the nbc classes;
- between any pair of vertices belonging to a same class, we set an edge with probability p_i;
- we reconnect each edge with a probability p_r (we exchange one extremity of the edge with a vertex randomly selected).

Finally we check the connectivity of the graph; if not, we add some edges until it becomes connected.

3.2 Validation of the kernels

In order to validate the first step and the use of k-means, we set n to 100, p_r to 0 and we vary p_i from 1 to 0.25 with a step of 0.25. The generated graphs have nearly no edges outside the classes, and the density within the classes decreases with p_i. We vary moreover the number of initial classes nbc which takes the values 2, 5, 10 and 20. For each set of parameters, we generate 100 graphs on which we apply the first step of the classification method.

In order to evaluate the quality of the kernels, we compute the average number of classes, the average percentage of clustered vertices and the average percentage of preserved pairs (percentage of pairs of vertices clustered together in the initial partition that are also together in the kernels). Table 1 gives the results obtained for the initial kernels (provided by the local function density) and the final kernels (after the application of the k-means method). We first observe that the lower the initial density of the classes and the greater the number of initial classes are, the more difficult it is to find the classes. Indeed the internal density of the corresponding classes is smaller and the classes themselves are less clear in the graph.

The initial creation of kernels does not produce the right number of kernels (except for the case $p_i = 1$). This number of classes is generally overestimated for $nbc = 2$ and $nbc = 5$ and underestimated for greater values of nbc.

The k-means process permits a significant improvement of the kernels, that are exactly or almost exactly found for $p_i = 1$, $p_i = 0.75$ with $nbc = 2, 5$ and 10, $p_i = 0.5$ with $nbc = 2$ and 5. The results remain good for the cases $p_i = 0.75$ with $nbc = 20$, $p_i = 0.5$ with $nbc = 10$ and $p_i = 0.25$ with $nbc = 2$ and 5. For the other cases, that correspond to fuzzier initial classes, the method gives satisfying results and the almost exact number of classes even if the initial kernels were bad (for instance, for the case $p_i = 0.25$ with

$p_i = 1$	$nbc = 2$	$nbc = 5$	$nbc = 10$	$nbc = 20$
initial kernels	2—98—100	5—92—100	10—82.8—100	19.9—67—100
final kernels	2—100—100	5—100—100	10—100—100	19.9—99.9—99.8
$p_i = 0.75$	$nbc = 2$	$nbc = 5$	$nbc = 10$	$nbc = 20$
initial kernels	2.04—47—100	4.99—49.9—100	8.8—51.9—99.8	15.4—51.2—69.5
final kernels	2—100—100	5.01—99.8—99.99	9.99—99.9—99.9	19.5—99.5—64.9
$p_i = 0.5$	$nbc = 2$	$nbc = 5$	$nbc = 10$	$nbc = 20$
initial kernels	2.25—36.8—100	5.1—44—99.8	10.2—46.6—85.2	12.5—45—65.1
final kernels	2.02—100—99.5	5.1—100—99.1	11.2—99.7—80	18.8—98.5—46.3
$p_i = 0.25$	$nbc = 2$	$nbc = 5$	$nbc = 10$	$nbc = 20$
initial kernels	2.9—28.3—77	9.3—35.9—57.8	9.3—35.5—51.6	2.6—11.8—67.3
final kernels	2.8—99.5—82.1	10.3—98.6—52.4	15.4—97.9—35.6	18.6—97.1—27

Table 1. Application of the first step to random graphs ($p_r = 0$): average numbers of classes, percentages of clustered vertices, percentages of preserved pairs.

nbc = 20, there exist in average 2.6 initial kernels, but the *k*-means method permits to increase this number until 18.6). In every case, *k*-means increases considerably the percentage of clustered items, always located between 95% and 100%. We do the same study with p_i fixed to 1 and p_r varying from 0.1 to 0.5 with a step of 0.2. The results are given in Table 2. The increasing of p_r makes the classes less dense but also the external density higher. The classes become quickly fuzzy with the augmentation of p_r. We notice once again the capacity of the *k*-means method to find back the initial classes since the number of found kernels is close to *nbc* and the values of the average percentage of preserved pairs are large.

$p_r = 0.1$	$nbc = 2$	$nbc = 5$	$nbc = 10$	$nbc = 20$
initial kernels	2—50—100	5—52—100	9.8—50—100	16.7—57.6—100
final kernels	2—100—100	5—100—100	9.99—99.98—99.96	19.2—99.6—97.4
$p_r = 0.3$	$nbc = 2$	$nbc = 5$	$nbc = 10$	$nbc = 20$
initial kernels	1.5—33.9—97.5	4.6—40.5—98.9	9—44.6—99.8	15.4—51.2—99.9
final kernels	1.5—100—100	4.5—99.4—95.3	9.2—98.8—95.2	17.4—97.1—76.6
$p_r = 0.5$	$nbc = 2$	$nbc = 5$	$nbc = 10$	$nbc = 20$
initial kernels	1.25—32—89.9	3.5—29.9—90.2	7.6—34.7—95	13.1—38.4—76.2
final kernels	1.1—100—97.8	3.4—99.3—79.5	8.5—97.9—74.6	18.8—95.6—45.8

Table 2. Application of the first step to random graphs ($p_i = 1$): average numbers of classes, percentages of clustered vertices, percentages of preserved pairs.

3.3 Behaviour of the extension step

In order to validate the second main step of the method, we studied its behaviour on two sets of 100 random graphs, with 100 vertices and 10 initial classes, generated as describes previously; the set A with $p_i = 1$ and $p_r = 0.5$ and the set B with $p_i = 0.5$ and $p_r = 0$. These two cases correspond to classes of density around 0.5; for the set A, the density in the graph is equal to 0.09, for the set B, the graphs are twice less dense with a density of 0.04. On an average, the first step of the method builds 8.5 kernels for the set A and 11.2 for the set B.

We apply the second step on these kernels, α varying in $[0.1; 2.5]$ with a step of 0.1. We first notice that when the parameter α increases, the classes become smaller (the average cardinality decreases from 100 to 12 for the set A and from 100 to 9 for the set B), less overlapping (the average number of classes per vertex decreases from 8.5 to 1 for the set A and from 11 to 1 for the set B) and denser (the average density of the classes increases from 0.1 to 0.4 for the set A and from 0.05 to 0.5 for the set B). Indeed, if α is close to 0, the criterion is totally relaxed, and the classes are extended to the whole graph. If, on the contrary, α is large, there is no extension and the classes are reduced to the kernels given by the first step. We notice that the behaviour of the extension is quite the same for the two sets of graphs, except that the classes builded on the set A are larger and more overlapping than those of set B.

The value of the parameter α must be chosen according to the initial graph and the requirement of the user. We must then select values that correspond to classes of limited cardinality but with a sufficient overlapping.

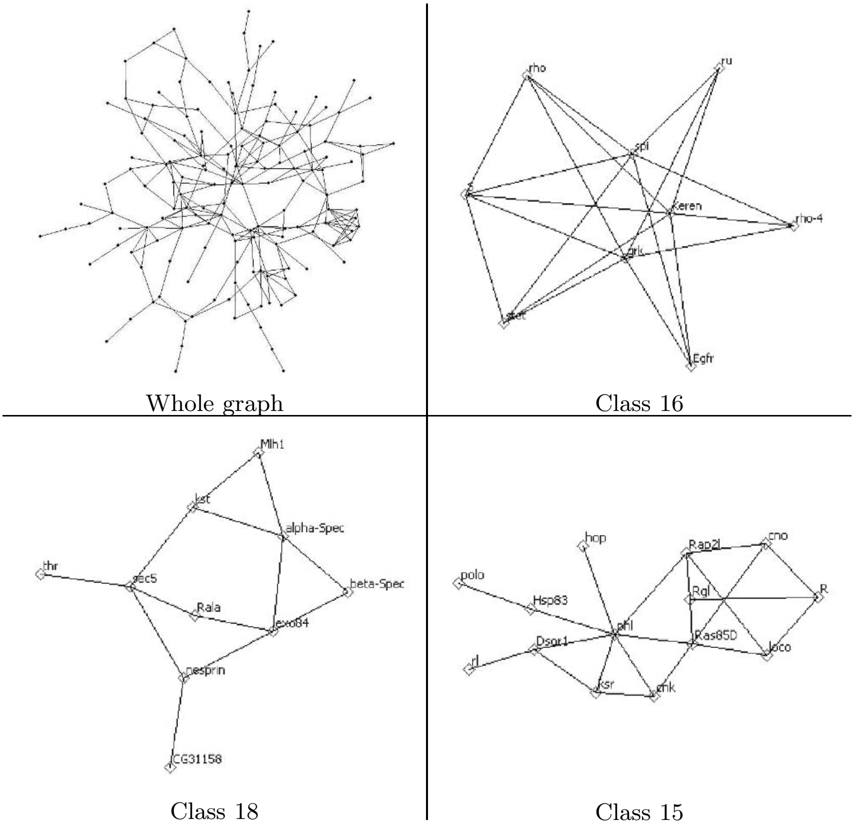
The parameter α allows to build a hierarchy of the classes, the classes obtained for increasing values of α being included one into another (see Denœud (2006) for more details).

4 Application to a protein interaction network

In this section, we apply the proposed method to a small protein interaction network from drosophilia. This graph has 149 vertices and 229 edges, corresponding to a low density of 0.02.

The first step of the method provides 18 classes, with 96 % of clustered vertices, an average cardinality of the classes of 8 and an average density of 0.35. Once again, we notice that the k-means method permits a clear increase of the number of kernels since there were only 8 initial kernels and 21% of clustered vertices. Concerning the extension step, we found that the value 1.5 for the parameter α is adapted for this graph (see Denœud (2006)). It produces classes with 9.1 vertices on the average and with moderate overlap (1.14 classes per vertex). The average density is equal to 0.33, which is large compared with the graph density. Figure 1 represents the whole graph, and three classes extracted by the method (the vertices are labelled by the name of the corresponding proteins).

In order to validate the method from a biological point of view, we compared the classes with cellular processes (we used the on-line database Go-ToolBox (Martin et al. (2004))). We found that 14 classes among the 18 correspond to cellular processes since they contain a majority of proteins belonging to a same biological function.



Whole graph

Class 16

Class 18

Class 15

Fig. 1. A real interaction network and three classes.

For instance, Class 15 contains 15 proteins, of which 14 are anotated (in the biological sense; see the Introduction). All these 14 proteins belong to the cellular process of *signal transduction*. In the neighbourhood of this class, only 50% of the proteins have also this function. Class 16 contains 9 proteins in which 8 are anotated: 7 belong to the process of *nervous system development*, and none in the neighbourhood of the class shares this function. Class 18 contains, among its 10 proteins, 7 anoted proteins which belong in majority to the function of *exocytosis*, while only one protein in the neighbourhood of the class is involved in this process.

5 Conclusion

The presented method permits to build an overlapping clustering in a graph, without imposing the number of classes. It consists in two steps, the first uses an adaptation of k-means in order to find disjoint kernels of the classes, the second consists in the extension of the kernels to overlapping classes. We found that the first step was effective to find dense classes initially set in random graphs. The second step permits, according to the value of the parameter α, to adapt the method following the characteristics of the graph or the application.

The method behaves also well when applied to real protein interaction networks. Some extended studies about the biological interpretation of the classes built by the method have been conducted, confirming that a large majority of the classes corresponds indeed to cellular functions.

References

ARABIE, P., HUBERT, L.J. and DE SOETE, G. (1996): *Clustering and Classification.* World Scientific, Singapore, New Jersey, London, Hong Kong.

BROSSIER, G.(2003): Les éléments fondamentaux de la classification. In: G. Govaert (Ed.): *Analyse des données*, Hermès Lavoisier, Paris, 235-262.

BRUN, C., WOJCIK, J., GUENOCHE, A. and JACQ, B. (2002): Étude bioinformatique des réseaux d'interactions : PRODISTIN, une nouvelle méthode de classification des protéines. In: J. Nicolas, C. Thermes (Eds.): *Actes des Journées Ouvertes: Biologie, Informatique et Mathématiques (JOBIM).* Rennes: IMPG, 171-182.

BRUN, C., HERRMANN, C. and GUENOCHE, A. (2004): Clustering proteins from interaction networks for the prediction of cellular functions. *BMC Bioinformatics 5:95.*

COLOMBO, T., QUENTIN, Y. and GUENOCHE, A. (2003): Recherche de zones denses dans un graphe : application aux gènes orthologues. In: *Knowledge Discovery and Discrete Mathematics Colloquium (Actes des Journées Informatiques de Metz)*, INRIA, 203-212.

DENŒUD, L. (2006): Étude de la distance de transfert entre partitions et recherche de zones denses dans un graphe. PhD thesis, University of Paris 1.

DICE, L.R. (1945): Measures of the amount of ecologic association between species. *Ecology, 26, 297-302.*

DIDAY, E. (1971): Une nouvelle méthode en classification automatique et reconnaissance des formes : la méthode des nuées dynamiques. *Revue de Statistique Appliquée, 19 (2), 19-33.*

HANSEN, P. and JAUMARD, B. (1997): Cluster analysis and mathematical programming. *Mathematical Programming 79, 191-215.*

MARTIN, D., BRUN, C., REMY, E., MOUREN, P., THIEFFRY, D. and JACQ, B. (2004): GOToolBox: functional analysis of gene datasets based on Gene Ontology. *Genome Biology, 5 (12), Article R101.* http://genomebiology.com/2004/5/12/R101.

WISHART, D. (1969): Mode Analysis: A generalisation of gearest geighbour which reduces chaining effects. In: A.J. Cole (Ed.): *Numerical Taxonomy.* London: Academic Press, 282-311.

Species Clustering via Classical and Interval Data Representation

Marie Chavent

Université Bordeaux 1,
Institut de Mathématiques de Bordeaux, UMR CNRS 5251,
351 cours de la libération,
33405 Talence cedex, France
chavent@math.u-bordeaux1.fr

Abstract. Consider a data table where n objects are described by p numerical variables and a qualitative variable with m categories. Interval data representation and interval data clustering methods are useful for clustering the m categories. We study in this paper a data set of fish contaminated with mercury. We will see how classical or interval data representation can be used for clustering the species of fish and not the fishes themselves. We will compare the results obtained with the two approaches (classical or interval) in the particular case of this application in Ecotoxicology.

1 Introduction

Interval data representation can be very useful to study groups of objects described by quantitative variables. Describing a group of objects on each variable by an interval of values rather than by a mean value, allows to reflect the variability that underlies the observed measurement. Many data analysis techniques have been extended to treat such new data description (see for instance Bock and Diday (2000)). But a question frequently asked while applying these techniques is the following: 'Are the results obtained with intervals different than those obtained with means?'. Of course it is very difficult to answer this question not only because the data tables are different but also because the techniques are different. We will however try to answer this question in the particular case a real application in Ecotoxicology and in the context of clustering. This application is concerned by mercury contamination of fish in rivers of French Guyana (Chavent et al. (2000)). Our problem was to define a partition of the different species of fish according to their mercury concentrations in fives organs (gills, liver, intestine, stomach, kidney). A first partition was calculated with point-valued data and a second one with interval-valued data. The two partitions were compared not numerically (because no numerical comparison between the two partitions is possible) but according to an external partition (the diet of the species) and according to a fuzzy partition of the species (obtained by clustering the fishes themselves).

Let consider the general case of a data table where n objects are described by p variables, one of them is qualitative with m categories and the $p-1$ others are quantitative. The problem is to find a partition in K clusters, not of the n objects, but of the m categories. In the application, the data table describes $n = 67$ fishes of $m = 10$ different species by 5 quantitative variables (their mercury contaminations in fives organs). We present here three different approaches to find a partition of the 10 species into homogeneous clusters.

- clustering the 67 fishes described by the five quantitative variables. It gives a fuzzy clustering of the 10 species,
- clustering the 10 species described by mean values on the five variables,
- clustering the 10 species described by intervals on the five variables.

2 The data

The data were collected by researchers of the EPOC[1] laboratory: 265 fishes of 36 different species were catch in 1997 in several French Guyana rivers and a sample of 67 fishes of 10 species and 3 diet were selected (see Table 1).

Carnivorous	Omnivorous	Detritivorous
Ageneiosus brevifilis (7)	Leporinus fasciatus (3)	Doras micropoeus (8)
Cynodon gibbus (7)	Leporinus frederici (3)	Platydoras costatus (10)
Hoplias amara (10)		Pseudoancistrus barbatus (7)
Potamotrygon hystrix (4)		Semaprochilodus varii (8)

Table 1. Diet and frequency of each species in the sample

The researchers of the EPOC laboratory measured for each of the 67 fishes the length, the weight and the mercury concentration in μg/g in the muscle and in 5 organs. After several pre-treatments (descriptive statistics, variable selection....), we retained the data table described Table 2.

	species	diet	ln(liver/muscle)	…	ln(stomach/muscle)
1	Ageneiosus brevifili	Carnivorous	-0,12	…	NA
2	Cynodon gibbus	Carnivorous	1,59	…	0,22
3	Leporinus frederici	Omnivorous	-0,04	…	-1,77
⋮	⋮	⋮	⋮	…	⋮
66	Doras micropoeus	Detritivore	0,8	…	-0,89
67	Doras micropoeus	Detritivore	1,34	…	-1,45

Table 2. Extract of the data table

[1] UMR CNRS 5805 EPOC (Environnements et Paloenvironnements OCéaniques)

The five quantitative variables of this data table are based on the ratio between the mercury concentration in the five organs and the mercury concentration in the muscle. These ratios were used because of the positive correlation between the mercury concentration variables. In a second time, the skewness of the distributions of the ratios has motivated the logarithmic transformation.

Figure 1 represents the 67 fishes in the first factorial plane calculated with these five quantitative variable. Each fish is numbered according to its species (from 1 to 10).

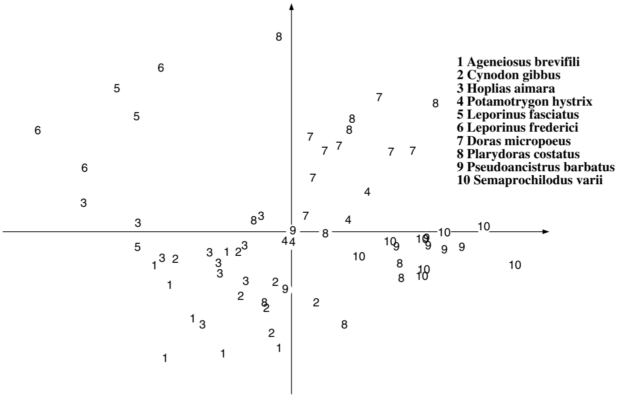


Fig. 1. The 67 fishes in the first factorial plane, numbered from 1 to 10 according to their species.

We notice on this figure that the fishes of the same species are rather close in the first factorial plane. As we will see in the next section when clustering the fishes, those in the same species are mostly in same clusters. The partition of the 67 fishes will then define a kind of fuzzy partition of the 10 species.

3 Fuzzy partition of the species

A partition in 4 clusters of the 67 fishes described by the five quantitative variables of Table 2, was performed by Ward hierarchical clustering. The Table 3 gives the proportion of fish of each species in each cluster. The diet of the species is also indicated. We notice that all the fishes of the three carnivorous species are in cluster1, and that this cluster contains no fish from another species. Obviously, a clustering of the 10 species should put the three carnivorous species in the same cluster. It means also that the carnivorous fishes have the same kind of behavior in term of mercury concentration. In the same way, we see that cluster2 contains only omnivorous fishes, and the of the omnivorous fishes are almost in cluster2 (only one of the three Leporinus fasciatus fishes is in cluster1). The two omnivorous species should then be in the same cluster in a partition of the species. Five species of detritivorous are in two different clusters and two species of detritivorous are difficult to assign to one of the four clusters. This result is not surprising because a doubt remains concerning the diet of these species.

	cluster1	cluster2	*cluster3*	*cluster4*	Diet
Ageneiosus brevifili	**100**	0	0	0	**carnivorous**
Cynodon gibbus	**100**	0	0	0	**carnivorous**
Hoplias aimara	**100**	0	0	0	**carnivorous**
Doras micropoeus	0	0	*100*	0	*detritivorous*
Leporinus fasciatus	33.33	66.67	0	0	omnivorous
Leporinus frederici	0	100	0	0	omnivorous
Pseudoancistrus barbatus	14.29	0	0	*85.71*	*detritivorous*
Semaprochilodus varii	0	0	0	*100*	*detritivorous*
PLATYDORAS COSTATUS	20	0	40	40	DETRITIVOROUS ?
POTAMOTRYGON HYSTRIX	50	0	25	25	DETRITIVOROUS ?

Table 3. Proportion of fish of each species in each cluster and the diet of the species

Table 4 gives the crisp partition of 8 of the 10 species deduced from Table 3. The two species Platydoras costatus and Potamotrygon hystrix are not assigned to one of those clusters.

cluster1	cluster2	*cluster3*	*cluster4*
(carnivorous)	(omnivorous)	*(detritivorus)*	*(detritivorus)*
Ageneiosus brevifili	Leporinus fasciatus	*Doras micropoeus*	*Pseudoancistrus barbatus*
Cynodon gibbus	Leporinus frederici		*Semaprochilodus varii*
Hoplias aimara			

Table 4. Crisp partition of 8 of the 10 species

4 Classical data description and divisive clustering

In order to describe the 10 species with the 5 mercury concentration variables, a new data table was constructed. The fishes of the same species were aggregated by calculating the mean value on each variable and Table 5 is the resulting classical data table.

The descendant hierarchical clustering method DIV (Chavent (1997)) was then applied to this data table and after three divisions, a four clusters partition of the 10 species was obtained (see Figure 2). This partition is not satisfactory according to the diet partition and according to the partition

species	ln(liver/musc)	ln(kidn/musc)	ln(gills/musc)	ln(intest/musc)	ln(stom/musc)
Ageneiosus brevifili	-0,39	-0,25	-1,54	-0,89	-1,25
Cynodon gibbus	1,05	0,24	-1,61	-1,29	-1,06
Hoplias aimara	0,26	0,764	-1,73	-1,36	-1,55
Doras micropoeus	1,72	2,11	-2,21	-0,78	-0,90
Leporinus fasciatus	-0,82	-0,28	-2,81	NA	-1,93
Leporinus frederici	-0,47	-0,65	-2,87	-1,61	-1,55
Pseudoancistrus barbatus	2,29	-1,00	NA	0,38	-0,24
Semaprochilodus vari	3,43	1,49	-1,64	0,02	-0,25
Platidoras costatus	1,58	1,51	-1,98	-0,28	-1,00
Potamitrigon hystrix	1,15	1,25	NA	-0,13	-0,76

Table 5. Point-type description of the 10 species

obtained by clustering the fishes (Table 4). The two omnivorous species (Leporinus fasciatus, Leporinus frederici) are not in the same cluster and the two clusters of detritivorous species (Doras micropoeus against Pseudoancistrus barbatus and Semaprochilodus varii) highlighted Table 3 and Table 4, do not appear in this partition.

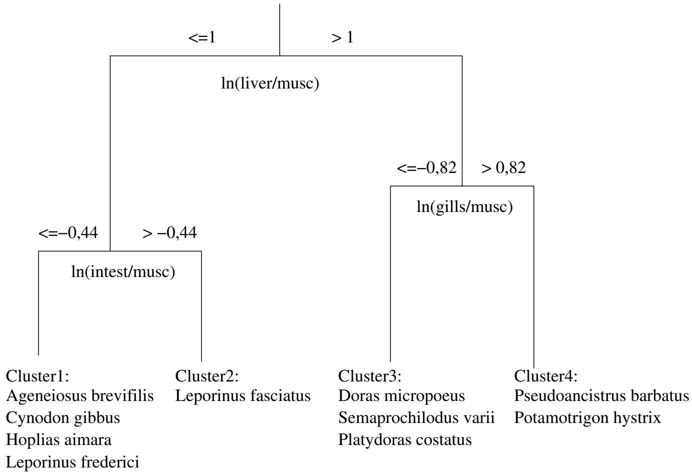


Fig. 2. Dendrogram of the upper hierarchy for classical data description.

The question was then: is this unsatisfactory result due to way the data were aggregated or to the clustering method itself? On order to answer this question, we applied an other clustering method, the Ward ascendant hierarchical clustering method, to the same data table. Figure 3 represents the 10 species described in Table 5, in the first factorial plane. Each species is numbered according to its cluster in the 4-clusters partition obtained with Ward. In this partition the two species (Leporinus fasciatus, Leporinus frederici) are in the same cluster. The inappropriate separation of these two species by DIV was perhaps due the monothetic constraint of this method. The three carnivorous species (Hoplias aimara, Cynodon gibbus, Potamotrygon hystrix) are well gathered in one cluster but the separation of the detritivorous species Doras micropoeus from the two other detritivorous species Pseudoancistrus barbatus and Semaprochilodus varii, again do not appear in this partition.

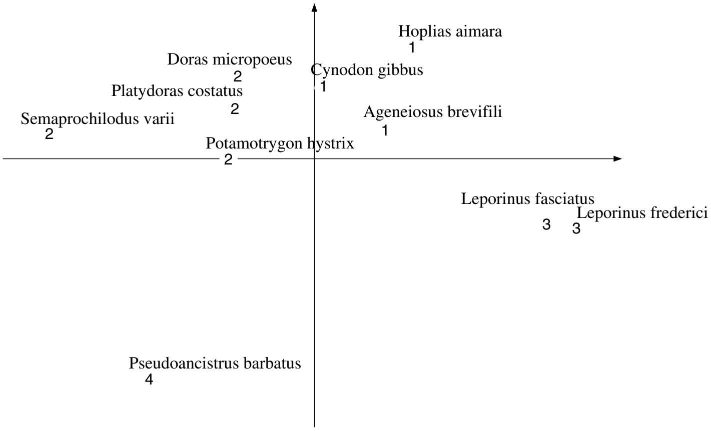


Fig. 3. First factorial plane of the 10 species (aggregated by the mean), numbered from 1 to 4 according to its cluster in the Ward partition.

5 Interval data description and divisive clustering

In a second time, the fishes of the same species were aggregated by calculating an interval of values on each variable. Table 6 is the resulting interval data table calculated with the DB2SO method (see Stephan (1998)) and the SODAS software (see for instance Diday and Esposito (2003)).

species	ln(liver/musc)	ln(kidn/musc)	ln(gills/musc)	ln(intest/musc)	ln(stom/musc)
Ageneiosus brevifili	[-0.80:0.34]	[-1.50:0.35]	[-1.88:-1.21]	[-1.45:-0.48]	[-1.49:-1.05]
Cynodon gibbus	[0.12:1.59]	[-0.51:1.18]	[-1.91:-1.44]	[-1.75:-0.68]	[-1.61:0.22]
Hoplias aimara	[-0.44:0.90]	[-0.17:1.60]	[-1.98:-1.53]	[-2.17:-0.71]	[-2.36:-0.93]
Doras micropoeus	[1.34:2.12]	[1.47:2.69]	[-2.38:-2.21]	[-1.99:0.39]	[-1.45:-0.24]
Leporinus fasciatus	[-0.98:-0.58]	[-0.32:0.35]	[-3.00:-2.63]	NA	[-2.11:-2.76]
Leporinus frederici	[-0.82:-0.04]	[-0.95:-0.19]	[-3.27:-2.55]	[-1.74:-1.42]	[-2.03:-0.55]
Pseudoancistrus barbatus	[1.26:2.84]	[-1.00:1.00]	NA	[-0.31:0.68]	[-0.71:0.12]
Semaprochilodus vari	[2.70:3.96]	[1.11:1.91]	[-1.79:-1.40]	[-0.91:0.52]	[-0.74:0.22]
Platidoras costatus	[0.41:2.42]	[-0.02:2.75]	[-2.90:-1.27]	[-1.22:0.38]	[-1.41:-0.49]
Potamitrigon hystrix	[0.66:2.01]	[0.77:2.15]	NA	[-0.50:0.23]	[-0.80:-0.69]

Table 6. Interval type description of the 10 species

The divisive clustering method DIV for interval data description (Chavent (1997)), was applied to the 10 species described in Table 6. After three divisions, a four clusters partition of the 10 species was obtained (see Figure

4). This partition is more in adequation with the fuzzy partition obtained by clustering the fishes (Table 4) than those obtained with the classical descriptions. The two omnivorous species are alone in one cluster. The three carnivorous species are alone in one cluster and the two clusters of detritivorous species (Doras micropoeus against Pseudoancistrus barbatus and Semaprochilodus varii) are found. The two detritivorous species Platydoras costatus and Potamitrigon hystrix that were not assign clearly to one cluster in the fuzzy partition (Table 3), are put together with the detritivorous species Doras Micropoeus. For all these reasons, this partition is more satisfactory than the one obtained with the classical data representation.

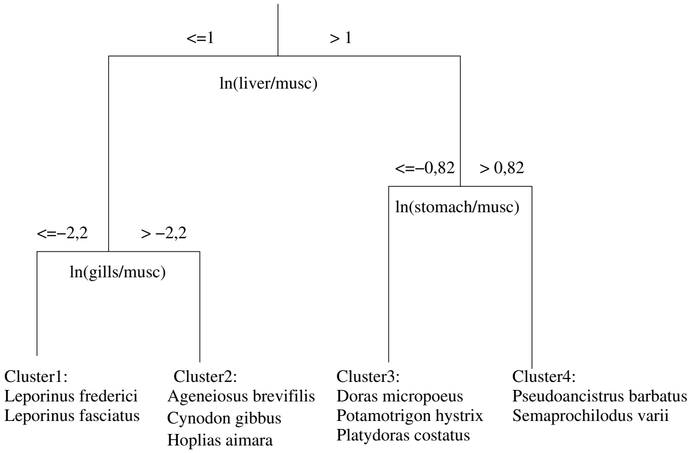


Fig. 4. Dendrogram of the upper hierarchy for interval data description.

Figure 5 gives an idea of the variation of the fishes of the 10 species. Rectangles were drawn on the Figure 1 in order to represent the min-max variation of the fishes of each species (numbered from 1 to 10) in each dimension of the first factorial plane. This figure helps understanding the partition obtained with DIV and the interval data description. The Semaprochilodus varii and the Pseudoancistrus barbatus for instance are in the same cluster because of the similarity between their positions and between their dispersions. The rectangle Platydoras castatus (8) shows an broad variability of the fishes of this species. It was assigned to the same cluster than the rectangle Doras micropoeus (7) but this important variability questions on the signification of the proximity between the two species. The fuzzy partition of the species gives more precise results concerning the difficulty of clustering this species.

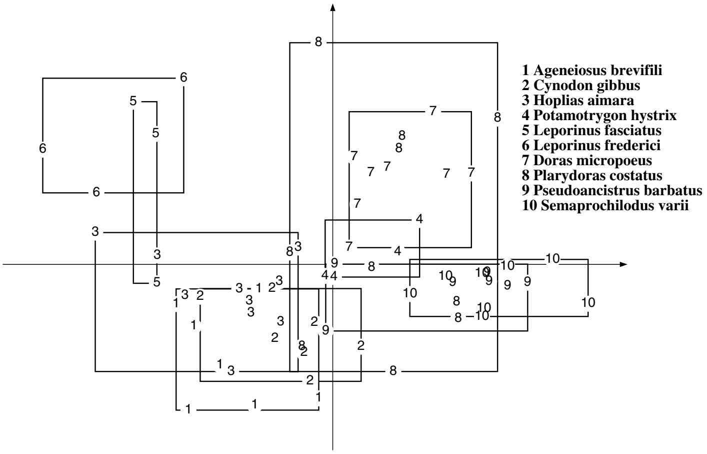


Fig. 5. Min-max variation of the fishes of each species in the two dimensions of the first factorial plane.

6 Conclusion

This case study in Ecotoxicology was a good illustration of the use of interval data representation for clustering aggregated data. We proposed a three steps methodology: clustering the 67 fishes to find a fuzzy partition of the species, clustering the species with point-type descriptions and clustering the species with interval-type descriptions. We compared the three partitions and we concluded that the partition obtained with the interval-type description is more in adequation with the diet of the species and with the fuzzy partition. This is a good result in a particular case showing the interest of interval data representation. Concerning the Ecotoxicological application, this study highlighted the discriminant power of the diet in term of mercury concentration and the existance of two clusters of detritivorous species.

Acknowledgments

The author thanks A. Boudou and R. Maury-Brachet (UMR CNRS 5805 EPOC laboratory of University Bordeaux 1) for providing data and collaborating on this application.

References

BOCK, H.-H. and DIDAY, E. (eds.) (2000): *Analysis of symbolic data. Exploratory methods for extracting statistical information from complex data.* Springer Verlag, Heidelberg.

CHAVENT, M. (1997), *Analyse des données symboliques, une méthode divisive de classification.* PhD thesis of Paris IX-Dauphine University.

CHAVENT, M. (1998): A monothetic clustering method. *Pattern Recognition Letters, 19, 989-996.*

DIDAY, E., ESPOSITO, F. (2003): An introduction to symbolic data analysis and the SODAS software. *Intell. Data Anal. 7(6): 583-601.*

CHAVENT, M., LACOMBLEZ, C., BOUDOU, A., MAURY-BRACHET, R. (2000): Contamination par le mercure et classification d'espéces en Ecotoxicologie: approche classique, approche symbolique. *La revue Modulad, Décembre 2000, 19–32.*

STEPHAN, V. (1998): *Construction d'objets symboliques par synthèse des résultats de requêtes SQL.* PhD thesis of Paris IX-Dauphine University.

Looking for High Density Zones in a Graph

Tristan Colombo[1] and Alain Guénoche[2]

[1] Laboratoire de Chimie Bactérienne
31 Av. Joseph Aiguier, 13009 Marseille, France, *tcolombo@bp-soft.com*

[2] Institut de Mathématiques de Luminy
163 Av. de Luminy, 13009 Marseille, France, *guenoche@iml.univ-mrs.fr*

Abstract. The aim of this paper is to introduce new methods to build dense classes of vertices in a graph. These classes correspond to connected parts having a proportion of inner edges which is higher than the average on the whole graph. They are progressively built ; a kernel of each class is first established, then they are extended to connected elements and finally to a partition. Several density fonctions are compared. A Monte-Carlo validation of this method is made from random graphs fulfilling some density conditions.

1 Introduction

With the development of the sequencing of complete bacterial genomes, we know all the gene sequences of more than a hundred species. But in many cases, their protein functions remain unknown. Establishing these functions is one of the priority tasks in genomics. The biological hypothesis is that proteins encoded by genes inherited from a common ancestor have identical or similar functions. Hence, the detection of these phylogenetic links, and the clustering of the related sequences, is an essential step for the identification of protein functions.

This evolutionary relation, called *homology* (Fitch (1970)), can be decomposed in two types of phylogenetic events : *paralogy* and *orthology.* They depend on the fact that the evolutionary path between *homolog* genes goes through a duplication event or not. This distinction is important since it is generally admitted that the proteins encoded by *orthologous* genes (without duplication) have conserved the same function whereas the proteins from *paralogous* genes (with duplication) have generally acquired different functions. Thus, the distinction between these two types is a fundamental step for the functional prediction process.

One way to bypass this pitfall is to analyse the links between genes, orthology being estimated using an evolutive distance. All the gene sequences included in genome G_1 are compared to those of G_2 and only pairs of genes having the smallest distance values are retained ; this relation is denoted BH (for best hit). It is not symmetrical and only *nearest reciprocal neighbors* are retained, making a BBH relation (for best bidirectional hit). It has been reinforced by Fitch (2000), defining *isorthology.* It is inferred between BBH genes A and B, respectively from genomes G_1 and G_2, looking for the nearest genes, A' from A in G_1 and B' from B in G_2, considered as their paralogs. If $d(A, A') > d(A, B)$ and $d(B, B') > d(A, B)$, then A and B are qualified *isortholog*.

These relations enable us to construct a graph Γ where the vertices are all the genes of the considered genome set. As the orthology relation is by definition a transitive one, the graph Γ should consist of disjoint complete subgraphs, that are disjoint cliques. Because of errors estimating evolutionary distances, the connected components are generally not cliques and genes having different functions may be observed in the same component because of artificial edges. Consequently, the orthologous classes can be detected looking for *dense zones* in Γ, that are classes of vertices having a high percentage of internal edges. These zones, also called *quasi-cliques* (Matsuda and al. (1999)), may constitute hypothetical functional classes.

The clustering methods based on density have been introduced by Wishart in 1976; the idea is to build classes around elements having many neighbors in a threshold graph associated to a distance value. They have not been largely used, because the simple degree as density function gives unexpected results. Recently, (Bader and Hogue (2003)) for simple graphs, have reactivated this approach, without comparison between density functions. The proposed method is also based on the evaluation of a density function for each vertex. Our algorithm is progressive with, in a first step, the identification of kernels defined as connected vertices having a locally maximal density and, in a second step, the extension of the *kernels* through out the addition of connected vertices according to two strategies ; the first one establishes partial classes, and the second one builds a complete partition. The number of classes is not required as in many other clustering methods since the number of kernels is kept as the number of classes.

The choice of a density function is critical for the efficiency of the algorithm. So, we will first describe in section 2 four density functions, testing them on simple graphs with a given number of dense zones to be recovered. The algorithm is detailed in section 3. Performances of the different functions are compared by simulations on random graphs in section 4.

2 Density functions

Let X be the set of the n vertices, E the set of the m edges and $\Gamma = (X, E)$ the corresponding graph. It is assumed to be connected. For any part Y of X, let $\Gamma(Y)$ be the set of vertices out of Y that are adjacent to Y

$$\Gamma(Y) = \{x \in X - Y \text{ such that } \exists y \in Y, (x, y) \in E\}.$$

The neighborhood of x is $\Gamma(x)$. The degree of a vertex x is denoted $Dg(x) = |\Gamma(x)|$ and let δ be the maximum degree in the graph.

For each vertex x, we evaluate a density value denoted $De(x)$. The density function De is a map from X to $\mathbb{R}_+$ varying increasingly with the number (or the percentage) of edges in the neighborhood of a vertex. By definition, all the vertices having degree 1 will get a density equal to 0, to avoid inappropriate or undefined values in the following computations. We propose four functions, which will be compared in section 4 :

- The Wishart's one was the simple degree ; here, we normalize it :

$$De_1(x) = \frac{Dg(x)}{\delta}.$$

  This function gives a central place to vertices with a high degree, a place they don't necessarily have, especially in protein graphs.
- The average degree in the neighborhood of x :

$$De_2(x) = \frac{Dg(x) + \sum_{y\in\Gamma(x)} Dg(y)}{(1 + Dg(x))}.$$

  This function counts the same way the edges adjacent to x and those that are adjacent to a neighbor of x. In order to overcome this drawback, we consider :
- The rate of triangles going through x. Let $N_t(x)$ be the number of triangles containing x :

$$N_t(x) = |\{(y, z) \in E \text{ such as } (x, y) \in E \text{ and } (x, z) \in E\}|.$$

  This number is divided by the maximum value expected for a vertex of degree $Dg(x)$.

$$De_3(x) = \frac{N_t(x)}{\frac{1}{2}.Dg(x).(Dg(x) - 1)}.$$

  This function is the most often used in similar approaches (Bader and Hogue (2003)). A vertex having only connected neighbors, making so a clique, will have a maximum density value equal to 1.0. At the contrary, when some vertex pairs in $\Gamma(x)$ are not connected, this function decreases very quickly. In order to give more density to the vertices which have many links, we introduce a new function :
- The percentage of edges in the neighborhood of x, that is the number of edges adjacent to x plus those making triangle, divided by the maximal number of edges in the neighborhood of a vertex of degree $Dg(x)$.

$$De_4(x) = \frac{Dg(x) + N_t(x)}{\frac{1}{2}.Dg(x).(Dg(x) + 1)}.$$

Other density functions have been tested : For the first one the Czekanovski-Dice distance on Γ is evaluated and the density is estimated from the average distances in the neighborhood of any vertex. It provides also satisfying results very similar to those of De_4. The second one, based on the core index of vertices (Batagelj et al. (1999)), gives poor results.

3 Algorithm for dense classes

The searched classes have to be connected in Γ, their elements sharing high density values. Our initial idea was to select a density threshold and to consider the partial subgraph whose vertices have a density greater than this threshold ; thus the classes would be the connected components of this threshold graph. This approach does not give satisfactory results and we adopt a progressive strategy, considering the *local maximum values* of the density function.

3.1 Kernels of the classes

A kernel, denoted K, is a connected part of Γ, obtained by the following algorithm : we start from the local maximum values of the density function that are larger than the average and we consider the partial subgraph of Γ reduced to those vertices.

$$\forall x \in K, \forall y \in \Gamma(x) \text{ we have } De(x) \geq De(y).$$

The initial kernels are the connected components of this graph. More precisely, if several vertices with equal maximum value are adjacent, they belong to the same kernel ; otherwise the initial kernels are singletons. Then, we assign recursively to each kernel K all the vertices (i) having a density greater than or equal to the average density value over X and (ii) that are adjacent to only one kernel. Doing so, we avoid any ambiguity in the assignment, postponing the decision when several are possible. Let p be the number of initial kernels that establishes the number of partial and complete classes.

3.2 Extensions to dense classes

In a second step, we extend the kernels by adding other elements, those that could be assigned to several ones or those having a density lower than the average. We have implemented the following two strategies :

Partial extension The principle of this extension rule is to aggregate to a kernel all the connected elements while its average degree increases. Let C be a class initially restricted to its kernel K ; we compute :

- $\overline{Dg}(C)$ the average degree of C, considering only the internal edges,
- for each element x of $\Gamma(C)$, its number c_x of connections to C and
- c_{max} the maximum of the c_x over $\Gamma(C)$.

If $c_{max} \geq \overline{Dg}(C)$ all the elements having c_{max} connections are added to C. If at least one element is added, this procedure is repeated until there is no element increasing the average inner degree.

Thus, the classes are still connected. Generally, at the end of this procedure all the vertices are not clustered, but these zones would have a density value higher than those making a complete partition. We also observe that an element may be added to several kernels and so the final classes are not necessarily disjoint.

Complete extension In order to cluster all the vertices, we have developed another strategy for the kernel extension. Let us consider that we have p kernels denoted K_i with k_i elements. Let L be the set of the vertices that remain to be classified; they are examined in the decreasing density order. We assign each element to the kernel to which it is mainly connected. Let x be the current vertex :

- for each kernel K_i we compute the number $c_i(x)$ of its connections to x: $c_i(x) = |\Gamma(x) \bigcap K_i|$.
- x is connected to the kernel K_j such that $c_j(x)$ is maximal and, in case of ties, to the one for which k_j is minimal or to both if overlapping classes are admitted ;
- the quantities c_i and k_i are updated.

Doing so, each element is assigned to a single kernel. The decision in case of ties to place x in the class with the smallest number of internal edges tends to give balanced classes.

3.3 Complexity

Kernel computation is in $O(m) \approx O(n\delta)$. For the extension step we start by evaluating the average internal degree of each kernel and the number of connections of the adjacent elements ; this step is in $O(p\delta)$.

- In the first case, at each iteration, we retain the elements having a sufficient degree and we update the average class degree ; this procedure has complexity $O(n\delta)$. The number of iterations being bounded by δ, the time complexity of the first extension rule is in $O(n\delta^2)$;
- in the second case, we assign at each iteration only one element, and we update the p values c_i and s_i examining at most δ edges, which gives $O(n\delta)$ for all the classes.

Thus, the complexity of the whole extension procedure is in $O(np\delta^2)$.

Compared to other methods computing first the minimum number of edges between vertices or a score function, as the number of shortest paths going through any edge (Newman (2001)), this algorithm is very efficient. It allows the treatment of large graphs ($n \approx 10000$), for which the linearity in n is essential.

4 Validation by simulation

In order to evaluate the ability of this method to recover high density areas in a graph, we compare the performances of the four density functions and the two extension procedures. First, we have developed a random generator of graphs in which there are classes having more internal edges than the external ones, according to given probabilities.

4.1 Generator of random graphs

The generator of random graphs depends on four parameters :

- N : the number of vertices,
- p : the number of wanted classes in the graph,
- d_i : the average internal density within the classes,
- d_e : the average density of the external edges.

In order to get such a random graph, we begin with a random partition of the N elements in p classes, denoted $C_1, .., C_p$. This initial partition P is stored in a vector $p_1, ..p_N$, where p_k is the class number of the k-th element. Next, for each pair of elements, we select at random a real number between 0 and 1, and we add the corresponding edge if and only if this number is lower than or equal to d_e (resp. d_i) when the two elements are in different (resp. identical) classes. This procedure does not guarantee that our graphs will have precisely p dense zones ; they may be decomposed, or rearranged according to the random edges. Similarly, the real density values are not necessarily equal to d_i and d_e but we have observed that, on average, these parameters are correctly fitted.

4.2 Quality of the classes compared to the initial partition

There are three levels of classes successively built :

- the initial kernels,
- the partial classes corresponding to the extended kernels (we have suppressed the multiple assignment possibility which is very rare),
- the complete classes obtained by full assignment to the most connected kernel.

Let N_c be the number of classified vertices at each level. They are distributed in p' classes denoted $C'_1, ..C'_{p'}$ realizing a partition P'. We first aim at mapping the classes of P' onto those of P evaluating $n_{i,j} = |C_i \bigcap C'_j|$. We set that the *corresponding* class to C'_j is the class C_k in P, that contains the greatest number of elements of C'_j, that is the one such that $n_{k,j} \geq n_{i,j}$ for all i from 1 to p.

In order to measure the accuracy of the computed classes, we evaluate three criteria.

- τ_c : the percentage of clustered elements in P' ($\frac{N_c}{N}$).
- τ_e : the percentage of elements in one of the p' class which belong to its corresponding class in P.
- τ_p : the percentage of joined pairs in P' coming from the same class in P.

Remark : The last two criteria may reach their maximum value (1.0) even when partitions P and P' are not identical, but when two classes of P' are included in one of P. These classes of P' will have the same corresponding class in P and all their elements will be considered as well classified and the rate of pairs will be equal to 1.

4.3 Results

These results are average values obtained from 200 graphs of 100 vertices distributed in 3 classes, with an internal density $d_i = 0.5$ and an external density $d_e = 0.1$. Such a gap seems to give easy problems, that are classes easy to recover. But assuming that the p classes have the same cardinality, there are $d_i\frac{n(n-p)}{2p}$ intra-class edges and $d_e\frac{n^2(p-1)}{2p}$ inter-class edges. So there will be around 808 + 333 edges in our graphs, corresponding to an average density of 0.232. It means that the internal density is approximatively twice the average over the whole graph.

The rows of Table 1 correspond to the three types of computed classes and the columns to the four density functions. Each cell contains the values of the three criteria τ_c, τ_e and τ_p. The last row indicates the average number of classes obtained in P'.

	De_1			De_2			De_3			De_4		
	τ_c	τ_e	τ_p	τ_c	τ_e	τ_p	τ_c	τ_e	τ_p	τ_c	τ_e	τ_p
Kernels	.26	.78	.65	.21	.89	.80	.26	.96	.93	.25	.95	.92
Partial extension	.42	.84	.74	.38	.91	.85	.48	.97	.95	.50	.97	.95
Complete extension	1.0	.68	.59	1.0	.65	.55	1.0	.93	.91	1.0	.95	.93
Nb. of classes	2.4			3.5			4.6			5.4		

Table 1. Average results of the 4 density functions obtained from 200 graphs of 100 vertices distributed in 3 classes. They are randomly generated with internal density $d_i = 0.5$ and external density $d_e = 0.1$.

The superiority of functions De_3 and De_4 is obvious. The function De_1 is not enough discriminant to generate a sufficient number of local maximum values. Since they determine the number of kernels, the classes are badly defined compared with the initial ones. The performances of functions De_3 and De_4 are satisfying, at each level. On average, 25% of the elements belong to the kernels and 50% are the extended classes. More than 95% of the joined elements come from the same initial class. For the complete extension, more than 90% of the assignments are correct.

4.4 Extended results

In order to study the variation of the results on more difficult problems we fix the external density (equal to .1 then .2) and the internal one varies from .4 to .7 (an internal density greater than .7 always gives easy problems). The average results of function De_4 are established again on 200 graphs of 100 vertices (cf Table 2).

		Kernels			**Partial extension**			**Complete extension**		
d_e	d_i	τ_c	τ_e	τ_p	τ_c	τ_e	τ_p	τ_c	τ_e	τ_p
.1	.4	.25	.93	.88	.44	.94	.90	1.0	.89	.84
.1	.5	.27	.96	.93	.48	.97	.95	1.0	.94	.91
.1	.6	.28	.97	.96	.52	.98	.97	1.0	.95	.93
.1	.7	.31	.98	.97	.59	.99	.98	1.0	.95	.93
.2	.4	.24	.74	.57	.42	.74	.60	1.0	.62	.49
.2	.5	.25	.85	.75	.44	.88	.80	1.0	.76	.66
.2	.6	.25	.91	.85	.45	.93	.89	1.0	.80	.72
.2	.7	.27	.93	.90	.50	.96	.93	1.0	.81	.74

Table 2. Average results for function De_4 obtained from 200 graphs of 100 vertices distributed in 3 classes when the internal density varies.

When there are few connections between classes ($d_e = .1$) whatever is the internal density, the results are satisfying : 25% to 31% of the elements are classified in kernels and 44% to 59% in the extended classes. Globally, more than 90% of the assignments are correct. But when the number of external edges increases ($d_e = .2$), the results are not so good, except when the internal density is high. When the gap between the density values is lower than .3, one can expect that the classes will not be precisely recovered.

Other simulation results and biological applications can be read in the Colombo thesis (2004). They permit to clarify the orthology relation between genes inherited from a common ancestor and to predict some biological cellular functions.

Acknowledgements Thanks to Yves Quentin who makes clear the motivations of this work which has been realized with the help of the ACI-IMPBio, provided by CNRS.

References

BADER, G.D. and HOGUE, C.W. (2003): An automated method for finding molecular complexes in large protein interaction networks, *BMC Bioinformatics*, 4, 2.

BATAGELJ, V. and MRVAR, M. (1999): Partitioning approach to visualisation of large graphs, *Lecture Notes in Computer Science* 1731, Springer, 90-97.

COLOMBO, T. (2004): *Algorithmes pour la recherche de classes de gènes en relation fonctionnelles par l'analyse de proximités et de similarités de séquences*, Thèse de l'Université d'Aix-Marseille II.

FITCH, W.M. (1970): Distinguishing homologous from analogous proteins, *Syst. Zool.*, 19, 99-113.

FITCH, W.M. (2000): Homology : a personal view on some of the problems, *Trends in Genetics*, 16.

MATSUDA, H., ISHIHARA, T. and HASHIMOTO, A. (1999): Classifying molecular sequences using a linkage graph with their pairwise similarities, *Theoretical Computer Science*, 210, 305-325.

NEWMAN, M.E.J. (2001): Scientific Collaboration Networks : Shortest paths, weighted networks and centrality, *Phys. Rev.*, 64.

WISHART, D. (1976): Mode analysis : generalization of nearest neighbor which reduces chaining effects, *Numerical taxonomy*, Academic Press, 282-311.

Block Bernoulli Parsimonious Clustering Models

Gérard Govaert[1] and Mohamed Nadif[2]

[1] HEUDIASYC, UMR CNRS 6599, Université de Technologie de Compiègne, BP 20529, 60205 Compiègne Cedex, France, *gerard.govaert@utc.fr*

[2] CRIP5, Université Paris Descartes, 45 Rue des Saint-Pères, 75270 Paris, France, *mohamed.nadif@univ-paris5.fr*

Abstract. When the data consists of a set of objects described by a set of binary variables, we have embedded the block clustering problem of binary table in the mixture approach. In using a Bernoulli model and adopting the classification maximum likelihood principle we perform an adapted version of the block CEM algorithm. In this paper, we propose different parsimonious models by imposing constraints on the Bernoulli parameter.

1 Introduction

Although many clustering procedures such as hierarchical clustering, k-means (Forgy, 1965) or the dynamic cluster method (*nuées dynamiques*) (Diday, 1971, 1974), aim to construct an optimal partition of objects or, sometimes, of variables, there are other methods, called block clustering methods, which consider simultaneously the two sets and organize the data into homogeneous blocks.

A wide variety of procedures have been proposed for finding patterns in data matrices. These procedures differ in the pattern they seek, the type of data to which they apply, and the assumption on which they rely. Let us mention the works of Hartigan (1975), Bock (1979), Marchotorchino (1987), Govaert (1983, 1984, 1995), Arabie and Hubert (1990) and Rirschard et al. (2001) who have proposed some algorithms dedicated to different kinds of matrices.

These last years, block clustering has become an important challenge in data mining context. These kinds of methods have practical importance in a wide of variety of applications such as text mining and market basket data analysis. Typically, the data that arise in these applications are arranged as a two-way contingency or co-occurrence table. In some cases the values of data are binary indicating for example the presence or absence of a word in a document.

In this paper, we will focus on these kinds of data. The data which we consider is noted $\mathbf{x}$; it is a $n \times d$ data matrix defined by $\mathbf{x} = \{(x_{ij}); i \in I, j \in J\}$, where I is a set of n objects and J a set of d variables. We will study the block clustering problem in embedding it in the mixture approach.

We will review the *block mixture model* (Govaert and Nadif, 2003) which takes into account the block clustering situation and describe a co-clustering algorithm. This one is based on the alternated application of Classification EM (Celeux and Govaert, 1992) on intermediate data matrices. To propose this algorithm, we set this problem in the classification maximum likelihood (CML) approach (Symons, 1981).

The paper is organized as follows. In Section 2, we review the *Crobin* algorithm proposed by Govaert (1983) for block clustering binary data. In Section 3, we give the necessary background CML approach and describe the different steps of the CEM algorithm. In Section 4, as we focus on binary data, we start by recalling the block general Bernoulli model and describe the associated block CEM algorithm. Section 5 is devoted to parsimonious models obtained by imposing some constraints on the general Bernoulli mixture model. Some numerical experiments are reported in the Section 6. Finally, the last section summarizes the main points of this paper.

For convenience, we represent a partition of I into g clusters by $\mathbf{z} = (\mathbf{z}_1, \ldots, \mathbf{z}_n)$ where $\mathbf{z}_i$, which indicates the component of the row i, is represented by $\mathbf{z}_i = (z_{i1}, \ldots, z_{ig})$ with $z_{ik} = 1$ if row i is in cluster k and 0 otherwise. Then, the k^{th} cluster corresponds to the set of rows i such that $z_{ik} = 1$. The term $n_{k.} = \sum_i z_{ik}$ denotes the cardinality of the k^{th} cluster. We will use similar notation for a partition $\mathbf{w}$ into m clusters of the set J and the term $n_{.\ell} = \sum_j w_{j\ell}$ denotes the cardinality of the ℓ^{th} cluster. In the following, to simplify the notation, the sums and the products relating to rows, columns, row clusters or column clusters will be subscripted respectively by letters i, j, k or ℓ without indicating the limits of variation, which will be thus implicit. For example, the sum $\sum_i$ stands for $\sum_{i=1}^{r}$ and $\sum_{i,j,k,\ell}$ stands for $\sum_{i=1}^{r} \sum_{j=1}^{s} \sum_{k=1}^{g} \sum_{\ell=1}^{m}$.

2 *Crobin* algorithm

When the data are binary, we can define the values of $\mathbf{x}$ by $x_{ij} = 1$ if the object i possesses the j^{th} attribute and $x_{ij} = 0$ otherwise. Given $\mathbf{x}$, the problem consists in optimizing the following criterion

$$W(\mathbf{z}, \mathbf{w}, \mathbf{a}) = \sum_{i,j,k,\ell} z_{ik} w_{j\ell} |x_{ij} - a_{k\ell}|,$$

where $a_{k\ell} \in \{0, 1\}$. To this end, Govaert (1983) has proposed the *Crobin* algorithm described hereafter.

1. Start from an initial position $(\mathbf{z}^{(0)}, \mathbf{w}^{(0)}, \mathbf{a}^{(0)})$.
2. Compute $(\mathbf{z}^{(c+1)}, \mathbf{a}^{(c+1)})$ starting from $(\mathbf{z}^{(c)}, \mathbf{w}^(c), \mathbf{a}^{(c)})$:
   (a) Compute $\mathbf{z}^{(c+1)}, \mathbf{a}^{(c+\frac{1}{2})}$ starting from $\mathbf{z}^{(c)}, \mathbf{a}^{(c)}$.
   (b) Compute $\mathbf{w}^{(c+1)}, \mathbf{a}^{(c+1)}$ starting from $\mathbf{w}^{(c)}, \mathbf{a}^{(c+\frac{1}{2})}$.
3. Iterate the steps 2 until the convergence.

In steps 2(a) and 2(b), for finding optimal partitions $\mathbf{z}^{(c+1)}$ and $\mathbf{w}^{(c+1)}$, we used respectively the dynamic cluster algorithm (Diday et al. (1980)) to optimize the following criteria deduced from $W(\mathbf{z},\mathbf{w},\mathbf{a})$

$$W(\mathbf{z},\mathbf{a}|\mathbf{w}) = \sum_{i,k} z_{ik} \sum_{\ell} |u_{i\ell} - n_{.\ell} a_{k\ell}|,$$

where $u_{i\ell} = \sum_j w_{j\ell} x_{ij}$, and

$$W(\mathbf{w},\mathbf{a}|\mathbf{z}) = \sum_{j,\ell} w_{j\ell} \sum_{k} |v_{kj} - n_{k.} a_{k\ell}|,$$

where $v_{kj} = \sum_i z_{ik} x_{ij}$.

The step 2(a) is carried out by the application of the dynamic cluster algorithm using the $n \times m$ matrix $(u_{i\ell})$, the L_1 distance and kernels of the form $(n_{.1}a_{k1},\ldots,n_{.m}a_{km})$. Alternatively, the step 2(b) is carried out by the application of the dynamic cluster algorithm using the $g \times d$ matrix (v_{kj}), the L_1 distance and kernels of the form $(n_{1.}a_{1\ell},\ldots,n_{g.}a_{g\ell})$. Thus, at the convergence, we obtain homogeneous blocks of 0 or 1 by reorganizing rows and columns according to the partitions $\mathbf{z}$ and $\mathbf{w}$. Hence, each block $\mathbf{x}_{k\ell}$, defined by the elements x_{ij} for i belonging to the k^{th} cluster and j to the ℓ^{th} cluster is characterized by $a_{k\ell}$ which is the highest frequency value. To interpret the results, some empirical statistics can be performed to evaluate the quality of the partition into blocks. For instance, we can define easily values $(1-\varepsilon_{k\ell})$, each one of them corresponds to the proportion of block $\mathbf{x}_{k\ell}$ values equal to $a_{k\ell}$ and measures therefore the degree of homogeneity of $\mathbf{x}_{k\ell}$.

Note that different variants of this algorithm can be proposed, for example (1) start from an initial position $(\mathbf{z}^{(0)},\mathbf{w}^{(0)},\mathbf{a}^{(0)})$ (2) Compute $(\mathbf{z}^{(c+1)},\mathbf{w}^{(c+1)},\mathbf{a}^{(c+1)}$ starting from $(\mathbf{z}^{(c)},\mathbf{w}^{(c)},\mathbf{a}^{(c)})$: (2a) Compute $\mathbf{z}^{(c+1)}$ starting from $(\mathbf{w}^{(c)},\mathbf{a}^{(c)})$ (2b) Compute $\mathbf{w}^{(c+1)}$ starting from $(\mathbf{z}^{(c+1)},\mathbf{a}^{(c)})$ (2c) Compute $\mathbf{a}^{(c+1)}$ starting from $(\mathbf{z}^{(c+1)},\mathbf{w}^{(c+1)})$. (3) Iterate the steps (2) until the convergence. ¿From our experiments, this version and *Crobin* give similar results.

One of the advantages of block clustering methods is to summary the initial data matrix $\mathbf{x}$ by a simpler data matrix $(a_{k\ell})$ having the same structure. Moreover, these methods are scalable. On the other hand, as for the k-means method the hypothesis which are implicitly supposed are often ignored. Here, for instance, the results of the *Crobin* algorithm are bad when the proportions of partitions are dramatically different which leads to think that *Crobin* assumes equal proportions of clusters. Moreover the degree of homogeneity obtained are generally very close. It is one of the aims of this paper to explore this aspect and to propose a general framework allowing to formalize the hypothesis we need to define the block clustering approach. Some parsimonious models will be described and allows us to take into account particular constraints on the proportions and the degree of homogeneity. Numerical experiments will be presented to show the interest to handle different variants of a model in the block clustering context.

3 Mixture model and clustering

In the model-based clustering (see for instance McLachlan and Peel, 2000), it is assumed that the data are generated by a mixture of underlying probability distributions, where each component k of the mixture represents a cluster. Thus, the density of the observed data $\mathbf{x}$ is expressed as

$$f(\mathbf{x};\boldsymbol{\theta}) = \prod_i \sum_k \pi_k \varphi_k(\boldsymbol{x}_i;\boldsymbol{\alpha}_k) \tag{1}$$

with $\boldsymbol{x}_i = (x_{i1},\ldots,x_{id})$ and $\boldsymbol{\theta} = (\pi_1,...,\pi_g,\boldsymbol{\alpha}_1,...,\boldsymbol{\alpha}_g)$ where $(\pi_1,...,\pi_g)$ are the mixing proportions and $(\boldsymbol{\alpha}_1,...,\boldsymbol{\alpha}_g)$ the parameters of the density components.

The clustering problem can be studied under mixture model using two different approaches: the maximum likelihood (ML) approach and the classification maximum likelihood (CML) approach (Symons, 1981). In this paper we focus on the second approach.

The ML approach estimates the parameters of the mixture and the partition is derived from these parameters using the maximum a posteriori principle (MAP). In the CML approach, the partition is added to the parameters to be estimated. The maximum likelihood estimation of these new parameters leads to optimize in $\boldsymbol{\theta}$ and $\mathbf{z}$ the complete data log-likelihood

$$L_C(\mathbf{z},\boldsymbol{\theta}) = L(\boldsymbol{\theta};\mathbf{x},\mathbf{z}) = \log f(\mathbf{x},\mathbf{z};\boldsymbol{\theta}) = \sum_{i,k} z_{ik} \log\left(\pi_k \varphi_k(\boldsymbol{x}_i;\alpha_k)\right).$$

This optimization can be done by the Classification EM (CEM) algorithm (Celeux and Govaert, 1992), a variant of EM (Dempster, Laird and Rubin, 1977), which converts the posterior probabilities t_{ik}'s to a discrete classification in a C-step before performing the M-step.

In clustering context, the use of the mixture model deals to find the component from which each row arises. The CEM algorithm allows us to achieve this goal and the different steps of CEM in this situation are

- E-step: compute the posterior probabilities $t_{ik}^{(c)} \propto \pi_k\varphi_k(\boldsymbol{x}_i;\boldsymbol{\alpha}_k)$;
- C-step: the k^{th} cluster of $\mathbf{z}^{(c+1)}$ is defined with $z_{ik}^{(c+1)} = 1$ if $k = \text{argmax}_{k=1,\ldots,g}\; t_{ik}^{(c)}$ and $z_{ik}^{(c+1)} = 0$ otherwise;
- M-step: by standard calculations, one arrives at the following re-estimations parameters $\pi_k^{(c+1)} = \frac{n_{k.}^{(c+1)}}{n}$ where $n_{k.}^{(c+1)}$ is the cardinality of the k^{th} cluster of $\mathbf{z}^{(c+1)}$, and $\boldsymbol{\alpha}_k$ which depends on the used distribution.

4 Block mixture model for binary data

To study the block clustering problem, we have extended (Govaert and Nadif, 2003) the mixture model to propose a block mixture model defined by the following probability density function (pdf)

$$f(\mathbf{x};\boldsymbol{\theta}) = \sum_{(\mathbf{z},\mathbf{w})\in\mathcal{Z}\times\mathcal{W}} \prod_i \pi_{z_i} \prod_j \rho_{w_j} \prod_{i,j} \varphi(x_{ij};\boldsymbol{\alpha}_{z_i w_j})$$

where $\boldsymbol{\theta} = (\boldsymbol{\pi},\boldsymbol{\rho},\boldsymbol{\alpha}_{11},\ldots,\boldsymbol{\alpha}_{gm})$, $\boldsymbol{\pi} = (\pi_1,\ldots,\pi_g)$ and $\boldsymbol{\rho} = (\rho_1,\ldots,\rho_m)$ are the mixing proportions and $\varphi(x,\boldsymbol{\alpha}_{k\ell})$ is a pdf defined on the real set $\mathbb{R}$. In our situation, we assume that for each block $k\ell$ the values x_{ij} are distributed according the Bernoulli distribution $\mathcal{B}(\alpha_{k\ell})$ for which the probability mass function is $\varphi_{k\ell}(x_{ij};\alpha_{k\ell}) = (\alpha_{k\ell})^{x_{ij}}(1-\alpha_{k\ell})^{(1-x_{ij})}$.

To tackle the simultaneous partitioning problem, we will use the CML approach, which aims to maximize the complete data log-likelihood associated to the block mixture model. With our model, the complete data are $(\mathbf{z},\mathbf{w},\mathbf{x})$ and the classification log-likelihood is given by

$$\begin{aligned} L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta}) = &\sum_{i,k} z_{ik}\log\pi_k + \sum_{j,\ell} w_{j\ell}\log\rho_\ell \\ &+ \sum_{i,k,j,\ell} z_{ik}w_{j\ell}\log\frac{\alpha_{k\ell}}{1-\alpha_{k\ell}} + \sum_{k,\ell} n_{k.}n_{.\ell}\log(1-\alpha_{k\ell}) \end{aligned}$$

The maximization of the L_c criterion can be performed with an alternated optimization using the following maximizations:

1. Maximization of $L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta})$ w.r. to $\mathbf{z}$ for fixed $\boldsymbol{\theta}$ and $\mathbf{w}$: $L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta})$ can be written $\sum_{i,k} z_{ik}A_{ik} + \sum_\ell n_{.\ell}\log\rho_\ell$ where

$$A_{ik} = \log\pi_k + \sum_\ell u_{i\ell}\log\frac{\alpha_{k\ell}}{1-\alpha_{k\ell}} + \sum_\ell n_{.\ell}\log(1-\alpha_{k\ell}),$$

   $u_{i\ell} = \sum_j w_{j\ell}x_{ij}$ and $n_{k.} = \sum_i z_{ik}$. It can be easily shown that $z_{ik}=1$ if $k = \mathrm{argmax}_k A_{ik}$ and 0 otherwise $\forall i$.
2. Maximization of $L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta})$ w.r. to $\mathbf{w}$ for fixed $\boldsymbol{\theta}$ and $\mathbf{z}$: in a similar way, $L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta})$ can be written $\sum_{j,\ell} w_{j\ell}B_{j\ell} + \sum_k n_{k.}\log\pi_k$ where

$$B_{j\ell} = \log\rho_\ell + \sum_k v_{jk}\log\frac{\alpha_{k\ell}}{1-\alpha_{k\ell}} + \sum_k n_{k.}\log(1-\alpha_{k\ell}),$$

   $v_{jk} = \sum_i z_{ik}x_{ij}$ and $n_{.\ell} = \sum_j w_{j\ell}$. It can be easily shown that $w_{j\ell}=1$ if $\ell = \mathrm{argmax}_\ell B_{j\ell}$ and 0 otherwise $\forall j$.
3. Maximization of $L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta})$ w.r. to $\boldsymbol{\theta}$ for fixed $\mathbf{z}$ and $\mathbf{w}$: $L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta})$ takes the following form

$$\sum_k n_{k.}\log\pi_k + \sum_\ell n_{.\ell}\log\rho_\ell + \sum_{k\ell}\left(y_{k\ell}\log\frac{\alpha_{k\ell}}{1-\alpha_{k\ell}} + n_{k.}n_{.\ell}\log(1-\alpha_{k\ell})\right)$$

   where $y_{k\ell} = \sum_{ij} z_{ik}w_{j\ell}x_{ij}$. We obtain $\pi_k = \frac{n_{k.}}{n}$, $\rho_\ell = \frac{n_{.\ell}}{d}$ and $\alpha_{k\ell} = \frac{y_{k\ell}}{n_{k.}n_{.\ell}}$.

The two first maximizations (1 and 2) correspond to the E-step and C-step of the CEM algorithm, they allow us to construct the pair partition $(\mathbf{z},\mathbf{w})$ before the M-step (3). We obtain a block version of CEM called block CEM.

5 Parsimonious Bernoulli models

From the classical Gaussian mixture model, Banfield and Raftery (1995) have proposed different variants of this model. They have considered a parametrization of the covariance matrix Σ_k in terms of its eigenvalue decomposition, $\Sigma_k = \lambda_k D_k A_k D_k^T$ (the superscript T denotes matrix transposition), where λ_k defines the volume of the k^{th} cluster, D_k is an orthogonal matrix, which defines its orientation and A_k is a diagonal matrix with determinant 1, which defines its shape. This parametrization allows one to propose many general criteria and the simplest one corresponds to spherical clusters and equal volumes that lead to the famous k-means criterion. From our block Bernoulli models, we can apply a similar parametrization on the parameters $\alpha_{k\ell}$. Next, we propose some parsimonious models.

5.1 Model $[\varepsilon_{k\ell}]$

The Bernoulli parameter can be break down into two parameters $a_{k\ell}$ and $\varepsilon_{k\ell}$. With this formulation $a_{k\ell}$ indicates the majority value in the block $k\ell$ and $\varepsilon_{k\ell}$ the degree of homogeneity. The Bernoulli pdf can be written

$$\varphi_{k\ell}(x_{ij};(a_{k\ell},\varepsilon_{k\ell})) = (\varepsilon_{k\ell})^{|x_{ij}-a_{k\ell}|}(1-\varepsilon_{k\ell})^{1-|x_{ij}-a_{k\ell}|}$$

if we replace each $\alpha_{k\ell}$ by $a_{k\ell} \in \{0,1\}$ and $\varepsilon_{k\ell} \in [0,1/2]$ with

$$\begin{cases} a_{k\ell} = 1 \text{ and } \varepsilon_{k\ell} = 1-\alpha_{k\ell} \text{ if } \alpha_{k\ell} \in [1/2,1] \\ a_{k\ell} = 0 \text{ and } \varepsilon_{k\ell} = \alpha_{k\ell} \text{ if } \alpha_{k\ell} \in [0,1/2(. \end{cases}$$

After simple calculations, the associated complete data log-likelihood can be written as

$$\begin{aligned} L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta}) = & \sum_{i,j,k,\ell} z_{ik}w_{j\ell}|x_{ij}-a_{k\ell}|\log\frac{\varepsilon_{k\ell}}{1-\varepsilon_{k\ell}} \\ & + \sum_{i,j,k,\ell} z_{ik}w_{j\ell}\log(1-\varepsilon_{k\ell}) \\ & + \sum_k n_{k.}\log\pi_k + \sum_\ell n_{.\ell}\log\rho_\ell, \end{aligned} \tag{2}$$

and the different steps of the algorithm become

1. Maximization of $L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta})$ w.r. to $\mathbf{z}$ for fixed $\boldsymbol{\theta}$ and $\mathbf{w}$: the term A_{ik} can be written

$$A_{ik} = \log\pi_k + \sum_\ell |u_{i\ell} - n_{.\ell}a_{k\ell}|\log\frac{\varepsilon_{k\ell}}{1-\varepsilon_{k\ell}} + \sum_\ell n_{.\ell}\log(1-\varepsilon_{k\ell}).$$

2. Maximization of $L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta})$ w.r. to $\mathbf{w}$ for fixed $\boldsymbol{\theta}$ and $\mathbf{z}$: the term $B_{j\ell}$ can be written

$$B_{j\ell} = \log \rho_\ell + \sum_k |v_{kj} - n_{k.}a_{k\ell}| \log \frac{\varepsilon_{k\ell}}{1-\varepsilon_{k\ell}} + \sum_k n_{k.} \log(1-\varepsilon_{k\ell}).$$

3. Maximization of $L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta})$ w.r. to $\boldsymbol{\theta}$ for fixed $\mathbf{z}$ and $\mathbf{w}$: $L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta})$ can be written

$$\begin{aligned}&\sum_k n_{k.} \log \pi_k + \sum_\ell n_{.\ell} \log \rho_\ell \\ &+ \sum_{k\ell} \left(|y_{k\ell} - n_{k.}n_{.\ell}a_{k\ell}| \log \frac{\varepsilon_{k\ell}}{1-\varepsilon_{k\ell}} + n_{k.}n_{.\ell} \log(1-\varepsilon_{k\ell}) \right).\end{aligned}$$

Then we obtain $\pi_k = \frac{n_{k.}}{n}$, $\rho_\ell = \frac{n_{.\ell}}{d}$, $a_{k\ell} = 0$ if $\frac{y_{k\ell}}{n_{k.}n_{.\ell}} < 0.5$ and 1 otherwise and $\varepsilon_k = \frac{|y_{k\ell} - n_{k.}n_{.\ell}a_{k\ell}|}{n_{k.}d}$.

5.2 Model $[\varepsilon_k]$

In this model, we impose that the $\varepsilon_{k\ell}$'s of the k^{th} cluster are equal for $\ell = 1,\ldots,m$, then L_c becomes

$$\begin{aligned}L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta}) = &\sum_{i,k} z_{ik} \log \pi_k + \sum_{i,k} w_{j\ell} \log \rho_\ell \\ &+ \sum_{i,j,k,\ell} z_{ik}w_{j\ell} \left(|x_{ij} - a_{k\ell}| \log \frac{\varepsilon_k}{1-\varepsilon_k} + \log(1-\varepsilon_k) \right),\end{aligned}$$

and the different steps of the algorithm are

1. Maximization of $L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta})$ w.r. to $\mathbf{z}$ for fixed $\boldsymbol{\theta}$ and $\mathbf{w}$: the term A_{ik} can be written $A_{ik} = \log \pi_k + \log \frac{\varepsilon_k}{1-\varepsilon_k} \sum_\ell |u_{i\ell} - n_{.\ell}a_{k\ell}| + d \log(1-\varepsilon_k)$.
2. Maximization of $L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta})$ w.r. to $\mathbf{w}$ for fixed $\boldsymbol{\theta}$ and $\mathbf{z}$: the term $B_{j\ell}$ can be written $B_{j\ell} = \log \rho_\ell + \sum_k |v_{kj} - n_{k.}a_{k\ell}| \log \frac{\varepsilon_k}{1-\varepsilon_k} + \sum_k n_{k.} \log(1-\varepsilon_k)$.
3. Maximization of $L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta})$ w.r. to $\boldsymbol{\theta}$ for fixed $\mathbf{z}$ and $\mathbf{w}$: $L_c(\mathbf{z},\mathbf{w},\boldsymbol{\theta})$ can be written as

$$\begin{aligned}&\sum_k n_{k.} \log \pi_k + \sum_\ell n_{.\ell} \log \rho_\ell \\ &+ \sum_{k,\ell} \left(|y_{k\ell} - n_{k.}n_{.\ell}a_{k\ell}| \log \frac{\varepsilon_k}{1-\varepsilon_k} + n_{k.}n_{.\ell} \log(1-\varepsilon_k) \right).\end{aligned}$$

The parameters of the model are defined in this case by $\pi_k = \frac{n_{k.}}{n}$, $\rho_\ell = \frac{n_{.\ell}}{d}$, $a_{k\ell} = 0$ if $\frac{y_{k\ell}}{n_{k.}n_{.\ell}} < 0.5$ and 1 otherwise and $\varepsilon_k = \frac{\sum_\ell |y_{k\ell} - n_{k.}n_{.\ell}a_{k\ell}|}{n_{k.}d}$.

Note that the same model can be used when the parameter $\boldsymbol{\varepsilon}$ depends only on $\mathbf{w}$. It suffices to consider the data matrix $\mathbf{x}^T$.

5.3 Model $[\varepsilon]$

With this model, the expression of $L_c(\mathbf{z}, \mathbf{w}, \boldsymbol{\theta})$ takes the form

$$\log \frac{\varepsilon}{1-\varepsilon} \, W(\mathbf{z}, \mathbf{w}, \mathbf{a}) + D,$$

where $W(\mathbf{z}, \mathbf{w}, \mathbf{a}) = \sum_{i,j,k,\ell} z_{ik} w_{j\ell} |x_{ij} - a_{k\ell}|$ is the criterion optimized by *Crobin* and D, equal to $nd \log(1-\varepsilon) - n \log g - d \log m$, does not depend on $(\mathbf{z}, \mathbf{w})$. So that, maximizing $L_c(\mathbf{z}, \mathbf{w}, \theta)$ is equivalent to maximizing $\log \frac{\varepsilon}{1-\varepsilon} W(\mathbf{z}, \mathbf{w}, \mathbf{a})$ or minimizing $W(\mathbf{z}, \mathbf{w}, \mathbf{a})$ because $\log \frac{\varepsilon}{1-\varepsilon} \leq 0$.

Notice that at the convergence, the parameter ε can be estimated from the optimal pair $(\mathbf{z}^*, \mathbf{w}^*)$ by $\sum_{i,j,k,\ell} z^*_{ik} w^*_{j\ell} |x_{ij} - a_{k\ell}|/nd$ and $(1-\varepsilon)$ represents the global degree of homogeneity. The value 1 corresponds to a perfect clustering into blocks. In steps 2(a) and 2(b), for finding an optimal $\mathbf{z}^{(c+1)}$ and $\mathbf{w}^{(c+1)}$, we used respectively the dynamic cluster algorithm to optimize the following criteria $W(\mathbf{z}, \mathbf{a}|\mathbf{w}) = \sum_{i,k} z_{ik} \sum_{\ell} |u_{i\ell} - n_{.\ell} a_{k\ell}|$, where $u_{i\ell} = \sum_j w_{j\ell} x_{ij}$, and $W(\mathbf{w}, \mathbf{a}|\mathbf{z}) = \sum_{j,\ell} w_{j\ell} \sum_k |v_{kj} - n_{k.} a_{k\ell}|$, where $v_{kj} = \sum_i z_{ik} x_{ij}$.

6 Synthetic data

To illustrate these models and the performance of associated algorithms, we carried out Monte Carlo simulations. In our experiments we selected nine types of data arising from 3×2-component mixture model corresponding to three degrees of cluster overlap (well separated (+), moderately separated (++) and ill-separated (+++)), and three data dimensions ($n \times d = 50 \times 30$, $n \times d = 100 \times 60$ and $n \times d = 200 \times 120$).

Parameters were selected so as to obtain error rates respectively in $[.01, .05]$ for the well-separated, in $[.12, .17]$ for the moderately and in $[.20, .24]$ for the ill-separated situations. For each of these 27 data structures we generated 30 samples, for each sample we ran the block CEM algorithm according the models $[\varepsilon]$, $[\varepsilon_k]$ and $[\varepsilon_{k\ell}]$ 500 times starting from random situations, and then selected the best solution for each variant in comparing the obtained partitions and the simulated ones. In our experiments the used parameters are $\boldsymbol{\pi} = (0.2, 0.3, 0.5)$, $\boldsymbol{\rho} = (0.7, 0.3)$ and $\mathbf{a} = (1, 0; 0, 1; 1, 1)$. Moreover, we consider the following situations:

- M1: $(\varepsilon_{k\ell} = \varepsilon; \forall k, \ell)$,
- M2: $(\varepsilon_{k\ell} = \varepsilon_k; \forall \ell)$,
- M3: $(\varepsilon_{k\ell})$ without constraint.

The simulation results are summarized in Table 4 which displays the error rates for each situation. The main findings arising from these first experiments are:

- It appears clearly that the simplest model $[\varepsilon]$ is interesting when the size of data is small (see, $n \times d = (50, 30)$). It outperforms $[\varepsilon_k]$ and $[\varepsilon_{k\ell}]$

in all situations of overlap even when the data are simulated according these models. This difference is reduced when the size increases (see, $n \times d = (100, 60), (200, 120)$) and the performances of $[\varepsilon_k]$ and $[\varepsilon_{k\ell}]$ are close. Furthermore, when the clusters are not separated and the degrees of overlap are dramatically different, the model $[\varepsilon]$ has difficulties to propose a clustering into 3×2 blocks.

- When the data size is medium ($n \times d = (100, 60)$), the model $[\varepsilon_k]$ is frequently better than $[\varepsilon]$ when the data are not distributed according M1.
- When the data are large enough and without any information about the clusters, the general model $[\varepsilon_{k\ell}]$ could be recommended.

Table 1. Means and standard errors (in parentheses) of error rates from the same random situations by the block CEM algorithm applied with the three models on data simulated according M1, M2 and M3.

size	degree of overlap	M1			M2			M3		
		$[\varepsilon]$	$[\varepsilon_k]$	$[\varepsilon_{k\ell}]$	$[\varepsilon]$	$[\varepsilon_k]$	$[\varepsilon_{k\ell}]$	$[\varepsilon]$	$[\varepsilon_k]$	$[\varepsilon_{k\ell}]$
(50,30)	+	.03(.01)	.06(.01)	.07(.04)	.08(.05)	.08(.05)	.10(.05)	.09(.03)	.12(.07)	.13(.08)
	++	.14(.04)	.26(.13)	.34(.19)	.15(.04)	.18(.07)	.19(.07)	.20(.05)	.27(.12)	.29(.13)
	+++	.23(.01)	.35(.13)	.36(.10)	.24(.03)	.26(.05)	.32(.05)	.27(.10)	.42(.14)	.39(.15)
(100,60)	+	.04(.00)	.05(.01)	.05(.03)	.04(.01)	.03(.01)	.04(.03)	.09(.03)	.05(.05)	.04(.08)
	++	.15(.02)	.18(.02)	.20(.05)	.13(.03)	.12(.03)	.15(.03)	.13(.03)	.12(.03)	.15(.08)
	+++	.23(.03)	.26(.08)	.31(.12)	.26(.04)	.27(.06)	.31(.06)	.25(.04)	.30(.06)	.29(.07)
(200,120)	+	.04(.01)	.02(.00)	.02(.00)	.02(.01)	.02(.01)	.01(.01)	nr*	.02(.01)	.01(.00)
	++	.15(.01)	.16(.02)	.18(.05)	.18(.03)	.17(.02)	.17(.03)	nr	.14(.03)	.14(.02)
	+++	.22(.02)	.22(.02)	.27(.05)	.24(.03)	.24(.03)	.30(.05)	nr	.32(.06)	.27(.08)

*nr indicates no result.

7 Conclusion

Setting the problem of block clustering under the CML approach, we have proposed parsimonious Bernoulli block models. Then we have compared the different associated block CEM algorithms. Numerical experiments show the interest to handle different variants of a model in the block clustering context according the size data. For the small size, the simplest is recommended. When we have not any information about the degree of overlap according the clusters and the size of data is large enough, the general model is appropriated. Then it will be necessary to study the problem of the choice of the model. In this paper, we have considered the block clustering for binary data under the CML approach and, as in Govaert and Nadif (2005, 2006), it would be interesting to study the block clustering approach under the ML and fuzzy approaches and to extend to other models.

References

ARABIE, P. and HUBERT, L.J. (1990): The bond energy algorithm revisited. *IEEE Transactions on Systems, Man, and Cybernetics 20, 268-274.*

BANFIELD J.D. and RAFTERY, A.E. (1993): Model-based Gaussian and non Gaussian clustering. *Biometrics 49, 803-821.*

BOCK, H.-H. (1979): Simultaneous clustering of objects and variables. In: E. Diday (Eds): *Analyse des Données et Informatique.* INRIA, 187–203.

CELEUX, G. and GOVAERT, G. (1992): A classification EM algorithm for clustering and two stochastic versions. *Computational Statistics and Data Analysis 4, 315–332.*

DEMPSTER, A.P., LAIRD, N.M. and RUBIN, D.B. (1977): Maximum likelihood from incomplete data via the EM algorithm (with discussion). *Journal of the Royal Statistical Society B 39, 1-38.*

DIDAY, E. (1971): Une Nouvelle Méthode en Classification Automatique et Reconnaisance des Formes *Revue de Statistique Appliquée, 19, 2.*

DIDAY, E., SCHROEDER A. and OK, Y. (1974): The Dynamic Clusters Method in Pattern Recognition. *Proceedings of IFIP Congress.*

DIDAY, E. and coll. (1980): *Optimisation en Classification Automatique.* Le Chesnay INRIA.

FORGY, E.W. (1965): Cluster Analysis of Multivariate Data: Efficiency versus Interpretability of Classification. *Biometrics, 21 (3), 768-769.*

GOVAERT, G. (1983): *Classification croisée.* Thèse d'état, Université Paris 6, France.

GOVAERT, G. (1984): Classification de tableaux binaires. In: E. Diday (Eds): *Data analysis and informatics 3.* Amsterdam, North-Holland, 223-236.

GOVAERT, G. (1995): Simultaneous clustering of rows and columns. *Control and Cybernetics 24, 437-458.*

GOVAERT, G. and NADIF, M. (2003): Clustering with block mixture models. *Pattern Recognition 36, 463-473.*

GOVAERT, G. and NADIF, M. (2005): An EM algorithm for the block mixture model. *IEEE Transactions on Pattern Analysis and Machine Intelligence 27, 643-647.*

GOVAERT, G. and NADIF, M. (2006): Fuzzy Clustering to estimate the parameters of block mixture models. *Soft Computing, 415-422.*

HARTIGAN, J.A. (1975): *Clustering Algorithms.* Wiley & Sons, New York.

MARCHOTORCHINO, F. (1987): Block seriation problems: A unified approach. *Applied Stochastic Models and Data Analysis 3, 73-91.*

MCLACHLAN, G.J. and PEEL, D. (2000): *Finite Mixture Models.* Wiley & Sons, New York.

RITSCHARD, G., ZIGHED, D. and NICOLOYANNIS, N. (2001): Maximisation de l'association par regroupement de lignes ou de colonnes d'un tableau croisé. *Revue de Mathmatiques & Sciences Humaines 39, 81-97.*

SYMONS, M.J. (1981): Clustering criteria and multivariate normal mixture. *Biometrics 37, 35-43.*

Cluster Analysis Based on Posets [⋆]

Melvin F. Janowitz

DIMACS, Rutgers University
96 Frelinghuysen Road, Piscataway, NJ 08854-8018, USA *melj@dimacs.rutgers.edu*

Abstract. When dissimilarities are measured in a space other than the reals, it is argued that previous models for cluster analysis are not adequate. Possible new models will be explored. It is also shown that formal concept analysis may be viewed as a special case of a Boolean dissimilarity coefficient. A persistent underlying theme involves generalized notions of adjoints of order preserving mappings between posets.

1 Background

A basic knowledge of cluster analysis will be assumed at the outset. This can be obtained by consulting where needed one of the standard references (Gordon (1999), Jain and Dubes (1988), Mirkin (1996)). Any needed background from the theory of partially ordered sets may be obtained from Birkhoff (1967), Davey and Priestley (1990), Szasz (1963). An interesting mathematical model for cluster analysis was presented in Jardine and Sibson (1971). We shall not reproduce it here, but do point out that the current discussion has its origins in that text. The basic input to a clustering algorithm is a finite nonempty set E equipped with a finite collection of attributes that the members of E may possess. These attributes can be numerical, nominal or binary. The attributes are then converted to a *dissimilarity coefficient* (DC). This is a mapping $d\colon E \times E \to \Re_0^+$, the non-negative reals, that satisfies $d(a,b) = d(b,a)$, and $d(a,a) = 0$ for all $a, b \in E$. To say that d is *definite* is to say that also $d(a,b) = 0 \Longrightarrow a = b$. Finally, the DC d is an *ultrametric* if it also satisfies

$$d(a,b) \leq \max\{d(a,c), d(b,c)\} \text{ for all } a, b, c \in E.$$

The idea is that lower values of d correspond to pairs of members of E that are more similar (i.e., less dissimilar).

The T-transform: Let $\Sigma(E)$ denote the set of reflexive symmetric relations on E, ordered by set inclusion. Associated with any DC d, there is a

[⋆] Talks based on preliminary versions of short portions of this paper were given at Ecole Nationale Supérieure des Télécommunications de Bretagne on October 30, 2004, at DIMACS on March 9, 2005, at the SFC meeting at the Université du Quebec à Montréal on May 31, 2005, at the IFCS meeting at the University of Ljubljana in August, 2006, at the University of Louisville on September 27, 2006 and at the joint BCS-CSNA meeting at Brunel University on February 2, 2007.

mapping $Td\colon \Re_0^+ \to \Sigma(E)$ defined by $Td(h) = \{(a,b) : d(a,b) \le h\}$ for all $h \in \Re_0^+$. It is easy to show and well known that $Td(h)$ is an equivalence relation for all $h \in \Re_0^+$ if and only if d is an ultrametric. Thus ultrametrics yield nested sequences of equivalence relations. For that reason, a cluster algorithm may be viewed as a transformation $d \mapsto C(d)$ of a DC d into an ultrametric $C(d)$.

In Janowitz (1978), we replaced $\Re_0^+$ with a join semilattice L having a smallest member 0 having more than one member. The *join* of x and y is denoted by the symbol $x \vee y$. We defined an L-dissimilarity coefficient to be a mapping $d\colon E \times E \to L$ such that $d(a,b) = d(b,a)$, and $d(a,a) = 0$ for all $a, b \in E$. d is *definite* if also $d(a,b) = 0 \Longrightarrow a = b$. Finally, the DC d is an *ultrametric* if it also satisfies $d(a,b) \le d(a,c) \vee d(b,c)$ for all $a, b, c \in E$. The T-transform associated with an L-dissimilarity coefficient d is defined as expected by taking it to be the mapping $Td\colon L \to \Sigma(E)$ defined by $Td(h) = \{(a,b) : d(a,b) \le h\}$ for all $h \in L$. This was the original setting, but it is not quite what is needed.

Single-linkage clustering is one of the standard clustering algorithms. Here is how it operates. If $u = C(d)$, we take $Tu(h) = \gamma \circ Td(h)$, where γ is the transitive closure operator. It was shown in Janowitz (1978) that if u is to be an ultrametric and if $h \wedge k$ exists in L, then $Tu(h \wedge k)$ must equal $Tu(h) \cap Tu(k)$. This says that $\gamma \circ Td(h \wedge k)$ must equal $\gamma \circ Td(h) \cap \gamma \circ Td(k)$. But γ defined on $\Sigma(E)$ does not have this property (Janowitz (1978), Lemma 6.1, p. 65). Thus we must either abandon single linkage clustering as a cluster method or modify our model.

2 The modified model

We choose here to change our perspective a bit. First of all, we assume nothing past the fact that the place L in which dissimilarities are measured should be a partially ordered set (poset) having a smallest member 0 and having more than one member. The idea behind the concept of a dissimilarity coefficient is that $d(a,b)$ should provide an *ordinal* measure in L of the dissimilarity between a and b. To say that $d(a,b) \le d(x,y)$ is to say that the pair $\{a,b\}$ is more similar (less dissimilar) then the pair $\{x,y\}$. Another way of viewing this is that at level $d(a,b)$, we deem the pair $\{a,b\}$ to be a *candidate* for clustering. If $h \ge d(a,b)$, we want (a,b) also to be a candidate for clustering. The idea of a clustering algorithm then is that at each level h, it somehow decides which candidates for clustering actually get clustered. This will all be clarified later in the paper by examining a specific algorithm. The point to the model introduced in Jardine and Sibson (1971) is that a cluster algorithm may be viewed as a transformation from one type of DC to a second type. We need to modify this be introducing the notion of a *general dissimilarity coefficient.* Before doing this, we need to mention some machinery from the theory of partially ordered sets.

Order filters of L will play an important role in what follows. An *order filter* of L is a subset F of L such that $h \in \mathsf{F}$, $h \leq k \Longrightarrow k \in \mathsf{F}$. If L has a largest member, we require that any order filter be nonempty; otherwise, we allow the empty set to be an order filter. The set $\mathcal{F}(L)$ of order filters of L is ordered by the rule $\mathsf{F} \leq \mathsf{G} \Longleftrightarrow \mathsf{G} \subseteq \mathsf{F}$. This may seem strange but the point is that we want $x \mapsto \mathsf{F}_x$ to be an embedding. Here F_x is the principal filter generated by x, and is defined by $\mathsf{F}_x = \{y \in L : y \geq x\}$. Since $\mathsf{F} \vee \mathsf{G} = \mathsf{F} \cap \mathsf{G}$ and $\mathsf{F} \wedge \mathsf{G} = \mathsf{F} \cup \mathsf{G}$, it is true that $\mathcal{F}(L)$ is a complete distributive lattice with smallest element the order filter F_0, and largest element the empty filter or a filter consisting of the largest member of L.

Definition 1. The terminology *ordinary* DC will be used to denote a dissimilarity $d \colon E \times E \to L$. We now define a *general* dissimilarity coefficient to be a mapping $D \colon E \times E \to \mathcal{F}(L)$ that satisfies

(GD1) $D(a,b) = D(b,a)$.
(GD2) $D(a,a) = \mathsf{F}_0$ for all $a \in E$.

Definite DCs are those that also satisfy

(GD3) $D(a,b) = D(a,a) = D(b,b)$ implies $a = b$.

Note that we are using a capital letter D to clearly distinguish this type of DC from the usual $d \colon E \times E \to L$. It will sometimes be useful to replace (GD2) with

(GD2′) $D(a,a) = \bigwedge\{D(a,b) : b \in L,\ a \neq b\}$, or
(GD2″) Calculate $D(a,a)$ using the same formula as for $D(a,b)$ with $b \neq a$.

An ordinary DC $d \colon E \times E \to L$ has an associated general DC $D_d \colon E \times E \to \mathcal{F}(L)$ defined by $D_d(a,b) = \mathsf{F}_{d(a,b)}$, For that reason, we can be sloppy about terminology, and just use the notation d versus D to specify whether we are dealing with ordinary or general DCs. Unless otherwise specified, the reader should assume that any given DC is a general DC.

Here then is the situation for a given general DC D taking values in $\mathcal{F}(L)$. The poset L is where the dissimilarities are measured, and it is where the clusters are indexed. The DC D takes values in $\mathcal{F}(L)$ because $D(a,b)$ designates the members of L at which $\{a,b\}$ is a candidate for forming a cluster. These values naturally constitute an order filter of L. We let $\Sigma(E)$ denote the symmetric relations of E. If D satisfies (GD2), there is no harm in specifying that $\Sigma(E)$ should in fact represent the reflexive symmetric relations of E. We now define a mapping $SD \colon E \times E \to \Sigma(E)$ by the rule $SD(h) = \{(a,b) : h \in D(a,b)\}$. Thus $SD(h)$ yields the pairs (a,b) that are candidates for clustering at level h. Evidently

$$h \leq k \Longrightarrow SD(h) \subseteq SD(k).$$

Such a mapping will be called an *L-stratified clustering.* For any L-stratified clustering S, there is an associated general DC D_S defined by $D_S(a,b) = \{h \in L : (a,b) \in S(h)\}$. Evidently $D \mapsto SD$ is a bijection whose inverse is given by $S \mapsto D_S$. There is a fundamental connection between a general DC D and its associated L-stratified clustering SD. It is given by

$$h \in D(a,b) \iff (a,b) \in SD(h). \tag{1}$$

A moment's reflection should convince the reader that to say that $SD(h)$ is a transitive relation for all h is equivalent to saying that

(GD4) $D(a,b) \leq D(a,c) \cap D(b,c) = D(a,c) \vee D(b,c)$ for all $a,b,c \in E$.

The point is that $h \in D(a,c) \cap D(b,c)$ should force $h \in D(a,b)$. In terms of the binary relation $S(h)$, we are saying that $(a,c),(b,c) \in S(h)$ should imply that $(a,b) \in S(h)$. This leads us to call a general DC an *ultrametric* if it satisfies (GD1), some variant of (GD2), and (GD4). A cluster method may now be taken as a mapping $D \mapsto C(D)$, where D is a general DC and $C(D)$ an ultrametric. Please recall that there is nothing that precludes the original DC D from having the property that each $D(x,y)$ is a principal filter. Indeed, suppose $d\colon E \times E \to L$ is an ordinary DC, and D is its associated general DC. The notation is consistent because

$$(a,b) \in Td(h) \iff d(a,b) \leq h \iff h \in \mathsf{F}_{d(a,b)} = D(a,b) \iff (a,b) \in SD(h).$$

The design of useful clustering algorithms when faced with dissimilarities measured in an arbitrary poset L is of course a critical issue to be dealt with. Only rudimentary progress has been made. We deal here primarily with two types of algorithms. Those that can be lifted from existing cluster algorithms based on real-valued dissimilarities, and those that ignore L, and concentrate entirely on calculations whose input consists of the relations of the form $\{S(h) : h \in L\}$. The prime example of the latter is single-linkage clustering which takes each $S(h)$ and calculates its transitive closure.

3 Boolean dissimilarities

With the introduction of the notion of a general dissimilarity coefficient, we now have the possibility of having a dissimilarity taking values in the same space as that of the attributes specified by the data. The formation of an ordinary DC inevitably forms a summary of the attribute data, thus destroying information. By using a general DC, one can retain the underlying attribute information before doing any clustering, albeit at some cost in computing efficiency. When dealing with binary attributes, we agree to call D a Boolean dissimilarity when L is a Boolean algebra. We now consider a rather special situation. Suppose the set E is equipped with k binary attributes. We want to use these attributes to define a DC taking values in $L = 2^k$. Let $a, b \in E$ with $a \neq b$. If a has attributes $(a_1, a_2, \ldots, a_k)$ and b has attributes $(b_1, b_2, \ldots, b_k)$, we want to define a dissimilarity $D(a, b) = (d_1, d_2, \ldots, d_k) \in 2^k$, where d_i is computed entirely from a_i and b_i. How shall we do this? Since a DC is supposed to be a measure of how dissimilar a and b are, it is clear that if $a_i \neq b_i$, we want $x_i = 1$. There are now only two remaining cases to consider: $a_i = b_i = 0$ and $a_i = b_i = 1$. Eliminating the trivial case where every d_i is necessarily 1, there are only three possibilities for distinct a, b:

$D_1(a, b) = (x_1, x_2, \ldots, x_k)$ where $x_i = 1$ if $a_i \neq b_i$ and 0 otherwise.
$D_2(a, b) = (y_1, y_2, \ldots, y_k)$ where $y_i = 0$ if $a_i = b_i = 1$ and 1 otherwise.
$D_3(a, b) = (z_1, z_2, \ldots, z_k)$ where $z_i = 0$ if $a_i = b_i = 0$ and 1 otherwise.

Note that we need not really consider D_3, as D_2 and D_3 are symmetric with respect to negation of attributes. Note further that the definitions of the output DC may vary from coordinate to coordinate of 2^k. Routine computation establishes the following result. thus showing that each D_i is an ultrametric.

Theorem 1. *For* $i = 1,\ 2,$ *or* 3 *and* $h = (h_1, h_2, \ldots, h_k)$*,* $\{(a, b) : D_i(a, b) \leq h\}$ *is a transitive relation.*

An ordinary DC called the "simple matching coefficient" for objects a and b is related to our D_1. It just counts up the number of attributes in which x, y differ and divides by the total number of attributes. Thus if we compute $D_1(a, b)$ and just add up the resulting vector and divide by k, we have the result of the simple matching coefficient. There are of course other rather different ways of measuring a dissimilarity between binary attributes.

Example 1. We illustrate with an intuitive, easily understood example. Consider the set E consisting of the first nine integers. We wish to classify E by considering various properties that these integers might enjoy. These properties are the *attributes* we shall utilize. Here are some we might consider:

o	odd
s	perfect square
p	prime
c	perfect cube
t	multiple of three

This leads to the attributes presented in Table 1. The idea is that an entry of 1 indicates presence of the attribute, while 0 denotes absence.

The reader should bear in mind that cluster analysis does not give a definitive structure for a data set; it just suggests plausible clusters. The D_1 DC appears in Table 2, and the D_2 DC in Table 4. Table 3 puts in bold type the links that are made at level 11100, thus demonstrating the induced clusters {18}, {2457}, {369}. Let's look at the effect of D_1 and D_2 at level 11110. Here D_2 has as its only nontrivial cluster {369}, while D_1 has this cluster together with its complement {124578}. To obtain the clusters for 01110, one would intersect the clusters for 01111 with those for 11110.

object	o	s	p	c	t
1	1	1	0	1	0
2	0	0	1	0	0
3	1	0	1	0	1
4	0	1	0	0	0
5	1	0	1	0	0
6	0	0	0	0	1
7	1	0	1	0	0
8	0	0	0	1	0
9	1	1	0	0	1

Table 1. Attributes for the first nine integers.

D_1	1	2	3	4	5	6	7	8	9
1	00000	11110	01111	10010	01110	11011	01110	11100	00011
2	11110	00000	10001	01100	10000	00101	10000	00110	11101
3	01111	10001	00000	11101	00001	10100	00001	10111	01100
4	10010	01100	11101	00000	11100	01001	11100	01010	10001
5	01110	10000	00001	11100	00000	10101	00000	10110	01101
6	11011	00101	10100	01001	10101	00000	10101	00011	11000
7	01110	10000	00001	11100	00000	10101	00000	10110	01101
8	11100	00110	10111	01010	10110	00011	10110	00000	11011
9	00011	11101	01100	10001	01101	11000	01101	11011	00000

Table 2. Illustration of the D_1 coefficient for the nine integers.

D_1	1	2	3	4	5	6	7	8	9
1	**00000**	11110	01111	10010	01110	11011	01110	**11100**	00011
2	11110	**00000**	10001	**01100**	**10000**	00101	**10000**	00110	11101
3	01111	10001	**00000**	11101	00001	**10100**	00001	10111	**01100**
4	10010	**01100**	11101	**00000**	**11100**	01001	**11100**	01010	10001
5	01110	**10000**	00001	**11100**	**00000**	10101	**00000**	10110	01101
6	11011	00101	**10100**	01001	10101	**00000**	10101	00011	**11000**
7	01110	**10000**	00001	**11100**	**00000**	10101	**00000**	10110	01101
8	**11100**	00110	10111	01010	10110	00011	10110	**00000**	11011
9	00011	11101	**01100**	10001	01101	**11000**	01101	11011	**00000**

Table 3. D_1 clusters at level 11100.

There are of course many other ways of defining a Boolean dissimilarity. For example, one could use a symmetric order preserving Boolean function $f\colon 2^k \times 2^k \to 2^j$ where $1 \leq j \leq k$.

D_2	1	2	3	4	5	6	7	8	9
1	00101	11111	01111	10111	01111	11111	01111	11101	00111
2	11111	11011	11011	11111	11011	11111	11011	11111	11111
3	01111	11011	01010	11111	01011	11110	01011	11111	01110
4	10111	11111	11111	10111	11111	11111	11111	11111	10111
5	01111	11011	01011	11111	01011	11111	01011	11111	01111
6	11111	11111	11110	11111	11111	11110	11111	11111	11110
7	01111	11011	01011	11111	01011	11111	01011	11111	01111
8	11101	11111	11111	11111	11111	11111	11111	11101	11111
9	00111	11111	01110	10111	01111	11110	01111	11111	00110

Table 4. Illustration of the D_2 coefficient for the nine integers.

4 Formal concept analysis

It turns out that the notion of Boolean dissimilarity fits nicely into a general theory designed to help gain insight into the structure of complicated data sets. Before seeing how this works, we present a quick general introduction to the subject. An early reference to what we discuss can be found in Birkhoff (1967). An elegant, but much more formal treatment occurs in Ganter and Wille (1999). A concise lattice theoretic introduction may be found in Davey and Priestley (1960). The elementary treatment we give here will be self-contained, and will follow that of Davey and Priestley (1960). Let $\mathfrak{G}$ and $\mathfrak{M}$ be nonempty sets, with $\perp \subseteq \mathfrak{G} \times \mathfrak{M}$ a binary relation from $\mathfrak{G}$ to $\mathfrak{M}$. The triple $(\mathfrak{G}, \mathfrak{M}, \perp)$ is called a *formal context*. The members of $\mathfrak{G}$ are called *objects* and the members of $\mathfrak{M}$ are called *attributes*. For $A \subseteq \mathfrak{G}$, $B \subseteq \mathfrak{M}$, we let

$$A^{\perp} = \{m \in \mathfrak{M} : a \perp m \text{ for all } a \in A\},\ B^{\perp} = \{g \in \mathfrak{G} : g \perp b \text{ for all } b \in B\}.$$

The pair (A, B) is called a *formal concept* of the context $(\mathfrak{G}, \mathfrak{M}, \perp)$ if $B = A^{\perp}$ and $A = B^{\perp}$. A is called the *extent* and B the *intent* of (A, B). We agree to let $\underline{\mathfrak{B}} = \underline{\mathfrak{B}}(\mathfrak{G}, \mathfrak{M}, \perp)$ denote the collection of all concepts of the context $(\mathfrak{G}, \mathfrak{M}, \perp)$.

Example 2.

(a) Take $\mathfrak{G}$ and $\mathfrak{M}$ to be the set I of integers, and $a \perp b$ to mean that $a-b$ is a multiple of 2. If E is the set of even integers and O the set of odd integers, assume $A \neq \emptyset$. Then if $A \subseteq E$, $A^{\perp} = E$, if $A \subseteq O$, $A^{\perp} = O$, and otherwise $A^{\perp} = \emptyset$. Thus $\underline{\mathfrak{B}}$ has four elements: $(\emptyset, I)$, $(I, \emptyset)$, (E, E), and (O, O).

(b) Now let $\mathfrak{G} = \mathfrak{M}$ denote the positive integers. Write $g \perp m$ to indicate that g is a factor of m. For $A \neq \emptyset$, $A \subseteq \mathfrak{G}$, $A^{\perp}$ is then the set of common multiples of A, and if $B \subseteq \mathfrak{M}$, $B^{\perp}$ is the set of common divisors of the members of B.

(c) Let $\mathfrak{G}$ be a collection of real-valued functions, and $\mathfrak{M}$ the real numbers. Define $\perp$ by $f \perp x$ if $f(x) = 0$. Let $A \neq \emptyset$. Thus if $A \subseteq \mathfrak{G}$, $A^{\perp}$ is the set of numbers that are roots of all functions in A. If B is any set of real numbers, $B^{\perp}$ is the set of functions that have every member of B as a root.

(d) Here is an example with a slightly different flavor. Let $\mathfrak{G}$ denote the first 9 integers, and $\mathfrak{M}$ the following set of attributes: {odd (o), perfect square (s), prime (p), perfect cube (c), multiple of three (t) }. Say that $i \perp m$ if the integer i has the property m. Thus $\perp$ is given by Table 1. The number 1 in row i and column j denotes the fact that object i has attribute j, while a 0 denotes the fact that it does not have attribute j. This of course is just Example 1 from Section 3 recast into a Formal Concept Analysis problem. We will have more to say about this example in a moment, but first we need to develop some machinery.

There is nothing new in the next Theorem. All of it can be found in Birkhoff (1967), for example, in the discussion of Galois connections. For completeness, we include a proof of this theorem.

Theorem 2. *Let $(\mathfrak{G}, \mathfrak{M}, \perp)$ be a formal context. For $A \subseteq \mathfrak{G}$, $B \subseteq \mathfrak{M}$, the following assertions are true:*

(i) $A_1 \subseteq A \Longrightarrow A^{\perp} \subseteq A_1^{\perp}$, *and* $B_1 \subseteq B \Longrightarrow B^{\perp} \subseteq B_1^{\perp}$.
(ii) $A \subseteq A^{\perp\perp}$ *and* $B \subseteq B^{\perp\perp}$.
(iii) $A^{\perp} = A^{\perp\perp\perp}$ *and* $B^{\perp} = B^{\perp\perp\perp}$.
(iv) $A = A^{\perp\perp} \Longleftrightarrow A = B^{\perp}$ *for some* $B \subseteq \mathfrak{M}$, *and* $B = B^{\perp\perp} \Longleftrightarrow B = A^{\perp}$ *for some* $A \subseteq \mathfrak{G}$.

Proof: (i) If $A_1 \subseteq A$, and $a \perp b$ for all $a \in A$, then surely $a \perp b$ for all $a \in A_1$. This shows that $A^{\perp} \subseteq A_1^{\perp}$; a similar argument establishes the remaining assertion of (i).

(ii) If $a \in A$, then $a \perp b$ for all $b \in A^{\perp}$. But this says that $a \in (A^{\perp})^{\perp} = A^{\perp\perp}$. Similarly, $B \subseteq B^{\perp\perp}$.

(iii) By (ii), $A \subseteq A^{\perp\perp}$. Apply $\perp$ to this and use (i) to get that $A^{\perp\perp\perp} \subseteq A^{\perp}$. A second application of (ii) shows that $A^{\perp} \subseteq (A^{\perp})^{\perp\perp}$. Similarly, $B^{\perp} = B^{\perp\perp\perp}$.

(iv) If $A = A^{\perp\perp}$, take $B = A^{\perp}$, and note that $A = B^{\perp}$. If $A = B^{\perp}$ for some $B \subseteq \mathfrak{M}$, then $A = A^{\perp\perp}$ follows from (iii). The final assertion follows in a symmetric way.

For $A \subseteq \mathfrak{G}$, $B \subseteq \mathfrak{M}$, we say that A is *closed* if $A = A^{\perp\perp}$; similarly, B is closed if $B = B^{\perp\perp}$. The closed sets are often of interest. Theorem 2 has a number of important consequences. If $A \subseteq \mathfrak{G}$, then $A^{\perp}$ is a closed subset of $\mathfrak{M}$, and if $B \subseteq \mathfrak{M}$, then $B^{\perp}$ is a closed subset of $\mathfrak{G}$. The mapping $A \mapsto A^{\perp}$ from the closed subsets of $\mathfrak{G}$ onto the closed subsets of $\mathfrak{M}$ is one-one and order inverting. It has as its inverse the mapping $B \mapsto B^{\perp}$ from the closed subsets of $\mathfrak{M}$ into the closed subsets of $\mathfrak{G}$. An easily understood (but not computationally efficient) method for finding all closed sets can be based on the following observations:

Corollary 1. *If $A \subseteq \mathfrak{G}$, then $A^{\perp} = \bigcap\{a^{\perp} : a \in A\}$.*
If $B \subseteq \mathfrak{M}$, then $B^{\perp} = \bigcap\{b^{\perp} : b \in B\}$.

Thus to determine all sets of the form $A^{\perp\perp}$, we simply form the closed sets $m^{\perp}$ with $m \in \mathfrak{M}$, and then look at all nonempty intersections of these sets. We illustrate the technique with Example 2 (d). We begin by considering the singleton members of $\mathfrak{M}$.

$$\begin{array}{ll}
o^{\perp} = \{1,3,5,7,9\} & s^{\perp} = \{1,4,9\} \\
p^{\perp} = \{2,3,5,7\} & c^{\perp} = \{1,8\} \\
t^{\perp} = \{3,6,9\} &
\end{array}$$

If we form nonempty intersections of these sets, we can find the remaining closed subsets. This will establish the extents of the various formal concepts.

$$\begin{array}{ll}
\{o,s\}^{\perp} = \{1,9\} & \{o,p\}^{\perp} = \{3,5,7\} \\
\{o,c\}^{\perp} = \{1\} & \{o,t\}^{\perp} = \{3,9\} \\
\{s,c\}^{\perp} = \{1\} & \{s,t\}^{\perp} = \{9\} \\
\{p,t\}^{\perp} = \{3\} &
\end{array}$$

To obtain the corresponding intents, we note that

$$\begin{array}{lll}
\{1,3,5,7,9\}^{\perp} = \{o\} & \{1\}^{\perp} = \{o,s,c\} & \{1,4,9\}^{\perp} = \{s\} \\
\{3,9\}^{\perp} = \{o,t\} & \{2,3,5,7\}^{\perp} = \{p\} & \{3,5,7\}^{\perp} = \{o,p\} \\
\{1,8\}^{\perp} = \{c\} & \{9\}^{\perp} = \{o,s,t\} & \{3,6,9\}^{\perp} = \{t\} \\
\{3\}^{\perp} = \{o,p,t\} & \{1,9\}^{\perp} = \{o,s\} &
\end{array}$$

Because of Theorem 2, the concepts can be partially ordered by the rule $(A_1, B_1) \leq (A_2, B_2) \iff A_1 \subseteq A_2$. When this is done, the set $\underline{\mathfrak{B}}(\mathfrak{G}, \mathfrak{M}, \perp)$ forms a complete lattice with meet operation $\wedge$ and join operation $\vee$ such that

$$\begin{array}{l}
(A_1, B_1) \wedge (A_2, B_2) = (A_1 \cap A_2, (B_1 \cup B_2)^{\perp\perp}) \\
(A_1, B_1) \vee (A_2, B_2) = ((A_1 \cup A_2)^{\perp\perp}, B_1 \cap B_2).
\end{array}$$

When this is done, the largest element of $\underline{\mathfrak{B}}$ is $(\mathfrak{G}, \emptyset)$ and its smallest element is $(\emptyset, \mathfrak{M})$. The immediate goal for a given formal context will generally be to explicitly determine the lattice $\underline{\mathfrak{B}}$ of formal concepts. For the nine integer example, the nonempty extents of this lattice are depicted in Figure 1. The effect of making $D_2(x,x) = 0$ for all x is illustrated in Figure 2. Note that it just involves the insertion of six new singleton clusters to Figure 1.

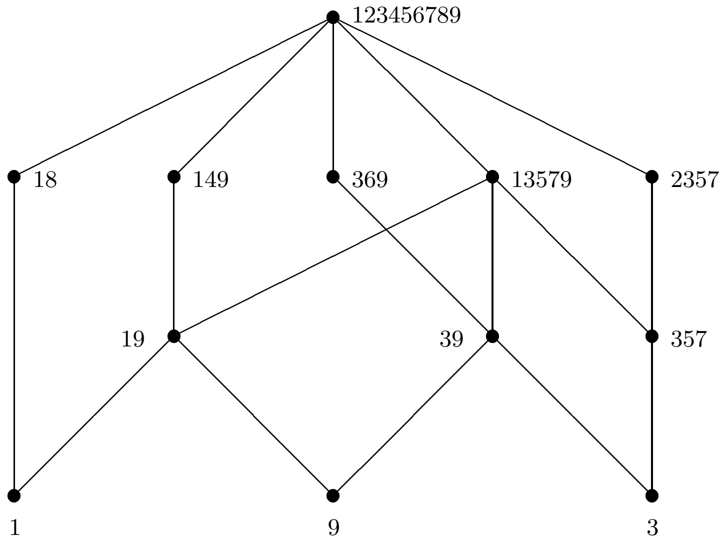


Fig. 1. Nonempty extents of the nine integer example.

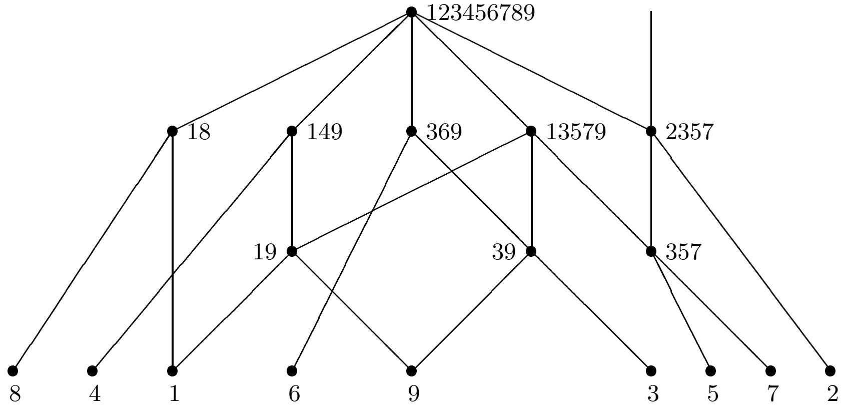


Fig. 2. Clusters for the nine integer example. with $D_2(x,x) = 0$.

But something interesting has happened. If we define D_2 using (GD2″), the cluster analysis results coincide with the formal concept approach. Since this is generally true, Formal Concept Analysis may be viewed as a special case of a Boolean dissimilarity. This also suggests that in the formal concept setting, it may be useful to associate with selected singleton attributes a bipartition. This would relate the analysis to D_1 as well as D_2.

The connection between Boolean dissimilarities and formal concept analysis may now be easily described. Starting with a Boolean DC $D\colon E{\times}E \to 2^k$, we form a formal concept by taking as objects the pairs of members of E, and saying that xy has attribute i if the i-th component of $D(x,y)$ is 1. Suppose on the other hand that $(\mathfrak{G}, \mathfrak{M}, \perp)$ is a formal context. The associated Boolean DC is what we called D_2 in Section 3. Though this relates formal concept analysis to a type of cluster analysis, it most certainly does not imply that the subject should be approached from that vantage point.

5 Complete linkage clustering with DCs measured in a poset

We just sketch a possible algorithm for complete linkage clustering based on a poset L. We assume we are working with a finite set, and that L is the image of the input DC D.

Step 1. At each minimal element m of L, form $S(m)$ by making all possible mergers of objects linked at level m.

Step 2. At each level k, assume all levels $h < k$ have been processed, and that $S(h)$ has been formed. Preserve all clusters formed by any $S(h)$, where k covers h in the sense that $k > h$ and there is no j such that $k > j > h$. When clusters A, B have a nonempty intersection, we merge them to form a single cluster $A \cup B$, repeating this as needed. Now look at the extra links implied by level k. If they lead to any new clusters, form them. The complete linkage algorithm states that A and B are merged only if all links between members of A and members of B have been formed, or if A, B overlap.

We mention that this implementation is based on the Jardine-Sibson algorithm for complete linkage clustering as explained in Jardine and Sibson (1971).

References

BIRKHOFF, G. (1967): *Lattice Theory.* Third Ed., American Mathematical Society, Providence.

DAVEY, B.A. and PRIESTLEY, H.A. (1990): *Introduction to Lattices and Order.* Cambridge Univeristy Press, Cambridge.

GANTER, B. and WILLE, R. (1999): *Formal Concept Analysis. Mathematical Foundations.* Springer-Verlag, Berlin.

GORDON, A.D. (1999): *Classification.* Second Ed., Chapman & Hall, London.

JAIN, A.K. and DUBES, R.C. (1988): *Algorithms for Clustering Data.* Prentice-Hall, Englewood Cliffs.

JANOWITZ, M.F. (1978): An order theoretic model for cluster analysis, *SIAM Journal of Applied Mathematics 34, 55-72.*

JARDINE, N. and SIBSON, R. (1971): *Mathematical Taxonomy.* Wiley, New York.

MIRKIN, B. (1996): *Mathematical Classification and Clustering.* Kluwer, Dodrecht.

SZASZ, G. (1963): *Introduction to Lattice Theory.* Third Ed., Academic Press, Budapest.

Hybrid *k*-Means: Combining Regression-Wise and Centroid-Based Criteria for QSAR

Robert Stanforth[1,2], Evgueni Kolossov[1] and Boris Mirkin[2]

[1] ID Business Solutions
2 Occam Court, Guildford, GU2 7QB, UK, {*RStanforth, EKolossov*}*@id-bs.com*

[2] School of Computer Science, Birkbeck, University of London
London WC1E 7HX, UK, *mirkin@dcs.bbk.ac.uk*

Abstract. This paper further extends the 'kernel'-based approach to clustering proposed by E. Diday in early 70s. According to this approach, a cluster's centroid can be represented by parameters of any analytical model, such as linear regression equation, built over the cluster. We address the problem of producing regression-wise clusters to be separated in the input variable space by building a hybrid clustering criterion that combines the regression-wise clustering criterion with the conventional centroid-based one.

1 Introduction

This paper addresses the issues emerging in regression-wise prediction when the sample is not homogeneous or the dependence between the response and input variables is not linear. This type of problem emerges, for example, in the quantitative analysis of relationships between structural features of chemical compounds and their biological activities; this field of research is conventionally referred to as Quantitative Structure-Activity Relationships (QSAR). In such a situation traditional methods of cluster-analysis such as *k*-means clustering may not work very well because they capture overall similarities rather than those related to the prediction. In early 70s, E. Diday proposed that a similar way of carrying out cluster analysis can be performed in such situations too (see, for example, Diday (1974)). The nature of the 'centroid' must just be redefined in such a way that any analytical data model, including those of regression or principal component analyses, can become the 'centroid', or 'kernel' of a cluster of entities under consideration (Diday (1974, 1989)).

It is exactly this approach that we are going to pursue for building a clustering better suited for prediction. There is an issue in using the regression-wise clustering for predicting compound activities: the clusters are built in the augmented space of input-response variables, but prediction is to be made based on only the input variables. When, in a typical situation, projections of the clusters to the input variables space overlap, the determination of which of the regression models to apply to an observation may become of an issue. To address this, we further advance in the kernel-based approach to combine the regression-wise with conventional centroid-based clustering so that the clusters found may be more separated in the space of input variables. The combined clustering criterion is referred to as the hybrid k-means criterion here. To assure that no overlaps may occur at all, we supplement the hybrid model with a post-processing option involving one iteration of the centroid-based k-means applied to the results of the hybrid model in the input variables space so that the resulting clusters are indeed separated in the input space. Another post-processing option involves application of the conventional k-means until convergence.

We present experimental results showing that such a modification indeed reduces the prediction error and find that there is an intermediate value of the hybrid model mixing coefficient leading to the best results.

The remainder is organised as follows. The hybrid criterion is introduced in section 2, after the conventional and regression-wise k-means clustering are defined. Our extension of k-means methods to the hybrid model is described in section 3. Section 4 presents experimental results and conclusions based on them.

2 The hybrid *k*-means criterion

We first present a brief recap on the formulation of the k-means algorithm, in preparation for deriving the variants that we shall use. The k-means family of algorithms iteratively optimise a model (of the dataset under consideration) as K clusters. This cluster model comprises a membership element, assigning each member of the dataset to one of the clusters, and a centroid element, which describes each cluster. Iteration of the algorithm proceeds by alternately optimising memberships (leaving centroids fixed) and optimising centroids (leaving memberships fixed).

The optimisation within the k-means algorithm is performed according to a loss function or 'criterion' to be minimised. In the standard 'distance-wise' formulation of k-means in the linear space of some finite feature set V, the loss function is the summary squared Euclidean distance from each point $\mathbf{x}_i$ in the dataset to the centroid $\mathbf{c}_{k(i)}$ of its assigned cluster $k(i)$.

$$L_{dist}(X, C, k) = \sum_{i=1}^{N} \sum_{v \in V} (x_{i,v} - c_{k(i),v})^2 \tag{1}$$

At each step the minimisation of this loss function can be solved directly. With the membership function k fixed, the optimal $c_{k,v}$ is the mean value of $x_{i,v}$ over those points for which $k(i) = k$; in other words the optimal $\mathbf{c}_k$ is the centroid of cluster k, justifying the terminology. On the other hand, with the centroids fixed, the optimal cluster $k(i)$ to which point $\mathbf{x}_i$ may be assigned is the cluster k whose centroid $\mathbf{c}_k$ is closest to $\mathbf{x}_i$. The algorithm terminates when the loss function fails to decrease, so the cluster model has 'converged' to a local (although not in general global) optimum. (Termination will necessarily occur eventually because there are only finitely many configurations of the membership function k.)

Variants of the k-means algorithm can be constructed by introducing different loss functions (Diday (1974)). To remain within the 'alternating optimisation' spirit of the k-means algorithm, we consider loss functions of the following form:

$$L(X, C, k) = \sum_{i=1}^{N} l(\mathbf{x}_i, \mathbf{c}_{k(i)}) \tag{2}$$

This formulation encompasses standard distance-wise k-means via taking l to be the squared Euclidean distance. Note that in general, however, the generalised 'centroids' $\mathbf{c}_k$ need not lie in the same space as the data points $\mathbf{x}_i$ as explained in Diday (1974, 1989).

Regression-wise k-means fits perfectly within this form. This variant of k-means applies to data points $\mathbf{x}_i$ in the linear space of some feature set V as before, but augmented with an associated output or 'activity' value y_i. The intention is that the activity values will be modelled as functions of the feature values x_v. Instead of approximating each point in a cluster k by the cluster's centroid $\mathbf{c}_k$, we model the cluster using a linear regression model $y \approx \sum_v a_{k,v} x_v + b_k$. We then use a squared-error loss function, measuring the summary squared distance along the activity component in augmented feature-activity space from each point to its cluster's regression hyperplane:

$$L_{reg}([X, \mathbf{y}], [A, \mathbf{b}], k) = \sum_{i=1}^{N} (y_i - (\mathbf{a}_{k(i)}^T \mathbf{x}_i + b_{k(i)}))^2 \tag{3}$$

With the membership function k fixed, the optimal cluster regression models $[\mathbf{a}_k, b_k]$ can again be computed directly, in this case by solving the following linear system (which is none other than the normal equations for multivariate linear least squares regression; see, for example, Tabachnik and Fidell (2006)):

$$\begin{aligned} \sum_{i:k(i)=k} \mathbf{x}_i \mathbf{x}_i^T \mathbf{a}_k \; + \sum_{i:k(i)=k} \mathbf{x}_i b_k \; &= \sum_{i:k(i)=k} \mathbf{x}_i y_i \\ \sum_{i:k(i)=k} \mathbf{x}_i^T \mathbf{a}_k \; + \sum_{i:k(i)=k} b_k \; &= \sum_{i:k(i)=k} y_i \end{aligned} \tag{4}$$

As usual, with the generalised 'centroids' fixed, the optimal membership assignment k is that which assigns each point $[\mathbf{x}_i, y_i]$ to the cluster k minimising the loss $(y_i - (\mathbf{a}_{k(i)}^T \mathbf{x}_i + b_{k(i)}))^2$, i.e. the cluster whose regression hyperplane is closest (along the activity axis).

If (for a cluster k) the linear system (4) turns out to be singular, for example because the size of the cluster has fallen below the number of features, then the only option is to 'dissolve' the cluster: its generalised centroid $[\mathbf{a}_k, b_k]$ is left undefined, and it is excluded from the pool when cluster memberships are reassigned in the next and subsequent iterations.

It is straightforward to see that k-means criteria are additive in the sense that, given two loss functions of the above form, their sum is also a valid k-means criterion of this form, distributing over the contributions from each point in the dataset. We may then define 'hybrid k-means' to be k-means clustering performed according to the following combined loss function:

$$L_{hyb} = (1 - p)L_{dist} + pL_{reg} \tag{5}$$

3 Methods

Regression-wise k-means can be viewed as training a composite model for activity (y) values in terms of the feature values ($\mathbf{x}$). The model is composite in the sense that, on each cluster, a separate linear model is computed to be applied on that cluster.

This approach is particularly useful if the activity depends on the feature values via a number of distinct mechanisms, with different mechanisms applying in different regions of feature space. It can also be useful if activity depends on the feature values in a non-linear fashion: the regression-wise clustering will effectively partition the model's non-linear hypersurface in augmented feature-activity space into approximately linear regions.

It should therefore, in principle, be possible for such a composite model resulting from regression-wise k-means to be used for prediction of activity for new points in feature space (whose activity value is not a priori known). Applying the composite model to a new point $\mathbf{x}$ would consist of the following steps:

1. **Classification:** Determine the cluster k to which $\mathbf{x}$ should belong.
2. **Evaluation:** Evaluate the predicted activity as $y = \mathbf{a}_k^T\mathbf{x} + b_k$ according to the regression model for cluster k.

The difficulty with this approach, when based on regression-wise clustering, lies in the classification step. Determination of cluster membership according to the regression-wise k-means criterion (3) is defined for a point $[\mathbf{x}, y]$ in the augmented space, but this dependence on y is circular as y is the unknown we are trying to predict in the first place.

The essence of the problem is that the clusters are defined in augmented space, and so can overlap substantially when projected onto feature space. The solution is to use the hybrid k-means criterion defined in the previous section: the algorithm will then run with a dominant element of distance-wise k-means, promoting separation of clusters in feature space, but retaining a contribution (proportion p) of the regression-wise criterion to guide the clusters towards regions of linearity.

To enable the prediction-time classification in feature space only, we can then follow the hybrid k-means (once it has converged) with one additional iteration with $p = 0$, i.e. according to the distance-wise criterion only. This supplementary iteration – updating memberships then updating centroids/models – will guarantee that the cluster partitioning is defined in terms of feature space only (with no dependence on activity), and that the cluster-specific linear regressions are optimal for the clusters thus defined.

An alternative resolution to compare would be to follow the hybrid k-means (again once it has converged) with as many additional iterations with $p = 0$ as are required until it converges again. This is effectively pure distance-wise k-means, but with hybrid k-means run as a preprocessing step; this is an attempt to orient the initial clusters towards regions of feature space on which separate linear models for activity apply.

Regression-wise and hybrid k-means share with standard distance-wise k-means the requirement for an initial cluster assignment k.

We propose that this initialisation be achieved using Anomalous Pattern Clustering (which also determines the number K of clusters to use), as incorporated into the so-called Intelligent k-means Algorithm by Mirkin (2005). This Anomalous Pattern Clustering, which itself makes use of a variant of 2-Means to extract the initial clusters, should be applied using the standard distance-only criterion L_{dist}.

4 Results

Ten datasets, each with 5000 points in ten-dimensional feature space augmented with one activity component, were generated randomly. Each dataset was generated with an underlying structure of five clusters, with the clusters' sizes chosen uniformly at random within the simplex of possible relative sizes. Each cluster was assigned a randomly generated mean and spread tensor, based on which the cluster contents were generated according to the multivariate normal distribution. Each cluster was also randomly assigned a linear activity model and an activity error variance; activity values for the points in the cluster were generated according to this linear model with random perturbations according to the error variance.

Each dataset was clustered according to the hybrid k-means algorithm using the criterion derived in section 2, the clustering having first been initialised according to Anomalous Pattern Clustering. Results were output at this stage, and again after one supplementary iteration of k-means, the Minimum distance assignment, was performed with no regression-wise contribution. The k-means algorithm was then allowed to proceed with no regression-wise contribution until convergence was achieved again, after which the results were output for a third and final time.

This procedure was repeated (for each dataset) for several values of p, the relative proportion of the regression-wise contribution.

At each stage, the following results were generated:

1. Regression-wise criterion, expressed as an explained proportion:

$$1 - L_{reg}/L_{reg(worst)}$$

2. Hybrid criterion, expressed as an explained proportion:

$$1 - L_{hyb}/L_{hyb(worst)}$$

3. Distance-wise criterion, expressed as an explained proportion:

$$1 - L_{dist}/L_{dist(worst)}$$

4. Mean relative error of prediction: mean value of

$$|y_{i;predicted} - y_i|/y_i$$

   over all structural features i, where the predicted value is according to the regression model of the structure's cluster in the current configuration.

In the above, the 'worst' configuration (used for normalising the criterion values) is that obtained using a single cluster and a constant (flat) regression model, leading to the maximum (worst) possible value of the criterion.

Table 1 below presents the mean relative errors of prediction for all datasets at all three stages, for the various values of p under consideration. Mean values over all ten datasets are also included.

The prediction results for the 'original' hybrid k-means (top value in each cell) show a strong decreasing trend (i.e. improvement) as p starts to increase from zero. This is unsurprising, as the relative prediction error closely corresponds to the regression-wise k-means criterion (3). Note that this stage's 'prediction' results have a somewhat artificial advantage as they are based on a cluster assignment that in turn depends on prior knowledge of the activity values. Even so, as p continues to increase towards 50%, the decreasing trend in prediction errors is not maintained (and even reversed for several datasets), suggesting that a relentlessly large regression-wise contribution is not aiding the modelling, and that retaining a distance-wise contribution is significantly beneficial in divining the underlying structure of the dataset.

As we would expect, performing the supplementary distance-only iteration of k-means causes the predictive results (centre value in each cell) to worsen. This is because we are now effectively forgoing our 'unfair' prior knowledge of the activity values and basing the cluster selection on feature values and cluster centroids alone. Here we observe, for most of the datasets (and for the mean), a trend in which the predictive power improves as p starts to increase from zero then worsens again as p becomes too large. For any dataset, a value of p specific to that dataset should then be chosen to minimise the prediction errors, expressing the optimal trade-off between regression-wise guidance and distance-wise cluster separation.

	Mean Relative Prediction Error					
Dataset	p=0	p=0.1	p=0.2	p=0.3	p=0.4	p=0.5
1	2.0865	1.5636	1.3256	0.6849	0.5302	0.5673
	2.0865	2.0125	1.9633	1.9708	2.0743	2.1780
	2.0865	2.0839	2.0534	2.0534	2.0560	2.0560
2	0.8609	0.5667	0.5353	0.5582	0.5423	0.5381
	0.8609	0.9819	0.8698	1.0281	0.9231	0.9405
	0.8609	0.8909	0.8902	0.7512	0.7509	0.9148
3	0.4919	0.4072	0.4075	0.4104	0.4080	0.3762
	0.4919	0.5639	0.5516	0.5514	0.5533	0.5389
	0.4919	0.4919	0.4919	0.4919	0.4919	0.4919
4	0.5529	0.4333	0.4219	0.4185	0.4197	0.4200
	0.5529	0.5528	0.5628	0.5153	0.5456	0.5564
	0.5529	0.5528	0.5628	0.5327	0.5330	0.5328
5	0.3150	0.1817	0.1864	0.1747	0.1545	0.1539
	0.3150	0.3199	0.3152	0.3202	0.3149	0.3157
	0.3150	0.3200	0.3200	0.3201	0.3201	0.3200
6	0.8154	0.5500	0.3196	0.3512	0.3738	0.3651
	0.8154	0.8012	0.5979	0.5677	0.5954	0.5942
	0.8154	0.8214	0.5770	0.4339	0.4048	0.5913
7	0.7504	0.4859	0.3332	0.4152	0.5957	0.5252
	0.7504	0.5733	0.5748	0.5082	0.5879	0.6339
	0.7504	0.5743	0.5539	0.5583	0.5814	0.5814
8	0.3790	0.2697	0.2257	0.2110	0.2088	0.2026
	0.3790	0.4102	0.4276	0.4605	0.4666	0.4322
	0.3790	0.3964	0.3955	0.3997	0.3960	0.3584
9	0.1625	0.1607	0.1624	0.1628	0.1633	0.2831
	0.1625	0.1625	0.1625	0.1625	0.1605	0.2176
	0.1625	0.1625	0.1625	0.1625	0.1625	0.1625
10	1.5186	0.5604	0.5875	0.4545	0.4115	0.5007
	1.5186	0.9123	0.9341	0.8754	0.9154	1.0462
	1.5186	0.8755	0.9003	0.9253	0.9155	0.9140
Mean	0.7933	0.5179	0.4505	0.3841	0.3808	0.3932
	0.7933	0.7290	0.6959	0.6960	0.7137	0.7453
	0.7933	0.7170	0.6908	0.6629	0.6612	0.6923

Table 1. Mean Relative Prediction Errors at different values *p*. Three reals in each cell present: original error of the hybrid model (top), that after one distance-wise iteration (middle), and the error of the hybrid model post-processed with the distance-wise *k*-means until convergence (bottom).

The alternative scheme of carrying through the supplementary distance-only k-means until convergence is achieved again yields similar, even slightly better, results. (See bottom value in each cell.) The point at which continuing to increase the proportion p of regression-wise contribution starts to have a detrimental effect tends to occur later than it did with only a single supplementary distance-only iteration (around 0.4 rather than 0.3). This can be explained by the fact that performing a greater amount of distance-based post-processing is better able to overcome a heavier regression-wise bias in the initial processing.

Overall, the following conclusions can be made from these experiments:

1. The proposed hybrid-based method indeed allows for a significant, 10%-20%, reduction of the relative prediction error. On average, the error decreases from 79% at only the centroid-based k-means to 66%.
2. On average, the option of post-processing with the conventional centroid-based k-means works better. However, when the error of the hybrid model is high (as at datasets 1 and 10), the option of applying the Minimum distance rule once only leads to better results.
3. The best reduction of the error is achieved with the value of the compromise coefficient p at about 0.3.

Table 2 presents the values of the regression-wise, hybrid, and distance-wise k-means criteria (averaged over the ten datasets) at the three stages of analysis. The values in this table demonstrate the degree to which the distance-wise criterion is boosted, to the detriment of the regression-wise criterion, as the supplementary distance-only k-means iterations are performed.

Criterion	Values of the criteria at					
	p=0	p=0.1	p=0.2	p=0.3	p=0.4	p=0.5
regression-wise	0.9777 0.9777 0.9777	0.9954 0.9806 0.9812	0.9962 0.9798 0.9828	0.9965 0.9792 0.9816	0.9967 0.9783 0.9817	0.9967 0.9783 0.9825
hybrid	0.6533 0.6533 0.6533	0.9672 0.9548 0.9555	0.9824 0.9676 0.9707	0.9880 0.9719 0.9744	0.9910 0.9736 0.9771	0.9927 0.9751 0.9794
distance-wise	0.6533 0.6533 0.6533	0.6388 0.6512 0.6531	0.6239 0.6494 0.6529	0.6108 0.6461 0.6514	0.5979 0.6455 0.6540	0.5814 0.6442 0.6540
Mean predictive error	0.7933 0.7933 0.7933	0.5179 0.7290 0.7170	0.4505 0.6959 0.6908	0.3841 0.6960 0.6629	0.3808 0.7137 0.6612	0.3932 0.7453 0.6923

Table 2. Values of the criterion of each of the considered models, centroid-wise, regression-wise and the hybrid one, at different values p. Three reals in each cell present: the criterion value after running the hybrid k-means to convergence then stopping (top), that after a supplementary step of one distance-wise iteration (middle), and the error of the hybrid model post-processed with the distance-wise k-means until convergence (bottom).

Obviously our conclusions are based on a rather limited set of experiments. In the future, we are going to, first, extend the simulated data models to other common distributions and, second, apply the hybrid model to real data.

References

DIDAY, E. (1974): Optimization in non-hierarchical clustering. *Pattern Recognition 6 (1), 17-33.*

DIDAY, E., CELEUX, G., GOVAERT, G., LECHEVALLIER, Y., and RALAMBONDRAINY, H. (1989): *Classification Automatique des Données.* Dunod, Paris.

MIRKIN, B. (2005): *Clustering for Data Mining: A Data Recovery Approach.* Chapman & Hall/CRC, Boca Raton, Fl.

TABACHNICK, B.G. and FIDELL, L.S. (2006): *Using Multivariate Statistics (5th Edition).* Allyn & Bacon, Boston.

Partitioning by Particle Swarm Optimization

Javier Trejos–Zelaya[1] and Mario Villalobos–Arias[2]

[1] CIMPA, Escuela de Matemática, Universidad de Costa Rica
2060 San José, Costa Rica, *jtrejos@cariari.ucr.ac.cr*

[2] CIMPA, Escuela de Matemática, Universidad de Costa Rica
2060 San José, Costa Rica, *mvillalo@cariari.ucr.ac.cr*

Abstract. We propose a clustering algorithm using particle swarm optimization (PSO) for partitioning a set of objects in K clusters, by defining a familiy of agents–partitions, each agent is defined by K centroids in a p–dimensional space; a centroid has an associated cluster, which is defined by the allocation of the objects to the nearest centroid. The agents move in the space according to PSO principles, that is, they move with random intensity in the direction of a vector called velocity, which results from the random sum of the best past position of this agent, the best overall agent, and the last direction. We compare the performance of the method with other heuristics also proposed by the authors, and with two classical methods.

1 Introduction

Let $\Omega = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset \mathbb{R}^p$ be a set of n objects described by p quantitative or numerical variables. The search of the best partition $P = (C_1, \ldots, C_K)$ of Ω in K classes is generally made by the minimization of the within–clusters inertia or variance criterion:

$$W(P) = \sum_{k=1}^{K} \sum_{i \in C_k} \|\mathbf{x}_i - \mathbf{g}_k\|^2 \tag{1}$$

where $\mathbf{g}_k$ is the centroid or barycenter of cluster C_k and $\mathbf{g}$ is the overall center of gravity or barycenter of Ω, $\|\cdot\|$ being a norm that defines an Euclidean distance. It is well known that this minimization is equivalent to the maximization of the between–clusters inertia: $B(P) = \sum_{k=1}^{K} \|\mathbf{g}_k - \mathbf{g}\|^2/|C_k|$, where $\mathbf{g}$ is the overall center of gravity or barycenter of Ω and $|C_k|$ is the cardinality of C_k. The monotonicity of criterion (1) implies that the number K of clusters must be defined in advance.

The search of the best partition in K clusters of Ω is a combinatorial problem with exponential complexity, and hence efficient heuristics should be used in order to find good quality solutions in a reasonable amount of time. Among the most popular methods, there is the k–means or dynamical clusters, or any of its variants; it has the disadvantage that is is deterministic and therefore depends directly on the initial partition, hence it may often be traped in local minima. Some authors use also hierarchical clustering (for example, using Ward's criterion) and cut the dendrogram in an appropriate level in order to have the desired number of clusters; however, hierarchical methods also obtain local minima because most ascending algorithms are greedy, and moreover the hierarchical tree imposes inclusion constraints and makes an approximation of the original distance by an ultrametric distance (see Diday et al. (1982)).

Trejos et al. (1998) deal with the problem of partitionng using modern combinatorial optimization heuristics, such as simulated annealing, tabu search and genetic algorithms, obtaining clearly better results than k–means and Ward's hierarchical methods. Pacheco et al. (2004) has made an exhaustive comparison, with a Monte–Carlo simulation study, confirming the results. More recently, Trejos et al. (2004) have used the ant colony optimization heuristic with very good results. We have also applied these techniques in some other clustering problems, such as binary data in Piza et al (2000), and two–mode or crossed classification (Trejos and Castillo (2000), Castillo and Trejos (2002)), obtaining always better results than traditional methods. We have also studied the use of these heuristics in some other data analysis problems, such as multidimensional scaling (Groenen et al. (2000), Trejos et al. (2000)), non linear regression (Villalobos et al. (2006)) and oblique varimax rotations (Trejos (1993)), with good results.

In this article, we deal with the problem of minimizing $W(P)$ by the use of Particle Swarm Optimization (PSO), Kennedy and Eberhart (2000). Next section contains a brief description of the methods to be compared. In section 3 we present the general model of PSO. In section 4 we show the details of our implementation of PSO in partitioning, and we present comparative results in section 3. Finally, in section 6 we conclude and suggest some forthcoming work.

2 Methods and heuristics in partitioning

We will not describe the well known k–means and Ward's hierarchical methods which are very well presented in Diday et al. (1982). We will only describe shortly the methods defined by the authors, based on combinatorial optimization heuristics.

2.1 Simulated annealing

It is an iterative optimization method that uses an external parameter T called temperature, that controls the acceptation of new states that give worse cost values, by the use of the so called Metropolis rule: a new state is accepted if $W(P)$ decreases, otherwise it is accepted with probability $exp(-\Delta W/T)$. It can be demonstrated —with a Markov chain modeling— that the method converges asymptotically to the global optimum. For a good implementation, four parameters must be handled (initial temperature, final temperature, decreasing rate of temperature, and length of Markov chains), as well as an easy way to compute $\varDelta W$. In partitioning (see Trejos et al. (1998)), a new partition (or state) P' is generated from a current partition P by a transfer: (i) an object $\mathbf{x}_i$ is chosen at random in $\varOmega$, (ii) a class index k is chosen at random, (iii) put $\mathbf{x}_i$ in class C_k. For convergence, reversibility (P can be generated from P' with the same probability that P' is generated from P) and connectivity (any partition can be generated from the current partition by a finite number of transfers) are satisfied; also, all neighborhoods have the same size $n(K-1)$.

2.2 Tabu search

It is an optimization technique based on the search of neighbors of a state and the choice of the best neighbor, whether or not it is better than the current state. A tabu list is handled, in order to avoid the access to states similar to states recently visited. It may include some variants that improve the technique (aspiration criteria, use of elite states, random generation of neighbors, ...) For partitioning (see Trejos et al. (1998)), we define a state as a partition of $\varOmega$ in K classes and a move consists in building a neighborhood of partitions defined by the transfer of a single object into a new class. The tabu list contains the indicators of classes where the transfered objects belong.

2.3 Genetic algorithm

It is a multiagents method that handles symultaneously a set of solutions that are combined. First, by a roulette wheel mechanism the best solutions are selected, and then they are combined by means of two operations: crossover and mutation. In partitioning (see Trejos et al. (1998)), states are represented by a string of n characters in $\{1, 2, \dots, K\}$, and the fitness function to be maximized is $B(P)$, which is used when the roulette wheel is applied. We have defined the following crossover: with probability p_c two parents are chosen and a class index is randomly chosen for the better one; then, this class index is imposed to the corresponding objects of the second parent, defining a son. For the mutation, choose an agent at random with probability p_{m_1} and choose an object with probability p_{m_2}, then modify randomly the class membership of that object (this corresponds to a single transfer).

2.4 Ant colonies

Ant colony optimization is based on the metaphore of the way that ants search for food. It is also a multiagent method and is based on a reinforcement step, for growing up the probability of good solutions (or parts of solutions). This is performed with a pheromone trail and a visibility part, that define the probability of changing from state. Applied to partitioning (see Trejos et al. (2004)), we handle a population of ants that modify its associated partition, local visibility is the inverse of the distance between two objects, and the pheromone trail is reinforced if two objects belong to the same class and the between–inertia $B(P)$ of the corresponding partition is big. The last two terms are used in the definition of the probability of choosing an object from another one.

3 Particle swarm optimization

Particle swarm optimization (PSO) as presented in Kennedy and Eberhart (2000) is based on the iterative use of a set of agents or particles that correspond to states in an optimization problem, moving each agent in a numerical space looking for the optimal position. A particularity of PSO is that agents communicate and hence —as in a social system— an agent with a good position (measured by its objective function value) *influences* on the other ones, *attracting* them.

3.1 Principles of PSO

A set of M particles is handled in a multidimensional space and it is inteded to model social behavior, in the sense that each particle tries to improve its performance according to its own experience and the experience of its environment. Indeed, each particle has three tendencies: (i) remember its best historial position, so that in a conservative way the particle will try to go back to this position; (ii) following the particle's inertia, will try to continue with its present tendency; and (iii) will try to imitate its best neighbor.

3.2 Modeling PSO

If $z^m(t)$ represents the m–th particle, then its velocity in iteration $t+1$ is defined as

$$v^m(t+1) = \alpha v^m(t) + r_1(z^{m*} - z^m(t)) + r_2(z^* - z^m(t)) \qquad (2)$$

where $v^m(t)$ is the velocity in the preceding iteration, z^{m*} is the best historial position ever obtained by m, z^* is the best particle ever obtained during the algorithm, r_1 and r_2 are random numbers, and α is a parameter. So, we define the new position of particle m as

$$z^m(t+1) = z^m(t) + v^m(t+1). \qquad (3)$$

Some authors (see Clerc (1998)) have studied conditions for non divergence of PSO. However, there is not yet a proof of convergence to the global optimum.

4 Use of PSO in numerical clustering

In Goddard et al. (2002) we had stated the basic ideas about the application of PSO in partitioning.

We propose an algorithm for the minimization of $W(P)$ using PSO. For this, the user defines M agents which correspond to M partitions of Ω in K classes, agent m being a set of K centroids $\mathbf{g}_1^m, \ldots, \mathbf{g}_K^m \in \mathbb{R}^p$. Centroid $\mathbf{g}_k^m$ has a class C_k^m associated, by the allocation of the objects in Ω to the nearest centroid. The centroids move according to the principles of PSO described in equations (2) and (3), and each partition is redefined by allocation to the nearest centroid. PSO consists then in moving, by a random intensity, K centroids in $\mathbb{R}^p$ for each agent (or particle), in the direction resulting for the velocity; there is a parameter $V_{\max}$ that bounds the move in order to avoid explosion in the space.

```
Algorithm PSOClus.
Initiate: read data; define $M$, $V_{\max}$, $\alpha$, $maxiter$
Do $M$ times: select at random $K$ objects in $\Omega$ as initial centroids and allocate the remaining objects to the nearest centroid; this makes $K$ classes.
Calculate the barycenters $\mathbf{g}_1^1, \ldots, \mathbf{g}_K^1, \ldots, \mathbf{g}_1^M, \ldots, \mathbf{g}_K^M$ of the classes
Initialize the best value of each particle (denoted $(\mathbf{g}_1^{m*}, \ldots, \mathbf{g}_K^{m*})$)
Initialize the overall leader according to (1), denoted $(\mathbf{g}_1^*, \ldots, \mathbf{g}_K^*)$
Repeat for $t = 1, 2, \ldots$ until convergence or $maxiter$ times:
  For $m = 1$ until $M$ do:
    Let $r_1 = random(0,1)$, $r_2 = random(0,1)$, $r_s = r_1 + r_2$
    Let $r_1 = 4.1 * r_1/r_s$, $r_2 = 4.1 * r_2/r_s$
    For $k = 1$ until $K$ do:
      For $j = 1$ until $p$ do:
        Let $v_{kj}^m(t) := \alpha v_{kj}^m(t-1) + r_1(g_{kj}^{m*} - g_{kj}^m(t-1)) + r_2(g_{kj}^* - g_{kj}^m(t-1))$
        If $v_{kj}^m(t) > V_{\max}$ then $v_{kj}^m(t) := V_{\max}$
          Else, if $v_{kj}^m(t) < -V_{\max}$ then $v_{kj}^m(t) := -V_{\max}$
        enf-if
        $g_{kj}^m(t) := g_{kj}^m(t-1) + v_{kj}^m(t-1)$
      end-for($j$)
    end-for($k$)
  Allocate all $n$ objects in $\Omega$ to the nearest centroid $\mathbf{g}_k^m$.
  Update vector $(\mathbf{g}_1^{m*}, \ldots, \mathbf{g}_K^{m*})$
  end-for($m$)
Update vector $(\mathbf{g}_1^*, \ldots, \mathbf{g}_K^*)$
end-repeat-for($t$)
```

5 Comparative results

We have compared our PSOClus algorithm with other partitioning methods based on heuristics: simulated annealing (SA), tabu search (TS), genetic algorithm (GA), and ant colony optimization (ACO), as well as with k–means (KM) and Ward's hierarchical ascending method (cutting the dendrogram at the level with the desired number of clusters). Table 4 contains the results of running the methods on 4 data tables: Scholar notes, Amiard's fishes, Thomas' sociomatrix and Fisher's iris. Each method has been applied a number of times indicated in parentheses under the method's name. K denotes the number of classes and W the best value of within-inertia obtained in all runs. The table contains the percentage of times that the corresponding method has obtained the best value of W in all runs; column corresponding to Ward's method indicates whether or not this method has obtained the best solution.

	Scholar Notes (9 × 5)							
K	W	PSO	SA	TS	GA	ACO	kM	Ward
		(100)	(150)	(1 000)	(100)	(25)	(10 000)	
2	28.2	92	100	100	100	100	12	No
3	16.8	57	100	100	95	100	12	No
4	10.5	51	100	100	97	100	5	Yes
5	4.9	29	100	100	100	100	8	Yes
	Amiard's fishes (23 × 15)							
K	W	PSO	SA	TS	GA	ACO	kM	Ward
		(100)	(150)	(200)	(100)	(25)	(10 000)	
3	32213	51	100	100	87	100	8	No
4	18281	23	100	100	0	100	9	No
5	14497	6	100	97	0	68	1	Yes
	Thomas' sociomatrix (24 × 24)							
K	W	PSO	SA	TS	GA	ACO	kM	Ward
		(100)	(150)	(200)	(100)	(25)	(10 000)	
3	271	7	100	100	85	100	2	No
4	235	7	100	100	24	96	0.15	No
5	202	7	100	98	0	84	0.02	No
	Fisher's Iris (150 × 4)							
K	W	PSO	SA	TS	GA	ACO	kM	Ward
		(100)	(150)	(1 000)	(100)	(25)	(10 000)	
2	0.99	76	100	100	100	100	100	No
3	0.52	79	100	76	100	100	4	No
4	0.38	55	55	60	82	100	1	No
5	0.32	28	0	32	6	100	0.24	No

Table 1. Comparative results for PSOClus with simulated annealing (SA), tabu search (TS), genetic algorithm (GA), ant colonies (ACO), k–means (kM) and Ward's dendrogram cut at K clusters on 4 real data tables; W is the best value of criterion obtained for all methods and the table contains the percentage of times that value W was obtained for each method.

We also performed a Monte Carlo study by the generation of random Gaussian tables in $[0,1]^6$, with four factors and each one with two levels: the number of objects (105,525), the number of classes (3,7), the cardinality of the classes (equal vs. different cardinalities), and the variance of the variables (equal vs. different variances). The PSOClus method was tested over these 16 tables, as well as the other partitioning methods. All methods used the same initial partition (those based on multipagents used this partition for the first element of the population and the remaining initial partitions were at random).

Table 4 contains the best value of $W(P)$ found by any method and the percentage of times (in 100 runs) that any method reached this value; for Ward's hierarchical method, it is reported only if the method found the best value.

n	K	W^*	PSO	SA	TS	GA	ACO	kM	Ward
Equal cardinalities									
			Equal variances						
105	3	5,42	100	100	99	100	100	91	yes
105	7	5,15	1	100	74	82	100	19	yes
525	3	5,99	94	100	100	100	100	98	yes
525	7	5,34	1	100	82	88	100	45	yes
			Different variances						
105	3	13,15	1	100	99	100	100	13	no
105	7	9,90	0	100	51	69	75	1	no
525	3	15,81	1	100	51	82	99	2	no
525	7	8,26	0	100	100	94	100	53	no
Different cardinalities									
			Equal variances						
105	3	5.01	99	100	100	100	100	91	yes
105	7	5.55	1	0	0	35	36	3	yes
525	3	5.67	84	8	100	100	100	95	yes
525	7	5.65	1	0	0	22	38	2	yes
			Different variances						
105	3	11.73	12	100	100	100	100	95	no
105	7	7.63	0	0	0	37	85	6	no
525	3	13.82	1	3	100	100	100	59	no
525	7	7.46	0	0	0	21	54	0	no

Table 2. Best value of $W(P)$ for 100 runs of simulated annealing (SA), tabu search (TS), genetic algorithm (GA), ant colonies (ACO), k–means (KM) and Ward's dendrogram cut at K clusters (one single run), on 16 simulated data tables; W is the best value of criterion obtained for all methods and the table contains the percentage of times that value W was obtained for each method.

Table 3 contains the average values of $W(P)$ for the 100 runs.

n	K	W^*	PSO	SA	TS	GA	ACO	kM	Ward
				Equal cardinalities					
				Equal variances					
105	3	5.42	5.42	5.42	5.53	5.42	5.42	6.42	5.42
105	7	5.15	6.21	5.15	5.97	5.29	5.15	7.78	5.15
525	3	5.99	5.99	5.99	5.99	5.99	5.99	6.15	5.99
525	7	5.34	6.87	5.34	5.87	5.65	5.34	7.20	5.34
				Different variances					
105	3	13.15	13.31	13.15	13.18	13.15	13.15	13.50	13.85
105	7	9.90	11.26	9.90	10.10	9.95	9.90	12.79	10.17
525	3	15.81	15.88	15.81	15.81	16.01	15.81	16.14	16.41
525	7	8.26	9.61	8.26	8.26	8.36	8.26	8.63	9.37
				Different cardinalities					
				Equal variances					
105	3	5.01	5.01	5.01	5.01	5.01	5.01	6.38	5.01
105	7	5.55	7.62	7.77	11.72	6.77	6.75	10.26	5.55
525	3	5.67	5.67	5.70	5.67	5.67	5.67	5.89	5.67
525	7	5.65	7.49	8.11	10.71	6.89	6.35	8.48	5.66
				Different variances					
105	3	11.73	11.77	11.73	11.73	11.73	11.73	12.23	11.86
105	7	7.63	9.77	8.65	9.82	8.37	7.68	9.76	7.69
525	3	13.82	13.84	13.87	13.82	13.82	13.82	14.12	14.2
525	7	7.46	9.18	9.76	10.17	8.53	7.72	9.27	8

Table 3. Average of $W(P)$ for 100 runs of simulated annealing (SA), tabu search (TS), genetic algorithm (GA), ant colonies (ACO), k–means (KM) and Ward's dendrogram cut at K clusters (one single run), on 16 simulated data tables; W is the best value of criterion obtained for all methods and the table contains the percentage of times that value W was obtained for each method.

All programs are in Delphi using Pascal code. We do not make an exhaustive report on running times, but we can say the k–means is very fast (a few seconds for 100 runs), simulated annealing may last 30 seconds, PSOClus and the genetic algorithm may last in average about 2 minutes in 100 runs, and the slower is tabu search, which may last 16 minutes on the same data sets.

6 Concluding remarks

From Tables 4 and 3, it can be seen that PSO obtains generally better results than classical methods, like k–means and Ward's hierarchical. But, compared to other heuristics, simulated annealing, genetic algorithm and ant colonies behave much better. Table 4 shows also that PSOClus has problems in some situations, mainly when variances are different among clusters, but also when dealing with a large number of clusters.

However, there is still some work to be done on calibrating the parameters that could improve these results. Indeed, as with all this kind of heuristics, there is some dependence on the parameters, which in PSOClus are α, $r_1 + r_2$, *maxiter*, $V_{\max}$ and M, the population size. At the present time, we are studying the behavior of the method with respect to these parameters, and possibly this study may improve the performances of the PSOClus.

References

CASTILLO, W. and TREJOS, J. (2002): Two-mode partitioning: review of methods and application of tabu search. In: K. Jajuga, A. Sokolowski and H.H. Bock (Eds.): *Classification, Clustering and Data Analysis.* Springer, Berlin: 43-51.

CLERC, M. (1998): Some math about particle swarm optimization, internet document at `http://clerc.maurice.free.fr/pso/`

DIDAY, E., LEMAIRE, J., POUGET, J. and TESTU, F. (1982): *Eléments d'Analyse des Données.* Dunod, Paris.

GODDARD, J., DE LOS COBOS, S., PIZA, E. and TREJOS, J. (2002): Clasificación de datos numéricos mediante optimización por enjambres de partículas. In: *5th International Conference on Operations Research*, La Habana, Cuba, 4–8 March 2002.

GROENEN, P.J.F., MATHAR, R. and TREJOS, J. (2000): Global optimization methods for multidimensional scaling applied to mobile communications. In: W. Gaul, O. Opitz and M. Schader (Eds.): *Data Analysis. Scientific Modeling and Practical Application.* Springer, Berlin, 459-469.

KENNEDY, J. and EBERHART, R.C. (2000): *Intelligent Swarm Systems.* Academic Press, New York.

PACHECO, A., MURILLO, A., PIZA, E. and TREJOS, J. (2004): Evaluación de heurísticas de optimización combintoria en clasificación por particiones. In: *II Congreso Andino de Investigación de Operaciones*, Cartagena, Colombia, 14–19 Mar 2004.

PIZA, E., TREJOS, J. and MURILLO, A. (2000): Clustering with non-Euclidean distances using combinatorial optimisation techniques. In: J. Blasius, J. Hox, E. de Leeuw and P. Schmidt (Eds.): *Science Methodology in the New Millenium*, CD-Rom paper Nr. P090504, ISBN 90-801073-8-7.

TREJOS, J. (1993): A simulated annealing implementation for oblique varimax rotations. In: J. Janssen and C. H. Skiadas (Eds.): *Applied Stochastic Models and Data Analysis*, Vol. II. World Scientific, Singapur, 981-989.

TREJOS, J., MURILLO, A. and PIZA, E. (1998): Global stochastic optimization for partitioning. In: A. Rizzi, M. Vichi and H.H. Bock (Eds.): *Advances in Data Science and Classification.* Springer, Berlin, 185-190.

TREJOS, J., MURILLO, A. and PIZA, E. (2004): Clustering by ant colony optimization. In: D. Banks, L. House, F.R. McMorris, P. Arabie and W. Gaul (Eds.): *Classification, Clustering, and Data Mining Applications.* Springer, Berlin, 25-32.

TREJOS, J. and CASTILLO, W. (2000): Simulated annealing optimization for two-mode partitioning, In: W. Gaul and R. Decker (Eds.): *Classification and Information at the Turn of the Millenium.* Springer, Berlin, 133-142.

TREJOS, J.; CASTILLO, W., GONZÁLEZ, J. and VILLALOBOS, M. (2000): Application of simulated annealing in some multidimensional scaling problems. In: H.A.L. Kiers, J.-P. Rasson, P.J.F. Groenen and M. Schader (Eds.): *Data Analysis, Classification and Related Methods*. Springer, Berlin, 297-302.

VILLALOBOS, M., TREJOS, J. and DE LOS COBOS, S. (2006): Aplicación de la búsqueda tabú en regresión no lineal. *Revista de Matemática: Teoría y Aplicaciones 13 (1), 79-92.*

Part III

Conceptual Analysis of Data

Concepts of a Discrete Random Variable

Richard Emilion

Laboratoire MAPMO, Université d'Orléans, B.P. 6759
45067 Orléans Cedex 2, France, *richard.emilion@univ-orleans.fr*

Abstract. A formal concept is defined in the literature as a pair (extent, intent) with respect to a context which is usually empirical, as for example a sample of transactions. This is somewhat unsatisfying since concepts, though born from experiences, should not depend on them. In this paper we consider the above concepts as 'empirical concepts' and we define the notion of concept, in a context-free framework, as a limit intent, by proving, applying the large number law, that :
Given a random variable $\mathcal{X}$ taking its value in a countable σ-semilattice, the random intents of empirical concepts, with respect to a sample of $\mathcal{X}$, converge almost everywhere to a fixed deterministic limit, called a concept, whose identification shows that it only depends on the distribution $P_{\mathcal{X}}$ of $\mathcal{X}$. Moreover, the set of such concepts is the σ-semilattice generated by the support of $\mathcal{X}$ and has even a structure of σ-lattice: the lattice concept of a random variable.
We also compute the mean number of concepts and frequent itemsets for a hierarchical Bernoulli mixtures model. Last, we propose an algorithm to find out maximal frequent itemsets by using minimal winning coalitions of $P_{\mathcal{X}}$.

1 Introduction

An important component of data mining is rule induction, that is extraction of useful if-then rules from data, and a key step in this induction consists in mining what is usually called frequent itemsets (FI's) as introduced in Agrawal et al. (1993 and 1994). In order to understand the ideas beyond these mining algorithms, it is helpful to use the notions of Galois connections, intent, extent, closed sets and so on.
A pair (extent, intent) was called concept by Wille (1980) but this notion of concept, widely used in various domains (artificial intelligence, robotics, psychology, software engineering, text mining and so on), depends on the extent which is, roughly speaking, a random sample. In the present paper we call it (random) empirical concept and we define concepts as limit of empirical intents, showing that these limits are no more random and do not depend on the sample. In other words concepts are defined with respect to a random variable rather than to a sample of this random variable.
The paper is organized as follows: Random variables taking their values in a σ-semilattice are introduced in Section 2. Random empirical Galois lattices are defined in Section 3 where is also proved the convergence of random intents and is defined the concept lattice of a random variable. In Section 4, the average number of empirical concepts and of frequent itemsets is computed

for a hierarchical Bernoulli mixtures model. Frequent itemsets and winning coalitions are studied in Section 5, providing an algorithm for mining maximal frequent itemsets.
The present paper answers a question of Edwin Diday who has also a definition of concept as an intent (Bock et al. (2000)). It is dedicated to Edwin Diday who introduced us to several interesting problems.

2 Notations and terminology

2.1 σ-semilattice

Our set of observations, say $\mathcal{L}$, is taken very general in order to cover a wide area of applications. It can be a subset of real numbers, real vectors, real functions, fuzzy sets, power set, words of a language, real cumulative distribution functions, real stochastic processes, and so on.
Let $(\mathcal{L}, \leq, \wedge)$ be a countable semilattice, that is

- $\mathcal{L}$ is a countable set
- $\leq$ is a partial order relation on the set $\mathcal{L}$
- $\wedge$ is an infimum operator .

We will asume in addition that $\mathcal{L}$ is a σ-semilattice : for any (countable) subset $\mathcal{A} \subseteq \mathcal{L}$, there exists a largest element in $\mathcal{L}$, denoted by $\bigwedge_{L\in\mathcal{A}} L$, which is lower than any $L \in \mathcal{A}$.
Without loss of generality, it can also be assumed that there exists a largest element in $\mathcal{L}$, denoted by $\mathbf{1}$, and by convention

$$\bigwedge_{L\in\mathcal{A}} L = \mathbf{1} \quad \text{if} \quad \mathcal{A} = \emptyset.$$

If $(L_n)_{n\geq 1}$ is a *decreasing* sequence in $\mathcal{L}$, then we will say that this sequence is convergent and that its limit is

$$\bigwedge_{n=1}^{\infty} L_n.$$

2.2 $\mathcal{L}$-valued random variable

Let $(\Omega, \mathcal{B}, P)$ be a probability space. Let

$$\mathcal{X} : \Omega \longrightarrow \mathcal{L}$$

be a (discrete) $\mathcal{L}$-valued random variable (r.v.) whose distribution probability $P_{\mathcal{X}}$ is a probability measure on $\mathcal{L}$ defined as usual as follows:

$$\forall L \in \mathcal{L}, P_{\mathcal{X}}(L) = P(\mathcal{X} = L) = P(\omega \in \Omega : \mathcal{X}(\omega) = L).$$

The *support* of $P_{\mathcal{X}}$ will play a key role: it is defined as the set

$$\mathcal{S}_{\mathcal{X}} = \{L \in \mathcal{L} : P_X(L) > 0\}.$$

For any $n \in \{1, 2, 3...\}$ a $n-$sample of $\mathcal{X}$ is a sequence $\mathcal{X}_1, \ldots, \mathcal{X}_n$ where the $\mathcal{X}_i$'s are independent and identically distributed (iid) r.v.'s distributed as $\mathcal{X}$.

2.3 Data mining context

To join the terminology of data mining (which comes from marketing) with the preceding setting, it suffices to take

$$\mathcal{L} = (\mathcal{P}(J), \subseteq, \cap),$$

the power set of a (large) finite set J of *items*. Any any $L \in \mathcal{L}$ is then a subset of J and is usually called an *itemset*.

The random variable $\mathcal{X}$ modellizes random transactions made by customers, the itemset $X(\omega) \in \mathcal{L}$ representing the *random set* of items bought by a customer.

The following simple example illustrates what is usually called a binary context. Take $J = \{a, b, c, d, e\}$ and $n = 10$ transactions (1 means that the item was bought and 0 not). The last column in Table 1 below contains the random set that will be considered in our approach.

a	b	c	d	e	*random set*
0	1	0	0	1	$\mathcal{X}_1(\omega) = \{b, e\}$
1	1	0	0	1	$\mathcal{X}_2(\omega) = \{a, b, e\}$
0	1	1	1	0	$\mathcal{X}_3(\omega) = \{b, c, d\}$
1	0	0	1	0	$\mathcal{X}_4(\omega) = \{a, d\}$
0	1	1	0	1	$\mathcal{X}_5(\omega) = \{b, c, e\}$
1	1	1	1	0	$\mathcal{X}_6(\omega) = \{a, b, c, d\}$
0	0	1	1	1	$\mathcal{X}_7(\omega) = \{c, d, e\}$
1	1	0	1	0	$\mathcal{X}_8(\omega) = \{a, b, d\}$
0	1	1	1	1	$\mathcal{X}_9(\omega) = \{b, c, d, e\}$
0	0	1	1	0	$\mathcal{X}_{10}(\omega) = \{c, d\}$

Table 1. Binary context and random set.

3 Galois lattice for semilattices

The notion of Galois connection (GC) was early introduced in Ore (1944), it is also mentionned in the book by Birkhoff (1967), chapter 5. We first note

that Barbut and Monjardet elegant and general definition of a Galois lattice (GL), stated for a GC between lattices (Barbut et al. (1970), pages 13 and 25), can be extended to a GC between semilattices:
Let $< \mathcal{E}, \leq, \wedge >$ and $< \mathcal{F}, \leq, \wedge >$ be two semilattices, a GC between $\mathcal{E}$ and $\mathcal{F}$ is a pair of mappings (f, g) verifying

$$f: \mathcal{E} \longrightarrow \mathcal{F} \text{ and } g: \mathcal{F} \longrightarrow \mathcal{E} \text{ are decreasing,} \tag{1}$$

$$h = g \circ f : \mathcal{E} \longrightarrow \mathcal{E} \text{ and } k = f \circ g : \mathcal{F} \longrightarrow \mathcal{F} \text{ are extensive,} \tag{2}$$

$$i.e.\ \forall x \in \mathcal{E}, x \leq h(x) \text{ and } \forall y \in \mathcal{F}, y \leq k(y).$$

These definitions imply that

$$f \circ h = f,\ h \circ h = h,\ g \circ k = g,\ k \circ k = k. \tag{3}$$

Let

$$I_h = \{x \in \mathcal{E} : h(x) = x\} \text{ (resp. } I_k = \{y \in \mathcal{F} : k(y) = y\})$$

be the set of *closed* (or invariant) elements of $\mathcal{E}$ (resp. of $\mathcal{F}$).
It can be seen that the restriction of f to I_h is a one-to-one mapping into I_k, its inverse being the restriction of g to I_k.
The *Galois lattice* (GL) $\mathcal{G}$ induced by the GC (f, g) is defined as the set of nodes

$$\mathcal{G} = \{(x, f(x)),\ x \in I_h\},$$

which has a *lattice* structure if $\leq, \vee$ and $\wedge$ are defined as follows:

$$(x, f(x)) \leq (x', f(x')) \text{ iff } x \leq x' \text{ and } f(x') \leq f(x),$$

$$(x, f(x)) \vee (x', f(x')) = (g(f(x) \wedge f(x')), f(x) \wedge f(x')),$$

$$(x, f(x)) \wedge (x', f(x')) = (x \wedge x', f(x \wedge x')).$$

It is easily seen that

$$\mathcal{G} = \{(x, f(x)), x \in I_h\} = \{(g(y), y), y \in I_k\}.$$

The mapping f (resp. g) is called an intent (resp. an extent).
As any pair (x, y) of the GL satifies $y = f(x)$ and $x = g(y)$, Wille (1980) then proposed to call such a pair a *concept.*
It is worthwhile to mention that the name of Galois appears here because of the analogy with a fundamental result in the celebrated Galois theory on the one-to-one correspondance between the intermediate fields of a field extension and the subgroups of its Galois group (see e.g. Stewart (1975), page 114).

3.1 Binary GL

Let $\mathcal{I}$ be a set of objects and $\mathcal{J}$ a set of properties. Let $\mathcal{R}$ be a binary relation on $\mathcal{I} \times \mathcal{J}$: $i\mathcal{R}j$ iff object i has property j.
For any non-emptyset $A \in \mathcal{E} = \mathcal{P}(\mathcal{I})$ let

$$f(A) = \{j \in \mathcal{J} : i\mathcal{R}j \text{ for all } i \in A\} \text{ and } f(\emptyset) = \mathcal{J} \tag{4}$$

be the the *intent* or the *description* of A, that is the set of properties satisfied by all objects of A. For any non-empty set $B \in \mathcal{F} = \mathcal{P}(\mathcal{J})$ let

$$g(B) = \{i \in \mathcal{I} : i\mathcal{R}j \text{ for all } j \in B\} \text{ and } g(\emptyset) = \mathcal{I} \tag{5}$$

be the *extent* of B, that is the set of objects satisfying all the properties given by B. The pair (f, g) is a popular example of GC, it is called a binary GC.

3.2 Explicit formulas for a general GC

Let $\mathcal{E} = \mathcal{P}(\mathcal{I})$, where $\mathcal{I})$ denote a countable set of objects. In most concrete situations, only the descriptions $d(i), i \in \mathcal{I}$, which belong to a general $\sigma-$semilattice $\mathcal{L}$, are given. A natural question to ask is the existence of a GC (f, g) such that $f(\{i\}) = d(i)$ with explicit fomulas generalizing formulas (4) (5) of the binary case. The solution exists, and is unique if the GC is supposed maximal (that is not dominated by a GC):

Theorem (Diday - Emilion (1997), (2003)) *There exists a unique maximal* GC (f, g) *between* $\mathcal{E} = \mathcal{P}(\mathcal{I})$ *and* $\mathcal{L}$ *verifying* $f(\{i\}) = d(i)$. *It is given by the formulas*:

$$f(A) = \wedge_{i \in A} d(i) \text{ for any non-empty } A \in \mathcal{E}, \tag{6}$$

$$f(\emptyset) = 1,$$

$$g(L) = \{i \in \mathcal{I} : L \leq d(i)\} \text{ for any } L \in \mathcal{L}. \tag{7}$$

Note that (6) and (7) imply

$$h(A) = g(f(A)) = \{i \in \mathcal{I} : \bigwedge_{j \in A} d(j) \leq d(i)\} \text{ for any } A \in \mathcal{E}, \tag{8}$$

$$k(L) = f(g(L)) = \bigwedge_{i \in \mathcal{I} : L \leq d(i)} d(i) \text{ for any } L \in \mathcal{L}. \tag{9}$$

In the binary case, $\mathcal{L} = (\mathcal{P}(\mathcal{J}), \subseteq, \cap)$ is isomorphic to $(\{0, 1\}^{\#J}, \leq, \wedge)$, therefore (6) and (7) generalize (4) and (5)

4 Random Galois lattices

4.1 Random empirical Galois lattices

As above, let $\mathcal{X} : \Omega \longrightarrow \mathcal{L}$ be a (discrete) $\mathcal{L}$-valued random variable (r.v.). Let $\mathcal{X}_1, \ldots, \mathcal{X}_n, \ldots$ be a sequence of iid r.v.'s distributed as $\mathcal{X}$.
For any $n = 1, 2, \ldots$, consider the following random Galois connections:

$$< \mathcal{E}_n, \leq, \wedge > = < \mathcal{P}\{1, 2, \ldots, n\}, \subseteq, \cap >,$$

$$< \mathcal{F}, \leq, \wedge > = (\mathcal{L}, \leq, \wedge),$$

$$f_n(A) = \bigwedge_{i \in A} \mathcal{X}_i, \; g_n(L) = \{i \in \{1, 2, \ldots, n\} : L \leq \mathcal{X}_i\},$$

$$h_n = g_n \circ f_n, k_n = f_n \circ g_n$$

for any $A \in \mathcal{E}_n$ and $L \in \mathcal{L}$.
Note that

$$h_n(A) = \{i \in \{1, 2, \ldots, n\} : \bigwedge_{j \in A} \mathcal{X}_j \leq \mathcal{X}_i\}$$

while

$$k_n(L) = \bigwedge_{i \in \{1, 2, \ldots, n\} : L \leq \mathcal{X}_i} \mathcal{X}_i.$$

4.2 Convergence of random empirical intents

We are now in a position to state the announced result on the convergence of random empirical intents with the identification of the deterministic limit.

Theorem 1. *For any $L \in \mathcal{L}$ the random intents $k_n(L) = \bigwedge_{i=1,\ldots,n : L \leq \mathcal{X}_i} \mathcal{X}_i$ converge a.e. towards the following deterministic limit:*

$$k_\infty(L) = \lim_{n \longrightarrow \infty} \downarrow k_n(L) = \bigwedge_{L' \in S_{\mathcal{X}} : L \leq L'} L'.$$

Proof. For any $L \in \mathcal{L}$, let $1_{(\mathcal{X}_i = L)}(\omega) = 1$ if $\mathcal{X}_i(\omega) = L$ and $= 0$ otherwise. Since the r.v.'s $1_{(\mathcal{X}_i = L)}$ so defined are i.i.d. with expectation $P_{\mathcal{X}}(L)$, the large number law provides a nullset $N_L \subseteq \Omega$, $N_L \in \mathcal{B}$, such that $P(N_L) = 0$, which satisfies

$$\forall \omega \notin N_L, \; \frac{1}{n} \sum_{i=1}^{n} 1_{(\mathcal{X}_i = L)}(\omega) \longrightarrow P_{\mathcal{X}}(L).$$

In particular for any $L \in S_{\mathcal{X}}$, since $P_{\mathcal{X}}(L) > 0$, we have

$$\forall \omega \notin N_L, \; \sum_{i=1}^{n} 1_{(\mathcal{X}_i = L)}(\omega) \geq 1$$

for n large enough. Therefore

$$\forall \omega \notin N_L,\ \exists i \geq 1 : \mathcal{X}_i(\omega) = L$$

that is

$$\forall \omega \notin N_L,\ L \in \{\mathcal{X}_i(\omega), i = 1, 2, \ldots\}.$$

As $S_{\mathcal{X}}$ is countable, the set $N = \bigcup_{L \in S_{\mathcal{X}}} N_L$ belongs to $\mathcal{B}$, $P(N) = 0$ and

$$\forall \omega \notin N,\ S_{\mathcal{X}} \subseteq \{\mathcal{X}_i(\omega), i = 1, 2, \ldots\}. \tag{10}$$

On the other hand, for any $i = 1, 2, \ldots$, let

$$N_i = \{\omega : \mathcal{X}_i(\omega) \notin S_{\mathcal{X}}\}.$$

Then, we have

$$P(N_i) = 0$$

since $\mathcal{L} \backslash S_{\mathcal{X}}$ is countable and

$$P(N_i) = P(\mathcal{X}_i \notin S_{\mathcal{X}}) = P(\mathcal{X} \notin S_{\mathcal{X}}) = \sum_{L \notin S_{\mathcal{X}}} P(\mathcal{X} = L) = 0$$

by definition of $S_{\mathcal{X}}$. Now, by definition of the N_i's we have

$$\forall \omega \notin \bigcup_{i=1}^{\infty} N_i,\ \mathcal{X}_i(\omega) \in S_{\mathcal{X}}\ \forall i = 1, 2, \ldots$$

or equivalently

$$\forall \omega \notin \bigcup_{i=1}^{\infty} N_i,\ \{\mathcal{X}_i(\omega), i = 1, 2, \ldots\} \subseteq S_{\mathcal{X}}. \tag{11}$$

So, if we let $N_0 = N \cup \bigcup_{i=1}^{\infty} N_i$, then $P(N_0) = 0$ and (10), (11) imply

$$\forall \omega \notin N_0, \{\mathcal{X}_i(\omega), i = 1, 2, \ldots\} = S_{\mathcal{X}},$$

that is, shortly,

$$\{\mathcal{X}_i, i = 1, 2, \ldots\} = S_{\mathcal{X}} \quad a.e. \tag{12}$$

Note that (12) holds for any random variable taking its value in a countable set. Observe now that (12) implies that for any $L \in \mathcal{L}$

$$\{\mathcal{X}_i, i = 1, 2, \ldots\ : L \leq \mathcal{X}_i\} = \{L' \in S_{\mathcal{X}} : L \leq L'\}\ a.e.$$

and thus

$$\bigwedge_{i=1,2\ldots,:L \leq \mathcal{X}_i} \mathcal{X}_i = \bigwedge_{L' \in S_{\mathcal{X}} : L \leq L'} L'\ a.e..$$

This completes the proof.

4.3 Limit GL

Obviously, the above closure operator k_∞ can be obtained by the following limit GC

$$< \mathcal{E}, \leq, \wedge > = < \mathcal{P}\{1, 2, \ldots, \}, \subseteq, \cap >,$$
$$< \mathcal{F}, \leq, \wedge > = (\mathcal{L}, \leq, \wedge),$$
$$f_\infty(A) = \bigwedge_{i \in A} \mathcal{X}_i,$$
$$g_\infty(L) = \{i \in \{1, 2, \ldots\} : L \leq \mathcal{X}_i\},$$
$$h_\infty = g_\infty \circ f_\infty, k_\infty = f_\infty \circ g_\infty$$

for any $A \in \mathcal{E}$ and $L \in \mathcal{L}$.
So,

$$h_\infty(A) = \{i \in \{1, 2, \ldots\} : \bigwedge_{j \in A} \mathcal{X}_j \leq \mathcal{X}_i\},$$

while

$$k_\infty(L) = \bigwedge_{i \in \{1,2,\ldots\} : L \leq \mathcal{X}_i} \mathcal{X}_i.$$

Hence the random limit GL can be defined as the lattice:

$$\mathcal{G}_\infty = \{g_\infty(L), k_\infty(L)), L \in \mathcal{L}\}.$$

Note that the extent $g_\infty(L)$ is random and depends on the sample $(\mathcal{X}_i)_{i=1,2,\ldots}$ while the intent is deterministic and does not depend on the sequence $(\mathcal{X}_i)_{i=1,2,\ldots}$.

4.4 Concepts, concept lattice

Definition: A concept of the r.v. $\mathcal{X}$ is an element of $\mathcal{L}$ such that

$$L = \bigwedge_{L' \in S_\mathcal{X} : L \leq L'} L'.$$

The set of concepts will be denoted by $\mathcal{C}(\mathcal{X}, \mathcal{L})$, shortly, $\mathcal{C}$. The random set of empirical intents w.r.t. a sample $\mathcal{X}_1, \ldots, \mathcal{X}_n$ of $\mathcal{X}$ will be denoted by $\mathcal{C}(\mathcal{X}_1, \ldots, \mathcal{X}_n, \mathcal{L})$, shortly, $\mathcal{C}_n$:

$$\mathcal{C}_n = k_n(\mathcal{L}) = \{k_n(L), L \in \mathcal{L}\}.$$

The above theorem states that

$$k_\infty(\mathcal{L}) = \{k_\infty(L), L \in \mathcal{L}\} = \mathcal{C}(\mathcal{X}, \mathcal{L}) \ a.e..$$

Since we have $L \leq k_\infty(L) \leq k_{n+1}(L) \leq k_n(L)$ we see that $k_n(L) = L \Rightarrow k_\infty(L) = L$, in other words

$$k_n(\mathcal{L}) \subseteq k_{n+1}(\mathcal{L}) \subseteq k_\infty(\mathcal{L}). \tag{13}$$

Proposition 1. *$\mathcal{C}(\mathcal{X},\mathcal{L})$ is the σ-semilattice generated by $S_{\mathcal{X}}$.*
In particular, if $P(\mathcal{X}=L)>0$ then L is a concept.

Note however that L such that $P(\mathcal{X}=L)=0$ can be a concept:
let $\mathcal{L}=\{0,c,a,b,1\}$ where 0 (resp. 1) is the lowest (resp. largest) element of $\mathcal{L}$, $a\not\leq b$, $b\not\leq a$, $c=a\wedge b$ and let $\mathcal{X}$ be such that $P(\mathcal{X}=a)=1/2$, $P(\mathcal{X}=b)=1/2$. Then c is a concept and $P(\mathcal{X}=c)=0$.
Further, observe that

Proposition 2. *i) If $L<1$ is a concept then $P(L\leq\mathcal{X})>0$*
ii) $\{L'\in S_{\mathcal{X}}:L\leq L'\}=\{L'\in S_{\mathcal{X}}:k_\infty(L)\leq L'\}$
iii) $P(L\leq\mathcal{X})=P(k_\infty(L)\leq\mathcal{X})$
iv) If $k_\infty(L)<1$ then $P(L\leq\mathcal{X})>0$
v) 1 is a concept iff $P(\mathcal{X}=1)>0$
vi) If $k_\infty(L)=1$ then $P(L\leq\mathcal{X})>0$ iff $P(\mathcal{X}=1)>0$

Note that $P(L\leq\mathcal{X})>0$ means that for a.a. sample, L appears infinitely often *within* an itemset. Also, the converse of i) need not be true (use iv)).

Proposition 3. *$\mathcal{C}(\mathcal{X},\mathcal{L})$ is a σ-lattice.*

5 Average number of concepts for hierarchical Bernoulli mixtures

Consider the case where

$$\mathcal{L}=(\mathcal{P}(J),\subseteq,\cap)$$

the power set of a (large) finite set $J=\{1,\ldots,r\}$ of r items, $\mathcal{P}(J)$ being identified to $\{0,1\}^r$. Suppose that the distribution of the r.v.

$$\mathcal{X}=(\mathcal{X}^{(1)},\ldots,\mathcal{X}^{(j)},\ldots,\mathcal{X}^{(r)})$$

is a finite mixture of products of Bernoulli's $\otimes_{j=1}^r B(p_{U,j})$, where the r.v. $U\in\{1,\ldots,K\}$ is a latent class variable and the weight vector $q=(q_1,\ldots,q_K)$ of the mixture has a Dirichlet distribution $D(\gamma_1,\ldots,\gamma_K)$. This precisely means that we have the following hierarchical mixture model (HMM):

$$\mathcal{X}|_{U=u,q}\sim\sum_{u=1}^{K}q_c\otimes_{j=1}^{r}B(p_{u,j}), \tag{14}$$

$$P(U=u|q)=q_u, \tag{15}$$

$$q\sim D(\gamma_1,\ldots,\gamma_K). \tag{16}$$

The following generalizes some results in Lhote et al. (2005) and Emilion et al. (2005):

Proposition 4. *For the HMM defined by equations (14), (15), (16)*

$$E(\#\mathcal{C}_n) = \sum_{u=1}^{K} \frac{\gamma_u}{\gamma} \sum_{i=0}^{n} \sum_{B \in \mathcal{P}(J)} \binom{n}{i} (1 - \prod_{j \in B} p_{u,j})^{n-i} \prod_{j \in B} p^i_{u,j} \prod_{j \notin B} (1 - p_{u,j})^i,$$

and

$$\lim_{n \to \infty} \uparrow E(\#\mathcal{C}_n) = 2^r = \#\mathcal{C}.$$

For such a model we can similarly compute the mean number of closed frequent itemsets.

6 Maximal frequent itemsets

6.1 Empirical frequent itemsets

We return now to the case of a general σ-semilattice whose elements are still called itemsets.
Let $\alpha \in (0,1)$ be a fixed treshold.
An itemset L is said empirically frequent (w.r.t the empirical context $\mathcal{X}_i, i = 1, \dots n$) iff

$$\#g_n(L) = \#\{i \in \{1, \dots, n\} : L \leq \mathcal{X}_i\} \geq n\alpha.$$

As

$$\#g_n(L) = \sum_{i=1}^{n} 1_{L \leq \mathcal{X}_i}, \tag{17}$$

and the $\mathcal{X}_i$'s are i.i.d., we see that the r.v. $1_{L \leq \mathcal{X}_i}$ are Bernoulli i.i.d. and the r.v. $\#g_n(L)$ has a binomial distribution:

$$\#g_n(L) \sim \text{Binom}(n, p_L)$$

where

$$p_L = P(L \leq \mathcal{X}).$$

Hence

$$P(L \text{ empirical frequent}) = P(\#g_n(L) \geq n\alpha) = \sum_{k \geq n\alpha} \binom{n}{k} p_L^k (1 - p_L)^{n-k}.$$

The average number of empirical frequent itemsets is then equal to

Proposition 5.

$$\sum_{L \in \mathcal{L}} P(L\ (n, \alpha) - \textit{frequent}) = \sum_{L \in \mathcal{L}} \sum_{k \geq n\alpha} \binom{n}{k} p_L^k (1 - p_L)^{n-k}.$$

6.2 Frequent itemsets

By the large number law, (17) implies:

$$\lim_{n\to\infty} \frac{\#g_n(L)}{n} = P(L \leq \mathcal{X}) \ a.e.$$

so that we are lead to the following:

Definition: $L \in \mathcal{L}$ is an α-frequent itemset iff

$$P(L \leq \mathcal{X}) \geq \alpha.$$

A maximal α-frequent itemset is an α-frequent itemset which is maximal (for the order $\leq$ in $\mathcal{L}$) among the α-frequent itemsets.

6.3 Minimal winning coalitions

We now propose an algorithm to find out maximal frequent itemsets by using minimal coalitions of $P_{\mathcal{X}}$.
Since $\mathcal{X}$ is countable, let

$$S_{\mathcal{X}} = \{L_1, \ldots, L_r, \ldots\},$$

and let

$$p_r = P(\mathcal{X} = L_r) > 0.$$

An α-winning coalition is a subset of $\{1, \ldots, r, \ldots\}$, say A, such that

$$\sum_{r \in A} p_r \geq \alpha.$$

A minimal α-winning coalition is an α-winning coalition which is minimal (for the inclusion order) among the α-winning coalitions.
Algorithms for finding minimal coalitions were intensively studied in games theory (see e.g. Matsu et. al (2000)). They can be applied to find out maximal frequent itemsets due to the following:

Theorem 2. *i) If L is a maximal frequent itemset then $L = \bigwedge_{r\in A} L_r$ where A is an α-minimal coalition.*
ii) Conversely if A is an α-minimal coalition then $L = \bigwedge_{r\in A} L_r$ is frequent.

It is easy to construct an example where A is an α-minimal coalition but $L = \bigwedge_{r\in A} L_r$ is not maximal.

6.4 Algorithm

The above theorem can be applied to the empirical measure (which is an estimator of $P_{\mathcal{X}}$), from a finite table of observed itemsets such as the one in Subsection 2.3:

- Find the distinct itemsets $L_1, \ldots, L_k$, and their respective frequency $p_1, \ldots, p_k$
- Find the α-minimal winning coalitions from $p_1, \ldots, p_k$
- For each of such a coalition, say A, compute $L = \bigwedge_{r \in A} L_r$
- The list of such L contains all the maximal frequent itemsets which can be extracted from this list.

Such an algorithm will be of interest if the number r of distinct itemsets is much lower than the total number of observed itemsets. Note that the step where are found minimal winning coalitions should be fast since it does not require any access to the dataset.

References

AGRAWAL, R., IMIELINSKI, T. and SWAMY, A. (1993): Mining association rules between sets of items in large databases. In: *ACM SIGMOD, Int'l Conf. on Managment of Data*, 207–216.

AGRAWAL, R. and SRIKANT, R. (1994): Fast algorithm for mining association. In: *20th. Intl'. Conf. VLDB*, 478–499

BARBUT, M. and MONJARDET, B. (1970): *Ordre et classification.* Hachette, Paris.

BIRKHOFF, G. (1967): *Lattice theory.* AMS Colloq. Public. Vol. XXV.

BOCK, H.-H. and DIDAY, E. (2000): *Analysis of Symbolic Data.* Springer Verlag, Berlin.

CASPARD, N. and MONJARDET, B. (2003): The lattice of closure systems. *Disc. Appl. Math. J., 127, 241-269.*

DIDAY, E. and EMILION, R. (1997): Maximal and stochastic Galois lattices. *C. R. Acad. Sci. Paris, 325, I (1), 261-266.*

DIDAY, E. and EMILION, R. (2003): Maximal and stochastic Galois lattices. *Disc. Applied Math. J, 27-2, 271-284.*

EMILION, R. and LÉVY, G. (2005): Size of random Galois lattices and number of frequent itemsets. *http://hal.archives-ouvertes.fr/hal-00013510.*

LHOTE, L., RIOULT, F. and SOULET, A. (2005): Average number of frequent (closed) patterns in Bernouilli and Markovian databases. In: *Fifth IEEE International Conference on Data Mining (ICDM'05), Houston, Texas*, 713–716.

MATSUI, T. and MATSUI, Y. (2000): A survey of algorithms for calculating power indices of weighted majority games. *J. Oper. Research Soc. Japan, 43.*

ORE, O. (1944): Galois connections. *Trans. Amer. Math. Soc. 55, 494-513.*

STEWART, J. (1975): *Galois Theory.* Chapman and Hall, New York.

WILLE, R. (1982): Restructuring Lattice Theory: An Approach Based on Hierarchies of Concepts. In: I. Rival (Ed.): *Ordered Sets.* Reidel, Dordrecht-Boston, 445–470.

Mining Description Logics Concepts with Relational Concept Analysis

Marianne Huchard[1], Amedeo Napoli[2], Mohamed Rouane Hacene[2], and Petko Valtchev[3]

[1] LIRMM, 161 rue Ada, 34392 Montpellier Cedex 5, France, *huchard@lirmm.fr*
[2] LORIA, Campus Sciences, BP 239, 54506 Vandœuvre-lès-Nancy Cedex, France, {*napoli, rouanehm*}*@loria.fr*
[3] Dépt. d'informatique, UQAM, CP 8888, succ. Centre-Ville, Montréal, Canada, H3C 3P8, *valtchev.petko@uqam.ca*

Abstract. *Symbolic objects* were originally intended to bring both more structure in data and more intelligibility in final results to statistical data analysis. We present here a framework of similar motivation, i.e., combining a data analysis method, — the concept analysis (FCA) — with a knowledge description language inspired by description logic (DL) formalism. The focus is hence on proper handling of relations between individuals in the construction of formal concepts. We illustrate the relational concept analysis (RCA) framework which complements standard FCA with a dedicated data format, a set of scaling operators, an iterative process for lattice construction, and translations to and from a DL language.

1 Introduction

Symbolic objects (SO) (Diday (1998)) were designed to meet the urgent need for processing of more realistically structured data, i.e., beyond mere real number vectors, in statistical data analysis, while representing the final results in a more intelligible manner. On data formats, beside the variety of value domains of the descriptive variables (taxonomic, interval, histogram, etc.), higher-level structure is also provided for, e.g., in *hordes* which provide for nesting of individuals. In the broader field of knowledge discovery from data, structure and intelligibility have been pursued through a symbiosis with knowledge representation (KR) (Brachman and Anand (1996))

Formal concept analysis (FCA) (Ganter and Wille (1999)) as data analysis paradigm also endorsed KR concerns. In fact its target FCA structure, the concept lattice, represents a natural framework for both taxonomies and conceptual hierarchies. While the standard FCA framework barely admits structure in the input datasets, recent trends targeted the complexly structured data. For example, a first trend admits explicit inter-individual links which, once expressed as first-class objects within a *power context family*, are dealt in a straightforward way, i.e., grouped into formal concepts representing new, and compound, relations (Prediger and Wille (1999)). Independently, and somewhat closer to the SO approach to structure, logic-based KR has been tentatively introduced in the *conceptual scaling* mechanism which enables the processing of non-binary data in FCA. Thus, in (Prediger and Stumme (1999)), a language of the description logic DL family (Baader et al. (2003)) was used to express conditions involving domain concepts and relations, which were then applied to individuals as binary attributes. It is noteworthy that the symbiosis of SO and FCA, i.e., the concept analysis of symbolic datasets, has been investigated as well (Polaillon (1998)).

Our own study on concept analysis of complex datasets is motivated by the rapidly growing need for interoperability between mining mechanism and modern KR environments, especially in the wake of the Semantic Web launch. In simple terms, this means mining tools must be able to process data expressed in languages, such as OWL and SWRL, and output the discovered knowledge in equally compatible formats. In this respect, our concerns combine, on the one hand, the adequate clustering of relational datasets, as logically-founded languages describe individuals by means of both unary predicates (*concepts*) and binary ones (relations, or *properties*), and, on the other hand, the design of compound expressions to intentionally describe the discovered clusters. As an approach for the concept analysis of relational data, we proposed a dedicated framework, called *relational concept analysis* (RCA), which offers simple solutions to both concerns. Moreover, the framework relies on three original components: a data format inspired by the entity-relationship conceptual data model, a scaling method applying various policies in the translation of inter-individual links into binary attributes, and an iterative lattice construction process allowing many separate individual sorts to be analyzed simultaneously.

The present paper summarizes the RCA theoretical foundations and illustrates its *modus operandi* using a small-size, albeit realistically structured dataset. The following Section 2 provides minimal background on FCA and then RCA, and briefly examines the composition of a DL language. Section 3 introduces the sample dataset, which is then analyzed w.r.t. two different scaling policies. The analysis processes based on wide and on narrow scaling are followed in Section 4 and Section 5, respectively.

2 From FCA to RCA

The following is a brief presentation of the RCA framework. Details may be found in (Huchard et al. (2007)) while an implementation is available within the Galicia platform[1].

2.1 Standard FCA

FCA is the process of abstracting conceptual descriptions from a set of individuals described by attributes (Ganter and Wille (1999)). Formally, a *context* $\mathcal{K}$

[1] http://sourceforge.net/projects/galicia/

associates a set of objects (O) to a set of attributes (A) through an incidence relation $I \subseteq O \times A$, i.e., $\mathcal{K} = (O, A, I)$. For example, in Section 3, a context is presented where objects are scientific publications (e.g., monographs, journal articles, conference papers, theses, etc.), whereas attributes are general topics (e.g., software engineering, lattice theory, etc.). The represented incidence relation is therefore to be interpreted as "speaks about" or "deals with".

In this settings, FCA focuses at the way objects group together on grounds of shared attributes. Intuitively, each subset of objects is examined together with the respective set of shared attributes (e.g., a set of publications determines a list of all common topics). Among all object sets, only maximal ones are kept, i.e., sets comprising *all* objects incident to the shared attributes. This is formalized by two applications mapping object sets to attribute ones and *vice versa*, both denoted $'$ hereafter. For instance, on objects, the $'$ application is defined as follows: $' : \mathcal{P}(O) \rightarrow \mathcal{P}(A); \quad X' = \{a \in A \mid \forall o \in X, oIa\}$.

A basic result states that maximal sets of objects, called *extents* in FCA, are in one-to-one correspondence to maximal sets on attributes, or *intents*. Furthermore, the pairs $(X, Y) \in \mathcal{P}(O) \times \mathcal{P}(A)$, of mutually corresponding sets, i.e., such that $X = Y'$ and $Y = X'$, called *(formal) concepts*, form a complete lattice with respect to the inclusion of the extents, i.e., the X part. Extracting the concept lattice $\mathcal{L}$ of a context $\mathcal{K}$ is the key task in FCA. Fig. 2 shows, on its right-hand side, the concept lattice of the publication context which is itself embedded in the table on the left-hand side (only the first four columns).

The classical FCA apparatus is limited to datasets that either originally represent binary relations or can be easily, i.e., with no significant precision loss, transformed to such relations. Indeed, the *conceptual scaling* mechanism translating non-binary attributes (e.g., numerical or nominal) into binary ones, amounts to replacing attribute values by predicates on them. For instance, the domain of *nbOfPages* attribute in publications could be split into the ranges *short*, *standard*, and *long* (paper), each of them expressed as a predicate (e.g, *nbOfPages*$\leq$ 6 for short one). Observe that the definition of the predicates precedes the scaling process and is usually the charge of a domain expert.

Unsurprisingly, the data stored in a relational database remains well beyond the reach of the above approach, and for some good reasons. First, the underlying entity-relationship (ER) conceptual data model admits several *entities*, i.e., sorts of individuals, that are connected by *relationships*, i.e., n-ary predicates on entities, whereas FCA typically focuses on a single set of individuals (although these may generate a family of contexts) and yields a single concept lattice. As an illustration, imagine a database modeling a collection of scientific publications, researchers, topics, author-to-paper links, references among publications, etc. Moreover, a natural way of analyzing such data would be to form concepts that reflect commonalities both in individual properties and in their links to other individuals, following, for instance, the way DL concepts are defined (see Section 2.3). Although approaches dealing with relations have been studied in FCA, none of them allows links and properties to be mixed in concept intents. To bridge the gap, we have proposed a relational FCA framework, called *relational concept analysis* (RCA), that basically adds a new data format, a set of scaling mechanisms for relational links and an iterative method for the simultaneous construction of a set of concept lattices.

2.2 RCA summary

The RCA data format, a *relational context family* (RCF), combines FCA and ER as it consists of a set of contexts and a set of binary relations, each involving the objects from two contexts of the RCF.

Definition 1. A *relational context family* $\mathcal{R}$ is a pair $(\mathbf{K}, \mathbf{R})$, where $\mathbf{K}$ is a set of contexts $\mathcal{K}_i = (O_i, A_i, I_i)$, $\mathbf{R}$ is a set of relations $r_k \subseteq O_i \times O_j$, where O_i and O_j are the object sets of the formal contexts $\mathcal{K}_i$ and $\mathcal{K}_j$.

Let now a relation r (e.g., authoring of papers by researchers) link objects from a context $\mathcal{K}_i$, the *domain* of r, to those of $\mathcal{K}_j$, its *range*. In order to scale upon r so that one can use the information it conveys in the concept analysis upon $\mathcal{K}_i$, we consider the conceptual structure, i.e., all (known) formal concepts, of $\mathcal{K}_j$. The concepts are turned into binary predicates just as in classical scaling. The key difference is that in assigning such a predicate to an object o_i from $\mathcal{K}_i$, instead of comparing an attribute value to a range of such values, a set of objects, i.e., the links of type r for o_i, denoted $r(o_i)$, is compared to the extent of a concept c_j on $\mathcal{K}_j$. For instance, to describe researchers with respect to the authored papers, these will be compared to the extents of the formal concepts on the entire papers collection (e.g., journal papers on statistics). Various relationships between $r(o_i)$ and the extent of c_j (e.g., inclusion, non-empty intersection, intersection of a certain size, etc.) may be required in order for o_i to acquire the corresponding attribute, invariably denoted by $r : c_j$. These are discussed in the next paragraph.

Relational scaling opens the way to lattice construction. However, the global analysis process is not one-shot, it rather proceeds iteratively, i.e., by successive steps alternating scaling and concept formation. Indeed, as no restriction is imposed in the relational structure of a RCF, there may well be circuits in the way contexts are related by relations, hence the mutual dependence between such contexts in the sense that each of them requires the other(s) to be processed first in order to provide the formal concepts required for scaling. To break the deadlock, a bootstrapping step is performed in the beginning of each RCA process, in which all object sorts get the lattice corresponding exclusively to their local properties (from the underlying contexts). In the subsequent steps, scaling is used to translate the already available structure, i.e., formal concepts, from the range context of a relation to the domain one. More precisely, the current lattices are first used to scale upon the relations of the RCF thus generating new attributes in the respective domain contexts. The lattices of the extended contexts are then constructed, possibly triggering a new scaling/construction step. Indeed, as the new attributes may yield new extents, the lattices and hence the scales they represent may evolve, hence the need to re-scale in order to keep the domain contexts in line with the evolution. The global process of iterative lattice construction, called MULTI-FCA, nevertheless converges to a set of lattices representing a fixed-point. Section 4 and Section 5 illustrate the way MULTI-FCA unfolds.

2.3 Description logics and relational scaling

Description logics (DL) are KR formalisms rooted in first order predicate logic that offer means to structure the otherwise flat logical representation, namely in terms of *concepts*, *roles*, and *individuals* (Baader et al. (2003)). DL languages allow expressions, or *descriptions*, to be composed out of other descriptions up to an arbitrary depth. A DL language is built on top of a collection of *primitive concept* and *role* names which denote the meaningful concepts and relations from a domain (e.g., Human, Female, Doctor, child, father, etc.), individual names (e.g., Ann) and constants ($\top$ and $\bot$).

Concepts are interpreted as sets of individuals (their *instances*) and roles as sets of individual pairs[2]. Further concepts and roles are defined by combining concept and role names, either primitive or already defined, via a set of constructors, e.g., conjunction ($\sqcap$), disjunction ($\sqcup$), negation ($\neg$). By definition, a role has a domain and a range concept and is inherited by the sub-concepts of the domain concept. It may be further restricted for every concept it applies to, for instance, by applying universal or existential quantifiers to the set of links. Thus, given a role r and a concept C, the following concept expressions can be composed: (*i*) $\forall\mathsf{r.C}$ (*value restriction*), (*ii*) $\exists\mathsf{r.C}$ (*full existential quantification*), and (*iii*) $\exists\mathsf{r}.\top$ (*limited existential quantification*). All these work as filters on the individuals: (*i*) collects those whose links of type r, if any, point exclusively to instances of the concept denoted by the expression C, (*ii*) those with at least one r link to such an instance, and (*iii*) those with at least one r link, regardless of the underlying concept. As an illustration, consider the expression of the concept of "all fathers *and* all parents of a female child whose children are all doctors" in DL:

$$\mathsf{Male} \sqcap \exists\mathsf{child}.\top \sqcup \mathsf{Human} \sqcap \exists\mathsf{child.Female} \sqcap \forall\mathsf{child.Doctor}$$

Individuals are represented in a DL language as constants (e.g., Ann) and characterized by a set of ground predicates, unary for the concepts they belong to and binary for the roles they possess (e.g., Human(Ann), Female(Ann), child(Ann, Mary)). Consequently, the translation of a collection of DL individuals into an RCF is immediate: First, each individual is assigned a unique

[2] See Baader et al. (2003) for formal definitions for DL syntax and semantics.

concept to express its very nature (e.g., Human) in the same way entities within an ER schema do. Such concepts are translated as contexts while their instances become the respective formal objects. Next, the remainder of the concepts an individual belongs to are translated as binary attributes and attached to the underlying context (e.g., Female for the context modeling Human). Finally, all roles become relations in the RCF whose domain and range contexts are determined following the individuals in the role pairs and the contexts comprising their respective translations.

The DL formalism has a direct impact on the RCA scaling mechanism as well. Indeed, as mentioned previously, given a relation r that connects objects from $\mathcal{K}_i$ to those from $\mathcal{K}_j$, the various ways to assign an attribute $r : c_j$, where c_j is a concept on $\mathcal{K}_j$, to objects o_i from $\mathcal{K}_i$ follow restriction constructors from DL. More precisely, we defined several scaling policies, termed *encoding schemes*, including a value-restriction-like scheme, called *strict narrow*, a full existential-like one, or *wide*, and a third one, called simply *narrow*, that amounts to a combination of both. Indeed, while strict narrow scheme only requires $r(o_i) \subseteq extent(c_j)$, the narrow adds the condition $r(o_i) \neq \emptyset$. The latter condition is implied by the requirement of a wide scheme, i.e., $r(o_i) \cap extent(c_j) \neq \emptyset$. The way narrow and wide encoding scheme work is illustrated below.

Given the forward translation from a DL language to an RCF and the above scaling policies, the reverse translation of the formal concepts yielded by RCA into a DL knowledge base is immediate.

3 Running example

The sample RCF is made of a single context and two binary relations. The *Papers* context assigns publications, as objects, to the topics they refer to — software engineering (*se*), lattice theory (*lt*) and man machine interface (*mmi*) — as attributes. The relation *cites* models citations while *develops* connects a long publication, e.g., a thesis, to a paper whose key ideas the former extensively develops. Fig. 2 depicts the RCF both as a conceptual schema and as a data graph made of links and individuals (to whom codes are assigned for subsequent use in the text). In order to eases the tracking of

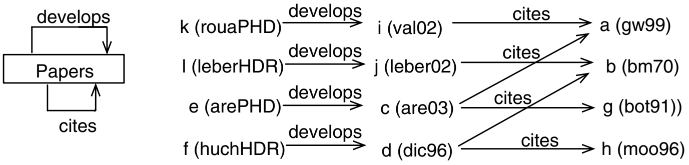


Fig. 1. Sample RCF. **Left:** As UML schema; **Right:** As data graph.

the gradual emergence of formal concepts, the example was stripped of a large number of papers and citation links. It nevertheless shows a complex, three-level link structure: Indeed, a set of four papers (level one) are substantial developments of four other papers (level two) which cite papers on level three. Moreover, though cycles in links are avoided, these are dealt with in much the same way. The RCF corresponds to a DL knowledge base with two roles (develops and cites) and four concepts, i.e., Papers, AboutLatticeTheory, AboutSoftwareEngineering, and AboutManMachineInterface.

With an object set $O = \{a..l\}$ and attribute set $A = \{lt, mmi, se\}$ the information content of the RCF can be summarized as follows (see Fig. 2):

- $I \subseteq O \times A$; $I = \{(a, lt), (b, lt), (g, mmi), (h, se)\}$,
- $cites \subseteq O \times O$; $cites = \{(c, a), (c, g), (d, b), (d, h), (i, a), (j, b)\}$,
- $develops \subseteq O \times O$; $develops = \{(e, c), (f, d), (k, i), (l, j)\}$.

Thus, initially, only level-three papers share descriptions and hence form concepts, e.g., a and b share the lt topic and therefore form the lattice theory publication concept. The lattice yielded by the paper context, regardless of the existing links, is given in Fig. 2 (on the right). Obviously, the aforementioned concept *c0*$= (\{a, b\}, \{lt\})$ is the only non-trivial one. This lattice, once translated into binary attributes by scaling, enables new groupings, e.g., of c, d, i, j which cite at least one paper on lattices. The resulting concept triggers yet further sharing, this time at level one, and the whole process goes on. The way the analysis process unfolds and its final result depend on the exact scheme used for scaling, as shown by the next two sections.

		lt	mmi	se
gw99	a	x		
bm70	b	x		
are03	c			
dic96	d			
arePHD	e			
hucHDR	f			
bot91	g		x	
moo96	h			x
val03	i			
leber02	j			
rouaPHD	k			
leberHDR	l			

Kpapers

	a	b	g	h
a				
b				
c	x		x	
d		x		x
e				
f				
g				
h				
i	x			
j		x		
k				
l				

cites

	c	d	i	j
a				
b				
c				
d				
e	x			
f		x		
g				
h				
i				
j				
k			x	
l				x

develops

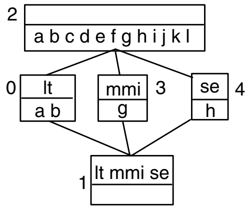


Fig. 2. Left: Initial RCF on papers; **Right:** Lattice 1 on papers.

4 Narrow scaling-based RCA

Intuitively, the narrow scheme favors compact lattices as potentially less objects will get the new attributes due to the stronger requirements.

Step 1 Narrow scaling upon *cites* and w.r.t. lattice in Fig. 2 adds five new attributes of the type *cites:c* to the context. However, given the non-empty citation link sets ($cites(c) = \{a, g\}$, $cites(d) = \{b, h\}$, $cites(i) = \{a\}$, $cites(j) = \{b\}$), only two of them, i.e., *cites:c2* and *cites:c0*, are effectively assigned to a paper. Thus, all level-two papers, i.e., c, d, i, j, get the attribute *cites:c2* in the scaled context as *c2* comprises the entire dataset, whereas only i and j get *cites:c0* as well. Correspondingly, the relation I is extended with the pairs (*c,cites:c2*), (*d,cites:c2*), (*i,cites:c2*), (*j,cites:c2*), (*i,cites:c0*), and (*j,cites:c0*) (see Fig. 3).

					cites				develops
		lt	mmi	se	2	0	3	4	2
gw99	a	x							
bm70	b	x							
are03	c				x				
dic96	d				x				
arePHD	e								x
hucHDR	f								x
bot91	g		x						
moo96	h			x					
val03	i				x	x			
leber02	j				x	x			
rouaPHD	k								x
leberHDR	l								x

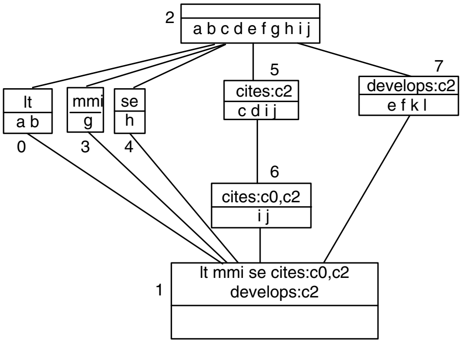


Fig. 3. Narrow scaling, step 1. **Left:** The scaled context; **Right:** Lattice 2.

Narrow scaling upon *develops* ranges only over papers having such links, i.e., e, f, k, and l. As none of the developed papers, i.e., c, d, i, and j, belongs to a non-trivial concept in the scaling lattice (i.e., other than *c2*), level-one papers only get the attribute *develops:c2*. The resulting scaled context and its lattice are given in Fig. 3. Three new concepts appear in the lattice:

- *c5=(*{*c,d,i,j*},{*cites:c2*}*)* – papers citing only papers of the RCF,
- *c6=(*{*i,j*},{*cites:c0, cites:c2*}*)* – papers citing only papers about lattice theory, i.e., in *c0*, as *c2* is redundant,
- *c7=(*{*e,f,k,l*},{*develops:c2*}*)* – developments of papers citing papers of the RCF.

Step 2 Given Lattice 2, richer than the initial one, narrow scaling is applied again upon *cites* and *develops*. While scaling upon *cites* does not add anything new, *develops* makes new incidences appear. First, as all the developed papers belong to the extent of*c5*, all the level-on papers also get the *develops:c5* attribute. Moreover, k and l also get *develops:c6*. The scaling yields a new RCF and its corresponding lattice, both given in Fig. 4.

					cites				develops		
		lt	mmi	se	2	0	3	4	2	5	6
gw99	a	x									
bm70	b	x									
are03	c				x						
dic96	d				x						
arePHD	e								x	x	
hucHDR	f								x	x	
bot91	g		x								
moo96	h			x							
val03	i				x	x					
leber02	j				x	x					
rouaPHD	k								x	x	x
leberHDR	l								x	x	x

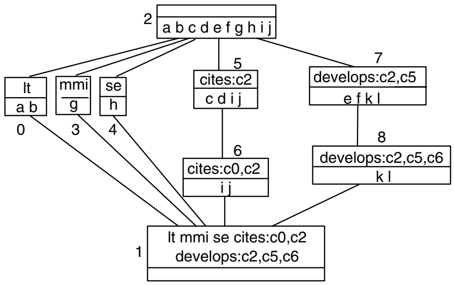


Fig. 4. Narrow scaling, step 2. **Left:** The scaled context; **Right:** Lattice 3.

The only new abstraction discovered at this stage is *c8* comprising publications that develop papers citing only papers about lattice theory. Step four terminates the analysis process, as no new concepts will be produced by further scaling. The interpretations of the formal concepts from the final lattice and their respective translations into DL are provided in Table 1.

5 Wide scaling-based RCA

The trace of the process with a wide scaling scheme starts immediately after the basic step of lattice construction on the unscaled context (see Fig. 2).

Step 1 When the inital lattice is used to scale upon *cites*, only the descriptions of papers c, d, i, j evolve. Hence all level-two papers get the attributes *cites:c0* and *cites:c2*, whereas c gets *cites:c3* as well, and d *cites:c4*. The result is to be seen in the scaled context in Fig. 5.

Applying wide scaling to *develops* and the initial lattice yields the same results as in the identical step of the narrow scaling-based process. Thus, *c2* being the only one whose extent comprises developed papers, all level-one papers get the attribute *develops:c2*. This yields the RCF depicted in Fig. 5 together with its lattice. The newly constructed concepts *c5*, *c6*, and *c7* represent, respectively, papers citing at least one paper about lattice theory,

Id	textual interpretation	translation into DL
$c0$	papers on lattice theory	AboutLatticeTheory
$c1$	papers having all properties	$\bot$
$c2$	all papers of the dataset	Paper
$c3$	papers on man machine interface	AboutManMachineInterface
$c4$	papers on software engineering	AboutSoftwareEngineering
$c5$	papers citing papers of the dataset	$C5 \equiv \exists\text{cites}.\top \sqcap \forall\text{cites}.\text{Paper}$
$c6$	papers citing only papers on lattice theory	$C6 \equiv \exists\text{cites}.\top \sqcap \forall\text{cites}.\text{AboutLatticeTheory}$
$c7$	papers developing only papers that cite only papers of the dataset	$\exists\text{develops}.\top \sqcap \forall\text{develops}.C5$
$c8$	papers developing only papers that cite only papers on lattice theory	$\exists\text{develops}.\top \sqcap \forall\text{develops}.C6$

Table 1. Narrow scaling. Interpretation of the mined concepts.

papers citing at least one paper both about man machine interface and lattice theory, and papers citing at least one paper both about software engineering and lattice theory. Furthermore, *c8* represents papers that develop at least one paper of the experiment.

					cites				develops
		lt	mmi	se	2	0	3	4	2
gw99	a	x							
bm70	b	x							
are03	c				x	x	x		
dic96	d				x	x		x	
arePHD	e								x
hucHDR	f								x
bot91	g		x						
moo96	h			x					
val03	i				x	x			
leber02	j				x	x			
rouaPHD	k								x
leberHDR	l								x

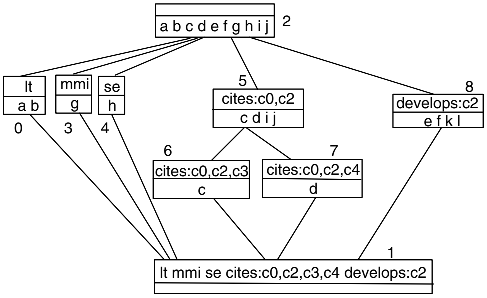
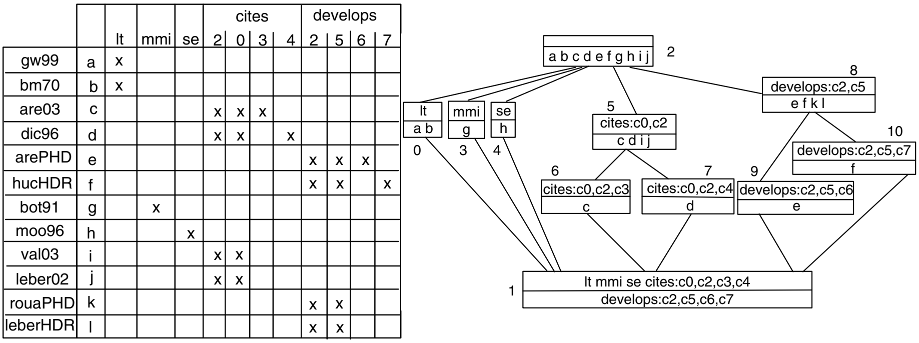


Fig. 5. Wide scaling. **Left:** RCF 2; **Right:** Lattice 2 on papers.

Step 2 Applying wide scaling to *cites* and the lattice of Fig. 5 does not bring any new incidence pair to the context. In contrast, scaling upon *develops* creates new attributes out of the concepts discovered at the previous step, i.e., *c5* to *c8*, and hence abstractions. Thus, all level-one papers get the *develops:c5* attribute as the extent of *c5* comprises all level-two concepts. In addition, *e* gets *develops:c6* and *f develops:c7*. The RCF of step 2 is drawn in Fig. 6 together with its lattice.



		lt	mmi	se	cites 2	cites 0	cites 3	cites 4	develops 2	develops 5	develops 6	develops 7
gw99	a	x										
bm70	b	x										
are03	c				x	x	x					
dic96	d				x	x		x				
arePHD	e								x	x	x	
hucHDR	f								x	x		x
bot91	g		x									
moo96	h			x								
val03	i				x	x						
leber02	j				x	x						
rouaPHD	k								x	x		
leberHDR	l								x	x		

Fig. 6. Wide scaling. **Left:** RCF 3; **Right:** Lattice 3 on papers.

The newly formed concepts *c9* and *c10* represent papers that develop papers citing: papers about man machine interface and lattice theory and papers about software engineering and lattice theory, respectively. Moreover, the already existing concept *c8* gets more focused as it turns out to represent papers that develop papers citing work on lattice theory. The interpretation and the translation into a DL format of all the concepts from the final lattice is presented in Table 2.

Id	textual interpretation	translation into DL
$c5$	papers citing one+ paper on lattice theory	∃cites.AboutLatticeTheory
$c6$	papers citing one+ paper on lattice theory and one+ on man machine interface	∃cites.AboutLatticeTheory ⊓ ∃cites.AboutManMachineInterface
$c7$	papers citing one+ paper on lattice theory and one+ paper on software engineering	∃cites.AboutLatticeTheory ⊓ ∃cites.AboutSoftwareEngineering
$c8$	papers developing one+ paper that cites one+ paper on lattice theory	∃develops.∃cites.AboutLatticeTheory
$c9$	papers developing one+ paper that cites one+ paper on both lattice theory and man machine interface	∃develops.∃cites.(AboutLatticeTheory ⊓ AboutManMachineInterface)
$c10$	papers developing one+ paper that cites one+ paper on both lattice theory and software engineering	∃develops.∃cites.(AboutLatticeTheory ⊓ AboutSoftwareEngineering)

Table 2. Wide scaling: interpretation of concepts (only unseen in Table 1).

6 Conclusion

The RCA framework illustrated here is a first step towards the complete interoperability between data mining and KR tools. Indeed, its input is fully compatible with the standard data models, e.g., the relational one, while its results are easily expressible in terms of a DL language. Therefore, the knowledge mined from the input data is directly available for reasoning and problem-solving.

Many issues with RCA are yet to be tackled: First, the scalability is still an open issue, since the size of lattices grows rapidly w.r.t. the growth of relations between contexts. Various tracks for preventing combinatorial explosion are currently explored, e.g. using reduced structures such as iceberg lattices or Galois sub-hierarchies. Next, algorithmic aspects are among primary concerns. For instance, efficiency could be further improved by replacing construction from scratch by incremental lattice maintenance. Finally, we are currently studying further scaling policies, e.g., the quantified existential restrictions providing upper/lower limits of the number of links to lay in a concept.

References

BAADER, F., CALVANESE, D., MCGUINNESS, D., NARDI, D., and PATEL-SCHNEIDER, P. (Eds.) (2003): *The Description Logic Handbook.* Cambridge University Press.

BRACHMAN, R. J. and ANAND, T. (1996): The Process of Knowledge Discovery in Databases. In: U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy (Eds): *Advances in Knowledge Discovery and Data Mining.* AAAI/MIT Press, 37–57.

DIDAY, E. (1998): Symbolic Data Analysis: a Mathematical Framework and Tool for Data Mining. In: A. Rizzi, M. Vichi, and H.-H. Bock (Eds.): *Advances in Data Science and Classification.* Springer, Berlin, 409–416.

GANTER, B. and WILLE, R. (1999): *Formal Concept Analysis, Mathematical Foundations.* Springer, Berlin.

HUCHARD, M., NAPOLI, A., ROUANE H., M., and VALTCHEV, P. (2007): A Proposal for Combining Formal Concept Analysis and Description Logics. In: S. Kuznetsov and S. Schmidt (Eds.): *Proc. of the 5th Intl. Conf. on Formal Concept Analysis.* Springer, Berlin.

POLAILLON, G. (1998): *Organisation et interprétation par les treillis de Galois de données de type multivalué, intervalle ou histogramme*, Thèse de doctorat, Université Paris IX Dauphine.

PREDIGER, S. and STUMME, G. (1999): Theory-driven Logical Scaling: Conceptual Information Systems Meet Description Logics. In: E. Franconi and M. Kifer (Eds.): *Proc. of the 6th Intl. Workshop on Knowledge Representation meets Databases.* Linköping, Sweden, 46–49.

PREDIGER, S. and WILLE, R. (1999): The Lattice of Concept Graphs of a Relationally Scaled Context. In: W. Tepfenhart and W. Cyre (Eds): *Proc. of the 7th Intl. Conf. on Conceptual Structures.* Springer, Berlin. 401–414.

Representation of Concept Description by Multivalued Taxonomic Preordonance Variables

Israël-César Lerman[1] and Philippe Peter[2]

[1] Irisa-Université de Rennes 1,
Campus de Beaulieu, 35042 Rennes Cédex, France, *lerman@irisa.fr*

[2] Lina, École Polytechnique de l'Université de Nantes
- La Chantrerie - BP 50609 - 44306 Nantes Cédex, France,
Philippe.Peter@polytech.univ-nantes.fr.

Abstract. Mathematical representation of complex data knowledge is one of the most important problems in Classification and Data Mining. In this contribution we present an original and very general formalization of various types of knowledge. The specific data are endowed with biological descriptions of phlebotomine sandfly species. Relative to a descriptive categorical variable, subsets of categories values have to be distinguished. On the other hand, hierarchical dependencies between the descriptive variables, associated with the mother → daughter relation, have to be taken into account. Additionally, an ordinal similarity function on the modality set of each categorical variable. The knowledge description is formalized by means of a new type of descriptor that we call "Taxonomic preordonance variable with multiple choice". Probabilistic similarity index between concepts described by such variables can be built.

1 Introduction

An early work (Lerman and Peter (1988), Lerman and Peter (1989)) is revisited here in a clearer, more synthetic and more accurate manner. In order to build similarity indices between complex descriptions, a mathematical representation of structured data by a knowledge expert is needed. This subject is becoming more and more important in Classification and Data Mining (Batagelj (1989), Bock and Diday (2000), Lerman (2000), Pennerath and Napoli (2006)). This work results from a collaboration with the late Jacques Lebbe.

This collaboration took place when Diday introduced the general idea of logical knowledge data analysis that he called "symbolic" data analysis Diday (1989). In this case and for a description of an objects set by attributes, the attribute value on a given object is not necessarily reduced to a single element of the scale associated with the concerned attribute. In other words the description system (attribute, single value) is left and substituted by the system (attribute, knowledge value). For example let us consider a knowledge value of a categorical attribute on a given object; this can be endowed with a logical formula on the category set, satisfied by the described object. Another example can be given by a probability distribution over the category set expressing uncertainty for the attributed value. In our subsequent development we consider only qualitative descriptions. A categorical attribute will also be called qualitative variable and a category value is expressed in terms of modality of the concerned qualitative variable.

Often, in “symbolic” data analysis papers qualitative data analysis is improperly interpreted as belonging to the “symbolic” domain. For this reason we prefer to speak in terms of “knowledge” data analysis. Furthermore, the notion of a classical data table which crosses an object set with an attribute set, is neglected and even rejected for knowledge description in (Diday (1989)). However, some evolution can be noticed in (Billard and Diday (2003)). From the begining (1988) the general notion of data table has played a fundamental part in our approach of knowledge data analysis. The only distinction considered is defined by the difference in nature of the cell content corresponding to the value of a descriptive variable on a given object. Semantic data relative to the scale associated with the value set of a given attribute can be recorded separately. On the other hand, logical relationships between descriptive variables have to be integrated in order to build the most synthetic attributes. In this paper we will be concerned by this type of construction leading to a very general and multivalued structured attribute called “taxonomic preordonance variable with multiple choice”.

This type of descriptive variable or “descriptor” has been obtained by a formalization of the expert knowledge of the biological descriptions of phlebotomine sandflies of French Guiana (Lebbe et al. (1987)). Descriptions are very complex. Each species is a class of specimens and its description must represent not only a prototype, but all possible variations in the species. Thus, the description by a qualitative variable of a given species, requires - most often - a subset of possible modalities. For sake of generality, we assume that the value of a given variable on a given species is defined by a probability distribution on a collection of modality subsets of this variable. Moreover, descriptive attributes are related by the mother → daughter relation; that is to say, if (v^0, v^1) is a such ordered pair of variables, v^1 is only defined when v^0 takes some of its values. Finally, we assume an ordinal similarity function on the modality set of each variable. A mathematical coding of this function in terms of a binary weighted relation is given in Section 3. In order to address the problem of conceptual knowledge description, Section 2 introduces the general notion of qualitative variable with multiple choice. The mother → daughter relations among the descriptive attributes lead to taxonomic variables organizing the initial qualitative variables (Section 4). By combining this structuration with local ordinal similarities, established on the respective modality sets of the different qualitative variables, we obtain the "taxonomic preordonance variable". Its construction and its mathematical coding are discussed in Section 5. In our description the components of a taxonomic preordonance variable are qualitative attributes with multiple choice. By integrating this descriptive property, "taxonomic preordonance variable with multiple choice" is derived. Section 6 is devoted to clarify the value set of a such variable. A similarity index between two concepts (or classes) described by this variable, has been proposed in (Lerman and Peter (1988), Lerman and Peter (1989)). Relative to a description by many taxonomic preordonance variables with multiple choice, a statistical normalization process was considered in order to establish a probabilistic similarity index. The latter is employed in the LLA (Likelihood of the Linkage Analysis) hierarchical classification method (Lerman (1993), Lerman and Peter (1988), Lerman and Peter (1989)). For concision reasons, these last aspects cannot be reported in this paper.

2 Qualitative variable with multiple choice

As mentioned above the data which have motivated this work are knowledge biological descriptions of species of phlebotomine sandflies of French Guiana (Lebbe et al. (1987)). Let us consider the 33rd variable of this description: "Aspect of individual duct". Its modalities are:

1. Smooth non-sclerotized
2. Smooth sclerotized
3. Transversely striated or annulated
4. With small prominent tubercles

The knowledge description of a given species (e.g. Lutzomyi carvalhoi) can be expressed as follows: "Specimens of this species have the value 1 and others of the same species have the value 3".

In these conditions, the value of the qualitative variable with multiple choice is defined by the modality subset $\{1, 3\}$, or equivalently by the conjunction 1&3. Thus, a qualitative variable with multiple choice is directly deduced from an ordinary qualitative (categorical) variable, for concept (one may also say class) description. For this, a given value is then defined in terms of a modality subset of the initial variable, or equivalently, in terms of a modality conjunction.

More formally, let us consider a universe $\mathcal{U}$ of elementary units (the whole set of phlebotomine sandflies specimens in our case) and suppose defined on $\mathcal{U}$ a partition where a distinct concept is associated with each of its classes. Let us denote by $\mathcal{C}$ the set of concepts or classes (the set of species in our case). Now, let us consider a classical qualitative (categorical) variable v defined on $\mathcal{U}$. For u belonging to $\mathcal{U}$, $v(u)$ is a single value of the modality set of v. Now, J coding this modality set, assume a collection of subsets of J:

$$\mathcal{P}_v(J) = \{J_1, J_2, ..., J_i, ..., J_k\} \tag{1}$$

so that for each concept c of $\mathcal{C}$, only one subset J_i of modality values can be met in c. The qualitative variable with multiple choice deduced from v and which we denote by $v_{\mathcal{C}}$, is defined as a mapping of $\mathcal{C}$ onto $\mathcal{P}_v(J)$

$$v_{\mathcal{C}} : \mathcal{C} \longrightarrow \mathcal{P}_v(J) \tag{2}$$

For generality reasons, we will consider a higher description level introducing a probability distribution $\{p_i \mid 1 \leq i \leq k\}$ on $\mathcal{P}_v(J)$. Therefore the $v_{\mathcal{C}}$ value can be written as follows:

$$(J_1, p_1)\&...\&(J_i, p_i)\&...\&(J_k, p_k) \tag{3}$$

or, more explicitly:

$$(\&\{j \mid j \in J_1\}, p_1)\&...\&(\&\{j \mid j \in J_i\}, p_i)\&...\&(\&\{j \mid j \in J_k\}, p_k) \tag{4}$$

This type of description introduces uncertainty in the concept recognition or can be associated with a partition of $\mathcal{C}$ in higher concepts (genus in our case) which can be described by (3). In this richer case the descriptive variable can be expressed in terms of probabilistic qualitative variable with multiple choice.

Because of the generalized data table formalization, the included value in the entry situated at the intersection of the c row and the $v_{\mathcal{C}}$ column is given by expression (3) or by that (4).

3 Preordonance structure on the modality set of a qualitative variable. Representation

A "preordonance" qualitative variable is a qualitative (categorical) variable whose modality set is endowed with an ordinal similarity. Formally, a preordonance is a total preorder (ranking with ties) on the set of unordered (or ordered) modality pairs. By denoting $J = \{1, 2, ..., j, ..., m\}$ the modality codes of the concerned variable, the total preorder is defined on the following set:

$$J^{\{2\}} = \{(j, h) \mid 1 \leq j \leq h \leq m\} \tag{5}$$

(Lerman and Peter (1985), Lerman (1987), Ouali-Allah (1991), Lerman (2000), Lerman and Peter (2003)). This total preorder is established by the expert knowledge by going from the highest ordinal similarity pairs to the lowest ones. For two pairs (j, h) and (j', h'), two cases have to be considered: either

$$(j, h) > (j', h') \tag{6}$$

or

$$(j, h) \sim (j', h') \tag{7}$$

In the first case j and h are assumed, without loss of generality, to be more similar than j' and h' and in the second case, j and h are assumed to be equally similar as j' and h'.

Let us consider the above example of the previous section ("Aspect of individual duct") where $J = \{1, 2, 3, 4\}$. By going from the most similar modality pair to the least similar one, the submitted preordonance by the expert is the following:

$$11 \sim 22 \sim 33 \sim 44 > 12 \sim 13 \sim 23 > 14 \sim 24 \sim 34 \tag{8}$$

where jh represents the pair $\{j, h\}, 1 \leq j \leq h \leq 4$.

The total preorder on $J^{\{2\}}$ is coded by means of the "mean rank function" given by the table:

$$\{r_{jh} \mid 1 \leq j \leq h \leq m\} \tag{9}$$

where the rank r_{jh} is computed with the following equation:

$$r_{jh} = l_1 + l_2 + ... + l_{p-1} + \frac{1}{2} \times (l_p + 1) \tag{10}$$

where l_q denotes the q^{th} class size of the total preorder on $J^{\{2\}}$ according to an increasing order and where jh belongs to the p^{th} class.

Then, in our example, the above table (9) becomes in our example:

$$\{8.5, 5, 5, 5, 2, 8.5, 5, 2, 8.5, 2, 8.5\} \tag{11}$$

4 Taxonomic variable organizing a set of dependent variables. Representation

Let us begin by an example and consider the variables 1, 18, 19 and 20 of Lebbe et al. (1987) that we denote v^1, v^{21}, v^{31}, and v^{32}, respectively. v^1 is the "Sex" attribute, v^{21} is defined by the "Number of style spines", v^{31} indicates the "Distribution of 4 style spines" and v^{32}, the "Distribution of 5 style spines". The value sets of these variables are:

{1: male, 2: female}, {1, 2, 3, 4, 5}, {1, 2, 3, 4, 5, 6} and {1, 2, 3, 4, 5}, respectively, where each integer code is associated with a modality value. We obtain the following taxonomic structure:

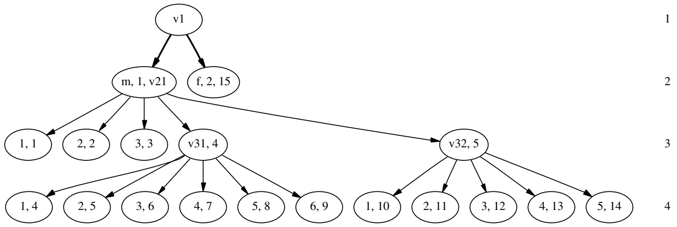


Fig. 1. Taxonomic variable.

Clearly, the variable v^{21} is defined only when the v^1 value is 1. On the other hand, the variables v^{31}, and v^{32} are defined only when the values of v^{21} are 4 and 5, respectively. The mother variable of v^{31} and v^{32} is v^{21}.

More generally, a taxonomic variable denoted ω, organizing a set of logically dependent variables consists of a sequence of collections of qualitative variables of the following form:

$$\omega = (\{v^1\}, \{v^{21}, v^{22}, ..., v^{2k_2}\}, ..., \{v^{p1}, v^{p2}, ..., v^{pk_p}\},\\ ..., \{v^{q1}, v^{q2}, ..., v^{qk_q}\}) \quad (12)$$

The first collection is necessarily reduced to one element: the variable v^1. This corresponds to the root of the tree representing the taxonomic variable ω. The variables $v^{p1}, v^{p2}, ..., v^{pk_p}$ are represented at the p^{th} level of this tree built in a descendant fashion. The set of variables $\{v^{p1}, v^{p2}, ..., v^{pk_p}\}$ can be divided into disjoint subsets (classes) according to mother variable concerned. More precisely, if $\{v^{pi}, ..., v^{pi'}\}$ $(i' > i)$ denotes a such subset, two of its elements v^{pj} and $v^{pj'}$ are characterized by the same mother variable $v^{(p-1)h}$. They are respectively defined on two distinct subsets of the described objects (specimens of phlebotomine sandflies in our case), where each subset is defined by one modality of $v^{(p-1)h}$.

In the above example ω is instanciated as follows:

$$\omega = (\{v^1\}, \{v^{21}\}, \{v^{31}, v^{32}\}) \quad (13)$$

The structure associated with this variable is represented in terms of a "ultrametric preordonance" (Lerman (1970), Lerman (2000)) on the set of taxonomy leaves (in the above example the cardinality of this set is 15). By considering a decreasing construction of the taxonomic tree from the root to the leaves, this total preorder on the set of unordered leaf pairs is such that, the higher the rank of a given pair, the lower the first node which underlies the two concerned leaves. Thus, in the above example, the pair $\{6, 8\}$ has the same rank as that of $\{10, 12\}$. The latter is greater than that of $\{7, 12\}$, which is equal to the $\{2, 3\}$ rank and so on ...

Now, let us denote by L the set of the taxonomy leaves, a ranking function r coding the total preorder defined by the utrametric preordonance is characterized by the following equation:

$$(\forall\{x, y, z\} \in P_3(L)), r(x, z) \geq min(r(x, y), r(y, z)) \tag{14}$$

where $P_3(L)$designates the set of all 3-subsets of L.

As in the general case (see Section 3), we adopt the notion of "mean rank" for the ranking function. Mathematical formula can be derived, relating the tree shape with the mean rank function (Lerman and Peter (1988)). The highest rank is assigned to the elements of the preorder class constituted by the the pairs having the form: $\{x, x\}$, $x \in L$. In these conditions, the taxonomic variable is interpreted as a particular case of a preordonance variable.

5 Taxonomic preordonance variable. Representation

Let us reconsider here the above ordinal similarity structure endowed with a taxonomic variable ω organizing a set of logically dependent qualitative variables. We further assume that the modality set $\mathcal{M}^{pi}$ of a given qualitative variable v^{pi} (see 12) is endowed with a total preordonance (see Section 3), $1 \leq i \leq k_p, 1 \leq p \leq q$. These preordonances are locally defined variable by variable, they have to be integrated in the taxonomic structure.

In these conditions, we have to build a total preordonance on the set of the taxonomy leaves, or - equivalently - on the set of the associated complete chains, going from the root to the leaves. This preordonance must take into account both the preordonance defined in the above Section 4 and those we have just mentioned.

Such a preordonance is built step by step, decreasingly, with respect to the resemblance between terminal modalities corresponding to the taxonomy leaves. The general principle consists in refining the ultrametric preordonance associated with the taxonomy by means of the preordonances locally defined on the modality sets of the different variables.

More clearly, let us begin by the above example (see Figure 1) and consider the leaf sets $A = \{4,5,6,7,8,9\}$ and $B = \{10,11,12,13,14\}$ associated with the modality sets of the variables v^{31} and v^{32}, respectively. By denoting $P_2(A)$ (resp., $P_2(B)$) the unordered pairs from A (resp., B), $P_2(A) \cup P_2(B)$ determines a unique class of the total preorder defined by the tree structure. This class comprises all the element pairs joined at the level 3 of the taxonomy. Preordonance structures on the modality sets of the variables v^{31} and v^{32} provide total preorders on $P_2(A)$ and $P_2(B)$, respectively. These, can be represented by rank functions. Specifically, one may consider the mean rank functions r_A and r_B defined on $P_2(A)$ and $P_2(B)$, respectively. In these conditions, a ranking function $r_{A\cup B}$ on $P_2(A) \cup P_2(B)$ is deduced from r_A and r_B as follows:

$$r_{A\cup B} : P_2(A) \cup P_2(B) \longrightarrow Val(r_A) \cup Val(r_B) \tag{15}$$

where $Val(r_A)$ (resp., $Val(r_B)$) is the value set of r_A (resp., r_B). Consequently, $r_{A\cup B}(\{x,y\})$ is defined by $r_A(\{x,y\})$ if $\{x,y\} \in P_2(A)$ and by $r_B(\{x,y\})$ if $\{x,y\} \in P_2(B)$.

Therefore, according to the value scale of $r_{A\cup B}$, a total preorder on $P_2(A) \cup P_2(B)$ is induced. This substitutes the unique class $P_2(A) \cup P_2(B)$.

Let us continue with the above illustrative example. The next preorder class construction is given by the preordonance variable v^{21}. Its modality set C appears at the level 3 of the taxonomy. $P_2(C)$ is endowed with a total preorder. In the latter we have to do the following substitutions:

$$\begin{aligned}
(\forall x \in \{1,2,3\}), \{x,4\} &\leftarrow \{\{x,y\} \mid y \in A\} \\
(\forall x \in \{1,2,3\}), \{x,5\} &\leftarrow \{\{x,y\} \mid y \in B\} \\
\text{for } \{4,5\} &\leftarrow \{\{x,y\} \mid \{x,y\} \in A \times B\}
\end{aligned} \tag{16}$$

where the different pairs included in a given class substitution are interpreted as equally similar.

Now, let us give a general expression of the construction of a taxonomic preordonance variable. We begin by ordering the set

$$\Delta(L) = \{\{x,x\} \mid x \in L\} \tag{17}$$

according to the leaf depth in the taxonomy: in other words, the deeper the leaf, the higher the ordinal similarity between the represented category and itself. Thus, in the above example, for these pairs we have

$$\begin{gathered}
\{4,4\} \sim \{5,5\} \sim \{6,6\} \sim \{7,7\} \sim \{8,8\} \sim \{9,9\} \\
\sim \{10,10\} \sim \{11,11\} \sim \{12,12\} \sim \{13,13\} \sim \{14,14\} \\
> \{1,1\} \sim \{2,2\} \sim \{3,3\} > \{15,15\}
\end{gathered}$$

(18)

Let us reconsider here the general expression 12 of the taxonomic variable ω. Let us indicate by $\mathcal{M}(v^{qj})$ the modality set of the variable v^{qj}, $1 \leq j \leq k_q$. These modality sets are figured by the deepest leaves of the tree depicting ω. Then the next step of refining the ω taxonomic preordonance consists in introducing the total preorders defined by the preordonance variables v^{qj} on the set of unordered modality pairs of $\mathcal{M}(v^{qj})$, that we denote by $P_2(\mathcal{M}(v^{qj}))$, $1 \leq j \leq k_q$. The unique class

$$\mathcal{P} = \bigcup \{P_2(\mathcal{M}(v^{qj})) \mid 1 \leq j \leq k_q\} \tag{19}$$

is refined according to the mean rank functions defined on the sets

$P_2(\mathcal{M}(v^{qj}))$, $1 \leq j \leq k_q$, respectively. The global ranking function $r_{\mathcal{P}}$ on the union $\mathcal{P}$ is defined directly from the partial mean rank functions (see the example above in this Section):

$$(\forall j, 1 \leq j \leq k_q), (\forall \{x, y\} \in P_2(\mathcal{M}(v^{qj}))), r_{\mathcal{P}}(\{x, y\}) = r_{qj}(\{x, y\}) \tag{20}$$

where r_{qj} designates the mean rank function on $P_2(\mathcal{M}(v^{qj}))$ associated with the preordonance variable v^{qj}.

Additionally, the ranking function, that we denote by R_U has to take into account the taxonomic structure. Consequently it can be written as follows:

$$(\forall \{x, y\} \in P_2(\mathcal{M}(v^{qj}))), R_{\mathcal{P}}(\{x, y\}) = r_{qj}(\{x, y\}) + card(L) \tag{21}$$

For all j, $1 \leq j \leq k_q$.

Thus, two modality pairs $\{x, y\}$ and $\{z, t\}$, belonging to two different sets $P_2(\mathcal{M}(v^{qj}))$ and $P_2(\mathcal{M}(v^{qj'}))$ $(j \neq j')$ are compared on the basis of their respective rank functions defined independently on the modality pairs of v^{qj} and on those of $v^{qj'}$. This is consistent with the similarity index construction (Lerman and Peter (1988), Lerman and Peter (1989)).

Now, let us consider the variable set $\{v^{p1}, v^{p2}, ..., v^{pk}\}$ introduced at the p^{th} level of the taxonomy (see 12). The respective modalities of each of these variables arise at the $(p+1)^{th}$ level. $\bigcup \{P_2(\mathcal{M}(v^{pi}) \mid 1 \leq i \leq k_p\}$ determines a unique class of the taxonomic preorder. For a given i $(1 \leq i \leq k_p)$, a total preorder is provided on $P_2(\mathcal{M}(v^{pi})$ by the preordonance variable v^{pi}. This refines the subclass $P_2(\mathcal{M}(v^{pi})$. Moreover, for $\{x, y\}$ belonging to $P_2(\mathcal{M}(v^{pi})$, if x (resp., y) is a node tree from which branches issue, the class of the terminal tree chains passing by x (resp., y) is substituted for x (resp., y) (see 16 in the above example). All the concerned pairs are interpreted as equally similar and the mean rank function value $r_{pi}(\{x, y\})$ deduced from the preordonance variable v^{pi}, is applied to all of these pairs. Denote $\mathcal{M}'(v^{pi}$ the extended value set and r'_{pi} the extended definition of the mean rank function r_{pi} on $P_2(\mathcal{M}'(v^{pi})$. From the set, denoted $\mathcal{R}'_p$, of rank functions

$$\mathcal{R}'_p = \{r'_{pi} \mid 1 \leq i \leq k_p\} \tag{22}$$

a unique rank function $r_{\mathcal{P}}$ is induced on

$$\mathcal{P} = \bigcup\{P_2(\mathcal{M}'(v^{pi})) \mid 1 \leq i \leq k_p\} \tag{23}$$

as follows:

$$\forall i, 1 \leq i \leq k_p, \forall\{x,y\} \in P_2(\mathcal{M}'(v^{pi})), r_{\mathcal{P}}(\{x,y\}) = r'_{pi}(\{x,y\}) \tag{24}$$

In these conditions, according to the $r_{\mathcal{P}}$ values, a total preorder on $\mathcal{P}$ is provided. Besides, $r_{\mathcal{P}}$ enables a consistent construction of a similarity index between described objects or concepts (Lerman and Peter (1989)). For this purpose we substitute for $r_{\mathcal{P}}$ a ranking function $R_{\mathcal{P}}$ which takes into account all the leaf pairs preceeding $\mathcal{P}$ in the taxonomic order, strictly. More clearly, by denoting P_{p+1} this set of leaf pairs

$$\forall\{x,y\} \in P_2(\mathcal{M}'(v^{pi})), R_{\mathcal{P}}(\{x,y\}) = r_{\mathcal{P}}(\{x,y\}) + card(P_{p+1}) \tag{25}$$

For all $i, 1 \leq i \leq k_p$.

In the case of the above example we have

$$P_3 = \{\{4,10\},\{4,11\},...,\{4,14\},\{5,10\},\{5,11\},...,\{9,13\},\{9,14\}\} \tag{26}$$

The above ranking function $R_{\mathcal{P}}$ is defined for all leaf pairs joined at p^{th} level (first junction). Each leaf can be associated with a terminal tree chain from the $(p+1)^{th}$ level. In these conditions, a global ranking function R is built from its $R_{\mathcal{P}}$ restrictions.

At the final step, the set of all complete chains of the tree represented by the leaf set, is provided with a total preorder. Consequently, the taxonomic variable is enriched and becomes a “taxonomic preordonance variable”, that we code by means of the ranking function R.

6 Taxonomic variable with multiple choice

The descriptive structure of the global variable considered here is defined in the previous Section 4. Nevertheless, the “value” of a given component variable v^{pi} of the taxonomy p level on a given concept is defined by a probabilistically weighted conjunction of conjunctions on the set $J = \mathcal{M}(v^{pi})$ of its modalities (see Formula (4) in Section 2). In Lebbe et al. (1987), only deterministic values are considered and, then, the value of such a variable v^{pi}

on a given concept is defined by a unique conjunction whose terms belong to J (or equivalently, as a subset of J) having the following form

$$\&\{j \mid j \in G\} \tag{27}$$

where G is a subset of J.

Let us begin with an example and imagine that for the above descriptive variables v^1, v^{21}, v^{31} and v^{32}, introduced in Section 4, one has the following values on a given concept (species in our case) c:

$$\begin{aligned}
v^1(c) &= 1 \\
v^{21}(c) &= (1\&2, 0.4)\&(2\&3\&4, 0.2)\&(4\&5, 0.4) \\
v^{31}(c) &= (1\&2, 0.4)\&(2\&3, 0.6) \\
v^{32}(c) &= (2\&3\&4, 0.8)\&(3\&5, 0.2)
\end{aligned} \tag{28}$$

Denote here by w the taxonomic variable organizing the preceeding variables (see Figure 1). One possible value of w on an element u drawn from c may be: $w(u) = 11\&12$, corresponding to $v^1(u) = 1$ and $v^{21}(u) = 1\&2$. Another possible value of w may be $w(u) = 12\&13\&141\&142$ corresponding to $v^1(u) = 1$, $v^{21}(u) = 2\&3\&4$ and $v^{31}(u) = 1\&2$.

The probability of the v^{21} value is 0.4 and that of the v^{31} value is $0.2 \times 0.4 = 0.08$. These values are obtained according to computational principle of a conditional probability.

More precisely, denoting $\vee$ the logical disjunction, the w value on a random unit u^* provided from the concept c, can be written:

$$\begin{aligned}
w(u^*) = (11\&12, 0.4) \vee (12\&13\&141\&142, 0.08) \\
\vee(12\&13\&142\&143, 0.12) \vee (141\&142\&152\&153\&154, 0.128) \\
\vee(141\&142\&153\&155, 0.032) \vee (142\&143\&152\&153\&154, 0.192) \\
\vee(142\&143\&153\&155, 0.048)
\end{aligned} \tag{29}$$

or, by using the coding of the taxonomy leaves with the integers 1 to 15 (see Figure 1),

$$\begin{aligned}
w(u^*) = (1\&2, 0.4) \vee (2\&3\&4\&5, 0.08) \\
\vee(2\&3\&5\&6, 0.12) \vee (4\&5\&11\&12\&13, 0.128) \\
\vee(4\&5\&12\&14, 0.032) \vee (5\&6\&11\&12\&13, 0.192) \\
\vee(5\&6\&12\&14, 0.048)
\end{aligned} \tag{30}$$

The weight sum is a probability sum and consequently, is equal to 1.

Now, the associated value of w on the concept c, can be put in the following form:

$$\begin{aligned} w(c) = (1\&2, 0.4) \wedge (2\&3\&4\&5, 0.08) \\ \wedge(2\&3\&5\&6, 0.12) \wedge (4\&5\&11\&12\&13, 0.128) \\ \wedge(4\&5\&12\&14, 0.032) \wedge (5\&6\&11\&12\&13, 0.192) \\ \wedge(5\&6\&12\&14, 0.048) \end{aligned} \tag{31}$$

where $\wedge$ is another notation for a conjunction.

Thus $w(c)$ consists of a probability distribution on a collection of leaf subsets of the taxonomic tree or, equivalently, on complete chain subsets of this tree.

The general case can be easily derived from the above illustration. Relative to the modality set $\mathcal{M}(v^{qi})$ of the qualitative variable v^{qi} appearing in the taxonomic variable ω (see 12), let us consider the possible values of v^{qi}. These values, can be put as follows (see Section 2):

$$\{(J_l, p_l) \mid 1 \leq l \leq m_{qi}\} \tag{32}$$

where $\{J_l \mid 1 \leq l \leq m_{qi}\}$ is a collection of m_{qi} modality subsets and where J_l occurs with the probability p_l, $1 \leq l \leq m_{qi}$, $(\sum\{p_l \mid 1 \leq l \leq m_{qi}\} = 1)$.

Now, consider for a given l, a modality x_l belonging to J_l for which a v^{qi} daughter variable $v^{(q+1)j}$ is defined. With its modality set designated by $\mathcal{M}(v^{(q+1)j})$ associate its values in the above form:

$$\{(J'_{l'}, p'_{l'}) \mid 1 \leq l' \leq m_{(q+1)j}\} \tag{33}$$

where $J'_{l'}$ is a modality subset of $\mathcal{M}(v^{(q+1)j})$ occuring in c with the probability $p'_{l'}$, $(\sum\{p'_{l'} \mid 1 \leq l' \leq m_{(q+1)j}\} = 1)$.

The joint probability of J_l and $J'_{l'}$ is obtained according to conditional probability principle by $p_l \times p'_{l'}$. This can also be expressed as follows:

$$Pr(\&\{x_l \& x'_{l'} \mid (x_l, x'_{l'}) \in J_l \times J'_{l'}\}) = p_l \times p'_{l'} \tag{34}$$

Thus $p_l \times p'_{l'}$ is the probability assigned to the conjunction of partial chains of the two elements x_l and $x'_{l'}$ belonging to J_l and $J'_{l'}$, respectively.

Finally and recursively, the value set of the taxonomic variable ω on c is obtained. This value set consists of a probabilized set of conjunctions of complete chains of the taxonomic tree. Note that each complete chain can be represented by its terminal leaf. Denoting as J the set of all leaves, one can easily see that the probabilized value of the concerned variable has the same general structure as that presented in Section 2 (see 3). More specifically, a given leaf conjunction is concerned by a unique sequence of the initial qualitative variables, totally ordered by the mother → daughter relation.

7 Conclusion

As claimed above, the conceptual notion of a data table remains fundamental in analyzing logical data knowledge. And, the only difference concerns the logical nature of a cell content, describing an object or a concept (class) with respect to a descriptive variable. In case of absence of missing data, the description complexity proceeds from two main causes. The first is associated with the complexity of the relation on the value set of the description endowed with the expert knowledge. The second results from the level knowledge of the description on the entities (objects or concepts) to be clustered according to their similarities. For a concept description by taxonomic preordonance variables with multiple choice, the structural aspects of the value scale have been studied in Sections 3, 4 and 5. Whereas, the formalization of the expert knowledge relative to the values of such descriptive variables on the described concepts, is given in Sections 2 and 6.

A rough similarity index between described concepts has been built in order to minutely take into account the two complexity origins mentioned above (Lerman and Peter (1988), Lerman and Peter (1989)). In case of a description by many multivalued taxonomic preordonance variables, the integration process of the rough similarity indices (taken variable by variable), into the LLA hierarchical classification method (Lerman (1993)), follows a general principle given in (Lerman and Peter (1985), Lerman (1987), Lerman (2000) Lerman and Peter (2003)). This approach is comprised in the hierarchical classification software named CHAVLH (Classification Hiérarchique par Analyse de la Vraisemblance des Liens en cas de variables Hétérogènes). Significant and interesting results have been obtained in the hierarchical classification of 142 species described by 61 taxonomic preordonance variables with multiple choice (Lerman and Peter (1988)).

Let us end by a general remark: taking into account the expert knowledge in building structured descriptive attributes enables to obtain more synthetic and more robust cluster organization; however, "explaining" the general features of a given "significant" cluster becomes more difficult.

References

BATAGELJ, V. (1989): Similarity Measures Between Structured Objects. In: *Studies in Physical and Theoretical Chemistry.* Elsevier, Amsterdam, Vol. 63, 25–40.

BILLARD, L. and DIDAY, E. (2003): From Statistics of Data to the Statistics of Knowledge: Symbolic Data Analysis. *Journal of the American Statistical Association, Vol. 98, No 462, 470-487.*

BOCK, H. H. and DIDAY, E. (eds) (2000): *Analysis of Symbolic Data: Explotary Methods for Extracting Statistical Information From Complex Data.* Springer-Verlag, Berlin.

DIDAY, E. (1989): Introduction à l'approche symbolique en analyse des données. *Recherche opérationnelle/Operations Research vol. 23, 2, 193-236.*

LEBBE, J., DEDET, J.P. and VIGNES, R. (1987): Identification assistée par ordinateur des phlébotomes de la Guyane Française. *Institut Pasteur de la Guyane Française, Version 1.02.*

LERMAN, I.C. (1970): *Les bases de la classification automatique* Gauthier-Villars, collection "Programmation", Paris.

LERMAN, I.C. (1987): Construction d'un indice de similarité entre objets décrits par des variables d'un type quelconque. Application au problème de consensus en classification. *Revue de Statistique Appliquée, XXXV(2), 39-60.*

LERMAN, I.C. (1993): Likelihood Linkage Analysis (LLA) Classification Method (Around an Example Treated by Hand.) *Biochimie 75, Elsevier editions, 379-397.*

LERMAN, I.C. (2000): Comparing Taxonomic Data. *Math. & Sci. hum.* 38^e *année, 150, 37-51.*

LERMAN, I.C. and PETER, P. (1985): Élaboration et logiciel d'un indice de similarité entre objets d'un type quelconque. Application au problème de consensus en classification. *Publication Interne Irisa* n^o *262.*

LERMAN, I.C. and PETER, P. (1988): Classification en présence de variables préordonnances taxonomiques à choix multiple. Application à la structuration des phlébotomes de la Guyane Française. *Publication Interne Irisa* n^o *426.*

LERMAN, I.C. and PETER, P. (1989): Classification of Concepts Described by Taxonomic Preordonnance Variables with Multiple Choice. In E. Diday Editor *Data Analysis, Learning symbolic and numerical knowledge* Nova Science Publishers, 73–87.

LERMAN, I.C. and PETER, P. (2003): Indice probabiliste de vraisemblance du lien entre objets quelconques, analyse comparative entre deux approches. *Revue de Statistique Appliquée, LI(1), 5-35.*

OUALI-ALLAH, M. (1991): *Analyse en préordonnance des données qualitatives. Application aux données numériques et symboliques*, thèse de doctorat de l'Université de Rennes 1.

PENNERATH, F. and NAPOLI, A. (2006): La fouille de graphes dans les bases de données réactionnelles au service de la synthèse en chimie organique. In: G. Ritschard and C. Djeraba (Eds.): *Extraction et Gestion des Connaissances.* Cépaduès, Toulouse, France, 517–528.

Recent Advances in Conceptual Clustering: CLUSTER3

Ryszard S. Michalski[1,2] and William D. Seeman[1]

[1] Machine Learning and Inference Laboratory
George Mason University, Fairfax, USA
[2] Institute of Computer Science
Polish Academy of Sciences, Warsaw, Poland
michalski@mli.gmu.edu, william.seeman@ngc.com

Abstract. Conceptual clustering is a form of unsupervised learning that seeks clusters in data that represent simple and understandable concepts, rather than groupings of entities with high intra-cluster and low inter-cluster similarity, as conventional clustering. Another difference from conventional clustering is that conceptual clustering produces not only clusters but also their generalized descriptions, and that the descriptions are used for cluster evaluation, interpretation, and classification of new, previously unseen entities. Basic methodology of conceptual clustering and program CLUSTER3 implementing recent advances are briefly described. One important novelty in CLUSTER3 is the ability to generate clusters according to the *viewpoint* from which clustering is to be performed. This is achieved through the *view-relevant attribute subsetting* (VAS) method. CLUSTER3's performance is illustrated by its application to clustering a database of automobile fatality accidents.

1 Introduction

Clustering is a fundamental methodology for discovering structure in data. A conventional approach to clustering seeks clusters of entities with high intra-cluster similarity and low inter-cluster similarity. The results of such clustering are therefore strongly dependent on the predefined similarity measure between entities. The similarity measure is typically computed on all attributes that characterize entities in the dataset. If it happens that some entities are very similar in terms of attributes that are irrelevant to the purpose of clustering, these entities may be put incorrectly into the same clusters, while entities that truly belong to the same cluster may be put into different clusters. Another aspect of conventional clustering is that its results are collections of clusters without any cluster descriptions. In many practical applications, however, the user seeks such descriptions, as they are useful for interpreting the obtained clusters and for classifying future entities.

There exists a large body of literature on clustering. The developed methods differ in terms of the similarity measures they use, the ways they generate clusters, and the way they represent them. For example, Gowda and Diday (1992) described a hierarchical clustering method which defines similarity

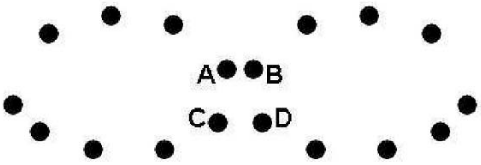


Fig. 1. Would you group points A and B, or C and D to the same of different clusters?

based on the position, span, and content of entities. Huang (1998) presented the k-modes algorithm as an extension to the well-known k-means algorithm and handles categorical data types. A popular AutoClass system (Cheeseman and Stutz (1996)) employs a Bayesian approach to partition the data into the most probable groupings. Conley et al. (2005) used a genetic algorithm approach to locate clusters of spatial point data with maximum density, and map the results geographically. The above examples are just a very small sampling of the many clustering methods that has been developed. A review of earlier clustering techniques can be found in Michalski and Stepp (1986) and a more recent one at Berkhin (2002).

This chapter presents recent advances in conceptual clustering that groups entities on the basis of their cohesion to concepts inherent in the data set, rather than a predefined measure of similarity among entities. Clustering entities in this manner attempts to reflect ways in which people group objects in a collection, when presented with clustering tasks. To illustrate conceptual clustering, consider Figure 1.

If clustering of entities represented as points in Figure 1 is done on the basis of their "similarity" defined by the reciprocal of the distance from each other, the points A and B, and C and D would be grouped into the same cluster, as they are closer to each other than to their other neighbors. A person, equipped with concepts of different shapes, however, would usually notice that these points are parts of two different ellipse-like configurations, and therefore would cluster them to different clusters. A program for conceptual clustering in which basic shapes are concepts in its background knowledge would also recognize the elliptical configurations of the points and cluster them accordingly. The above is a very simple illustration of the conceptual clustering approach, originally introduced by Michalski (1979).

The rest of this paper is organized as follows. Section 2 consists of several subsections, which present successively the cluster description language, the top-level algorithm for conceptual clustering, a multi-criterion clustering quality measure, and finally the view-relevant attribute subsetting (VAS) method that guides the clustering process according to its pre-defined goal. VAS and some other features have been implemented in CLUSTER3 that embodies recent advances in conceptual clustering. Section 3 illustrates CLUS-

TER3's performance by its application to a real-world problem of clustering fatality accidents. Conclusions and future directions are described in Section 4.

2 Conceptual clustering methodology

2.1 Cluster representation

The complexity of problem spaces and the overwhelming amount of data available for human interpretation creates the need for presenting results of clustering in easily interpretable forms. Conventional methods of clustering typically produce a collection of clusters or a hierarchy of clusters, without any description or explanation of the clusters. Such explanation-free results are often insufficient in practical applications, especially when the number of entities or the number of attributes describing them is large.

This chapter is concerned with conceptual clustering that seeks clusters in data that represent concepts described in a predefined form. It outputs a hierarchy of clusters as well as their generalized descriptions. The generated descriptions facilitate cluster interpretation and are useful for classifying future entities. In the method presented here, descriptions are in the form of conjunctive expressions in *Attributional Calculus*, a logic and simple representation language that combines elements of propositional, predicate and multi-valued calculus to facilitate inductive inference (Michalski (2004)). Such cluster descriptions are directly translatable to simple natural language statements, and thus are easy to interpret and understand. Geometrically, attributional descriptions correspond to one or more hyper-rectangles in a subset of the multi-dimensional space spanned over attributes present in the descriptions.

The fundamental building block of Attributional Calculus is an *attributional condition* or *selector*, whose simple form is:

[L *rel* R],

where L is an attribute, *rel* is relation, and R defines a subset of values from the attribute domain. Here are a few examples of selectors and their interpretation in natural language:

[house_type = colonial]	(the house type is colonial)
[color = blue $\vee$ red]	(the color is blue or red)
[size = 14..16]	(the size is between 14 and 16, inclusively)
[length $\geq$ 30]	(the length is greater than or equal 30)

where the measurement units for attributes "height" and "length" are specified in the attribute domain.

A logical conjunction of selectors constitutes an *attributional statement* or *complex*, and is used to form a description of a single cluster. Here is an example of a cluster description derived from the automobile fatality accidents dataset (see Section 3):

Cluster 1: [Road_profile = level] & [Road_cond = dry] & [Light_cond ≠ dawn]

2.2 Cluster hierarchies

To describe a large collection of objects in a simple and understandable way, people frequently organize it into a hierarchy. Biological taxonomies, government structures, and library catalogue systems are but a few examples of such hierarchical organizations. In data analysis, structuring dataset into a hierarchy is called clustering or unsupervised learning. For example, Ciampi et al. (2000) describe a method for creating hierarchies of data items and presenting them in the form of dendrograms that graphically illustrate similarities between entities. In data analysis and machine learning, it is often useful to also structure domains of attributes into hierarchies, e.g., see Michalski and Stepp (1983); Gowda and Diday (1992); Michalski (2004)).

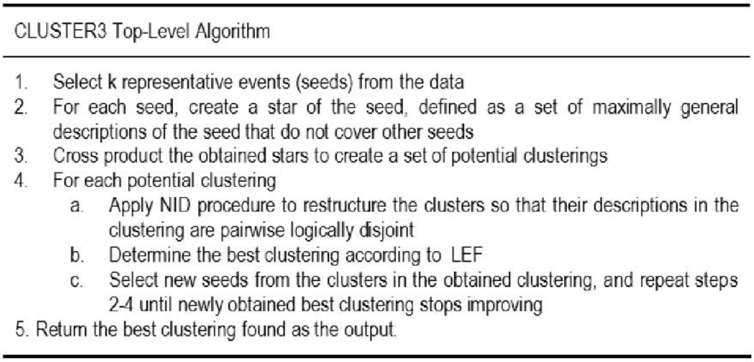
This chapter concerns a conceptual clustering method that structures datasets into hierarchies of clusters accompanied by their generalized descriptions. The method described here assumes that the dataset is in the form of a collection of *events*, which are vectors of attribute-value pairs (alternatively called records or datapoints). The height, h, of the hierarchy to be created is specified by the user. When h=1, the dataset is partitioned into groups accompanied by logically disjoint generalized descriptions of the groups. When h=2, the original dataset is partitioned into clusters, and then each cluster is partitioned again into the next-level clusters and again accompanied by their descriptions. This process continues at each level of the hierarchy in the case of larger values of h.

Descriptions of clusters that stem from the same parent in the hierarchy are logically pairwise disjoint. Because these descriptions are generalizations of events, when a new, previously unseen event needs to be classified, it is matched against all cluster descriptions at the lowest level, and the description it matches determines the conceptual lineage to which the event is classified.

2.3 The CLUSTER3 method

The most recent version of conceptual clustering has been implemented in the CLUSTER3 program. The program conducts a divisive process that iteratively applies the top-level algorithm presented in Figure 2 to consecutively smaller subsets of data at each level of hierarchy being developed. The number of clusters to be created, k, at each level of the hierarchy is a user-provided parameter, or is determined by the system (see explanation below).

Let us briefly explain the algorithm. It starts by selecting k events, called *seeds*, that serve as initial representatives for each of the desired k clusters. In the first iteration, seeds are selected randomly, but in subsequent iterations they are selected by interchangeably applying the *principle of representation*



```
CLUSTER3 Top-Level Algorithm

1. Select k representative events (seeds) from the data
2. For each seed, create a star of the seed, defined as a set of maximally general
   descriptions of the seed that do not cover other seeds
3. Cross product the obtained stars to create a set of potential clusterings
4. For each potential clustering
    a. Apply NID procedure to restructure the clusters so that their descriptions in the
       clustering are pairwise logically disjoint
    b. Determine the best clustering according to LEF
    c. Select new seeds from the clusters in the obtained clustering, and repeat steps
       2-4 until newly obtained best clustering stops improving
5. Return the best clustering found as the output.
```

Fig. 2. The CLUSTER3 top-level algorithm.

or the *principle of adversity.* The principle of representation calculates clusters' centers, defined by the most frequent values of attributes in the events within each cluster. Events closest to the centers are selected as new seeds. The principle of adversity chooses as new seeds the events that are furthest from the centers of cluster descriptions (Michalski (1979)) in order to see if the current clustering is stable.

For each seed, the algorithm creates a set, called a *star*, of generalized descriptions of the seed that do not cover other seeds, but cover the maximal number of other events. The next step assembles different clusterings (data clusters and their descriptions) by selecting different combination of descriptions from each of k stars. Descriptions of clusters in the obtained clusterings often logically overlap. The NID procedure ("Non-disjoint Into Disjoint") is then used to contract the descriptions to eliminate these overlaps. The events in the overlaps are assigned to single clusters determined as the best "hosts" for them on the basis of a multi-criterion measure of clustering quality, called LEF (see Sec. 2.4). The best clustering according to LEF is determined. New seeds are selected from the clusters, and the above steps are repeated until the generated clusterings stop improving. When this happens, the program outputs the best clustering found so far. To make the above process efficient, various heuristics are applied. If the assumed height of the hierarchy, h, is greater than 1, the above procedure is applied *multiple* times, each time to the clusters obtained at the previous step.

If the value of k (the desired number of clusters at each level) is not provided by the user, for each node of the hierarchy the program executes the clustering algorithm for k=2, 3, ..., MAX, and then selects the globally best clustering according to the *global LEF* (Sec. 2.4). A justification of this procedure is that people typically like to split data into a relatively small number of clusters (MAX $\leq$ 7) at each level of hierarchy. Since MAX is small, CLUSTER3 simply repeats the procedure for values of k within the indicated range, and determines the best clustering among all clusterings that have been generated, according to a measure of clustering quality that takes into consideration both the simplicity of cluster descriptions and their number.

2.4 Clustering quality measure

As described above, at each step of the process, clusterings (sets of clusters and their descriptions) are ranked according to a multi-criterion measure that takes into consideration the properties of the cluster descriptions and the relation of descriptions to events being clustered. This feature represents an important difference between conceptual and conventional clustering, as the latter evaluates clustering quality solely on the basis of intra-cluster and inter-cluster similarities. Several criteria are used, such as:

- ***Sparseness***: the reciprocal of the density of events within the area defined by the union of cluster descriptions. The program tries to minimize the sparseness by seeking clusters with the highest density, defined by the ratio of the number of observed events to the number of all events covered by the cluster.
- ***Simplicity***: the average simplicity of cluster descriptions in the clustering, measured by the reciprocal of the number of selectors in the descriptions and the complexity of selectors.
- ***Balance***: the deviation of the distribution of events in clusters from an equal distribution, measured by the sum of squares of differences between the number of events covered by each cluster and the average number of events in the clusters.
- ***Disjointedness***: the degree of difference between cluster descriptions, measured by the sum of the number of attributes across every pair of cluster descriptions sharing no values in the selectors.

A user combines some or all of these criteria into a single multi-criterion called *Lexicographic Evaluation Functional* (LEF). In a LEF, individual criteria are ordered based on their estimated relative importance, and each criterion i is accompanied by a tolerance τ_i in percentage. The tolerance on criterion i means that every clustering scoring more than τ_i percentage worse than the best scoring clustering on this criterion is rejected. Therefore, two clusterings are considered equivalent if for all criteria defined in the LEF they evaluate to within τ% of the best result evaluation for each respective elementary criterion. Equivalent clustering results are further evaluated based on the sum of the evaluation percentages for each measure, retaining the clustering with the highest overall percentage. More details on LEF used for clustering are in (Michalski and Stepp (1983)).

Figure 3 presents an abstract example of evaluation of three candidate clusters on a LEF consisting of three criteria, listed in individual rows, in the order defined by parameter "Order". The higher score indicates greater quality. After evaluating clusterings on criterion α, Clustering 3 is eliminated

Criterion	*Tolerance*(τ)	*Order*	*Clustering*1	*Clustering*2	*Clustering*3
			Evaluation	*Evaluation*	*Evaluation*
α	$\tau_\alpha = 10\%$	1	10.0	9.5	8.5
β	$\tau_\beta = 10\%$	2	18.5	20.0	–
γ	$\tau_\gamma = 10\%$	3	5.6	4.8	–

Fig. 3. An example of applying LEF to three clusterings.

because it scored below 10% worse than the best score on this criterion (obtained by Clustering 1). The evaluation of the remaining clusterings on the second criterion (β) did not eliminate any clustering, because their score was within 10% from each other. After evaluating them on the 3^{rd} criterion, γ, Clustering 1 was found to be the best.

When clustering with different number of clusters are evaluatated, a global LEF is used that involves an additional criterion, ”global simplicity” that prefers clusterings with fewer clusters to those with more clusters.

2.5 View-relevant attribute subsetting

One class of enhancements in guiding the clustering process involves providing additional information about attributes beyond defining their domain and type. To affect the amount of influence that each attribute (e.g. Zhao et al. (2004)) or a type of attribute, i.e., numerical vs. categorical (Huang (1998)) has on the evaluation of cluterings, an attribute ”weight” is assigned to attributes or attribute types. A more advanced type of enhancement is embodied in symbolic data analysis, described by Diday and Esposito (2003). The method employs symbolic variables whose values may include in addition to conventional values, also more complex values such as histograms, intervals, or membership functions.

This section briefly describes another type of enhancement, called *view-relevant attribute subsetting* (VAS), introduced in Seeman and Michalski (2006). The VAS method selects subsets of attributes that are most relevant to the *viewpoint* from which clustering is performed. It leverages the knowledge of the experimenter who has deep understanding of the nature of the attributes and their relationship to the intended goal of the clustering process. Based on this knowledge, a *viewpoint meta-attribute* M_v is defined for a viewpoint v as a higher level attribute that categorizes original attributes according to their relevance to the viewpoint. Each viewpoint meta-attribute bisects the attribute space into two subsets - the set of attributes M_v associated with the viewpoint and the complement to this set M_v^C. Viewpoint meta-attributes are combined using set operators to determine which attributes are to be included in the projection space for clustering. For example, consider the attributes describing automobiles and viewpoint meta-attributes presented in Figure 4.

Attributes	Performance	Aesthetic	Convenience
#Cylinders	+	−	−
#Gears	+	−	+
HasSpoiler	+	+	−
Color	−	+	−
HasCruiseControl	−	−	+

Fig. 4. Viewpoint categorization of attributes describing automobiles.

A collection of automobiles can be classified from different viewpoints, such as performance, aesthetic criteria or convenience of driving. A ”+” or ”-” in a cell indicates that an attribute is viewed as relevant or irrelevant, respectively, for the given viewpoint. From Figure 4, the following definitions are derived:

$$M_{performance} = \{\#Cylinders, \#Gears, HasSpoiler\}$$

$$M^C_{performance} = \{Color, HasCruiseControl\}$$

$$M_{performance} \cup M^C_{Aesthetic} = \{\#Cylinders, \#Gears\}$$

Given a viewpoint, CLUSTER3 selects from the original set of attributes a subset that is relevant for this viewpoint and uses only relevant attributes for clustering. This operation allows the program to cluster data from the given viewpoint, and also simplifies the process of clustering.

3 Initial experiments with CLUSTER3 on FARS Data

3.1 Problem description

The Fatality Analysis Reporting System (FARS) is a national data gathering initiative sponsored and maintained by the U.S. Department of Transportation National Highway Traffic Safety Administration and is described by Tessmer (2002) and the FARS Coding and Validation Manual (2004). FARS reports records of accidents with fatalities annually, and makes them publicly available under the URL: www-fars.nhtsa.dot.gov. Every accident record is described by groups of attributes referring to various aspects of the accident, called levels:

1) Accident level, which specifies general accident information (39 attributes)
2) Vehicle level, which specifies information about each vehicle involved in the accident (33 attributes)
3) Driver level, which specifies information referring to each driver's qualifications, history and relevant physical characteristics (20 attributes)
4) Person level, which specifies relevant characteristics about each person (including drivers) and their involvement in the accident (28 attributes).

Attributes	**Domain Type**	**Values**
AccidentMonth	*Ordinal*	*Jan* − *Dec*
RoadProfile	*Nominal*	*Level*; *Grade*; *Hillcrest*; *Sag*
RoadSurface	*Nominal*	*Concrete*; *Blacktop*; *Brick*; *Gravel*; *Dirt*; *Other*
RoadCondition	*Nominal*	*Dry*; *Wet*; *Snow*; *Ice*; *SandorDirtorOil*; *Other*
LightCondition	*Nominal*	*Daylight*; *Dark*; *DarkbutLighted*; *Dawn*; *Dusk*
DrunkDrivingInvolved	*Nominal*	*Yes*; *No*

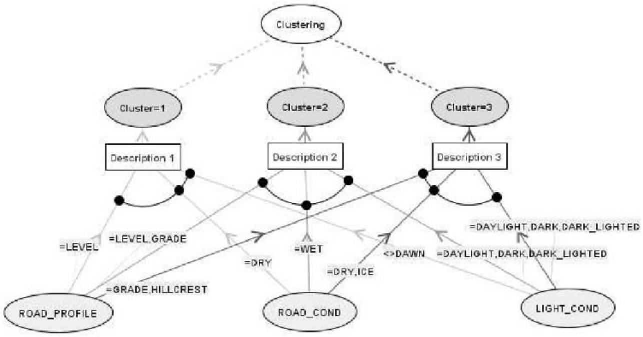
Fig. 5. FARS attributes used in disjoint and balanced clustering.

In the initial experiments described here, we used data from the 2004 reporting period and only relevant to the accident level. At the level, each record consists of values of 39 multi-type attributes. We pre-processed the data by only including records where the accident occurred in the state of New York during the month of April, reducing the dataset to 93 events.

The experiments described below explored some of the many possible clustering viewpoints (goals) related to these data. Goals are determined by changing the available parameters (number of clusters, number of hierarchy levels, etc.), modifying the LEF criterion, and using the VAS method. Specific goal-related information is provided in the next sections.

3.2 Experiments seeking disjoint and balanced clusterings

The first set of experiments sought a clustering with cluster descriptions that maximize disjointedness of cluster descriptions (by minimizing the number of shared values between attributes among cluster descriptions) while trying to maintain equal distribution of the data amongst the clusters. We selected the six attributes presented in Figure 5, which were judged as being independent. For example, there is most likely a high dependence between atmospheric conditions (rain, snow, etc) and road conditions (wet, dry, ice, etc.), therefore only the road condition attribute was included, and the atmospheric condition attribute was omitted. The multi-criterion clustering quality measure, LEF, consisted of the disjointedness criterion with tolerance $\tau = 10\%$, and then the balance criterion with tolerance $\tau = 20\%$. The initial experiments clustered data into varying numbers of clusters, (k), ranging from 2 to 6, and leaving all other parameters at their default levels. The clustering obtained for k=2 was influenced significantly by the values of the ”Road profile” attribute. The clustering obtained for k=3 proved to be the most interesting. The resulting cluster descriptions show a dependence not on the three possible values of the ”Road profile” attribute, as was expected based on the results for k=2, but rather on a combination of values from three attributes. This result demonstrates the ability of CLUSTER3 to discover complex relationships and present the results as easily interpretable descriptions.



Cluster 1:
[Road_profile=level] & [Road_cond=dry] & [Light_cond ≠ dawn]
Cluster 2:
[Road_profile=level v grade] & [Road_cond=wet] & [Light_cond=daylight v dark v dark_lighted]
Cluster 3:
[Road_profile=grade v hillcrest] & [Road_cond=dry v ice] & [Light_cond=daylight v dark v dark_lighted]

Fig. 6. A CAG visualizing clustering obtained in the initial experiments.

The results obtained and their visualization in the form of a *Concept Association Graph* (CAG) are displayed in Figure 6. In a CAG used for visualizing results from CLUSTER3, the top node represents a clustering, the nodes below represent data clusters, rectangular nodes below represent cluster descriptions. Links to nodes representing selectors constituting a cluster description are connected by arches to indicate that selectors are combined by conjunction. Remaining nodes represent attributes.

Links between attribute nodes and cluster descriptions are marked with the relation and the subset of attribute values from the corresponding selector. The graph shows a multi-attribute dependence for determining the clusters that maximize the combined disjointedness and balance criteria. The hierarchy obtained for k=4 through k=6 are further divisions of the result obtained for k=2. Figure 6 is a simplified form of a CAG, as it uses links of the same thickness, unlike CAGs representing results of supervised learning in which links' thickness represents some property associated with the condition, e.g., its consistency (Kaufman et al. (2006)).

3.3 Clustering data from the road state viewpoint

Results of previous experiments suggest a high dependence of cluster descriptions on attributes relevant to the state of the road viewpoint. Therefore, the next experiments involved building a hierarchy of clusters from the *road state* viewpoint. A meta-attribute $M_{roadstate}$ was defined whose values include original attributes relevant to this viewpoint, such as Road profile, Road condition, Road surface and Road alignment. The "Road alignment", with values {straight, curve} was not in the original set of attributes listed in Figure 5, but was added as also relevant to the road state viewpoint. The

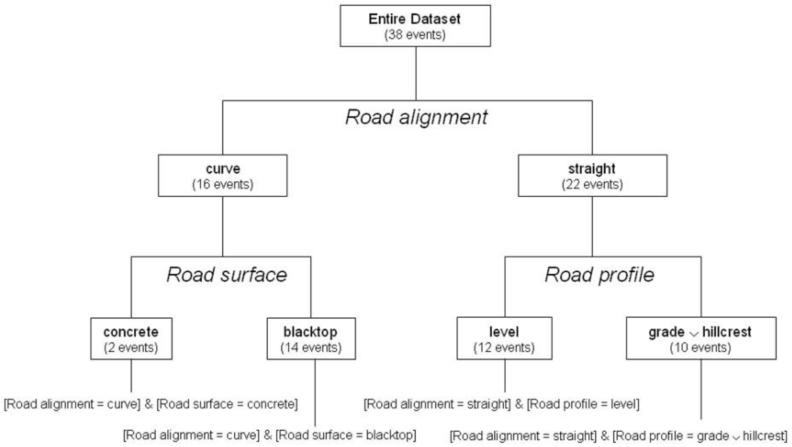

Entire Dataset
(38 events)
Road alignment
curve
(16 events)
straight
(22 events)
Road surface
Road profile
concrete
(2 events)
blacktop
(14 events)
level
(12 events)
grade ∨ hillcrest
(10 events)
[Road alignment = curve] & [Road surface = concrete]
[Road alignment = curve] & [Road surface = blacktop]
[Road alignment = straight] & [Road profile = level]
[Road alignment = straight] & [Road profile = grade ∨ hillcrest]

Fig. 7. Hierarchy of Clusters and Cluster Descriptions Generated by CLUSTER3 From the Roadway Viewpoint.

hierarchy depth was set to 2. The results are shown in the dendrogram-like diagram in Figure 7.

The hierarchy shows a clear division based first on the newly added attribute, ”Road alignment,” and then secondarily on either ”Road profile” or ”Road surface.” The inclusion of the ”Road profile” attribute as a key attribute in the second level strengthens the result of the first set of experiments, which relied heavily on this attribute. The inclusion of the ”Road surface” as a key attribute is surprising given that it did not play a significant role in the previous experiments. It is also worth noting the difference in relevance of attributes at the second level in the hierarchy. The attribute with the greatest contribution to minimizing sparseness with respect to fatal accidents on curved roads is ”Road surface”; whereas the attribute with the greatest contribution to minimizing sparseness with respect to fatal accidents on straight roads is ”Road profile”. The hierarchy presented in Figure 7 appears quite elegant and informative classification of accidents from the road state viewpoint.

4 Conclusion

This chapter reviewed the conceptual clustering methodology and presented some of its recent advances implemented in CLUSTER3. The program was illustrated by applying it to the problem of clustering of fatalities data from different viewpoints. The clustering generated consisted of clusters and cluster descriptions that are easy to interpret and understand. The obtained results appear highly satisfactory, and indicate that conceptual clustering can be a useful tool in practical applications, in particular in situations in which not only clusters but also their generalized descriptions are required.

5 Acknowledgements

The idea and the first algorithm for conceptual clustering were originally developed by Michalski during his visit to Professor Diday's Laboratory at INRIA in 1979. The process of incrementally improving conceptual clusters was based on the idea of dynamic clustering employed in Diday's research on clustering.

The authors thank Jarek Pietrzykowski and Janusz Wojtusiak for comments on earlier versions of this paper. Previous implementations of conceptual clustering, CLUSTER, CLUSTER2, and CLUSTER2C++, described by Michalski and Stepp (1983), Stepp (1984), and Fischthal (1997), respectively, have contributed significantly to the success of the version described in this chapter.

This research was conducted in the Machine Learning and Inference Laboratory of George Mason University, whose research activities are supported in part by the National Science Foundation Grants No. IIS 9906858 and IIS 0097476. The findings and opinions expressed here are those of the authors, and do not necessarily reflect those of the above sponsoring organization.

References

BERKHIN, P. (2002): Survey of clustering data mining techniques, Accrue Software, http://citeseer.ist.psu.edu/berkhin02survey.html.

CHEESEMAN, P. and STUTZ, J. (1996): Bayesian classification (AutoClass): theory and results In: *Advances in Knowledge Discovery and Data Mining.* Chapter 6, AAAI Press / MIT Press, 153–180.

CIAMPI, A., DIDAY, E., LEBBE, J., PERINEL, E. and VIGNES, R. (2000): Growing a tree classifier with imprecise data. *Pattern Recognition Letters 21 (9), 787–803.*

CONLEY, J., GAHEGAN, M. and MACGILL, J. (2005): A genetic approach to detecting clusters in point data sets. *Geograpical Analysis 37 (3), 286–315.*

DIDAY, E. and ESPOSITO, F. (2003): An Introduction to symbolic data analysis and the SODAS software. *Intelligent Data Analysis 7 (6), 583–601.*

FARS (2004): Coding and validation manual (2004): National Center for Statistics and Analysis, National Highway Traffic Safety Administration, Department of Transportation, Washington, D.C.

FISCHTHAL, S. (1997): A description and user's guide for CLUSTER2C++ a program for conjunctive conceptual clustering. Technical Report MLI 97-10, *Reports of the Machine Learning and Inference Laboratory*, George Mason University, Fairfax, VA.

GOWDA, K.C. and DIDAY, E. (1992): Symbolic clustering using a new similarity measure. *IEEE Transactions on Systems, Man, and Cybernetics 22 (2), 368–378.*

HUANG, Z. (1998): Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery 2 (3), 283–304.*

KAUFMAN, K., MICHALSKI, R.S., PIETRZYKOWSKI, J. and WOJTUSIAK, J. (2006): An integrated multi-task inductive database and decision support system VINLEN: An initial implementation and first results. In: *Proceedings of The 5th International Workshop on Knowledge Discovery in Inductive Databases, KDID'06, in conjunction with ECML/PKDD.* Berlin, Germany.

MICHALSKI, R.S. (1979): Conceptual clustering: a theoretical foundation and a method for partitioning data into conjunctive concepts. *Seminaries IRIA, Classification Automatique et Perception par Ordinateur, INRIA, France, 253–295.*

MICHALSKI, R.S. (2004): Attributional calculus: a logic and representation language for natural induction. Technical Report MLI 04-2, *Reports of the Machine Learning and Inference Laboratory*, George Mason University, Fairfax, VA.

MICHALSKI, R.S. and STEPP, R.E. (1983): Learning from observation: conceptual clustering. In: R.S. Michalski, J.G. Carbonell and T.M. Mitchell (Eds.): *Machine Learning: An Artificial Intelligence Approach.* Chapter 11, Morgan Kaufman, 331–364.

MICHALSKI, R.S. and STEPP, R. (1986): Clustering. In: S. Shapiro, D. Eckroth and G.A. Vallasi (Eds.): *Encyclopedia of Artificial Intelligence.* John Wiley & Sons, New York, 185–194.

SEEMAN, W.D. and MICHALSKI, R.S. (2006): The CLUSTER3 system for goal-oriented conceptual clustering: method and preliminary results. In: A. Zanasi, C.A. Brebbia and N.F.F. Ebecken (Eds.): *Data Mining VII: Data, Text and Web Mining, and their Business Applications (Proceedings of the Seventh International Conference on Data Mining).* WIT Press, 81–90.

STEPP, R.E. (1984): Conjunctive conceptual clustering: a methodology and experimentation, *PhD Thesis, UIUCDCS-R-84-1189*, Department of Computer Science, University of Illinois, Urbana.

TESSMER, J.M. (2002): FARS Analytic Reference Guide 1975 to 2002, National Highway Traffic Safety Administration, Department of Transportation, Washington, D.C.

ZHAO, H., ZHANG, G. and YANG, W. (2004): Categorical data clustering with evolutionary strategy weighting attributes. In: *Proceedings of the 5th World Congress on Intelligent Control and Automation.* Hangzhou, P.R. China.

Symbolic Dynamics in Text: Application to Automated Construction of Concept Hierarchies

Fionn Murtagh

Department of Computer Science
Royal Holloway, University of London
Egham, Surrey TW20 0EX, England
fmurtagh@acm.org

Abstract. Following a symbolic encoding of selected terms used in text, we determine symmetries that are furnished by local hierarchical structure. We develop this study so that hierarchical fragments can be used in a concept hierarchy, or ontology. By "letting the data speak" in this way, we avoid the arbitrariness of approximately fitting a model to the data.

1 Introduction

1.1 Symmetry Group and Alternating Permutation Ordinal Encodings in Symbolic Dynamics

In symbolic dynamics, we seek to extract symmetries in the data based on topology alone, before considering metric properties. For example, instead of listing a sequence of iterates, $\{x_i\}$, we may symbolically encode the sequence in terms of up or down, or north, south, east and west moves. This provides a sequence of symbols, and their patterns in a phase space, where the interest of the data analyst lies in a partition of the phase space. Patterns or templates are sought in this topology. Sequence analysis is tantamount to a sort of topological time series analysis.

Thus, in symbolic dynamics, the data values in a stream or sequence are replaced by symbols to facilitate pattern-finding, in the first instance, through topology of the symbol sequence. This can be very helpful for analysis of a range of dynamical systems, including chaotic, stochastic, and deterministic-regular time series. Through measure-theoretic or Kolmogorov-Sinai entropy of the dynamical system, it can be shown that the maximum entropy conditional on past values is consistent with the requirement that the symbol sequence retains as much of the original data information as possible. Alternative approaches to quantifying complexity of the data, expressing the dynamical system, is through Lyapanov exponents and fractal dimensions, and there are close relationships between all of these approaches (Latora and Baranger (1999)).

Later in this work, we will use a “change versus no change” encoding, using a multivariate time series based on the sequence of terms used in a document.

From the viewpoint of practical and real-world data analysis, however, many problems and open issues remain. Firstly (Bandt and Pompe (2002)), noise in the data stream means that reproducibility of results can break down. Secondly, the symbol sequence, and derived partitions that are the basis for the study of the symbolic dynamic topology, are not easy to determine. Hence Bandt and Pompe (2002) enunciate a pragmatic principle, whereby the symbol sequence should come as naturally as possible from the data, with as little as possible by way of further model assumptions. Their approach is to define the symbol sequence through (i) comparison of neighboring data values, and (ii) up-down or down-up movements in the data stream.

Taking into account all up-down and down-up movements in a signal allows a permutation representation.

Examples of such symbol sequences from Bandt and Pompe (2002) follow. They consider the data stream $(x_1, x_2, \ldots, x_7) = (4, 7, 9, 10, 6, 11, 3)$. Take the order as 3, i.e. consider the up-down and down-up properties of successive triplets. $(4, 7, 9) \longrightarrow 012; (7, 9, 10) \longrightarrow 012; (9, 10, 6) \longrightarrow 201; (6, 11, 3) \longrightarrow 201; (10, 6, 11) \longrightarrow 102$. (In the last, for instance, we have $x_{t+1} < x_t < x_{t+2}$, yielding the symbolic sequence 102.) In addition to the order, here 3, we may also consider the delay, here 1. In general, for delay τ, the neighborhood consists of data values indexed by $t, t - \tau, t - 2\tau, t - 3\tau, \ldots, t - d\tau$ where d is the order. Thus, in the example used here, we have the symbolic representation 012012201201102. The symbol sequence (or “itinerary”) defines a partition – a separation of phase space into disjoint regions (here, with three equivalence classes, 012, 201, and 102), which facilitates finding an “organizing template” or set of topological relationships (Weckesser (1997)). The problem is described in Keller and Lauffer (2003) as one of studying the qualitative behavior of the dynamical system, through use of a “very coarse-grained” description, that divides the state space (or phase space) into a small number of regions, and codes each by a different symbol.

Different encodings are feasible and Keller and Sinn (2005a, 2005b) use the following. Again consider the data stream $(x_1, x_2, \ldots, x_7) = (4, 7, 9, 10, 6, 11, 3)$. Now given a delay, $\tau = 1$, we can represent the above by $(x_{6\tau}, x_{5\tau}, x_{4\tau}, x_{3\tau}, x_{2\tau}, x_\tau, x_0)$. Now look at rank order and note that: $x_\tau > x_{3\tau} > x_{4\tau} > x_{5\tau} > x_{2\tau} > x_{6\tau} > x_0$. We read off the final permutation representation as (1345260). There are many ways of defining such a permutation, none of them best, as Keller and Sinn (2005a) acknowledge. We see too that our m-valued input stream is a point in $\mathbb{R}^m$, and our output is a permutation $\pi \in S_m$, i.e. a member of the permutation group.

Keller and Sinn (2005a) explore invariance properties of the permutations expressing the ordinal, symbolic coding. Resolution scale is introduced through the delay, τ. (An alternative approach to incorporating resolution scale is used in Costa et al. (2005), where consecutive, sliding-window based, binned or averaged versions of the time series are used. This is not entirely satisfactory: it is not robust and is very dependent on data properties such as dynamic range.) Application is to EEG (univariate) signals (with some discussion of magnetic resonance imaging data) (Keller et al. (2005)). Statistical properties of the ordinal transformed data are studied in Bandt and Pompe (2002), in particular through the S_3 symmetry group. We have noted the symbolic dynamics motivation for this work; in Bandt (2005) and other work, motivation is provided in terms of rank order time series analysis, in turn motivated by the need for robustness in time series data analysis.

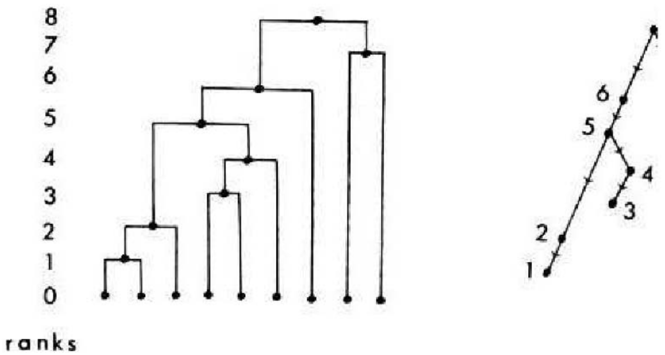


Fig. 1. Left: dendrogram with lower ranked subtree always to the left. Right: oriented binary tree associated with the non-terminal nodes.

Given the permutation representation used, let us note in passing that there is an isomorphism between a class of hierarchic structures, termed unlabeled, ranked, binary, rooted trees, and the class of permutations used in symbolic dynamics. Each non-terminal node in the tree shown in Figure 1 has one or two child nodes. This is a dendrogram, representing a set of $n-1$ agglomerations based on n initial data vectors. A packed representation (Sibson (1980)) or permutation representation of a dendrogram is derived as follows. Put lower ranked subtree always to the left; and read off oriented binary tree on non-terminal nodes (see Figure 1). Then for any terminal node indexed by i, with the exception of the rightmost which will always be n, define $p(i)$ as the rank at which the terminal node is first united with some terminal node to its right. For the dendrogram shown, the packed representation is: (125346879). This is also an inorder traversal of the oriented binary tree. The packed representation is a uniquely defined permutation of $1 \ldots n$. Dendrograms (on n terminals) of the sort shown in Figure 1, referred to as non-labeled, ranked

(NL-R) in Murtagh (1984), are isomorphic to either down-up permutations, or up-down permutations (both on $n-1$ elements).

1.2 Motivation for an Alternative Ordinal Symbolic Dynamics Encoding

In some respects we follow the work of Keller, Bandt, and their colleagues in using an ordinal coding to provide for an encoding of the data sequence. However in the following areas we need to adopt a different approach.

- We need to handle multivariate time series.
- We need to bypass the two alternative analyses that the ordinal symbolic encoding necessarily leads to, viz. either up-down or down-up.
- Biological verisimilitude is not strong with the ordinal encoding as discussed so far.

We look at each of these in turn.

To handle multivariate time series, Keller and Lauffer (2003), and Keller and Wittfeld (2004) find the best composite time series, using projections on the first factor furnished by correspondence analysis. Correspondence analysis uses a weighted Euclidean distance between profiles (or, using the input data, the χ^2 distance) and for time-varying signals such as EEG signals, it is a superior choice compared to, say, principal components analysis.

In Bandt and Groth (2005), the need for multivariate analysis is established. Among tentative steps towards this are window-based averages of distances.

It is immediate in any inequality, $x_t > x_{t-1}$, that reversing the inequality (e.g. through considering an axial symmetry in the time axis) can lead to a new and different outcome. When we have multivariate data streams, enforcing symmetry is very restrictive. We bypass this difficulty very simply by instead using a change/no change symbolic representation. Financial verisimilitude is lost in doing this (if up = gain, down = loss); but biological verisimilitude, and that of other areas, is aided greatly.

Based on their EEG analysis, Keller and Sinn (2005a) ask: “Does there exist a basic (individual) repertoire of ‘ordinal’ states of brain activity?”. As opposed to this, we target the hierarchy or branching fragment as the pattern that is sought, which suits the dendritic structures of the brain. While rank order alone is a useful property of data, we seek to embed our data (globally or locally) in an ultrametric topology, which also offers scope for p-adic algebraic processing. We move from real data, we take account of ordinal properties, and we end up with a topological and/or algebraic framework. This implies a data analysis perspective which is highly integrated and comprehensive. Furthermore, as an analysis pipeline, it is potentially powerful in bridging observed data with theoretically-supported interpretation.

2 The topological view: ultrametric embedding

1. We seek uncontestable local hierarchical structure in the data. The traditional alternative is to impose hierarchical structure on the data (e.g. through hierarchical clustering, or otherwise inducing a classification tree).
2. We seek to avoid having any notion of hierarchical direction. In practice this would imply that hierarchical "up" (e.g. agglomerative or bottom-up) and hierarchical "down" (e.g. divisive or top-down) should each be considered independently.
3. We may wish to accommodate (i.e., include in our analysis) outliers and random exceptional values in the data. More particularly: we want to handle power law distributions, characterized by independent but not identically distributed values. An example is Zipf's law for text.
4. Therefore, for text we will use the property of linearity of text: words are linearly ordered from start to finish. (Note that a hypertext could be considered as a counter-example.)

The approach to finding local hierarchical structure is described for time series data in Murtagh (2005). We use the same approach here. The algorithm is as follows. The data used is the sequence of frequencies of occurrence of the terms of interest – nouns, noun-substantives – in their text-based order. These terms are found using TreeTagger (Schmid (1994)).

In seeking to use free text, we will also take into consideration the strongest "given" in regard to any classical text: its linearity (or total) order. A text is read from start to finish, and consequently is linearly ordered.

A text endowed with this linear order is analogous to a time series. (This opens up the possibility to generalize the work described here to (i) speech signals, or (ii) music. We will pursue these generalizations in the future.)

3 Quantifying hierarchical structure in a linear ordered set: application

We proceed now to particular engineering aspects of this work. We require a frequency of occurrence matrix which crosses the terms of interest with parts of a free text document. For the latter we could well take documentary segments like paragraphs.

O'Neill (2006) is a 660-word discussion of ubiquitous computing from the perspective of human computing interaction. With this short document we used individual lines (as proxies for the sequence of sentences) as the component parts of the document. There were 65 lines.

Based on a list of nouns and substantives furnished by the part-of-speech tagger (Schmid (1994)) we focused on the following 30 terms:

`support` = { "agents", "algorithms", "aspects", "attempts", "behaviours", "concepts", "criteria", "disciplines", "engineers", "factors", "goals", "interactions", "kinds", "meanings", "methods", "models", "notions", "others", "parts", "people", "perceptions", "perspectives", "principles", "systems", "techniques", "terms", "theories", "tools", "trusts", "users" }.

This set of 30 terms was used to characterize through presence/absence the 65 successive lines of text, leading to correspondence analysis of the 65×30 presence/absence matrix. This yielded then the definition of the 30 terms in a factor space. In principle the rank of this space (taking account of the trivial first factor in correspondence analysis, relating to the centering of the cloud of points) is min($65 - 1, 30 - 1$). However through all zero-valued rows and/or columns, the actual rank was 25. Therefore the full rank projection of the terms into the factor space gave rise to 25-dimensional vectors for each term, and these vectors are endowed with the Euclidean metric.

Define this set of 30 terms as the support of the document. Based on their occurrences in the document, we generated the following *reduced* version of the document, defined on this support, which consists of the following ordered set of 52 terms:

`Reduced document` = "goals" "techniques" "goals" "disciplines" "meanings" "terms" "others" "systems" "attempts" "parts" "trusts" "trusts" "people" "concepts" "agents" "notions" "systems" "people" "kinds" "behaviours" "people" "factors" "behaviours" "perspectives" "goals" "perspectives" "principles" "aspects" "engineers" "tools" "goals" "perspectives" "methods" "techniques" "criteria" "criteria" "perspectives" "methods" "techniques" "principles" "concepts" "models" "theories" "goals" "tools" "techniques" "systems" "interactions" "interactions" "users" "perceptions" "algorithms"

This reduced document is now analyzed using the algorithm described earlier. Each term in the sequence of 52 terms is represented by its 25-dimensional factor space vector.

For successive triples, if the triple is to be compatible with the ultrametric inequality, we require the recoded distances to be one of the following patterns: 1,1,1 or 2,2,2 for an equilateral triangle; and 1,2,2 in any order for an isosceles triangle with small base.

The only other pattern is 1,1,2 (in any order) which is not compatible with the ultrametric inequality. (It is seen to represent the case of an isosceles triangle with large base.)

Out of 43 unique triplets, with self-distances removed, we found 31 to respect the ultrametric inequality, i.e. 72%. The ultrametricity of this document, based on the support used, was thus 0.72.

For a concept hierarchy we need an overall fit to the data. Using the Euclidean space perspective on the data, furnished by correspondence analysis, we can easily define a terms × terms distance matrix; and then hierarchically cluster that. Consistent with our analysis we recode all these distances, using the mapping onto $\{1, 2\}$ for unique pairs of terms.

Note that this is tantamount to having a window encompassing all of the reduced document. It is also interesting to check the ultrametricity coefficient here. This means therefore the ultrametricity coefficient in the window length

n case, versus the ultrametricity coefficient in the window length 3 case. The latter was seen to be (from exhaustive calculation) above, 0.72. For the window length n case, we sampled 2000 triplets, and found the ultrametricity coefficient to be 0.56. Since the linear order is of greater ultrametric (hence, hierarchical) structure, an evident question arises as to whether it should be used as the basis for a retrieved overall or global hierarchy. We do not do this, however, because the greater hierarchical structure comes as the cost of being overly fragmentary. Instead, we adopt the approach now to be described.

Approximating a global ultrametric from below, achieved by the single linkage agglomerative hierarchical clustering method (this best fit from below is optimal), and an approximation from above, achieved by the complete linkage agglomerative hierarchical clustering method (this best fit from above is non-unique and hence is one of a number of best fits from above), will be identical if the data is fully ultrametric-embeddable. If we had an ultrametricity coefficient equal to 1 – we found it to be 0.72 for this data – then it would not matter what agglomerative hierarchical clustering algorithm (among the usual Lance-Williams methods) that we select.

In fact, we found, with an ultrametricity coefficient equal to 0.72, that the single and complete linkage methods gave an identical result. This result is shown in Figure 2.

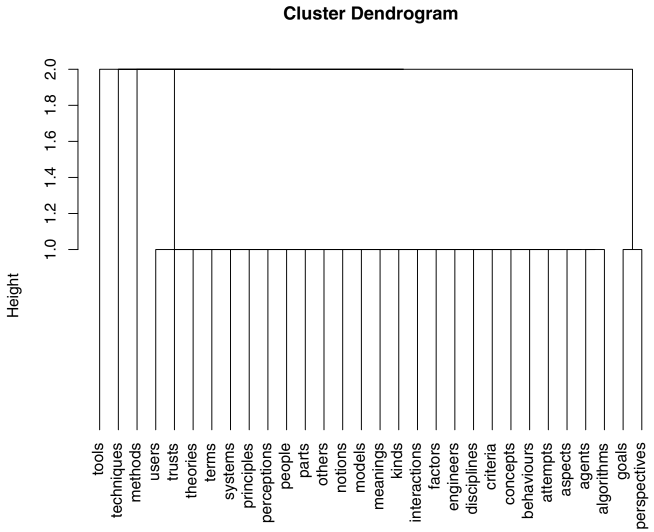


Fig. 2. Single (or identially, complete) linkage hierarchy of 30 terms, comprising the support of the document, based on (i) "no change/change" metric recoded (ii) 25-dimensional Euclidean representation.

References

BANDT, C. and POMPE, B. (2002): Permutation entropy: a natural complexity measure for Time Series. *Physical Review Letters, 88, 174102(4).*

BANDT, C. and SHIHA, F. (2005): *Order Patterns in Time Series.* Preprint 3/2005, Institute of Mathematics, Greifswald, www.math-inf.uni-greifswald.de/∼bandt/pub.html

BANDT, C. (2005): Ordinal time series analysis. *Ecological Modelling, 182, 229–238.*

BANDT, C. and GROTH, A. (2005): *Ordinal Time Series Analysis.* Poster Freiburg. www.math-inf.uni-greifswald.de/∼groth

COSTA, M., GOLDBERGER, A.L. and PENG, C.-K. (2005): Multiscale entropy analysis of biological signals. *Physical Review E, 71, 021906(18).*

DE SOETE, G. (1986): A least squares algorithm for fitting an ultrametric tree to a dissimilarity matrix. *Pattern Recognition Letters, 2, 133–137.*

KELLER, K. and LAUFFER, H. (2003): Symbolic analysis of high-dimensional time series. *International Journal of Bifurcation and Chaos, 13, 2657–2668.*

KELLER, K. and WITTFELD, K. (2004): Distances of time series components by means of symbolic dynamics, *International Journal of Bifurcation and Chaos, 693–704.*

KELLER, K. and SINN, M. (2005): *Ordinal Symbolic Dynamics.* Technical Report A-05-14, www.math.mu-luebeck.de/publikationen/pub2005.shtml

KELLER, K. and SINN, M. (2005): Ordinal analysis of time series. *Physica A 356, 114–120.*

KELLER, K., LAUFFER, H. and SINN, M. (2005): Ordinal analysis of EEG time series. *Chaos and Complexity Letters, 2.*

LATORA, V. and BARANGER, M. (1999): Kolmogorov-Sinai Entropy Rate versus Physical Entropy. *Physical Review Letters, 82, 520(4).*

MURTAGH, F. (1984): Counting Dendrograms: a Survey. *Discrete Applied Mathematics, 7, 191–199.*

MURTAGH, F. (2005): Identifying the ultrametricity of time series. *European Physical Journal B, 43, 573–579.*

O'NEILL, E. (2006): Understanding ubiquitous computing: a view from HCI, in Discussion following R. Milner, Ubiquitous Computing: How Will We Understand It?", *Computer Journal, 49, 390–399.*

SCHMID, H. (1994): Probabilistic part-of-speech tagging using decision trees. IN: *Proc. Intl. Conf. New Methods in Language Processing.* TreeTagger, www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/ DecisionTreeTagger.html

SIBSON, R. (1980): SLINK: an optimally efficient algorithm for the single-link cluster method. *Computer Journal, 16, 30–34.*

WECKESSER, W. (1997): Symbolic Dynamics in Mathematics, Physics, and Engineering, based on a talk by N. Tuffilaro, http://www.ima.umn.edu/∼weck/nbt/nbt.ps

Part IV

Consensus Methods

Average Consensus and Infinite Norm Consensus : Two Methods for Ultrametric Trees

Guy Cucumel

École des sciences de la gestion, Université du Québec à Montréal
C.P. 8888, Succ. Centre-Ville, Montréal (Québec), H3C 3P8, Canada
cucumel.guy@uqam.ca

Abstract. Consensus methods are widely used to combine hierarchies defined on a common set of n object. Many methods have been proposed during the last decade to combine hierarchies. One of these, the average consensus method, allows one to obtain a consensus solution that is representative of the initial profile of trees by minimizing the sum of the squared distances between this profile and the consensus solution. This problem is known to be NP-complete and one has to rely on heuristics to obtain a consensus result in such cases. As a consequence, the uniqueness and optimality of the solution is not guaranteed. The L_∞-consensus that yields to a universal solution in a maximum of n^2 steps is an alternative to the average consensus procedure. The two methods will be presented and compared on a numerical example.

1 Introduction

Given a profile $\mathbf{P} = (H_1, \dots, H_l, \dots, H_k)$ of k hierarchical classifications (e.g., n-trees or ultrametric trees) defined on a common set of n objects $\mathbf{S}$, a consensus hierarchy H_c is a single hierarchy that is representive, in a "certain sense" (Barthélemy and McMorris (1986)) of the entire profile $\mathbf{P}$ (Leclerc (1998) and Leclerc and Cucumel (1987)). Since the first algorithm proposed by Adams (1972), the use of consensus hierarchies has increased and methods and algorithms to combine classifications have been developed during the last decades of which some apply to ultrametric trees (Margush and McMorris (1981), Neumann (1983), Stinebrickner (1984), Finden and Gordon (1985), Barthélemy and McMorris (1986), Cucumel (1990) and Lapointe and Cucumel (1997)). In this paper we present two approaches: the average consensus (or L_2-consensus) and the infinite norm consensus (or L_∞-consensus).

2 The average consensus method

The average consensus originally proposed by Cucumel (1990) returns a consensus solution that minimizes the sum of distances, in the sense of Hartigan's (1967) distance, between each of the indexed hierarchies of the profile $\mathbf{P}$ and the consensus hierarchy. As there exists a one to one correspondence between dendrograms and ultrametrics defined on $\mathbf{S} = (1, \ldots, i, \ldots, n)$ it is equivalent to deal with dendrograms or with their corresponding ultrametric matrices. Let H_1 and H_2 be two ultrametric trees. Let u_1 and u_2 be the two associated ultrametrics. Let $\mathit{\Delta}$ be a distance between ultrametric trees (Hartigan (1967)):

$$\mathit{\Delta}(H_1, H_2) = \sum_{i=1}^{n} \sum_{j=1}^{n} [u_1(i,j) - u_2(i,j)]^2 \tag{1}$$

Now, let H_c be the average consensus hierarchy. H_c is the hierarchy H among all hierarchies defined on $\mathbf{S}$ that minimizes:

$$\sum_{l=1}^{k} \mathit{\Delta}(H_l, H) \tag{2}$$

Let u_c be the ultrametric associated with H_c. Problem (2) is equivalent to finding the ultrametric u_c among all ultrametrics defined on $\mathbf{S}$ that minimizes:

$$\sum_{l=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{n} [u_1(i,j) - u_2(i,j)]^2 \tag{3}$$

This problem is equivalent to find the ultrametric the closest (in the sense of the L_2 norm) to the obtained dissimilarity by computing the mean term to term of the ultrametrics associated to the hierarchies of the initial profile $\mathbf{P}$ (Lapointe and Cucumel(1997)). It is a NP-complete problem which can be resolved only by using a “ branch and bound ” algorithm (Chandon and De Soete (1984)). This algorithm, which is a generalization of the average linkage algorithm, leads to a solution which is not necessarily unique. When the number of objects to be classified is big, it is necessary to find an approached solution. The ultrametric associated to the hierarchy of the average linkage, to which leads one of the branches of the algorithm developed by Chandon and De Soete is a possible approached solution.

3 The L_∞-consensus

A new approach has been developed by Chepoi and Fichet (2000) who have proposed an algorithm which has a complexity in n^4 and leads to a unique solution: the consensus in infinite norm (L_∞-consensus). This method proceeds in the construction of two sequences of ultrametrics $w_1 < w_2 < \ldots < w_q$ and $v_1 > v_2 > \ldots > v_q$ such that at the convergence of the algorithm $w_q{=}v_q$. We present it in the steps below[1].

[1] We thank Bernard Fichet to have communicated to us a detailed version of the algorithm.

Let $u_1, u_2, \ldots, u_k$ be the k ultrametrics associated to the k indexed hierarchies of the profile $\mathbf{P}$ and let $\parallel . \parallel$ be the infinite norm.

Step 1

- compute $w = inf(u_1, u_2, \ldots, u_k)$, minimum term to term
- compute $v = sup(u_1, u_2, \ldots, u_k)$, maximum term to term
- compute w^* the subdominant of w
- compute $e = \parallel v - w \parallel /2$
- compute w_1 by deducting e to each term of v
- compute v_1 by adding e to each term of w^*
- let m be equal to 1

Step 2

- compute $e_m = \parallel v_m - w_m \parallel /2$
- if $e_m = 0$, end of the algorithm, $w_m = v_m$ is the L_∞-consensus

Step 3

- compute t_m by deducting e_m to each term of v_m
- compute $w_{m+1} = sup(w_m, t_m)$
- compute s_m by adding e_m to each term of w_m
- compute v_{m+1} the subdominant of $inf(v_m, s_m)$
- let m be equal to $m + 1$ and go back to step 2

The algorithm converges to a solution in a maximum of n^2 steps. As the complexity of the calculation of the subdominants is in n^2 also, the algorithm has a complexity in n^4.

4 Example

Both methods are applied for the search of a consensus between the three indexed hierarchies H_1, H_2 and H_3 of Figure 1[2] defined on the set $\mathbf{S}= \{x_1, x_2, x_3, x_4, x_5, x_6\}$. We also show an approached solution obtained by the algorithm of the average linkage (Figure 2).

The consensus in infinite norm (Figure 2) has the same structure as the majority consensus (Margush and McMorris (1981)) and retains only two subsets $\{x_1, x_2\}$ and $\mathbf{S}$ and thus accepts few compromises. The level associated with $\mathbf{S}$ (2.5) is lower than those who are associated to it in the initial hierarchies. As a result certain objects as x_4 and x_6 or x_2 and x_5 are much closer in the consensus than in the hierarchies of the initial profile.

The average consensus and its approached solution by the average linkage algorithm have both the same structure (Figure 2). It is interesting to

[2] This example was used by Chepoi and Fichet for computing a consensus in infinite norm.

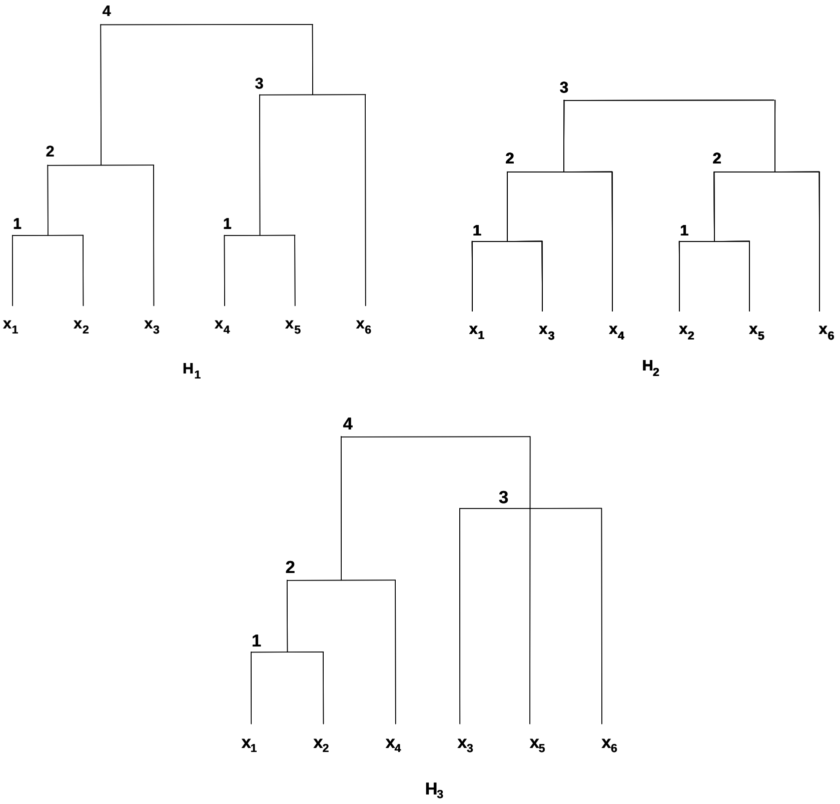
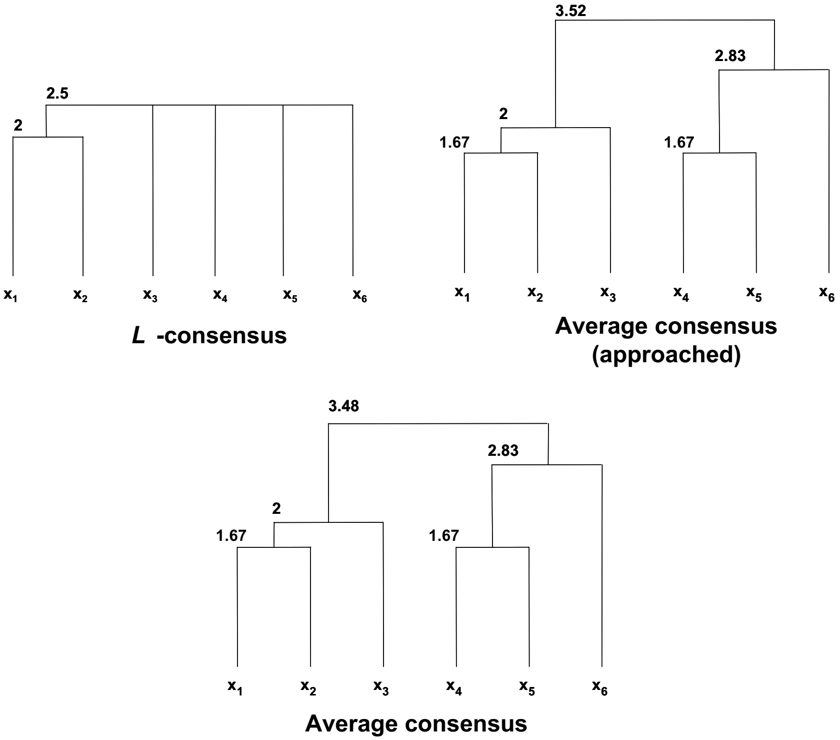


Fig. 1. Hierarchies H_1, H_2 and H_3.

note that in that case the approached solution is excellent because only the level associated to the subset **S** differs slightly from a consensus with the other one. The average consensus is inevitably binary by construction, what presents the inconvenience to force certain groupings. As for the consensus in infinite norm, the proximity of x_1 and x_2 in the hierarchies H_1 and H_3 is well represented. The relative proximities of x_1 and x_3 in the hierarchies H_1 and H_2 on one hand and of x_5 and x_6 in the three initial hierarchies on the other hand are also well represented in the consensus. The binary structure of the average consensus let represent x_2 and x_3 rather close one to the other what is questionable considering the relative positions of these two objects in the initial hierarchies (they are only close in H_1). It is the same for x_4 and x_5 who are only close in H_1.

Another way to compare the obtained consensus is to use the cophenetic correlation coefficient introduced by Sokal and Rohlf (1962) who measures



L -consensus

Average consensus (approached)

Average consensus

Fig. 2. L_∞-consensus and average consensus.

the similarity between two indexed hierarchies and who is independent of the consensus methods. This coefficient has been computed to measure the adequacy between each of the hierarchies H_1, H_2 and H_3 and each of the consensus (Table 1). For each consensus method, the mean of the cophenetic correlation coefficients associated to H_1, H_2 and H_3 has also been computed. As the results are very similar for the average consensus and its approached solution, we will only comment the cophenetic correlation coefficients obtained for the average consensus and the infinite norm consensus.

The average consensus is very similar to H_1 (0.993), and in fact both hierarchies have the same structure in terms of subsets, but is rather different from H_2 (-0.077) and from H_3 (0.082). The infinite norm consensus for its part is rather close to H_1 (0.531) and more close to H_3 (0.661) and different from H_2 (-0.077). Both methods derive solutions that are rather different from H_2 which is a waited result as H_1 and H_3 are more similar one to each other than they are to H_2. The means of the coefficients associated to H_1, H_2 and H_3 are 0.333 and 0.331 respectively for the average consensus and the infinite norm consensus. If we consider these means as global mesures of

	L_∞-consensus	Average consensus	Average consensus (approached)
H_1	0.531	0.993	0.993
H_2	-0.199	-0.077	-0.076
H_3	0.661	0.082	0.080
Mean	0.331	0.333	0.332

Table 1. Cophenetic correlation coefficients.

the similarity between each consensus solution and the profile **P**, the results are very similar.

5 Conclusion

On this example, when one considers the subsets obtained in the two consensus, the consensus in infinite norm seems a little bit drastic while the average consensus seems for its part too tolerant. When comparing the two methods with the cophenetic correlation coefficient, the L_∞-consensus seems to be more faithful to the profile **P**. The algorithm of construction of the consensus in infinite norm presents nevertheless the indisputable advantage to be polynomial. Empirical studies with real data and simulations would be necessary to highlight the advantages and the inconveniences of each of the methods.

References

ADAMS, E.N., III. (1972): Consensus techniques and the comparison of taxonomic trees. *Systematic Zoology, 21, 390-397.*

BARTHÉLEMY, J.-P. and McMORRIS, F.R. (1986): The median procedure for n-Trees. *Journal of Classification, 3, 329-334.*

CHANDON, J.-L. and DE SOETE, G. (1984): Fitting a least squares ultrametric to dissimilarity data : approximation versus optimisation. In E. Diday et al. (Eds.): *Data Analysis and Informatics III.* Elsevier Science Publishers B.V., Amsterdam, 213–219.

CHEPOI, V. and FICHET, B. (2000): L_∞-approximation via subdominants. *Journal of Mathematical Psychology, 44, 600-616.*

CUCUMEL, G. (1990): Construction d'une hiérarchie consensus à l'aide d'une ultramétrique centrale. In : *Recueil des textes des présentations du colloque sur les méthodes et domaines d'application de la Statistique 1990.* Bureau de la Statistique du Québec, Québec, 235–243.

FINDEN, C.R. and GORDON, A.D. (1985): Obtaining common pruned trees. *Journal of Classification, 2, 225-276.*

HARTIGAN, J.A. (1967): Representation of similarity matrices by trees. *Journal of the American Statistical Association, 62, 1140-1148.*

LAPOINTE, F.-J. and CUCUMEL, G. (1997): The average consensus procedure: combination of weighted trees containing identical or overlapping sets of objects. *Systematic Biology, 46, 306-312.*

LECLERC, B. (1998): Consensus of classifications: the case of trees. In: A. Rizzi, M. Vichi and H.-H. Bock (Eds.): *Advances in Data Science and Classification.* Springer-Verlag, Berlin, 81–90.

LECLERC, B. and CUCUMEL, G. (1987): Consensus en classification : une revue bibliographique. *Mathématiques et sciences humaines, 100, 109-128.*

MARGUSH, T. and McMORRIS, F.R. (1981): Consensus n-trees. *Bull. Mathematical Biology, 43(2), 239-244.*

NEUMANN, D.A. (1983): Faithful Consensus Methods for n-trees. *Mathematical Biosciences, 63, 271-287.*

SOKAL R.R. and ROHLF, F.J. (1962): The Comparison of dendrograms by objective methods. *Taxon, 11, 33-40.*

STINEBRICKNER, R. (1984): An Extension of intersection methods from trees to dendrograms. *Systematic Zoology, 33, 381-386.*

Consensus from Frequent Groupings

Bruno Leclerc

Centre d'Analyse et de Mathématique Sociales, EHESS
54 bd Raspail, 75270 Paris cedex 06, France, *leclerc@ehess.fr*

Abstract. Let $\mathcal{D}^* = (\mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_k)$ be a profile of classifications of a given set X. We aim to aggregate $\mathcal{D}^*$ into a unique consensus classification $\mathcal{D}$. Classifications considered here are sets of classes which are not included into each other. To any integer p comprised between 1 and k (both included), one makes correspond a *frequent grouping consensus function* F_p which returns the maximal subsets of X included in elements of at least p of the $\mathcal{D}_i$'s. We give some properties and three characterizations of such consensus rules.

1 Introduction

In his celebrated work on the "nuées dynamiques", Edwin Diday (1971) emphasized the importance of groupings appearing in many iterations of his algorithm, that he called "formes fortes". The present paper may be thought of as a study of the systems of classes obtained on this way, that will be called *frequent groupings.*

Considerations issued from another background will play a major role in this study. Let X be a finite set, and $R \subseteq (\mathcal{P}(X))^2$ a binary relation on the set of all subsets of X. In previous papers and communications (Domenach and Leclerc (2004b), Leclerc (2004), Leclerc (2005)), we established the uniqueness of a classification $\mathcal{D}$ (on the Moore family form) satisfying two conditions related with R and generalizing conditions stated by Adams (1986); such conditions ensure an admissible fitting of the nesting order of $\mathcal{D}$ (see Section 5 below) to R. It remains an existence problem, since such a classification $\mathcal{D}$ does not exist for any relation R. Indeed, Adams pointed out the existence in a specific case, related to hierarchies and to a unanimity rule. Here, we show that frequent groupings correspond to a generalized Adams consensus situation, in a fairly general frame. Moreover, the obtainment of the consensus classification $\mathcal{D}$ is close to the determination of "frequent items", now a major topic in association rules mining (cf. Hipp et al. (2000), Han and Kamber (2001)).

The paper is organized as follows. Main definitions, including frequent groupings consensus functions, are given in Section 2. In Section 3, we briefly mention the relation between our frequent groupings and the frequent items of the literature, with its interesting algorithmic consequences. Section 4 presents some properties of frequent groupings consensus functions, with a first characterization result, in the spirit of Arrow. In Section 5, we first recall the definitions of the implication and the nesting relations associated with a set of classes. Then, a characterization in terms of nestings (that is, in the spirit of Adams) is obtained, and its variant in tems of implications (or exact association rules) is stated.

2 Definitions

A classification, as considered here, consists of a family $\mathcal{D}$ of subsets (classes) of a given finite set X with n elements ($n \geq 2$). Moreover, the classification $\mathcal{D}$ is supposed here to be a *proper Sperner family*, that is $\mathcal{D} \neq \{X\}$ and $\mathcal{D} \neq \emptyset$, and classes of $\mathcal{D}$ are pairwise incomparable for inclusion. The classification $\mathcal{D}$ is a *covering of* X if the union of its classes is X and a *partition of* X if, moreover, any pair of classes has empty intersection. The set of all proper Sperner families on X is denoted as $\mathbf{S}$. A subset A of X is said to be a *grouping of* $\mathcal{D}$ if there exists at least one class C of $\mathcal{D}$ containing A.

Besides the elements of $\mathbf{S}$, another type of families will be considered below. A family $\mathcal{M}$ of subsets of X is a *Moore family* (or a *closure system*) if it satisfies the following two properties: (i) $X \in \mathcal{M}$, and (ii) for all $A, B \in \mathcal{M}$, $A \cap B \in \mathcal{M}$. The Moore family $\mu(\mathcal{D}) = \{\cap\mathcal{D}' : \mathcal{D}' \subseteq \mathcal{D}\}$ is associated to any (Sperner or not) family $\mathcal{D}$ (the obtainment of $\mathcal{M} = \{X\}$ corresponding to $\mathcal{D}' = \emptyset$).

Let $\mathcal{D}^* = (\mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_k) \in \mathbf{S}^k$ be a *profile* of such classifications. We aim to aggregate $\mathcal{D}^*$ into a unique Sperner family $\mathcal{D}$. Set $K = \{1, 2, ..., k\}$. We associate to the profile $\mathcal{D}^*$ a *grouping index* $g_{\mathcal{D}^*}$ on the set $\mathcal{P}(X)$ of all subsets of X, by setting, for any $A \subseteq X$,
$g_{\mathcal{D}^*}(A) = |\{i \in K : A \subseteq C \text{ for at least one class } C \text{ of } \mathcal{D}_i\}|$
So, $g_{\mathcal{D}^*}(A)$ is the number of those classifications in the profile $\mathcal{D}^*$ admitting A as a grouping.

We associate a consensus function $F_p : \mathbf{S}^k \longrightarrow \mathbf{S}$ to the $g_{\mathcal{D}^*}$ index and to any integer $p \in K$. A subset A of X is said a *p-frequent grouping* if $g_{\mathcal{D}^*}(A) \geq p$, and the *p-frequent groupings consensus* of $\mathcal{D}^*$, denoted $F_p(\mathcal{D}^*)$, is the Sperner family of all the maximal p-frequent groupings. Note that $F_k(\mathcal{D}^*)$ is the set of the subsets C of X with the form $C = \bigcap_{1 \leq i \leq k} C_i$, with $C_i \in \mathcal{D}_i$ for all $i \in K$, and maximal with this property. Then, if $\mathcal{D}^*$ is a profile of partitions, one finds the meet of the partitions of $\mathcal{D}^*$. On the other hand, $F_1(\mathcal{D}^*)$ is the set of those classes of $\bigcup_{1 \leq i \leq k} \mathcal{D}_i$ which are maximal for inclusion.

Though they are somewhat natural, such frequent groupings do not seem to have been often considered in the literature. As we mention in the introduction, the “formes fortes” of Edwin Diday (Diday (1971)) constitute a noticeable exception.

3 Frequent groupings and frequent items

Consider the special case where, in the profile $\mathcal{D}^*$, the family $\mathcal{D}_i$ reduces to a unique class C_i, for all $i = 1, ..., k$. It is equivalent to consider a database $\mathcal{D}^*$ whose transactions are the C_i's. Then, our frequent groupings correspond to the so-called *frequent itemsets* of $\mathcal{D}^*$.

The determination of frequent itemsets is an important topic in data mining (association rules mining). Many algorithms have been designed to obtain them, even in large databases. According to the previous observation, frequent groupings, as defined above, constitute a generalization of frequent itemsets. This observation has important consequences for algorithmic issues. Here, we just give the example of the adaptation of the “prototypal” algorithm *Apriori* (Agrawal and Srikant (1994)).

This algorithm proceeds with a tree exploration of $\mathcal{P}(X)$. The cutting of many branches, which allows to deal with great amounts of data, corresponds to a selection of potentially frequent itemsets. For each such itemset B, the database $\mathcal{D}^*$ is scanned to determine whether it has at least p elements C_i containing B. The adaptation of this procedure to frequent groupings is straightforward: one scans successively the families $\mathcal{D}_i$'s, with the new instruction to jump to family $\mathcal{D}_{i+1}$ as soon as a class containing B is found in $\mathcal{D}_i$.

The adaptation of other algorithms should be examined case by case.

4 Some properties, with an arrowian characterization

We first give some properties of the consensus function F_p. This function from $\mathbf{S}^k$ to $\mathbf{S}$ associates the Sperner family $F_p(\mathcal{D}^*)$ to any profile $\mathcal{D}^*$ of Sperner families. One easily verifies that, moreover:

- if all the $\mathcal{D}_i$'s are coverings of X, then $F_p(\mathcal{D}^*)$ is a covering of X,
- if all the $\mathcal{D}_i$'s are families of intervals of a fixed linear order L on X, then $F_p(\mathcal{D}^*)$ is a family of intervals of L,
- if all the $\mathcal{D}_i$'s are partitions of X, then $F_k(\mathcal{D}^*)$ is a partition of X,

For $p < k$, the consensus $F_p(\mathcal{D}^*)$ of a profile $\mathcal{D}^*$ of partitions of X is generally not a partition. For an example, set $X = \{a, b, c, d\}, k \geq 3$, and consider a profile $\mathcal{D}^*$)) of partitions, $k-2$ of them being equal to $\{\{a,b,c\},\{d\}\}$, and the remaining two being $\{\{a,b\},\{c\},\{d\}\}$ and $\{\{a\},\{b,c\},\{d\}\}$. One gets $F_{k-1}(\mathcal{D}^*) = \{\{a,b\},\{b,c\},\{d\}\}$, not a partition.

The conclusion about the fact that a subset A of X is or is not a grouping of $F_p(\mathcal{D}^*)$ depends only on the value of the index $g_{\mathcal{D}^*}(A)$, and not on the elements or subsets of $X - A$. More precisely, consider the following three properties for a consensus functions F from $\mathbf{S}^k$ to $\mathbf{S}$ (see, e.g., Monjardet (1990), Day and McMorris (2003)). The property (S) of *symmetry* ensures that the output of F does not depend on the order of the elements of a profile; here, given a permutation σ of K, and a profile $\mathcal{D}^* = (\mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_k)$, we set $\mathcal{D}^*_\sigma = (\mathcal{D}_{\sigma(1)}, \mathcal{D}_{\sigma(2)}, ..., \mathcal{D}_{\sigma(k)})$. The properties (UG) of *unanimity for groupings* and (NMG) of *neutral-monotony for groupings* are "arrowian" ones, where groupings are taken as elementary constituents of a Sperner family:

(S) For any profile $\mathcal{D}^*$ and permutation σ of K, $F(\mathcal{D}^*) = F(\mathcal{D}^*_\sigma)$;

(UG) $[A \subseteq X$ and $g_{\mathcal{D}^*}(A) = k] \Rightarrow [A$ is a grouping of $F(\mathcal{D}^*)]$;

(NMG) $[\mathcal{D}^*, \mathcal{E}^* \in \mathbf{S}^k, A, A' \subseteq X$ and $\{i \in K : A$ is a grouping of $\mathcal{D}_i\} \subseteq \{i \in K : A'$ is a grouping of $\mathcal{E}_i\}] \Rightarrow [A$ is a grouping of $F(\mathcal{D}_*) \Rightarrow A'$ is a grouping of $F(\mathcal{E}_*)]$.

Theorem 1. *A consensus function F: $\mathbf{S}^k \to \mathbf{S}$ is a p-frequent grouping consensus function for some $p \in K$ if and only if it satisfies properties (S), (UG) and (NMG).*

Proof. Obviously, every p-frequent grouping consensus function F_p satisfies properties (UG) and (NMG).

For the converse, consider a consensus function F satisfying conditions (UG) and (NMG). We say that a subset J of K is *decisive* for a profile $\mathcal{D}^*$ and for a subset A of X if $J = \{i \in K : A$ is a grouping of $\mathcal{D}_i\}$ and A is a grouping of $F(\mathcal{D}^*)$. According to (NMG), we then have, for any profile $\mathcal{E}^* \in \mathbf{S}^k$ and for any $A' \subseteq X$, $J \subseteq \{i \in K : A'$ is a grouping of $\mathcal{E}_i\}$ implies $[A'$ is a grouping of $F(\mathcal{E}^*)]$. That is, J is decisive for any subset of X and any profile, as well as any subset J' of K containing J. So, we just say that J is a decisive set.

It remains to determine these decisive sets. According to (UG), K is a decisive set. Let J be a decisive set of minimum cardinality p. If $p = k$, then $F = F_k$. Otherwise, let J' be another subset of cardinality p of K, and consider a permutation σ on K which maps J onto J'. Let $A \subset X$ and $\mathcal{D}^*$ be a profile such that $J = \{i \in K : A$ is a grouping of $\mathcal{D}_i\}$. Then, since J is decisive, A is a grouping of $F(\mathcal{D}^*)$. By property (S), A is a grouping of $F(\mathcal{D}^*_{\sigma)})$ too, and J' is again a decisive set of F. So, any subset of K of cardinality p (and, so, of cardinality at least p) is decisive while, by the minimality hypothesis on p, a subset of K with less than p elements is not. In other terms, $F = F_p$.

The previous result may be also derived from a general one on the latticial consensus in distributive lattices (Monjardet (1990)). Even thought it is generally interesting to obtain a specific result as a particularization of a general one, it is not straightforward in this case to proceed on this way and, for sake of brevity, we do not detail the involved steps.

5 Characterizations by nestings and implications

Two binary relations on $\mathcal{P}(X)$ are associated to any family $\mathcal{D}$ of subsets of X (here, $\mathcal{D}$ is not assumed to be Sperner).

The *implication* relation I is defined by:

for all $A, B \subseteq X$, $[(A, B) \in I] \iff [\text{for any } C \in \mathcal{D}, A \subseteq C \Rightarrow B \subseteq C]$.

So, $(A, B) \in I$ (also denoted by $A \rightarrow B$) means that every class containing A contains B too. It is equivalently said that the pair $A \rightarrow B$ is an (exact) *association rule*, or a *functional dependency* (see Caspard and Monjardet (2003) for results and survey on these implication relations).

The *nesting* order $Œ$ is defined by:

for all $A, B \subseteq X$, $[(A, B) \in Œ\,] \iff A \subset B$ and there exists $C \in \mathcal{D}$ such that $A \subseteq C$ and $B \not\subseteq C$.

So, $(A, B) \in Œ$ (also denoted $A \; Œ \; B$) means that the subset B is more general than A with regards to $\mathcal{D}$. See Domenach and Leclerc (2004a) about these nestings (or overhangings), introduced first by Adams (1986) in the particular case of hierarchies.

An important remark is that, by definition, a Sperner family $\mathcal{D}$ and its corresponding Moore family $\mu(\mathcal{D})$ have the same implication and nesting relations.

Given a profile $\mathcal{D}^* = (\mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_k)$ of families of subsets of X, we denote by $Œ_i$ and $\rightarrow_i$, respectively, the nesting and implication relations associated with the family $\mathcal{D}_i$. For $p \in K$, $Œ^{(p)} = \bigcup_{J \subseteq K, |J| \geq p} \bigcap_{i \in J} \; Œ_i$ is the set of all the pairs $(A, B) \in (\mathcal{P}(E))^2$ which belongs to at least p of the $Œ_i$'s. It was observed that, generally, $Œ^{(p)}$ is not a nesting relation. Then, in the aggregation of the profile $\mathcal{D}^*$ into a unique proper Sperner family $\mathcal{D}$, with nesting relation $Œ$, the equality $Œ = Œ^{(p)}$ cannot be required. Instead, consider the following two conditions, prompted by Adams ones.

(PN) $Œ^{(p)} \subseteq Œ$;

(QN) For all $C \in (\mathcal{D}$, $(C, X) \in Œ^{(p)}$;

The inclusion (PN) corresponds to a *preservation of nestings* (those appearing in at least p elements of the profile). The condition (QN) of *qualified nestings* may be thought of as a partial converse of (PN), where it is just required that the distinguished pairs (C, E) (which obviously are in Œ) are already nestings for at least p elements of the profile.

Theorem 2. *Let $\mathcal{D}^* \in \mathbf{S}^k$be a profile of proper Sperner families on X. Then, for each $p \in K$, the family $\mathcal{D} = F_p(\mathcal{D}^*)$ is the unique Sperner family on X satisfying Conditions (PN) and (QN) above.*

Proof. We first show that $F_p(\mathcal{D}^*)$ satisfies Conditions (PN) and (QN). Let $A, B \subseteq X$, with $(A, B) \in Œ^{(p)}$. We then have $A \subset B$, and there exists a subset J of K with cardinality at least p such that, for any $i \in J$, there is a class C_i of $\mathcal{D}_i$ for which $A \subseteq C_i$ and $B \nsubseteq C_i$. Choosing J such that $C = \bigcap_{i \in J} C_i$ is maximal for these properties, we obtain $C \in F_p(\mathcal{D}^*)$. We then have $A \subseteq C$ and $B \nsubseteq C$. So, $(A, B) \in Œ$, which corresponds to Condition (PN).

For any class C of $\mathcal{D}$, there exists by definition a subset J of K, with at least p elements, such that, for any $i \in J$, there is some $C_i \in \mathcal{D}_i$ with $C \subseteq C_i$. This implies $(C, X) \in Œ_i$ for any $i \in J$, and $(C, X) \in Œ^{(p)}$. So, the function F_p satisfies Condition (QN).

The uniqueness is the consequence of previous results not detailed here. One considers the Moore family $\mu(\mathcal{D})$, which has the same nesting relation as $\mathcal{D}$. One then applies a uniqueness result given in Domenach and Leclerc (2004b) and Leclerc (2004).

Since implications are more popular than nestings, in the literature as well as in applied fields, it is interesting to obtain a counterpart of Theorem 2 in terms of implications. We first derive two conditions (FI) and (NQI) from Conditions (PN) and (QN) above. Condition (FI) means that any implication pair of $\mathcal{D}$ is a *frequent implication* pair, in the sense that it is an implication pair in enough (precisely, $k - p$) elements of the profile $\mathcal{D}^*$. Condition (NQI) of *negatively qualified implications* means that, for any $C \in \mathcal{D}$, the pair (C, X) (which is not an implication pair of $\mathcal{D}$) cannot be an implication pair in many elements of the profile.

(FI) For all $A, B \subseteq X$, $A \to B$ implies $|\{i \in K : A \to_i B\}| \geq k - p$;

(NQI) For all $C \in \mathcal{D}$, $|\{i \in K : C \to_i X\}| < k - p$.

Proposition 1. *One has the equivalences (PN) $\iff$ (FI) and (QN) $\iff$ (NQI).*

Proof. (PN) implies (FI): let $A, B \subseteq X$ such that $A \to B$. If $B \subseteq A$, then (FI) is always satisfied. Otherwise, according to the properties of implication relations (see, e.g., Caspard and Monjardet (2003)), $A \to B$ implies $A \to A \cup B$, with $A \subset A \cup B$. Then, $(A, A \cup B) \notin Œ$ implies, by (PN), $|\{i \in K : A$

$Œ_i\ A \cup B\}| < p$ and, so, $|\{i \in K : A \to_i A \cup B\}| \geq k - p$, which again implies $|\{i \in K : A \to_i B\}| \geq k - p$, that is, Condition (FI).

(FI) implies (PN): let $A, B \subseteq X$ such that $(A, B) \in Œ^{(p)}$, that is $A \subset B$ and $|\{i \in K : A\ Œ_i\ B\}| \geq p$. Then, $|\{i \in K : A \to_i B\}| < k - p$, which, by (FI), implies that $A \to B$ is not satisfied, that is one has $A\ Œ\ B$.

(QN) $\iff$ (NQI): for $C \in \mathcal{D}$, we have the equivalences $(C, X) \in Œ^{(p)} \iff |\{i \in K : C\ Œ_i\ X\}| \geq p \iff |\{i \in K : C \to_i X\}| < k - p$.

Corollary 1. *Let $\mathcal{D}^* \in \mathbf{S}^k$ be a profile of proper Sperner families on X. Then, for each $p \in K$, the family $\mathcal{D} = F_p(\mathcal{D}^*)$ is the unique Sperner family on X satisfying conditions (FI) and (NQI).*

6 Conclusion

We defined a class of consensus rules by frequent groupings which apply to any profile $\mathcal{D}^* \in \mathbf{S}^k$ of proper Sperner families. We gave three characterizations of these rules. It remains to generalize these results by extending them to more general classification models. Since hierarchies involved in Adams results are not Sperner families, one may expect that such generalizations exist.

It was observed in Section 3 that reaching one of the corresponding systems of conditions requires to give up stability for partitions. In fact, in many domains of application, the obtainment of overlapping classes is not at all a drawback. The frequent groupings consensus may be a useful tool for the classification of data described by qualitative variables, less constraining than the search of a consensus partition initiated by Régnier (1965) and Mirkin (1975) (see also Barthélemy and Leclerc (1995)).

References

ADAMS III, E.N. (1986): N-trees as nestings: complexity, similarity and consensus. *Journal of Classification 3, 299-317.*

AGRAWAL, R. and SRIKANT, R. (1994): Fast algorithms association rules. *Proceedings of the 20th VLDB Conference, Santiago, Chile.* 1–7.

BARTHÉLEMY, J.P. and LECLERC, B. (1995): The median procedure for partitions. In: I.J. Cox, P. Hansen and B. Julesz (Eds.): *Partitioning data sets, DIMACS Series in Discrete Mathematics and Theoretical Computer Science 19.* American Mathematical Society, Providence, RI, 3–34.

CASPARD, N. and MONJARDET, B. (2003): The lattices of Moore families and closure operators on a finite set: a survey. *Discrete Applied Mathematics 127, 241-269.*

DAY, W.H.E. and MCMORRIS, F.R. (2003): *Axiomatic Consensus Theory in Group Choice and Biomathematics.* SIAM, Philadelphia.

DIDAY, E. (1971): Une nouvelle méthode en classification automatique et reconnaissance des formes : la méthode des nuées dynamiques. *Revue de Statistique Appliquée XIX, 19-33.*

DOMENACH, F. and LECLERC, B. (2004a): Closure Systems, Implicational Systems, Overhanging Relations and the case of Hierarchical Classification. *Mathematical Social Sciences 47, 349-366.*

DOMENACH, F. and LECLERC, B. (2004b): Consensus of classification systems, with Adams' results revisited. In: D. Banks, L. House, F.R. McMorris, P. Arabie and W. Gaul (Eds.): *Classification, Clustering and Data Mining Applications.* Springer, Berlin, 417–428.

HAN, J. and KAMBER, M. (2001): *Data mining: concepts and techniques.* Morgan Kaufmann Publishers, San Francisco.

HIPP, J., GUNTZER, U. and NAKHAEIZADEH, G. (2000): Algorithms for association rule mining - a general survey and comparison. *SIGKDD Explorations 2, 58-64.*

LECLERC, B. (2004): On the consensus of closure systems. *Annales du LAMSADE 3, 237-247.*

LECLERC, B. (2005): Implications, emboîtements et ajustements de classifications. In: V. Makarenkov, G. Cucumel, F.-J. Lapointe (Eds.): *Comptes rendus des 12èmes rencontres de la Société Francophone de Classification.* UQAM, Montréal, 17–20.

MIRKIN, B.G. (1975): On the problem of reconciling partitions. In: *Quantitative Sociology, International Perspectives on Mathematical and Statistical Modelling.* Academic Press, New York, 441-449.

MONJARDET, B. (1990): Arrowian characterizations of latticial federation consensus functions. *Mathematical Social Sciences 20, 51-71.*

RÉGNIER, S. (1965): Sur quelques aspects mathématiques des problèmes de classification automatique. *ICC Bulletin 4, 175-191*, repr. (1983) *Mathématiques et Sciences humaines 82, 13-29.*

Consensus of Star Tree Hypergraphs

Fred R. McMorris[1] and Robert C. Powers[2]

[1] Department of Applied Mathematics, Illinois Institute of Technology
Chicago, Illinois 60616, USA, *mcmorris@iit.edu*

[2] Department of Mathematics, University of Louisville
Louisville, Kentucky 40292, USA, *rcpowe01@louisville.edu*

Abstract. Popular methods for forming the consensus of several hypergraphs of a given type (e.g., hierarchies, weak hierarchies) place a cluster in the output if it appears sufficiently often among the input hypergraphs. The simplest type of tree hypergraph is one whose clusters are subtrees of a star. This paper considers the possibility of forming consensus by simply counting the frequency of occurances of clusters for star hypergraphs.

1 Introduction and definitions

In a thought-provoking paper, Diday (2004) extends the usual notion of a pyramid to the case of "spatial pyramids". An example of this is where instead of certain intervals of a path being considered as clusters, the clusters are certain convex subsets of an underlying grid graph. In previous work of ours (Lehel et al. (1998)), we have proposed the study of consensus of hypergraphs where the clusters are taken as convex subsets (i.e.,subtrees) of a tree. It would seem to be a reasonable research project to study the consensus of various spatial pyramids. But before undertaking this project, in this short note we study the consensus of the simplest type of tree hypergraph, namely those defined on a star. Although general tree hypergraphs do not have the nice visualization properties of Diday's spatial pyramids perhaps on "tree-like grids" a more spatial version may be possible. This too is left for future investigation.

We first recall some basic definitions: A (simple) *hypergraph* on the set S is a set of non-empty subsets (the *edges*) of S (the set of *vertices*). For H a hypergraph, we also require $S \in H$ and $\{x\} \in H$ for all $x \in S$. Since these kinds of hypergraphs result after applying standard clustering methods, we call an edge A of the hypergraph H ($A \in H$) a *cluster* of H and if $1 < |A| < |S|$ it is a *nontrivial cluster.* A *pyramid* on S is a hypergraph P with $A \cap B \in P \cup \{\emptyset\}$ for all $A, B \in P$, and there is a total ordering of S such that each cluster of P is an interval in this ordering. In this definition if "total ordering" is replaced by "tree" and "interval" by "subtree", a *tree hypergraph* on S results. A *star tree hypergraph* is a tree hypergraph where the underlying tree is a star graph (a graph with $n+1$ vertices, with n vertices of degree one and one vertex of degree n, the *central* vertex). For example,

$H = \{\{x, a, d\}, \{x, b, c\}\}$ is a star tree hypergraph based on the tree shown in the Figure 1.

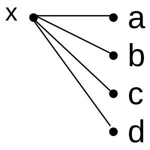


Fig. 1. A star.

To extend this a step further, we will modify a star tree hypergraph into a *subdivided star tree hypergraph* if the underlying star has each edge subdivided once. For example, $H = \{\{x, a\}, \{x, b, c, c'\}, \{a, a'\}, \{b, b'\}\}$ is a star-one tree hypergraph based on the tree shown in Figure 2.

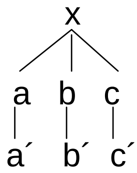


Fig. 2. A subdivided star.

2 Consensus of star tree hypergraphs

Let $\mathcal{S}$ be the set of all star tree hypergraphs with vertex set S and $|S| \geq 3$. We first make a couple of easy observations concerning star tree hypergraphs. If H is a star tree hypergraph, then $A \cap B \neq \emptyset$ for any two nontrivial clusters $A, B \in H$. In other words, there do not exist disjoint clusters. Indeed, the following statements are easy to prove.

Proposition 1 *A hypergraph H is a star tree hypergraph if and only if the intersection of all the nontrivial clusters of H is nonempty.*

Proposition 2 *If H is a subdivided star tree hypergraph, then there exists a bipartition $(\mathcal{C}_1, \mathcal{C}_2)$ of the set of nontrivial clusters of H such that*
i) if $\mathcal{C}_1 \neq \emptyset$, then the intersection of all the elements in $\mathcal{C}_1$ is nonempty;
ii) if $\mathcal{C}_2 \neq \emptyset$ and $A \in \mathcal{C}_2$, then $|A| = 2$ and $A \cap B = \emptyset$ for all $B \neq A$ in $\mathcal{C}_2$.

For our example $H = \{\{x,a\},\{x,b,c,c'\},\{a,a'\},\{b,b'\}\}$, the sets $\mathcal{C}_1 = \{\{x,a\},\{x,b,c,c'\}\}$ and $\mathcal{C}_2 = \{\{a,a'\},\{b,b'\}\}$ satisfy items i) and ii) of Proposition 2. Unfortunately, the converse of Proposition 2 is not true. For example, if $H = \{\{x,y,a\},\{x,z,b\},\{a,b\}\}$, then $\mathcal{C}_1 = \{\{x,y,a\},\{x,z,b\}\}$ and $\mathcal{C}_2 = \{\{a,b\}\}$ satisfy items i) and ii) of Proposition 2 but it is easy to see that H is not a subdivided star tree hypergraph. Therefore, in order to simplify our discussion we will focus on star tree hypergraphs.

We are concerned with *consensus functions* on $\mathcal{S}$, which are mappings

$$f : \mathcal{S}^k \to \mathcal{S}$$

where k is a fixed positive integer. Elements of $\mathcal{S}^k$ are called *profiles* and are denoted by $P = (H_1, \ldots, H_k)$. One method of consensus is to implement a *counting rule*, i.e. the definition of f is based on the existence of a nonnegative number q such that

$$A \in f(P) \ \Leftrightarrow |\{i : A \in H_i\}| > q$$

for all $P = (H_1, \ldots, H_k) \in \mathcal{S}^k$. In this case, the consensus function f is denoted by f_q. In particular, $f_{\frac{k}{2}}$ is called the *majority rule.* (See (Day and McMorris, 2003) for these and other types of consensus rules.) Unfortunately, in this case, the output of $f_{\frac{k}{2}}$ need not be a star tree hypergraph.

Example 1 *Let $S = \{x_1, x_2, x_3\}$ and consider the profile $P = (H_1, H_2, H_3)$ where $H_1 = \{x_1x_2, x_1x_3\}$, $H_2 = \{x_2x_1, x_2x_3\}$, and $H_3 = \{x_3x_1, x_3x_2\}$. (We use the notation $x_1x_2 \ldots$ for $\{x_1, x_2, \ldots\}$.) Then $f_{\frac{k}{2}}(P) = \{x_1x_2, x_1x_3, x_2x_3\}$. Since the three clusters that make up $f_{\frac{k}{2}}(P)$ have an empty intersection, it follows from Proposition 1 that $f_{\frac{k}{2}}(P)$ is not a star tree hypergraph.*

If we restrict the domain in a certain way, it is possible to force the output of $f_{\frac{k}{2}}$ to be a star tree hypergraph. This is similar to the situation we found (Lehel, et al., 1998) when trying to construct counting rules for pyramids. In that case we tried restricting each pyramid to fixed underlying linear order, but then found that **any** selection of clusters from the input profile of pyramids would result in a well-defined "consensus" pyramid. This required us to abandon the counting approach for pyramids. For star tree hypergraphs, we can make some progress. Towards this end, we first need some notation. Let H_0 denote the hypergraph on S with no non-trivial clusters and for any $H \in \mathcal{S}$ with $H \neq H_0$ and $T \subseteq S$, let

$$T \cap H = T \cap A_1 \cap A_2 \cap \ldots \cap A_r$$

where $A_1, A_2, \ldots, A_r$ are the nontrivial clusters of H. For any nonempty subset $\mathcal{S}'$ of $\mathcal{S}$, let

$$c(\mathcal{S}') = min\{|T| : T \subset S \text{ and } T \cap H \neq \emptyset \ \forall H \in \mathcal{S}' \text{ with } H \neq H_0\}.$$

If $c(\mathcal{S}') = 1$, then it is clear that $f_{\frac{k}{2}}(P) \in \mathcal{S}$ for all $P \in (\mathcal{S}')^k$.

Theorem 1 *For any nonempty subset $\mathcal{S}'$ of $\mathcal{S}$, if $c(\mathcal{S}') = 2$, then$f_{\frac{k}{2}}(P) \in \mathcal{S}$ for all $P \in (\mathcal{S}')^k$. Moreover, if $k \geq 3$, then there exists a subset $\mathcal{S}'$ of $\mathcal{S}$ such that $c(\mathcal{S}') = 3$ and $f_{\frac{k}{2}}(P) \notin \mathcal{S}$ for some $P \in (\mathcal{S}')^k$.*

Proof. If $c(\mathcal{S}') = 2$, then there exists $T \subseteq S$ such that $|T| = 2$ and $T \cap H \neq \emptyset$ for all $H \in \mathcal{S}'$ with $H \neq H_0$. Let $T = \{x, y\}$ for some $x, y \in S$. Suppose $A_1, A_2, \ldots, A_r$ are the nontrivial clusters of $f_{\frac{k}{2}}(P)$ for a profile $P \in (\mathcal{S}')^k$ and that $A_1 \cap A_2 \cap \ldots \cap A_r = \emptyset$. Then there exist clusters A_i and A_j such that $x \notin A_i$ and $y \notin A_j$. Since each cluster belongs to more than half the profile it follows that there exists H_ℓ such that $A_i, A_j \in H_\ell$. But then $T \cap H_\ell = \emptyset$ contrary to the above.

The second part of the theorem is essentially covered in Example 1. □

So we have seen that majority rule leads to a well-defined consensus method for star tree hypergraphs only if there is some restriction on the domain.

The ideas used to prove Theorem 1 can be used to establish the following result.

Theorem 2 *For any nonempty subset $\mathcal{S}'$ of $\mathcal{S}$, if $c(\mathcal{S}') = n$ where $n < k$, then$f_{\frac{(n-1)k}{n}}(P) \in \mathcal{S}$ for all $P \in (\mathcal{S}')^k$. Moreover, if $|S| \geq n+1$, then there exists a subset $\mathcal{S}'$ of $\mathcal{S}$ such that $c(\mathcal{S}') = n+1$ and $f_{\frac{(n-1)k}{n}}(P) \notin \mathcal{S}$ for some $P \in (\mathcal{S}')^k$.*

For the "moreover" part, create the profile

$$P = (H_1, \ldots, H_{n+1}, H_0, \ldots, H_0)$$

where $f_{\frac{(n-1)k}{n}}(P)$ contains the $(n+1)$ clusters $A_{n+1} = x_1x_2...x_n$, $A_1 = x_2x_3...x_{n+1}$, $A_2 = x_3...x_{n+1}x_1$, ..., $A_n = x_{n+1}x_1...x_{n-1}$ such that $A_i \in H_j$ if and only if $i \neq j$. Since the intersection of the $A_i's$ is empty it follows that $f_{\frac{(n-1)k}{n}}(P) \notin \mathcal{S}$.

The key idea for the first part is to use the pigeon hole principle which forces any n nontrivial output clusters to belong to one of the input hierarchies.

Using Theorem 2, we can now propose a new type of consensus rule for star tree hypergraphs. For any profile $P = (H_1, \ldots, H_k) \in \mathcal{S}^k$ define $g : \mathcal{S}^k \to \mathcal{S}$ by

$$g(P) = H_1 \cap \ldots \cap H_k \text{ if } c(\{P\}) = k$$

or

$$g(P) = f_{\frac{(n-1)k}{n}}(P) \text{ if } c(\{P\}) = n < k$$

where $\{P\} = \{H_1, \ldots, H_k\}$. The rule g is a counting rule where the threshold depends on the input profile.

To gain a better understanding of the consensus rule g, we propose a small list of axioms.

Anonymity (A) A rule $f : \mathcal{S}^k \rightarrow \mathcal{S}$ satisfies anonymity if $f(H_{\phi(1)}, \ldots, H_{\phi(k)}) = f(H_1, \ldots, H_k)$ for any profile $(H_1, \ldots, H_k)$ and permutation ϕ of $\{1, \ldots k\}$.

Monotone Neutrality (MN) A rule $f : \mathcal{S}^k \rightarrow \mathcal{S}$ satisfies monotone neutrality if for any two profiles P and P' and for any two nontrivial clusters A and B,

$$c(\{P\}) \geq c(\{P'\}) \text{ and } \{i : A \in H_i\} \subseteq \{i : B \in H_i'\}$$

implies that

$$B \in f(P') \text{ whenever } A \in f(P).$$

The counting rule g satisfies the axioms (A) and (MN). However, these axioms do not characterize g since the unanimity rule also satisfies (A) and (MN). An interesting problem is to give a complete characterization of g.

References

DAY, W.H.E. and McMORRIS, F.R. (2003): *Axiomatic Consensus Theory in Group Choice and Biomathematics*, SIAM Frontiers in Applied Mathematics, Philadelphia.

DIDAY, E. (2004): Spatial pyramidal clustering based on a tessellation. In: D. Banks, et al. (Eds.): *Classification, Clustering, and Data Mining Applications.* Springer, Berlin,105–120.

LEHEL, J., McMORRIS, F.R. and POWERS, R.C. (1998): Consensus methods for pyramids and other hypergraphs. In: C. Hayashi, et al. (Eds.): *Data Science, Classification, and Related Methods*, Springer, Tokyo, 187-190.

Part V

Data Analysis, Data Mining, and KDD

Knowledge Management in Environmental Sciences with $\mathcal{IKBS}$: Application to Systematics of Corals of the Mascarene Archipelago

Noel Conruyt and David Grosser

Laboratoire IREMIA, Université de la Réunion,
15 avenue René Cassin, BP 7151,
97715 Saint-Denis Msg. Cedex 9, France
Web: http://ikbs.univ-reunion.fr/
conruyt,grosser@univ-reunion.fr

Abstract. Systematics, the scientific discipline that deals with listing, describing, naming, classifying and identifying living organisms is a central point in environmental sciences. Expertise is becoming rare and for future biodiversity studies relying on species identification, environmental technicians will only be left with monographic descriptions and collections in museums.

With the emergence of knowledge management, it is possible to enhance the use of systematician's expertise, by providing them with collaborative tools to widely manage, share and transmit their knowledge. Knowledge engineering in Systematics means to revise taxa and descriptions of specimens. We have designed an Iterative Knowledge Base System – $\mathcal{IKBS}$ – for achieving these goals. It applies the scientific method in biology (conjecture and test) with a natural process of knowledge management. The product of such a tool is a collaborative knowledge base of a domain, that can evolve (by updating the knowledge) and be connected to distributed databases (bibliographic, photographic, geographic, taxonomic, etc.) that will yield information on species after the identification process of a new specimen.

This paper presents an overview of the methodology, the methods (identification tree and case-based reasoning) and the validation process used to build knowledge bases in Systematics. An application on corals of the Mascarene Archipelago is given as a case study.

1 Introduction

Today around the world, scientific databases are increasingly delivered on CD-ROM or through Internet (e.g. *World Biodiversity data-base* from ETI in Netherlands, *Reefbase* and *Fishbase* from ICLARM in the Philippines, *Hawaii Biological Survey databases*, *Coral Id* at AIMS, etc.). These applications are taxonomic and bio-geographic information systems with some identification keys for biologists (students, amateurs) and professionals (environment, tourism). In fact, they reproduce mostly electronically what already exists in books (i.e. textual descriptions, identification with diagnostic characters). This approach is interesting when the taxa are well known and stable, but it is not sufficient when the knowledge of groups evolves rapidly, which is particularly the case in the marine environment (corals, hydroids, sponges, etc.).

In such domains, products for knowledge management in Systematics are also needed, with a new methodology of knowledge extraction. This method is based on the re-examination of specimens in various collection in order to get more robust classifications (definition of the taxa) and identifications. In fact, the description of specimens is the key point for engineering Systematics: this descriptive information in the application can always be retrieved in the future and compared again with the museum sample collections. For young systematicians, this specimen-oriented approach brings more robustness to the learning process than working with old monographs based on conceptual species descriptions. Moreover, end-users of such a system (e.g. environmental technicians) can directly compare a newly collected specimen with the description of other specimens in collections.

We have developed a type of knowledge base that supports the above methodology. The tool that generates such applications is called $\mathcal{IKBS}$ (*Iterative Knowledge Base System*, Grosser (2002)). $\mathcal{IKBS}$ is a knowledge management system available on the Internet which is developed in the object-oriented language Java. This tool was co-designed with specialists and end-users for 15 years in different domains such as plant pathology diagnosis, Manago (1992) and computer aided Systematics, Conruyt (1994). For making descriptions, classifications and identifications, our knowledge bases rely not only on observed things (the database of specimen descriptions) but also on observable things (the knowledge of a descriptive model of the domain).

2 Knowledge acquisition

Three points have to be addressed for the knowledge acquisition process: descriptive model definition, questionnaire generation and case acquisition.

2.1 The descriptive model

The descriptive model represents all the observable characteristics (objects, attributes and values) pertaining to individuals belonging to a particular domain. It is organized in a structured scheme, the name of the domain being at the root of a description tree. Each node of the tree is an object (a component of the individual) defined by a list of attributes with their respective possible values. Designing a descriptive model is essentially an expert task.

To help them, we have set up logical rules for case description covering: decomposition, viewpoint, iteration, specialization, and contextual conditions, Le Renard et al. (1994). These rules were constructed from the analysis of the process followed by the experts to create monographs of organisms or diseases.

To serve as an example in coral Systematics, we present the descriptive model of the family *Pocilloporidae* (Figure. 1). The expert has defined 51 objects and 120 attributes. With them, biologists are able to describe 4 genera and 14 species and ecomorphs (see attribute called taxon in Figure 1).

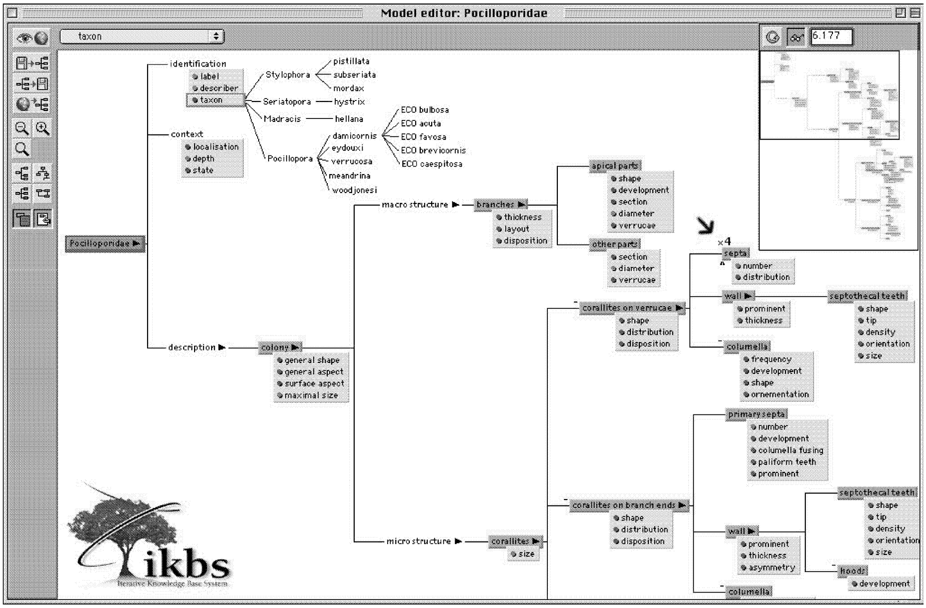


Fig. 1. Part of the descriptive model of the Family *Pocilloporidae*.

There are multiple benefits in such a representation. Viewpoints divide the descriptive model into homogeneous parts, thus giving a frame of reference for describing organisms at a particular level of observation (see object identification, context, description, macro and micro structure in Figure 1).

Sub-components introduce modularity into the descriptions making it possible to structure the domain from the more general to the more specific parts. This object representation of specimens is semantically better than the flat feature-value representation: in the former, local descriptions of attributes depend on the existence of parent objects, although in the latter the defined characters are independent of one another. Some of the possibly missing objects are marked with a minus sign (e.g. columella).

Figure 1 shows the partitioning dimension of objects (subpart links for disjoint classes). For some of them (i.e. septa), other dimensions such as multi-instantiation (× symbol) and specialization ($\wedge$ symbol) of objects can be seen. The former enables users to describe several sorts of the same object by descriptive iteration (there are 4 possible instances for "septa" in Figure 1) and the latter lets users name each sort with the help of the following classification tree of objects (specialization links in Figure 2).

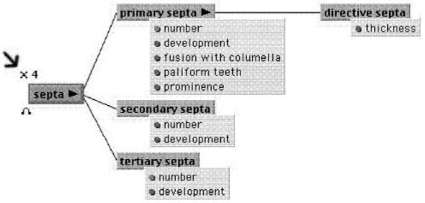


Fig. 2. Classification tree of object septa.

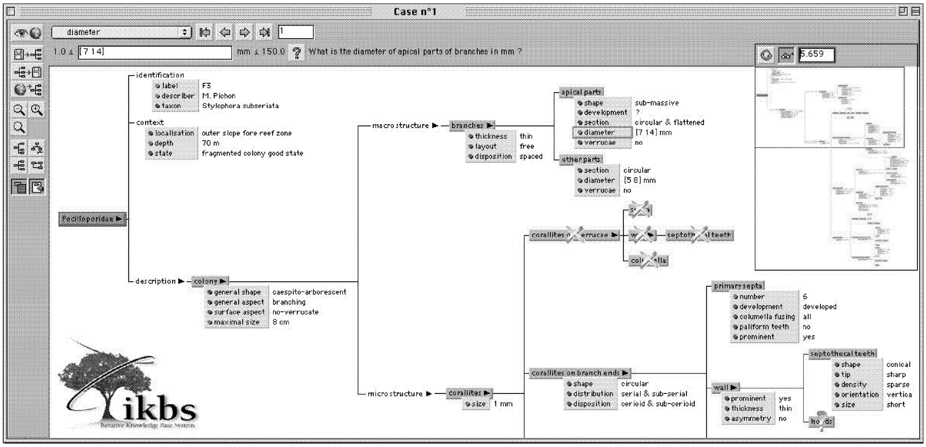
In fact, one of the roles of the descriptive model is to bring an observation guide to the end-user. The objects are linked together by relations that go from the most general to the most specific (from left to right), making the next description process easier for the non-specialist (see below).

2.2 The descriptions

The second phase of knowledge acquisition allows biologists less informed than the experts to acquire personal descriptions and create a case base. An identification name is associated to each specimen observation in order to form a description or a case (Figure 3).

The description process generates sub-trees of the descriptive model (see Figure 1 and Figure 3). Therefore, observed descriptions can be directly compared to one another by leafing through page by page: this navigation process is easier than viewing different lists of attribute-value pairs. In Figure 3, we illustrate possibilities of $\mathcal{IKBS}$ for rendering complete and comprehensive descriptions of a given sample. Different types of attribute are used: taxonomic (e.g. general shape of object colony), numerical intervals (e.g. diameter of apical parts) and multi-nominal values (e.g. section of apical parts). The latter shows variation in objects displaying a set of multiple elements.

The visualization of objects differs graphically according to their status: black if present, black with a cross if absent, symbol **?** if unknown (see object hoods at the bottom-right side of Figure 1 and Figure 3). And last, an object can be specialized (e.g. the septa of corallites on branch ends, see Figure 1): the result is a substitution of its name by a more precise one (e.g. primary septa, see Figure 3) with its associated attributes (inherited or not, see Figure 2). It is important for the user to visualize structured descriptions: so doing brings better clarity and comprehensibility to the acquisition phase. Different sort of values can be used to inform each attributes according to its type:



1.0 ≤ [7 14] mm ≤ 150.0 ? What is the diameter of apical parts of branches in mm ?

identification
- label: F3
- describer: M. Pichon
- taxon: Stylophora subseriata

context
- localisation: outer slope fore reef zone
- depth: 70 m
- state: fragmented colony good state

description ▶ colony ▶
- general shape: caespito-arborescent
- general aspect: branching
- surface aspect: no-verrucate
- maximal size: 8 cm

macro structure ▶ branches ▶
- thickness: thin
- layout: free
- disposition: spaced

apical parts
- shape: sub-massive
- development: ?
- section: circular & flattened
- diameter: [7 14] mm
- verrucae: no

other parts
- section: circular
- diameter: [5 8] mm
- verrucae: no

micro structure ▶ corallites ▶
- size: 1 mm

corallites on branch ends ▶
- shape: circular
- distribution: serial & sub-serial
- disposition: cerioid & sub-cerioid

primary septa
- number: 6
- development: developed
- columella fusing: all
- paliform teeth: no
- prominent: yes

wall ▶
- prominent: yes
- thickness: thin
- asymmetry: no

septothecal teeth
- shape: conical
- tip: sharp
- density: sparse
- orientation: vertica
- size: short

Fig. 3. Part of the description tree of a case of the Family *Pocilloporidae*: *Stylophora subseriata*.

textual type: Any character string is allowed by the system. See the first attribute *label* of the object identification that corresponds to the case identifier. **numerical type**: simple discrete value, interval value (e.g. *diameter* of the object apical parts between [7 14] mm) or set of values, noted {3, 14, 15} for instance. **nominal or ordinal type**: simple value in the attribute domain, disjunction or conjunction of simple values are allowed. See *section* of the object apical parts which is *circular & flattened* : this conjunction of values (AND) means that both values are observed simultaneously for this character. Conversely, a disjunction of values (OR) would be noted *circular | flattened*, and means that the observator is not sure about his choice and prefers to give an imprecise response. **hierarchical type**: hierarchical values are the nodes of the definition domain of an attribute. As for nominal and ordinal types, set of hierarchical values are allowed by 𝓘𝓚𝓑𝓢 (see attribute localisation of object context : the sub-value *fore reef zone* of value *outer slope*).

3 Knowledge processing

In Systematics, data to be processed may be more complex than those considered in conventional data analysis. This complexity cannot be captured by a simple data matrix representation composed by a set of attributes and values. Diversity and incompleteness must be taken into account, and the exception is the only valid rule. The descriptions of specimens are often highly structured (composite or specialised objects), noisy (erroneous or unknown data) and polymorph (variable or imprecise data). Consequently, the design of new symbolic/numeric methods of data analysis that masters this complexity is a challenge for $\mathcal{IKBS}$.

From the computer science viewpoint, we have adapted learning methods from inductive learning algorithm *C4.5*, Quinlan (1994) and case-based reasoning, Aamodt et al. (1994) fields. $\mathcal{IKBS}$ can be compared with AcknoSoft's *KATE*, Isoft's *RECALL* and TecInno's *CBR-Works.* These decision support systems have been designed to cope with industrial fields and very large databases, Manago et al. (1993). Our contribution was to develop new algorithms that exploit background knowledge to facilitate classification (class definition) and identification of natural organisms with the representation and processing of such reality.

$\mathcal{IKBS}$ proposes an easy-to-use on-line identification and classification tool developed in the Java programming language. It integrates two main approaches for finding the class (taxa) to which a specimen belongs. These approaches are based on decision trees (monothetic selection of characters) and case-based reasoning (polythetic selection of characters).

3.1 Identification Trees (IT)

The process of top-down induction of decision trees (DT) is well-known in the machine learning research field since, Quinlan (1993). Classically, DT constitute a particular sort of classifiers, i.e., solutions of the *classification* problem. For identification of biological objects needs, we propose an extension of classical DTs called *Identification Trees* (IT). ITs are used by biologists to: 1/ generate classification rules that correspond to conventional identification keys or 2/ used by themselves as an interactive process to identify new specimen.

Notations The input of the problem is made up of a set of variables (attributes) $\mathcal{A} = \{A_1, \ldots, A_p\}$. In our structured knowledge representation formalism, each attribute pertains to a structured object defined in a descriptive model, and represents a function from a universe D to specific set of values, $\mathrm{dom}(A_k)$, and a set of classes (categories) $\mathcal{C} = \{C_1, \ldots, C_p\}$. The target classifier assigns one (or more) classes from $\mathcal{C}$ to each individual x from D. The assignment is based on the values of x for the variables in $\mathcal{A}$, $x.A_i$. Each example x is usually viewed as a point of the description space $\times_{i=1,k}\mathrm{dom}(A_i)$. Within a typical application of ITs to specimen identification, the variables A_i represent observable characters (or attributes) while the individuals are specimens. In our case, all attributes can be continuous (numeric), discrete (nominal or ordinal), and categorical (hierarchical).

Principles To guess correctly the class of a previously unseen individual x, an IT checks a set of conditions on the attribute values in x, denoted $x.A_i$. The "questions asked" by the IT have one of the following forms: $x.A_i \# v$, where $x.A_i \in \text{dom}(A_i)$, and $\#$ is a generic comparison operator depending on the type of A.

If A is numeric then $\# \in \{=, \neq, \leq, >\}$. In the case of hierarchical attributes (see for instance the attribute *taxon* in Figure 1) specific and generic values have to be compared. This task is realized by a *generalization operator*, noted $\preccurlyeq$ that tests if the input value $x.A_i$ is generalized by (more specific than) the output value v.

For example, the input value $x.taxon =$ *Pocillopora damicornis* (species name) is generalized by $v =$ *Pocillopora* (genus name), because x is a kind-of *Pocillopora.* x is thus assigned to the partition denoted by v, noted $x.taxon \preccurlyeq v$.

An IT is graphically represented by a tree where vertices are labeled by variables A_i and edges by conditions. Moreover, all leaf nodes are labeled by a class from $\mathcal{C}$. When a new x is presented, the prediction engine walks a particular path in the tree and once at a leaf node, it outputs the node's label as the predicted class for x. At an inner node, it checks which of the conditions on adjacent edges holds to choose the next node and thus the next variable to examine.

"Best" attribute But the key step of IT algorithm is the choice of the "best" attribute that eventually leads to compact trees with high predictive accuracy. As the domain of the attribute is split into subdomains, the predictive power of a split may be measured by the homogeneity of the obtained subsets D_{n_i} with respect to the class labels of the member items. Information theory-based criteria have been widely used in split comparison, e.g., entropy reduction, Gini-index, χ^2, and variance reduction. However, in the real applications, any attribute does not have necessarily the same cost as the others. In order to take into account these differences, the best character selection procedure is based on character weighting depending on a linear combination of two factors:

1. The **observation cost**. Each descriptive model components (object or attribute) is weighted in a range [0 1], corresponding to the capability to easily observe the corresponding character.
2. The **discrimination power** of that character which is the entropy reduction.

"On-line" interactivity The user who consults the decision tree can come back to a previous answered question. Another sub-tree is proposed when selecting another character or answering unknown. This dynamic aspect is achieved by the indexing of a sub-set of cases at each node of the decision tree. At each node, the set of indexed cases can be viewed and the case-based strategy can be used (see below).

$\mathcal{IKBS}$'s identification tree algorithm adds some important functionalities to the well known decision tree builder *C4.5* : it works not only on discrete and continuous attributes, but also on structured objects, taxonomic attribute-values and multi-valued attributes, Conruyt et al. (1999).

3.2 Case-based reasoning

$\mathcal{IKBS}$ proposes an alternative method for specimen identification which allows users to inform any characters in any order (random-access keys). Characters which are not available for the specimen observation, or whose interpretation is not clear, can be avoided. Then an interpretation of this incomplete description is retrieved by selecting a subset of k-nearest cases from the descriptions set and by reusing solutions found in the subset by maximizing the probability of obtaining a correct identification or by generating a decision tree as seen above from the k selected cases.

The overall reasoning process behind CBR consists in solving new problems by retrieving and adapting the solutions to similar problems that have occurred in the past. The choice of the most appropriate case(s) from the case base whose solution will be reused to construct the solution of the new case, is driven by analogies in case descriptions, Aamodt (1994). These analogies are detected by a matching mechanism, which typically relies on a similarity assessment function. In cases where the target problem solution is restricted to a single dependent variable, the case-based reasoner may be seen as a particular sort of classifier, and compares to what is known as instance-based learning (IBL).

However, in the context of available background knowledge made up with object, attributes, relations and hierarchical values, the solution have to be situated in an ordered space and the cases structure must be taken into account. Moreover, as the case-based reasoner is heavily dependent on the structure and content of its collection of cases, the case search and matching processes need to be both effective and reasonably time efficient. In this context, two important issues have to be addressed: case retrieval and solution reuse.

Case Retrieval The aim is to find the set of known cases that match the new case at best, i.e., the *BestMatch* set. In our case, this amounts to look for most similar cases (in terms of our similarity assessment function). These cases are typically called *nearest neighbors* since they lay within a particular neighborhood of the new case in the description space. Thus, the retrieve task takes a (possibly partial) problem description, and ends with a complete *BestMatch* set.

The retrieval algorithm performs a complete search through the case base. Each case in the base is compared to the new case by means of similarity function (see 3.2). Depending on the similarity value, the current known case may be inserted into the set of current best matches.

In order to increase the chances of a correct prediction, the set of nearest neighbors *BestMatch* is considered of size greater than one. The exact value of $|BestMatch|$ is a parameter of the algorithm, the approach being known as the *k-nearest neighbors* learning (*k-NN*).

Similarity issues For the assessment of similarity between components in a n-dimensional Euclidean space where coordinates are discrete values, several measures have been proposed in the literature, including various metrics such as the Euclidean distance, Manhattan distance, etc.

In our study, the similarity measure has been derived from the Minkowski metrics to deals on the one hand with complex values such as unknown, hierarchical, interval and set values, and on the other hand with structured descriptions. Some aspects of the similarity measure that works with complex values are developed here. The mathematical details of the complete similarity measure definition can be found in Grosser (2002) and Grosser et al. (2000).

The similarity measure for complex values is defined on two levels: attribute or local level, and component or global level. For each variable $A_i \in \mathcal{A}$, the similarity factor between two descriptions x and y, denoted $sim(x.A_i, y.A_i)$, is defined as the combination of two factors d_P and d_C, reflecting respectively the *relative position* assessment and the *contents part* assessment of the two values.

The relative position factor d_P translates the distance between the two values $x.A_i$ and $y.A_i$ in an ordered (numerical values) or partially-ordered (hierarchical values) space. The contents factor d_C is based on the length of the intersection of the two values, in order to measure the extent of interval or set values common parts. The precise definition of these two factors depends on the type of A.

Formally, the local similarity is computed by:

$$\begin{aligned} sim(x.A_i, y.A_i) = \eta\ & d_P(x.A_i, y.A_i) + \\ \zeta\ & d_C(x.A_i, y.A_i) \end{aligned} \tag{1}$$

where $\eta,\ \zeta \geq 0$, $\eta + \zeta = 1$. η and ζ modulates the relative importance of d_P and d_C. Thus, $\forall (x, y) \in D^2$
$sim(x.A_i, y.A_i) \in [0\ 1]$, d_P and $d_C \in [0\ 1]$.

The contribution of each variable A_i is combined into a unique value characterizing the overall similarity of all the components. For this purpose, we use a linear combination of all attribute-level similarities.

$$Sim(x, y) = \frac{\sum_{i=1}^{i=p} \beta_i\, sim(x.A_i, y.A_i)}{p} \tag{2}$$

where p is the number of attributes and $\beta_i \geq 0$ is the weight of the attribute A_i.

Solution reuse and adaptation The aim of solution reuse is to predict a solution of the current problem from the solutions of the cases in the *BestMatch* set. For this purpose, a combination of the solutions in *BestMatch* is defined that represents a reasonable trade-off of several factors such as frequency of particular class in the set, rank of best matches, etc. The combination may be of type choice of one particular item. For example one may systematically choose the most frequent solution in the best matching set. Another way of combining is to use a weighted (linear) function of solutions with a threshold (for a Boolean dependent variable).

However in complex domains, without adaptation, CBR systems are restricted both in scope and application, Lieber et al. (1996). To reuse cases effectively in new situations, solutions must be adapted to account for differences between the new target and the retrieved cases (Bergmann and Wilke (1996)).

For classification of biological descriptions, the adaptation process may be seen as refining the solutions by exploiting certain contextual information of the new case but also available background knowledge of the domain. This knowledge makes it possible to eliminate certain clearly inapplicable solutions or maximizing the probability of obtaining a correct identification. In the example of the Figure 3, properties of the object "context", like *depth of harvest* or *biotic location* of the specimen, are not relevant for the retrieval process (they are not descriptive properties) but may be used to adapt the solutions. Additional knowledge in the form of classification rules can be defined in the descriptive model to express known facts like "only some particular species can live at a depth of more than 30 meters" or "this specie has never been found in outer slope". This knowledge makes it possible to eliminate *a priori* certain clearly inapplicable solutions.

4 Conclusion

This paper gives an overview that synthesizes different aspects of our research works in artificial intelligence (knowledge representation, processing and validation) developed for Systematics knowledge management. The concrete result of our research is the integrated object-oriented platform $\mathcal{IKBS}$ available on the Web.

Nowadays, expertise in natural sciences is rare and precious. It is therefore urgent to develop tools that will ensure that expertise be collected and safeguarded for transmission to future generations. If this is not done, we will be left only with monographic descriptions and museum collections. The Reengineering of Systematics with $\mathcal{IKBS}$ is our response among others, from a computer science offer viewpoint, to this problem of enhancing scientific databases and museum collections.

We would like to thank the Conseil Régional de la Réunion for supporting the project “Knowledge bases on corals of the Mascarene Archipelago”, and we are grateful to the experts Gérard Faure and Michel Pichon for their active collaboration.

References

AAMODT, A. and PLAZA, E. (1994): Case-based reasoning: foundational issues, methodological variations, and system approaches. *AI Communications, 1(7), 39-59.*

BERGMANN, R. and WILKE, W. (1996): PARIS: Flexible plan adaptation by abstraction and refinement. In: A. Voss, R. Bergmann and B. Bartsch-Sporl (Eds.): *Workshop on Adaptation in Case-Based Reasoning.* Budapest, Hungary.

CONRUYT, N. (1994): *Amélioration de la Robustesse des Systèmes d'Aide à la Description, à la Classification et à la Détermination des Objets Biologiques.* Thèse de doctorat, Université Paris-IX-Dauphine, 1994.

CONRUYT, N. and GROSSER D. (1999): Managing complex knowledge in natural sciences. *3rd International Conference on Case-Based Reasoning, 401–414.*

GROSSER, D. (2002): *Construction Itérative de Bases de Connaissances Descriptives et Classificatoires avec la Plate-forme a Objets IKBS : Application à la Systématique des Coraux des Mascareignes.* Thèse de doctorat, Université de la Réunion.

GROSSER, D., DIATTA, J. and CONRUYT, N. (2000): Improving dissimilarity functions with domain knowledge. *4th European Conference on Principles of Data Mining and Knowledge Discovery*, 409–415.

LE RENARD, J. and CONRUYT, N. (1994): *On the representation of observational data used for classification and identification of natural objects*, LNAI IFCS'93, 308–315.

LIEBER, J. and NAPOLI, A. (1996): Adaptation of synthesis plans in organic chemistry. In: A. Voss, R. Bergmann and B. Bartsch-Sporl (Eds.): *ECAI-Workshop on Adaptation in Case-Based Reasoning.* Budapest, Hungary.

MANAGO, M., ALTHOFF, K.D., AURIOL, E., TRAPHONER, R., WESS, S., CONRUYT, N. and MAURER, F. (1993): Induction and reasoning from cases. In: M.M. Richter, S. Wess, K.-D. Althoff and F. Maurer (Eds.): *1st European Workshop on Case-Based Reasoning.* Kaiserslautern, Germany, 13–318.

MANAGO M. and CONRUYT N. (1992): Using information technology to solve real world problems. *Lecture Notes in Computer Science Subseries.* Springer-Verlag 622, 22–37.

QUINLAN, J.R. (1993): *C4.5: Programs for Machine Learning.* Morgan Kaufmann, Los Altos (CA).

Unsupervised Learning Informational Limit in Case of Sparsely Described Examples

Jean-Gabriel Ganascia and Julien Velcin

Laboratoire d'Informatique de Paris 6, University Pierre and Marie Curie
104, avenue du Président Kennedy, 75016 Paris, France
{*jean-gabriel.ganascia, julien.velcin*}*@lip6.fr*

Abstract. This paper presents a model characterizing unsupervised learning from an information theoretic point of view. Under some hypothesis, it defines a theoretical quality criterion, which corresponds to the informational limit that bounds the learning ability of any clustering algorithm. This quality criterion depends on the information content of the learning set. It is relevant when examples are sparsely described, i.e. when most of the descriptors are missing. This theoretical limit of any unsupervised learning algorithm is then compared to the actual learning quality of different clustering algorithms (EM, COBWEB and PRESS). This empirical comparison is based on the use of artificial data sets, which are randomly degraded. Finally, the paper shows that the results of PRESS, an algorithm specifically designed to learn from sparsely described examples, are very closed to the theoretical upper bound quality.

1 Introduction

Many works have been achieved on unsupervised learning, both in data analysis and in artificial intelligence. Among the numerous applications of unsupervised learning, some are dealing with highly missing descriptions, i.e. with examples that are described on a very tiny part of the description space. A typical case consists in automatically classifying news items. There are many applications of such classification process in sociology or in technology watch. One of these applications we are working on was automatic stereotype learning.

The stereotype notion has been introduced by Walter Lippmann in his famous book "Public Opinion" (1922) to characterize the way partial information is crystallized in our mind. Lippmann says that each of us builds stereotype folders from general and partial information we gather through family discussions, school, newspapers, TV, rumors, etc. Then, these stereotypes are used to form opinions concerning public events about which we have in general no precise knowledge.

According to Lippmann's hypothesis, stereotypes are constructed from meagerly described data, which descriptions are mainly implicit. Therefore, stereotype learning is a concrete illustration of an unsupervised learning from sparsely described data. The aim here is precisely focused in exploring the

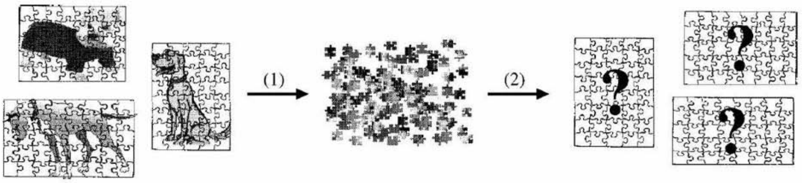


Fig. 1. Stereotype reconstruction.

way such unsupervised learning techniques may automatically reconstruct learning processes like, for instance, stereotype learning from examples.

However, since examples are sparsely described, the learning quality highly depends on the complementarity of examples. In the extreme case of very tiny descriptions, with one or two descriptors, it would be very difficult to learn correlations among descriptors, especially if the total number of descriptors is huge. How would it be possible to build correct classes from highly degraded examples? Our goal here is to determine the information theoretic limitations of such a learning process. In other words, it is to relate unsupervised learning quality to the amount of information present in the examples.

Apart the introduction and the conclusion, the paper contains two main parts: the first describes a theoretical model that defines the upper bound limit of the learning quality while the second provides an empirical evaluation of the theoretical model using artificial data sets.

1.1 Clustering as "Jigsaw Puzzle" reconstruction

The paper focuses on unsupervised learning from sparsely described examples. Our goal is to evaluate the limitations due to the sparseness, i.e. the result of the learning procedures when example descriptions are reduced to very few descriptors. In a word, it is like playing jigsaw puzzles. Let us precise our insight: degraded information can be seen as the pieces of some stereotypes, pieces which are mixed together (see step (1) in fig. 1).

Each fragment of information, i.e. a partially described example, is a piece of this puzzle. Some fragments can be over-duplicated whereas other fragments can be missing. Then, unsupervised learning corresponds to an attempt to automatically reconstruct the original jigsaw puzzles, i.e. the original stereotypes, from this mixing (see step (2) fig. 1). Our goal is to test the ability of clustering techniques to retrieve stereotypes, which is equivalent here to learn from a specific kind of sparsely described examples.

Let us note that this work is highly related to the notion of *informational limit*. This notion was recently studied by Srebro et al. (2005; 2006) in the context of Gaussian mixture learning. It can be viewed as the minimum information amount that permits to learn the complete model having generated the data. Underneath this amount of information, the problem is intractable, whatever the algorithm used. If there is enough information, the clustering becomes an easy task. If you reuse fig. 1, the minimum information corresponds to the minimum number of puzzle pieces that are required in order to be able to reconstruct the three initial jigsaw puzzles. Testing the algorithm learning ability is equivalent to compare two quality scores: the theoretical score corresponding to the informational limit and the practical score that is effectively obtained through the algorithms we use.

1.2 Evaluation on artificial data sets

Tests are done using randomly generated artificial data sets, which are degraded according to predetermined rules. Our goal is to confront the empirical results obtained using degraded artificial data sets with the theoretical upper bound of the learning quality computed within the theoretical model that is proposed here. More precisely, a set of initial descriptions is initially being given. It may also be randomly chosen, according to an initial attribute-value language. These initial descriptions are duplicated δ time and then randomly degraded (η is the proportion of descriptors that are destroyed) in order to obtain a set of partially described descriptions characterized by both the duplication rate δ and the degradation rate η. Once those artificial data sets have been built, the goal is to automatically reconstitute the initial descriptions using different unsupervised learning techniques and to compare the obtained clusters with the initial stereotypes. Undoubtedly, the quality of the rebuilt stereotypes is limited by the information given in the dataset, as we said in the former section. By making both the duplication rates δ of the initial descriptions and the degradation factor η varying, we shall observe the evolution of the learning quality. The main goal of this paper is to propose a mathematical model for the theoretical limit of the learning quality $L(\delta, \eta)$, i.e. the optimal quality of the learning procedure, and to compare it to the actual quality of rebuilt descriptions q using different unsupervised learning algorithms. It must be noted that this model is independent of the algorithms used. The only condition is that the data have to be categorical data.

2 Unsupervised learning model

Our goal consists in estimating the informational limits of the learning quality. More precisely, it is to compute the quantity of information given through the learning sets and then to evaluate the optimal learning quality $L(\delta, \eta)$, i.e. the theoretical ability to retrieve the initial descriptions from sets of degraded examples.

2.1 Artificial data sets

Given a description language $\mathcal{D}$ with n_a attributes, let us introduce a small set of full consistent descriptions, $I = \{i_1, i_2, i_3, \ldots i_{n_i}\}$, which stands, for instance, for the description of a stereotype set, as described in Velcin and Ganascia (2005). Here is the formal definition of what we call a full consistant description set:

Definition 1. A full consistent description set is a set $S = \{s_1, s_2, \ldots s_n, s_\top\}$ of non-redundant descriptions, i.e.:

- $\forall i \in [1, n], s_i \in \mathcal{D}$,
- $\forall d \in \mathcal{D}, (d \in s_i \wedge d \in s_j) \Rightarrow i = j$ (non-redundancy constraint),
- $s_\top$ is the empty-description that covers the examples rejected by the other stereotypes[1].

These $n_i = |I|$ initial descriptions may be randomly generated. They have to be full, with respect to the description language $\mathcal{D}$ (i.e. a value for each attribute, also called a descriptor), and consistent (i.e. no contradiction between the descriptors). Furthermore, descriptions have to be non-redundant, which means that they do not share any common descriptor (see def. 1).

At this step, let us note that two major simplifications are done:

1. The non-redundancy constraint is strong and may be relaxed in future works. However, it permits us not to take into account the factor n_i in the proposed model.
2. The datasets are considered noiseless. As we shall see, it is a point that is worthy of further considerations. But the noise problem seems to be a too complex one for the goal of this paper. That is the reason why it will be studied in future works.

The second step of the artificial data set generation is to duplicate the n_i descriptions δ times (δ is called the duplication rate), e.g. 150 times, making n_d artificial examples. Then, these $n_i \times \delta$ descriptions are arbitrarily degraded: descriptors belonging to each of those duplications are chosen at random to be destroyed. Here, the only parameter is the percentage of degradation η, i.e. the proportion of initial description descriptors that are destroyed. Finally, the generated learning set E contains $n_d = n_i \times \delta$ example descriptions, which altogether correspond to a degraded mixing of the n_i initial descriptions. Since there are n_a attributes, each initial description contains n_a descriptors (i.e. a specific value for each attribute). After description degradation, each example description contains on the average $n_a \times (1 - \eta)$ descriptors. Knowing that initial descriptions are randomly built and degraded, the information content of the artificial data set determines the optimal learning quality. Our purpose here is to evaluate the theoretical ability to retrieve the initial descriptions through duplicated degraded descriptions that constitute the new dataset E. Remember that the examples of this training set are similar to jigsaw puzzle pieces (see fig. 1), which may help to rebuild the overall initial images.

[1] This specific description $s_\top$is handled in the PRESS algorithm, but useless in the cases of the EM and COBWEB algorithms.

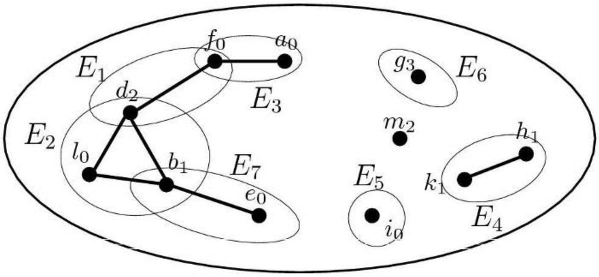


Fig. 2. A stereotype description space.

2.2 Statistical estimation

As previously said, we are interested in rebuilding the initial descriptions. Let us consider the descriptors of one of the initial descriptions as the vertices of an undirected graph whose edges correspond to the simultaneous presence of two descriptors in an example description of E. Fig. 2 presents the description space of the stereotype $s = \{a_0, b_1, d_2, \ldots m_2\}$ associated to the examples E_1 to E_7, that is a subset E' of E. Each of the δ degraded descriptions E_i is a complete subgraph of the initial graph, which is equivalent to a fragment of the initial description. The key point of our reasoning is that the recovered descriptors are those that belong to a subpart of the examples that form a kind of chaining. Consider examples E_1, E_2, E_3, E_7 on the left of fig. 2: they can be merged because they share at least one descriptor in common two by two. Nevertheless, the descriptors i_0, h_1 or g_3 are lost and the learning quality will not be optimal. This idea of "chaining" is exactly related to the notion of cognitive cohesion developed in Velcin and Ganascia (2005).

2.3 Probability to belong to an example description

Let us now consider the probability p that a descriptor belongs at least to one example description. The examples being randomly generated by degrading full initial descriptions containing n_a descriptors each, i.e. by keeping $n_a \times (1 - \eta)$ among n_a descriptors, then p corresponds logically to $(1 - \eta)$. We now propose a first naive estimation $\overline{L}(\delta, \eta)$ of the theoretical limit that corresponds to the greater bound of the quality criterion $L(\delta, \eta)$.

Since the examples are duplicated δ times, we have to compute the probability that a descriptor belonging to an initial description belongs to at least one degraded example among the δ copies of the stereotype. We may use a binomial distribution $\mathcal{B}(\delta; p)$. Let us recall that the binomial distribution $\mathcal{B}(\delta; p)$ gives the probability distribution of obtaining exactly k successes in δ independent Bernoulli trials, where the result of each Bernoulli trial is true with probability p and false with probability $(1 - p)$. The binomial distribution is therefore given by the following formula:

$$P(k) = \binom{k}{\delta} p^k (1-p)^{\delta-k}$$

In particular, if $k = 0$ and $k = 1$:

$$P(0) = \binom{0}{\delta} p^0 (1-p)^{\delta} = (1-p)^{\delta}$$

$$P(1) = \binom{1}{\delta} p^1 (1-p)^{\delta-1} = \delta p (1-p)^{\delta-1}$$

The first proposed estimation function $\overline{L}(\delta, \eta)$ corresponds to the probability that an initial stereotype descriptor belongs to at least one of the learning set examples:

$$\overline{L}(\delta, \eta) = 1 - P(0) = 1 - (1-p)^{\delta} = 1 - \eta^{\delta}$$

It can be seen as the upper bound of the expected quality of the recovered stereotypes.

2.4 Learning quality $L(\delta, \eta)$

This new step consists in refining this first estimation by evaluating the average number of merged examples of E', i.e. the average number of examples sharing two by two at least one descriptor of s (the examples E_1 to E_7 of fig. 2). In other words, the probability calculated in $\overline{L}(\delta, \eta)$ relies on the hypothesis that the descriptors can be found in every example description in E'. However, you have to take into account that a part of E' can be lost and that you have therefore to restrict your research area to the merged examples, i.e. to the examples sharing at least one descriptor with another example of the cluster. Having said this, it follows that the maximum number of merged examples is lower than the number of descriptors shared by two examples descriptions. Moreove, it appears that the number of merged examples is obviously bounded by the number of initial examples. Consequently, the average number of merged examples corresponds either to the minimal number ν of descriptors belonging simultaneously to at least two example descriptions, or to δ if $\delta < \nu$. Since n_a is the number of descriptors belonging to each initial description, ν corresponds to $[1 - P(0) - P(1)]n_a$ which is equivalent to $[1-(1-p)^{\delta-1}(1+\delta p-p)]n_a$. So, we can define a new quality function that is a better estimation for the learning capability, i.e. the probability that the initial description descriptors belong to the learned class:

$$L(\delta, \eta) = 1 - \eta^{\chi}$$

where $\chi = \min(\delta, [1 - (1-p)^{\delta-1}(1 + \delta p - p)] \times n_a)$. Since $p = (1 - \eta)$, this formula can be rewritten:

$$L(\delta, \eta) = 1 - \eta^{\min(\delta, [1-\eta^{\delta-1}(\eta+\delta(1-\eta))] \times n_a)}$$

This evaluation will be confronted with the experimental quality q calculated in the following part. Its notation will be sometimes simplified to L.

3 Evaluation

This second part is dedicated to the evaluation of the theoretical learning quality $L(\delta, \eta)$ using randomly generated data sets and three clustering algorithms: COWEB (Fisher (1987)), EM (Dempster et al. (1977)) and PRESS (Velcin and Ganascia (2005)). The platform WEKA (Witten and Frank (2005)) was used for the experiments with the first two algorithms.

3.1 Quality criterion

In this section, we define a new quality criterion q that compares the set S of clusters extracted with the clustering algorithms and the set I of initial descriptions having generated the data. This criterion is intended to empirically compare the extracted clusters with the original stereotypes from which the training examples where built. In the best case, if the original stereotypes are retrieved, the quality criterion is equal to 1; otherwise it is lower. This criterion was originally proposed in (Velcin, 2005) and relies both on S and I, but also on the dataset E having as parameters the variables n_a, n_i, the duplication rate δ and the degradation rate η. Apart the attribute number n_a, arbitrary fixed to 30, different values of the variables n_i, δ and η are tested in our experiments. The criterion q will be compared with the theoretical quality limit L presented in the previous part.

In order to set this criterion properly, let us consider the function $\mu(s)$ that relates each cluster s in S to a "most appropriate" initial description i in I. A "most appropriate" means an initial description having generated the major part of the examples covered by s. In the following, let us note $E_{|s}$ the subset of E that is covered by s, i.e. the set of examples being more similar to s than to the other clusters of S. Here is the definition of $q(I, S, E)$, whose notation is simplified by $q(S)$:

$$q(S) = \sum_{s \in S} sim(s, \mu(s)) \times \frac{|E_{|s}|}{|E|}$$

where sim is the classical jaccard similarity measure. Note that each cluster is weighted by its size. q is a way to evaluate the quality of the discovered clusters thanks to the original descriptions. It is on purpose that we do not compare the manner examples are clustered, as it is done by the entropy criterion usually used in clustering validity (He et al. (2002)).

3.2 Experiments

Experiments are conducted by varying the degradation and the duplication rates, η and δ for different values of n_i. The results are validated over 20 runs. First, let us consider table 1 showing the different values of $L(\delta, \eta)$, i.e. the expected quality which will be compared to the values of the criterion q presented above.

δ/η	0.7	0.8	0.9
5	0.83193	0.67232	0.227003
7	0.917646	0.790285	0.376967
10	0.971752	0.892626	0.565753
12	0.986159	0.931281	0.659667
15	0.995252	0.964816	0.759586
18	0.998372	0.981986	0.824050
20	0.999202	0.988471	0.853770
30	0.999977	0.998672	0.924240

Table 1. Some $L(\delta, \eta)$ values.

For three values of η, 0.7, 0.8 and 0.9, which corresponds respectively to 70%, 80% and 90% of degradation, we evaluated the empirical learning quality q of the learning process for three clustering algorithms. This quality value is compared to the estimated learning quality L (in dashed line). The duplication factor δ varies from 5 to 30, i.e. from a difficult task with poor information to a (quite) easy task. Figure 3 presents in two graphs the results obtained with COBWEB, EM and PRESS for $\eta = 0.8$ and $\eta = 0.9$ where n_i is equal to 5. The results for $\eta = 0.7$ are not presented here because it is perfectly consistent with the results presented in this paper.

Finally, Figure 4 shows the practical influence of the number n_i of initial stereotypes. The value of n_i does not effectively modify the q score, which confirms the predictions of our model.

As shown by the different diagrams, the theoretical model fits particularly well the PRESS program, while it is not the case with COBWEB and EM. This is not a surprise because PRESS is precisely dedicated to unsupervised learning from meagerly described examples. Its cognitive cohesion constraint is probably a key point to translates the "chaining" effect of sparse data, in its global search strategy. These results need three additional comments:

1. It clearly appears that PRESS actual learning quality is identical to the estimated learning quality, while it is not the case with COBWEB and EM. However, it must be recall that COBWEB and EM are not designed to learn from sparsely described examples. Moreover, the result they provide is a classification procedure and not a collection of cluster descriptions (like in conceptual clustering). The cluster description extraction

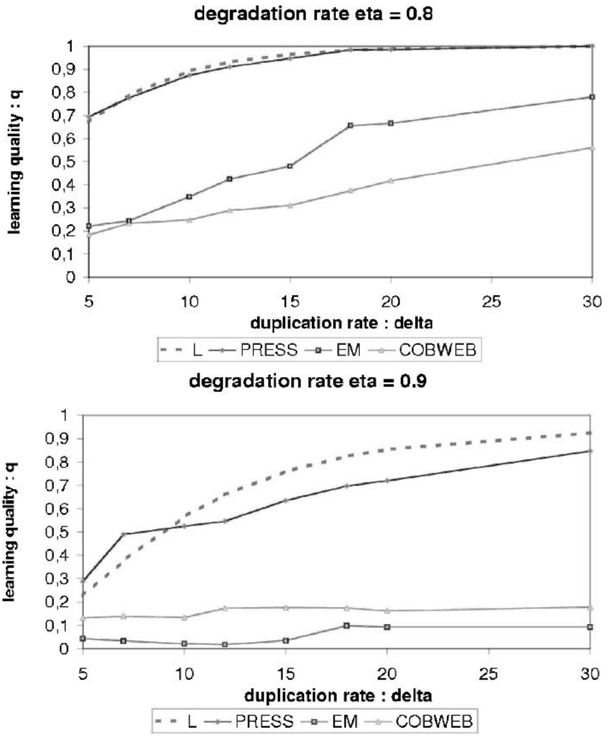


Fig. 3. Comparative results for $\eta = 0.8$ and $\eta = 0.9$ with $n_i = 5$.

from the clusterings they give (see the detailed technique in Velcin and Ganascia (2005)) is done *a posteriori*. Furthermore, the number of classes have to be discovered with a posteriori techniques (see Witten and Frank (2005)).

2. According to the estimated learning quality evaluation, the number of initial stereotypes does not seem to influence the learning quality. It seems to be confirmed by the experiments with the three algorithms, even if this is clearer with PRESS.

3. The PRESS algorithm is based on meta-heuristic optimization techniques. The results show that these techniques, based on a tabu search strategy, lead to a nearly optimal solution.

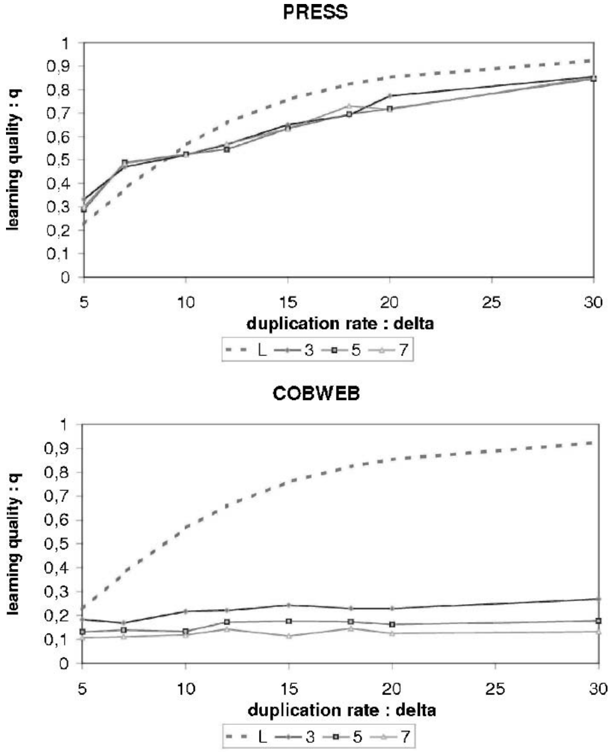


Fig. 4. Comparative results for $n_i = 3$, $n_i = 5$ and $n_i = 7$ with PRESS and COBWEB.

4 Conclusion and perspectives

4.1 Conclusion

The main result of this paper is that unsupervised learning processes can deal with very sparse descriptions and that the number of exemples compensates the sparseness.

A second important point is that a theoretical evaluation of the upper bound learning quality can be computed. We defined here a model to achieve this evaluation. Our experimentations confirm that this model is relevant.

The third result concerns the PRESS algorithm that we developed to learn from sparsely described examples. The experimentations show that PRESS is appropriate in the context of unsupervised learning from sparsely described data. Moreover, the obtained results show that the quality of the results is better than with classical clustering algorithm, COBWEB and EM here. Lastly, it appears that PRESS reaches the theoretical upper bound limit of the learning quality while other clustering algorithms don't. Undoubtedly, these empirical results are evidence for the relevancy of our model and confirm the efficiency of PRESS on sparsely described data.

Furthermore, the paper shows that the unsupervised stereotype extraction process can be modeled with algorithms. It can open many perspectives in social science or in social psychology, where the notion of stereotype plays a crucial role.

4.2 Perspectives

One of our future works will be to extend our model to noisy artificial data sets. Let us recall that, in the presented experiments, the artificial data sets are noiseless. As previously said, we are achieving some experimentations with noisy data, but our goal is not only to test the robustness of learning algorithms; it is to include the noise in the theoretical model. Once such generalization to noisy data will be done, it will be possible to define an evaluation criterion for unsupervised learning algorithms, which will not be based on supervised learning or on a measure, but only on the ability to recover an initial set of descriptions. Another perspective is to relax the non-redundancy constraint of stereotypes in order to consider a more general framework.

References

DEMPSTER, A.P., LAIRD, N.M. and RUBIN, D.B. (1977): Maximum likelihood from incomplete data via the EM algorithm. *Royal Statistical Society B 39 (1), 1-38.*

FISHER, D.H. (1987): Knowledge acquisition via incremental conceptual clustering. *Machine Learning (2), 139-172.*

HE, J., TAN, A.-H., TAN, C.-L. and SUNG, S.-Y. (2002): On Qualitative Evaluation of Clustering Systems. *Information Retrieval and Clustering*, Kluwer Academic Publishers.

LIPPMANN, W. (1922): *Public Opinion.* Wading River, Long Island.

ROSCH, E. (1975): Cognitive representations of semantic categories. *Journal of Experimental Psychology: General, number 104, 192-232.*

SREBRO, N., SHAKHNAROVICH, G. and ROWEIS, S. (2005): When is Clustering Hard? In: *PASCAL Workshop on Statistics and Optimization of Clustering Workshop.*

SREBRO, N., SHAKHNAROVICH, G. and ROWEIS, S. (2006): An investigation of computational and informational limits in Gaussian mixture clustering. In: *23rd International Conference on Machine Learning (ICML)* (preliminary version appeared as UTML-TR-2006-002, February 2006).

VELCIN, J. and GANASCIA, J.-G. (2005): Stereotype extraction with default clustering. In: *Proceedings of the 19th International Joint Conference on Artificial Intelligence.* Edinburgh, Scotland.

WITTEN, I.H. and FRANK, E. (2005): *Data Mining: Practical machine learning tools and techniques.* 2nd Edition, Morgan Kaufmann, San Francisco.

Data Analysis and Operations Research

Wolfgang Gaul

Institut für Entscheidungstheorie und Unternehmensforschung
Universität Karlsruhe (TH), D-76128 Karlsruhe, Germany
wolfgang.gaul@wiwi.uni-karlsruhe.de

Abstract. Data Analysis and Operations Research are two overlapping sciences as there are, e.g., many data problems in which optimization techniques from Operations Research have to be applied to detect best fitting structures (under suitable constraints) in the underlying data. On the other hand, Operations Research is often based on model formulations for which some model parameters might be unknown or even unobservable. In such cases Operations Research problems consist of a data collection and analysis part and an optimization part in which solutions dependent on model parameters (derived from available information via Data Analysis techniques) are calculated.

We give typical examples for research directions where Data Analysis and Operations Research overlap, start with the topic of pyramidal clustering as one of the fields of interest of Edwin Diday, and present methodology how selected problems can be tackled via a combination of techniques from both scientific areas.

1 Introduction

When the data analysis community is the target group for a contribution concerning Data Analysis (DA) and Operations Research (OR) it is not necessary to present a list of topics that describe which kinds of data problems are of interest (see, e.g., the Springer series "Studies in Classification, Data Analysis, and Knowledge Organization" the articles of which cover nearly all aspects in this context). From a methodology-oriented point of view most textbooks in OR deal with topics as Linear/Nonlinear (Convex) Programming, Integer/Combinatorial Programming, Multicriteria Decision Making/Goal Programming and the Analytic Hierarchy Process (AHP), Dynamic Programming, Stochastic Programming, Stochastic Processes' Applications (e.g., Markov Decision Processes, Queueing Theory), Simulation and Sensitivity Analysis, Forecasting as well as Graph Theory and Network Models (see, e.g., the 6th edition of Domschke, Drexl (2004) or the 8th editions of Hillier, Lieberman (2004) or Taha (2007)). Sometimes, questions concerning problem definition, data gathering, and OR model formulation (and dependencies between these tasks) are addressed but a combination of tools from DA and OR is rarely described, explicitly. Against this background the underlying paper emphasizes situations where DA and OR overlap and presents methodology how selected problems can be tackled via a combination of techniques from both scientific areas.

2 Situations for combining data analysis and operations research

2.1 Mixed integer programming for pyramidal clustering

As starting point pyramidal clustering is selected because of the contributions of Edwin Diday to this area (see, e.g., Diday(1986, 1987), Diday, Bertrand (1986)).

The pyramidal generalization of hierarchical classification allows a certain kind of overlapping of clusters (which the hierarchical counterpart does not) where – based on a total order on the set of objects to be clustered – those objects of different clusters that are minimal or maximal with respect to the given order are candidates for overlapping.

Let $I = \{1, ..., m\}$ denote the index set of objects of interest and δ_{ij} given non-negative empirical dissimilarities between pairs (i, j) of objects (a transformation of any measure of association between pairs of objects to non-negative dissimilarities is possible in all realistic empirical situations).

Empirical dissimilarities may not be available for all pairs of objects and don't have to fulfill conditions needed for representation of the objects via, e.g., hierarchies (the ultrametric condition) or pyramids (the pyramidal condition).

The PLSC (Pyramidal Least-Squares Classification) technique is based on the following mixed integer optimization problem:

Denote by $M \subset I^2$ the set of pairs of objects for which the empirical dissimilarities are missing. Choose an initial total order $\preceq$ on I. Describe this total order and the total orders generated in subsequent steps of the solution procedure by a vector $x = (..., x_{ij}, ...)$, with

$$\begin{array}{lll} x_{ij} \in \{0,1\}, & \forall i,j \in I & \\ x_{ii} = 1, & \forall i \in I & \text{(reflexivity)} \\ x_{ij} + x_{ji} = 1, & \forall i,j \in I & \text{(antisymmetry and completeness)} \\ x_{ij} + x_{jk} - x_{ik} \leq 1, & \forall i,j,k \in I & \text{(transitivity)} \end{array} \quad (1)$$

and solve the problem

$$F(d^x) = \sum_{(i,j) \in I^2 - M} (\delta_{ij} - d^x_{ij})^2 = min \quad (2)$$

$$\begin{array}{l} d^x_{ik} \geq max\{d^x_{ij} x_{ij} x_{jk} \ , \ d^x_{jk} x_{ij} x_{jk}\} \ , \qquad \forall i,j,k \in I \\ d^x_{ij} = 0 \Leftrightarrow i = j \ , \ d^x_{ij} = d^x_{ji} \ , \ d^x_{ij} \geq 0 \ , \forall i,j \in I \end{array} \quad (3)$$

The procedure suggested in Gaul, Schader(1994) to tackle (2) under the constraints (1) and (3) can be described as follows:

Select a total order x, set $y = x, F = \infty$. Step 1: Solve (2), (3). If $F(d^x) < F$, update $y = x, d^y = d^x, F = F(d^x)$, and go to Step 2; otherwise got to Step 3. Step 2: Take y and create a new total order x_{new} from y by using the DD (Doubles Décalages) method (the DD method updates an underlying total order, for a description see, e.g., Gaul, Schader (1994), appendix c), set $x = x_{new}$, and go to Step 1. Step 3: Take x and check whether the DD method can be continued. If not, STOP with the results y and d^y; otherwise create a new total order x_{new} from x by using the DD method, set $x = x_{new}$, and go to Step 1.

Remark:

The data problem of subsection 2.1 is to find pyramidal dissimilarities – that allow visualization of clustering structures in the set of underlying objects – which best fit given empirical dissimilarities (perhaps with missing values). For the solution OR methodology based on a mixed integer programming formulation is suggested. The situation can be explained by Tables 1a, b and Figures 1a, b taken from Gaul, Schader (1994).

	1	2	3	4	5	6
1	0					
2	7	0				
3	1	7	0			
4	5	4	5	0		
5	7	2	7	5	0	
6	4	6	4	3	6	0

Table 1a: Dissimilaritiy Data Between Pairs of Objects for $I = \{1, ..., 6\}$.

	3	1	6	4	2	5
3	0					
1	1	0				
6	4	4	0			
4	5	5	3	0		
2	7	7	6	4	0	
5	7	7	6	5	2	0

Table 1b: Rearranged Dissimilarity Data of Table 1a according to the Total Order $3 \prec 1 \prec 6 \prec 4 \prec 2 \prec 5$.

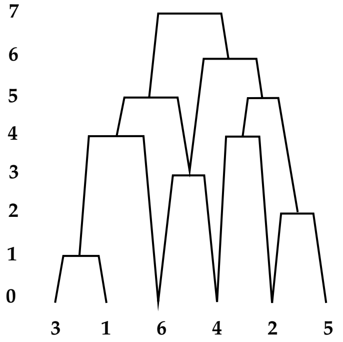


Figure 1a: Pyramidal Classification (Indexed Pyramid) of the Dissimilarity Data of Tables 1a,b.

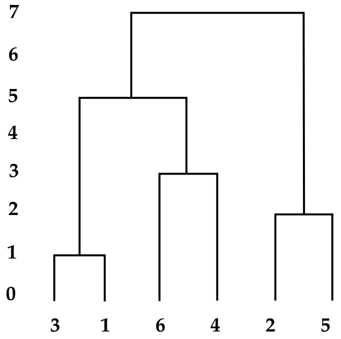


Figure 1b: Hierarchical Classification (Indexed Hierarchy, Complete-Linkage) of the Dissimilarity Data of Tables 1a,b.

Notice that the dissimilarities of Table 1a don't fulfill the ultrametric condition and that the rearrangement of these dissimilarities in Table 1b (according to the total order $3 \prec 1 \prec 6 \prec 4 \prec 2 \prec 5$) can be represented by the (indexed) pyramid of Figure 1a without any loss of information while the (indexed) dendrogram of Figure 1b gives a "poor" fit (of the dissimilarities of Table 1a (or Table 1b)).

2.2 Clustering of relations via combinatorial optimization

Let, again, $I = \{1, ..., m\}$ denote the index set of objects under consideration, and $S = \{1, ..., p\}$ the index set of given binary relations $R_1, ..., R_p$ on I. Different situations can be handled within this framework:

If S is a set of judging subjects, then $R_1, ..., R_p$ could be individual relations which result from paired comparisons with respect to the elements of I or $R_1, ..., R_p$ could be individual total orders or preorders – in other words: rankings – on the elements of I. $R_1, .., R_p$ could also be derived from a mixed data matrix $A = (a_{is}), i \in I, s \in S$, where a_{is} is the value of variable s with respect to object i. Here, S denotes a set of variables used to describe the elements of I. In this case the relations $R_s, s \in S$, are usually defined by

$$iR_sj :\Leftrightarrow a_{is} = a_{js} \text{ for a nominal variable } s,$$
$$iR_sj :\Leftrightarrow a_{is} \leq a_{js} \text{ for an ordinal or a cardinal variable } s,$$

where R_s is an equivalence relation or a complete preorder on I.

If for a relation R one uses the graph G_R with node set $N(G_R) = I$ and arc set $A(G_R) = \{(i, j) : i, j \in I \text{ and } iRj\}$, a well-known distance function for two relations R_1, R_2 is

$$d(R_1, R_2) := |A(G_{R_1}) \cup A(G_{R_2})| - |A(G_{R_1}) \cap A(G_{R_2})|.$$

With $T = \{1, ..., q\}$ as index set of target segments, i.e.,

$$S_t = \{t_1, ..., t_{p_t}\} \subset S, \quad t \in T,$$

and C_t as so-called central relation that best represents the relations contained in S_t one can now solve the problem

$$\sum_{t=1}^{q} \sum_{s \in S_t} d(R_s, C_t) = min \tag{4}$$

subject to constraints that, e.g., $\{S_1, ..., S_q\}$ is a partition of S and $C_1, ..., C_q$ are central relations on I of some specific type(s) (described by constraints similar to (1)).

Remark:

The data problem of subsection 2.2 is to find segments of similar relations and segment-specific central relations – that allow visualization of important relational structures – which best explain the information contained in a set of given relations. For the solution OR methodology based on combinatorial programming is suggested. A more detailed description and examples can be found in Gaul, Schader (1988).

2.3 Optimal positioning

Again, let $I = \{1, ..., m\}$ denote the index set of objects, $S = \{1, ..., p\}$ the index set of judging subjects, and $T = \{1, ..., q\}$ the index set of target segments to which similar subjects are assigned.

In an r-dimensional perceptual space in which the objects are represented by deterministic coordinate vectors $x_i = (x_{i1}, ..., x_{ir})', i \in I$, the target segments are described by stochastic ideal points $v_t = (v_{t1}, ..., v_{tr})', t \in T$, which are assumed to follow multivariate normal distributions $N(\mu_t, \sum_t)$. As it may be difficult for subjects to report about their ideal objects, the idea behind the presented perceptual space model is that subjects from a target segment sample an ideal point from their corresponding segment-specific ideal point distribution and give greater preferences to those objects that are nearer to their ideal points. Consequently, the notation

$$R_{i|I} = \{z \in \mathbb{R}^r : (z - x_i)'(z - x_i) \leq (z - x_j)'(z - x_j) \qquad \forall j \in I\} \tag{5}$$

describes what could be called *preference subset* for object i (which contains all points in $\mathbb{R}^r$ for which i is the closest object with respect to I) and

$$p_{ti|I} = Pr(v_t \in R_{i|I}) \tag{6}$$

gives the probability that subjects from segment t prefer object i to any other object from I.

Using λ_t as a relative size of segment t $\left(\sum\limits_{t=1}^{q} \lambda_t = 1\right)$

$$p_{i|I} = \sum_{t=1}^{q} \lambda_t p_{ti/I} \tag{7}$$

is the so-called overall share of choices for object i.

For $|I| = 2$ a closed form solution of the probability $p_{ti|I}$ of (6) is

$$p_{ti|\{i,j\}} = Pr(v_t \in R_{i|\{i,j\}}) = \Phi(\frac{x_j'x_j - x_i'x_i - 2(x_j - x_i)'\mu_t}{4(x_j - x_i)' \sum_t (x_j - x_i)}),$$

where Φ denotes the standard normal distribution,

for $|I| > 2$ an analytical solution of the probability expression (6) is not known (see, e.g., Baier, Gaul (1999), appendix, for hypercube approximation).

With $\Theta_t = (\mu_{t1}, ..., \mu_{tr}, \sigma_{t11}, \sigma_{t12}, ..., \sigma_{trr})'$ as parameter vector for $N(\mu_t, \Sigma_t)$, $t \in T$, and $\Theta = (\Theta_1', ..., \Theta_q')'$ as overall parameter vector the data collection and parameter estimation part of the optimal positioning problem can be described as follows:

Paired comparisons $Y = (y_{sij}), s \in S, i, j \in I$, with

$$y_{sij} = \begin{cases} 1 & , \text{ if subject } s \text{ prefers object } i \text{ to object } j, \\ 0 & , \text{ otherwise,} \end{cases}$$

are collected (Note that this notation allows for missing values in the data.). As segment-specific model parameters Θ_t are needed, an additional segmentation matrix $H = (h_{ts}), t \in T, s \in S$, with

$$h_{ts} = \begin{cases} 1 & , \text{ if subject } s \text{ belongs to segment } t, \\ 0 & , \text{ otherwise,} \end{cases}$$

is introduced (from which one gets the relative segment sizes, $\lambda_t = \sum_{s=1}^{p} h_{ts}/p$).

The parameter estimation part will not be explained in detail. A simultaneous technique for jointly determining Θ and H (based on the classification maximum likelihood method which incorporates a quasi-Newton procedure) is used. Notice that the negative log-likelihood function

$$\begin{aligned} -lnL(\Theta, H|Y) &= -\sum_{t=1}^{q}\sum_{i=1}^{m}\sum_{j\in I\setminus\{i\}}\Big(\sum_{s=1}^{p} h_{ts}y_{sij}\Big)ln(p_{ti|\{i,j\}}) \\ &= -\sum_{s=1}^{p}\sum_{t=1}^{q}h_{ts}\Big(\sum_{i=1}^{m}\sum_{j\in I\setminus\{i\}} y_{sij}ln(p_{ti|\{i,j\}})\Big) \\ &= -\sum_{s=1}^{p}\sum_{t=1}^{q}h_{ts}L_{ts}(\Theta_t|Y) \end{aligned} \tag{8}$$

allows simplifications for given H. The determination of H is improved by allocating subjects to segments in such a way that (8) is minimized.

Based on the estimated parameters overall shares of choices for the given objects (see (7)) can be predicted.

For optimal positioning of a new object assume that I is enlarged to $I_0 = I \cup \{0\}$ where 0 describes the additional alternative.

If the new object is positioned at $x_0 = (x_{01}, ..., x_{0r})'$ one gets the preference subset

$$R_{0|I_0}(x_o) = \{z \in \mathbb{R}^r : (z - x_0)'(z - x_0) \leq (z - x_j)'(z - x_j) \quad \forall j \in I_0\}$$

and

$$p_{0|I_0}(x_o) = \sum_{t=1}^{q} \lambda_t p_{t0|I_0}(x_o) \tag{9}$$

as overall share of choices for the new object (dependent on x_0).

Now, optimal positioning options for the new object can be obtained through maximizing (9) by one of the adequate positioning techniques listed in Baier, Gaul (1999), Table 3, which gives a quite complete overview concerning references up to the end of the nineties of the last century.

Remark:

The data problem of subsection 2.3 is to find segment-specific stochastic ideal points described by multivariate normal distributions – that allow visualization of the underlying choice situation in corresponding perceptual spaces

– which best explain the preferences of segments of subjects contained in given individual paired comparisons. OR methodology (e.g., a quasi-Newton procedure) is already incorporated in the classification maximum likelihood method for the estimation of the model parameters. For the generation of positioning options of a new object in the given perceptual space a standard hill-climbing algorithm of nonlinear programming was applied. A more detailed description with Monte Carlo experiment and application can be found in Baier, Gaul(1999).

2.4 Random variables in operations research models

This time, the starting point is an OR model – a linear program, say – in which some model parameters have to be viewed as random variables, which is the basic assumption of stochastic programming (see, e.g., Kall (1979) for an early and Kall, Wallace (1994) for a more recent textbook concerning this OR field). Let

$$\begin{aligned} c'x &= min \\ Ax &= b \\ x &\geq 0 \end{aligned} \tag{10}$$

be a standard linear program with known $m \times n-$matrix $A = (a_{ij}), b = (..., b_i, ...)' \in \mathbb{R}^m$, and $c = (..., c_j, ...)' \in \mathbb{R}^n$ for which the decisions $\{x : Ax = b, x \geq 0\}$ form a closed convex set. Assume there exist additional constraints

$$Bx = d \tag{11}$$

with $\widetilde{m} \times n$-matrix $B = (b_{ij})$ and $d = (..., d_i, ...)' \in \mathbb{R}^{\widetilde{m}}$ where – for simplicity – only d is assumed to be a random vector on a probability space $(\Omega, \mathfrak{S}, Pr)$ for which the expectation E_d exists. If the realization $d(\omega), \omega \in \Omega$, is known before the decision x has to be calculated, the problem is "easy". If x has to be determined before the realization of d is known

$$Q(x, d) = \inf\{cc'_+ y_+ + cc'_- y_- : y_+ - y_- = d - Bx, y_+ \geq 0, y_- \geq 0\}$$

describes a possibility for compensation (a so-called simple recourse compensation) with compensation costs $cc_+, cc_- \in \mathbb{R}^{\widetilde{m}}$. If $cc_+ - cc_- \geq 0$ the so-called two-stage stochastic programming problem with simple recourse

$$\begin{aligned} c'x + E_d[Q(x, d)] &= \min \\ Ax &= b \\ x &\geq 0 \end{aligned} \tag{12}$$

solves (10) and (11) in the sense that x is selected in such a way that nonconformity of Bx with d in (11) is optimally compensated.

Now, from the data problem point of view the probability distribution of d is of importance for the solution of (12). Here, it is assumed that the components d_i have finite discrete probability distributions (or that the corresponding distributions are approximated by finite discrete probability distributions), i.e.,

$$p_{ik} = Pr(d_i = d_{ik}) \quad , \quad k = 1, ..., r_i \quad , \quad i = 1, ..., \widetilde{m} \quad ,$$

(with lower (upper) bounds d_{i0} (d_{ir_i+1}) with $p_{i0} = 0$ $(p_{ir_i+1} = 0)$) are taken into consideration.

For a selected realization $d_{k^*} = (d_{1k_1^*}, ..., d_{\widetilde{m}\; k^*_{\widetilde{m}}})'$ of the vector d one solves the following dual problems

PRIMAL (k^*)

$$\begin{array}{lllll}
\displaystyle\sum_{j=1}^{n}\Big(c_j + \sum_{i=1}^{\widetilde{m}}\big(-(cc_+)_i + \big((cc_+)_i + (cc_-)_i\big)\sum_{k=1}^{k_i^*} p_{ik}\big)b_{ij}\Big)x_j = min & & & & \\
\displaystyle\sum_{j=1}^{n} a_{ij}x_j & & = b_i & , & i = 1, ..., m \\
\displaystyle\sum_{j=1}^{n} b_{ij}x_j \quad - s_{1i} & & = d_{ik_i^*} & , & i = 1, ..., \widetilde{m} \\
\displaystyle -\sum_{j=1}^{n} b_{ij}x_j & - s_{2i} & = -d_{i(k_i^*+1)} & , & i = 1, ..., \widetilde{m} \\
\qquad x_{ij} \geq 0, s_{1i} \geq 0, s_{2i} \geq 0 & & & &
\end{array}$$

DUAL (k^*)

$$\begin{array}{llllll}
\displaystyle\sum_{i=1}^{m} b_i u_i & \displaystyle +\sum_{i=1}^{\widetilde{m}} d_{ik_i^*} v_{1i} & \displaystyle -\sum_{i=1}^{\widetilde{m}} d_{i(k_i^*+1)} v_{2i} = max & & & \\
\displaystyle\sum_{i=1}^{m} u_i a_{ij} & \displaystyle +\sum_{i=1}^{\widetilde{m}} v_{1i} b_{ij} & \displaystyle -\sum_{i=1}^{\widetilde{m}} v_{2i} b_{ij} & \leq \mathfrak{c}_j & , & j = 1, ..., n \\
& \quad v_{1i} \geq 0 \quad , & v_{2i} \geq 0 & & &
\end{array}$$

$$\text{with } \mathfrak{c}_j = c_j + \sum_{i=1}^{\widetilde{m}}\big(-(cc_+)_i + \big((cc_+)_i + (cc_-)_i\big)\sum_{k=1}^{k_i^*} p_{ik}\big)b_{ij} \quad .$$

Notice that PRIMAL (k^*) and DUAL (k^*) are optimization problems on the grid given by the realizations of the vector d. For a selected k^* optimization is performed between d_{k^*} and d_{k^*+e}, $e = (1, ..., 1)'$.

If $\widetilde{x}$, $\widetilde{s}_1$, $\widetilde{s}_2$ (for PRIMAL (k^*)) and $\widetilde{u}$, $\widetilde{v}_1$, $\widetilde{v}_2$ (for DUAL (k^*)) are complementary optimal solutions and

$$\widetilde{v_{1i}} \leq \big((cc_+)_i + (cc_-)_i\big)p_{i_{k_i^*}} \quad , \quad i = 1, ..., \widetilde{m} \tag{13}$$

$$\widetilde{v_{2i}} \leq \big((cc_+)_i + (cc_-)_i\big)p_{i_{k_i^*+1}} \quad , \quad i = 1, ..., \widetilde{m} \tag{14}$$

then $\widetilde{x}$ is optimal for the two-stage stochastic programming problem with simple recourse (12), otherwise update

$$k_i^{*(new)} = k_i^{*(old)} + \begin{cases} (-1)\ , & \text{if (13) is violated,} \\ \quad 1\ , & \text{if (14) is violated,} \quad i = 1'..., \widetilde{m}\ , \\ \quad 0\ , & \text{otherwise,} \end{cases} \tag{15}$$

and solve PRIMAL $(k^{*(new)})$, DUAL $(k^{*(new)})$. Under reasonable assumptions an optimal solution is obtained after a finite number of iterations.

Remark:

A stochastic programming problem is solved by a finite sequence of "easier to handle" non-stochastic PRIMAL/DUAL problems. It is, of course, advantageous, when the PRIMAL/DUAL problems are of a special form for which fast solutions are already available.

Notice, that sometimes the dual problem of an initial linear program is of the form described by (10), (11). For an application of the described OR methodology to project scheduling via stochastic programming (in which project activity times are random variables) see Cleef, Gaul (1982) where the "easier to handle" PRIMAL/DUAL problems are based on network models (e.g., solving minimal cost flow problems by the "out-of-kilter" algorithm).

2.5 Web mining and recommender systems

Nowadays, contributions concerning DA and OR have to cope with web mining because the web as one of the fastest growing sources of information is a challenge for data analysts. Here, a recent reference is Gaul (2006) (see also Gaul (2004)) in which certain topics (concerning web data, data analysis techniques, and web mining applications) are presented that will not be repeated in this paper. However, at least recommender systems, e.g., for clickstream analysis (see, e.g., Gaul, Schmidt-Thieme (2000, 2002)), should be mentioned, explicitly, as – in the narrow sense – these systems tackle data problems. Here, data (input) has to be analysed in such a way by DA and/or OR techniques that recommendations (output) for target segments can be provided.

3 Conclusion

DA (Data Analysis) and OR (Operations Research) techniques are needed in quite a number of situations in which on the basis of underlying data (sometimes with missing values) "optimal" solutions for target groups have to be calculated. Thus, it seems to be worth while to consider research directions where DA and OR overlap. In this paper, a constrained optimization formulation for pyramidal clustering was the starting point for a collection of examples (in which, e.g., combinatorial programming, optimization techniques to calculate maximum likelihood estimates, algorithms for optimal positioning, and stochastic programming were applied) that describe situations where a combination of DA and OR has to be used to solve the underlying problems. As KIT (Karlsruhe Institute of Technology, a merger of the Forschungszentrum Karlsruhe and the Universität Karlsruhe) was elected as one of the best German universities in 2006, new courses will be established in 2007 and one of it is "Data Analysis and Operations Research". Here, hints and remarks concerning additional examples, situations, and solutions are welcome.

References

BAIER, D. and GAUL, W. (1999): Optimal product positioning based on paired comparisons data. *Journal of Econometrics, 89, 365-392.*

CLEEF, H.J. and GAUL, W. (1982): Project scheduling via stochastic programming. *Mathematische Operationsforschung und Statistik, Ser. Optimization, 13, 449-468.*

DIDAY, E. (1986): Orders and overlapping clusters by pyramids. In: J. De Leeuw, W. Heiser, J. Meulman and F. Critchley (Eds.): *Multidimensional Data Analysis.* DSWO, Leiden, 201-234.

DIDAY, E. (1987): *Orders and Overlapping Clusters by Pyramids.* Rapport de Recherche No 730. INRIA, Paris.

DIDAY, E. and BERTRAND, P. (1986): An extension of hierarchical clustering: the pyramidal representation. In: E.S. Gelsema and L.N. Kanal (Eds.): *Pattern Recognition in Practice II.* North-Holland, 411-424.

DOMSCHKE, W. and DREXL, A. (2004): *Einführung in das Operations Research (6. Auflage).* Springer.

GAUL, W. (2004): Market research and the rise of the web: the challenge. In: Y. (Jerry) Wind and P.E. Green (Eds.): *Market Research and Modeling: Progress and Prospects: A Tribute to P.E. Green. International Series in Quantitative Marketing.* Kluwer, 103-113.

GAUL, W. (2006): Challenges concerning web data mining (Invited Paper). In: A. Rizzi and M. Vichi (Eds.): *COMPSTAT 2006.* Physica, 403-416.

GAUL, W. and SCHADER, M. (1988): Clusterwise aggregation of relations. *Applied Stochastic Models and Data Analysis, 4, 273-282.*

GAUL, W. and SCHADER, M. (1994): Pyramidal classification based on incomplete dissimilarity data. *Journal of Classification, 11, 171-193.*

GAUL, W. and SCHMIDT-THIEME, L. (2000): Frequent generalized subsequences – a problem from web mining. In: W. Gaul, O. Opitz, and M. Schader (Eds.): *Data Analysis: Scientific Modeling and Practical Application. Studies in Classification, Data Analysis, and Knowledge Organization.* Springer, 429-445.

GAUL, W. and SCHMIDT-THIEME, L. (2002): Recommender systems based on user navigational behavior in the internet. *Behaviormetrika, 29, 1-29.*

HILLIER, F.S. and LIEBERMAN, G.J. (2004): *Introduction to Operations Research (8^{th} Edition).* McGraw-Hill.

KALL, P. (1979): *Stochastic Linear Programming.* Springer.

KALL, P. and WALLACE, S. (1994): *Stochastic Programming.* Wiley.

TAHA, H.A.(2007): *Operations Research: An Introduction (8^{th} Edition).* Prentice Hall.

Reduction of Redundant Rules in Statistical Implicative Analysis

Régis Gras and Pascale Kuntz

Équipe COnnaissances & Décision
Laboratoire d'Informatique de Nantes Atlantique, FRE CNRS 2729
Site École Polytechnique de l'Université de Nantes, La Chantrerie, BP 50609
44306 Nantes cedex 3, France
regisgra@club-internet.fr, pascale.kuntz@univ-nantes.fr

Abstract. Quasi-implications, also called association rules in data mining, have become the major concept to represent implicative trends between itemset patterns. To make their interpretation easier, two problems have become crucial: filtering the most interestingness rules and structuring them to highlight their relationships. In this paper, we put ourselves in the Statistical Implicative Analysis framework, and we propose a new methodology for reducing rule sets by detecting redundant rules. We define two new measures based on the Shannon's entropy and the Gini's coefficient.

1 Introduction

"If a question is more complex than another, then each pupil who succeeds in the first one should also succeed in the second one". Every teacher knows that this situation shows exceptions without throwing back the general tendencies. The evaluation and the structuration of such implicative relationships between didactic situations are the generic problems at the origin of the development of the Statistical Implicative Analysis (SIA, Gras (1979)). These problems, which have also drawn a great attention from psychologists interested in tests of ability (e.g. Loevinger (1947), Bernard and Poitrenaud (1999)), have known a significant renewed interest in the last decade in data mining. Indeed, quasi-implications, also called association rules in this field, have become the major concept in data mining to represent implicative trends between itemset patterns. In data mining, the paradigmatic framework is the so-called basket analysis where a quasi-implication $T_i \rightarrow T_j$ means that if a transaction contains a set of items T_i than it is likely to contain a set of items T_j too. For simplicity's sake, in the following, let us call "rule" a quasi-implication.

In data mining, rules are computed on large size databases. And, because of this scale change, two problems have become crucial: filtering the most interestingness rules and structuring them to highlight their relationships and make their interpretation easier.

From the seminal works of Agrawal *et al.* (1993) numerous algorithms have been proposed to mine set of relevant rules. However it is now well-known that they produce large sets which remain tricky to interpret. To overcome this difficulty three different ways have been explored. The first one consists in pruning rule sets by defining interestingness measures (Hilderman and Hamilton (1999)) or pre-defined patterns (Klementtinen *et al.* (1994)). The second one structures rule sets *via* clustering approaches (Lent *et al.* (1997), Vaillant (2006)) or graphical representations (Kuntz *et al.* (2000)). The third one considers the user as a full component of the discovery process which guides the computing heuristics *via* well-adapted interactive interfaces (Blanchard *et al.* (2007))

In this paper, we put ourselves in the SIA framework, and we propose a new methodology for reducing rule sets by detecting redundant rules. One of the major interest of SIA is to combine the two first approaches previously quoted in a coherent framework. A measure of interestingness, the implicative intensity (Gras (1979), Gras *et al.* (1996, 2001)), has been defined to evaluate the rule "surprisingness" *i.e.* the improbable small number of counter-examples in comparison with the data number. And, two modes of structuration have been developed: the implicative graph (Gras *et al.* (1996)) and the directed hierarchy (Gras and Kuntz (2005)). The directed hierarchy completes the graph model. It is composed of R-rules (rules of rules) which are rule extensions: their premisses and their conclusions can be rules themselves.

The work presented in this paper is a first attempt to characterize redundant rules and redundant R-rules in the SIA framework. We have defined two new measures based on the Shannon's entropy and the Gini's coefficient. For each rule pair, we evaluate the information quantity brought by one of the rules when the realization of the other is known.

The rest of the paper is organized as follows. Section 2 briefly recalls the main results of SIA. Section 3 analyzes the information brought by one rule on the other and measures this information by an adaptation of the conditional Shannon's entropy. In section 4, we propose a different measure based on the Gini's coefficient.

2 The SIA framework

Throughout this paper, we consider a set I of n individuals described by a finite set $A = \{a, b, c, ...\}$ of m attributes.

We first recall the definition of the implicative intensity proposed by Gras (1979) for simple rules of the form $a \rightarrow b$. Then, we present the generalization to R-rules.

2.1 The implicative intensity

Let us denote by $A \subset I$ the subset of individuals for which a is present, $\overline{A}$ its complementary in I and $n(A)$ the cardinal of A. To accept or reject the general trend to have b when a is present, it is quite common to consider the number $n_{a \wedge \overline{b}} = card\left(A \cap \overline{B}\right)$ of counter-examples of the rule $a \rightarrow b$. However, to quantify the "surprisingness" of this rule, this must be relativized according to n,n_a and n_b. Intuitively, it is all the more surprising to discover that a rule has a small number of counter-examples as the database is large.

Hence, the objective of the implicative intensity is to express the unlikelihood of $n_{a \wedge \overline{b}}$ in I. We compare the observed number of counter-examples $n_{a \wedge \overline{b}}$ with the expected number of counter-examples for an independent hypothesis. Let us assume that we randomly draw two subsets U and V in I with respectively n_a and n_b elements. We denote by $X_{a \wedge \overline{b}} = card\left(U \cap \overline{V}\right)$ the random variable associated with the number of counter-examples in this random model.

The distribution of $X_{a \wedge \overline{b}}$ depends on the random drawing pattern (Gras *et al.* (1996)). In practice, we consider a normal distribution; let $\widetilde{X}_{a \wedge \overline{b}}$ be the standardized random variable and $\widetilde{n}_{a \wedge \overline{b}}$ be the reduced-centered value of $n_{a \wedge \overline{b}}$.

Definition 1. The *implicative intensity* of the rule $a \rightarrow b$ is defined by

$$\varphi\left(a, b\right) = 1 - Pr\left(\widetilde{X}_{a \wedge \overline{b}} \leq \widetilde{n}_{a \wedge \overline{b}}\right)$$

if $n_b \neq n$; otherwise $\varphi\left(a, b\right) = 0$. The rule $a \rightarrow b$ is retained for a certain threshold α if $\varphi\left(a, b\right) \geq 1 - \alpha$.

Throughout this paper, we illustrate the numerical values obtained by the different measures on a database given in appendix A. We consider 5 binary variables v_1, v_2, ..., $v_5$observed on a set of 30individuals. The calculation of the implication intensity of the rule $v_3 \rightarrow v_1$ requires the number of counter-examples : $n_{v_3 \wedge \overline{v_1}} = 1$. Hence, by using the Poisson's law of parameter $21. \left(30 - 24\right) / 30 = 4.2$ we obtain $\varphi\left(v_3, v_1\right) = 1 - Pr\left(\widetilde{X}_{v_3 \wedge \overline{v_1}} \leq \widetilde{n}_{v_3 \wedge \overline{v_1}}\right) = 1 - 0.06 = 0.94$. Similarly, for the rule $v_5 \rightarrow v_4$ the implication intensity is equal to $\varphi\left(v_5, v_4\right) = 0.92$.

2.2 The R-rules

Roughly speaking, the R-rules are an extension of the classical binary rules $a \rightarrow b$ to rules of rules which may be complex themselves. To guide the intuition a parallel can be drawn from the proof theory with the logical implication: $(X \Rightarrow Y) \Rightarrow (Z \Rightarrow W)$ describes an implication between the two theorems $X \Rightarrow Y$ and $Z \Rightarrow W$ previously established.

Definition 2. The R-rule of degree 0 are attributes of A. The R-rules of degree 1 are the simple binary rules of the form $a \rightarrow b$. A R-rule of degree i,

$1 < i \leq p$, is a rule $R' \to R''$ between two R-rules R' and R'' whose respective degrees satisfy $j + k = i - 1$.

The R-rules allow to express different levels of abstraction: *(i)* descriptions (conjunction of R-rules of degree 0), *(ii)* implications between descriptors (R-rules of degree 1), *(iii)* implications between implications (some R-rules of degree greater than 1).

An extension of the implicative intensity, called *cohesion*, has been proposed to discover the R-rules $R' \to R''$ with a strong implicative relationship between the components of R' and those of R''. Intuitively, for a R-rule $(a \to b) \to (c \to d)$, the cohesion takes simultaneously into account the implicative strength of $a \to b$ and $c \to d$ but also of $a \to c$, $a \to d$, $b \to c$ and $b \to d$. We refer to Gras and Kuntz (2005) for an exact definition. Moreover, we have proposed a structuration of the R-rules by a directed hierarchy $\overrightarrow{H}$ which is an adaptation of the classical hierarchy: the nodes of $\overrightarrow{H}$ are R-rules, the intersection of two R-rules of $\overrightarrow{H}$ is either empty or equal to one of the R-rules, and for each R-rule of $\overrightarrow{H}$ of non null degree there exists a unique decomposition into two R-rules of $\overrightarrow{H}$ of lower degree.

3 Reduced and conditional Shannon's entropy

In this section we restrict ourselves to simple rules but our reasoning remains correct for R-rules. Our objective is to characterize redundant rules in rule subsets defined by a significant implicative intensity: $S_\varphi = \{R; \varphi(R) \geq 1 - \alpha\}$, where $\alpha \in [0, 1]$ is a fixed threshold.

Given any individual $i \in I$, we can associate with a rule $R \in S_\varphi$ the random variable X_R s.t. $X_R = 1$ if R is true for i and $X_R = 0$ otherwise. Let $I(R)$ be the subset of individuals with R true. The realization frequency of X_R in I is defined by $p_R = card(I(R))/n$. Let us remark that p_R is different from the confidence (estimation of the conditional probability) classically used in rule mining. For instance, the confidence of the rule $v_3 \to v_1$ is equal to $20/21 = 0.95$ whereas the rule frequency is equal to $p_{v_3 \to v_1} = 1 - 1/30 = 0.97$.

The quantity of information contained in the realization of the rule R can be measured by the classical Shannon's entropy $H(R)$:$H(R) = -p_R \log_2 p_R - (1 - p_R)\log_2(1 - p_R)$. $H(R)$ is the average information associated with the knowledge of the result of the random experiment which realizes R.

In data mining we are interested in cases where counter-examples are rare considering the database size; they correspond to great values of p_R. Consequently, we here restrict our analysis to rules R with $p_R \geq 0.5$. Theoretically, we could obviously consider a higher threshold. However, this does not avoid the discontinuity of the entropy in the vicinity of the threshold. The threshold 0.5 coincides with the first value for which the Shannon's entropy is equal to 0.

Definition 3. The reduced entropy of the rule R is

$$H\left(R\right)=-p_R\log_2 p_R-\left(1-p_R\right)\log_2\left(1-p_R\right)\ \text{ if } p_R\geq 0.5$$

and $H\left(R\right)=0$ otherwise. If $p_R=1$ then $H\left(R\right)=0$: the uncertainty is null as the rule R is certain. If $p_R=0.5$ than $H\left(R\right)=1$: the uncertainty is maximal.

Let us now consider two rules $R\in S_\varphi$ and $S\in S_\varphi$ and their associated variables X_R and X_S with their respective frequencies p_R and p_S. The frequencies of their negations are $p_{\overline{R}}=1-p_R$ and $p_{\overline{S}}=1-p_S$. Let $I\left(RS\right)$ be the individual subset in I with R and S simultaneously true, and $I\left(R\overline{S}\right)$ (resp. $I\left(\overline{R}S\right)$) be the individual subset with R true (resp. false) and S false (resp. true) . We define $p_{RS}=card\left(I\left(RS\right)\right)/n$, $p_{\overline{RS}}=1-p_{RS}$, $p_{R\overline{S}}=card\left(I\left(R\overline{S}\right)\right)/n$, $p_{\overline{R}S}=card\left(I\left(\overline{R}S\right)\right)/n$.

With the same argument as previously (definition 3), we can define the reduced conditional entropy $H\left(S\mid R\right)$ to measure the information growth in X_S when X_R is known.

Definition 4. Given the rule R, the reduced conditional entropy of the rule S is

$$H\left(S\mid R\right)=-p_{RS}\log_2\frac{p_{RS}}{p_R}-p_{R\overline{S}}\log_2\frac{p_{R\overline{S}}}{p_R}-p_{\overline{R}S}\log_2\frac{p_{\overline{R}S}}{p_{\overline{R}}}-p_{\overline{RS}}\log_2\frac{p_{\overline{RS}}}{p_{\overline{R}}}$$

if $p\geq 0.5$; and $H\left(S\mid R\right)=0$ otherwise.

If $H\left(S\mid R\right)=0$ then X_R brings no information on X_S. The difference $H\left(S\right)-H\left(S\mid R\right)$ is the information quantity on X_S contained in X_R when $p_{RS}\geq 0.5$.

To make the analysis of a rule set easier, we look for a threshold value which allows to automatically prune the redundant rules. We resort to a normalization of the different reduced entropies.

We set $h\left(R\right)=H\left(R\right)/\log_2 N$ where N is the number of values taken by the random variable X_R associated with a rule R. Here, $X_R=0$ or 1 and $N=2$. Let us recall that $H\left(R\right)\leq\log_2 N$ (Roubine, 70). The equality holds for the maximal incertitude ($p_R=\left(1-p_R\right)=0.5$). Then, $h\left(R\right)\leq 1$. Moreover, if $h\left(S\right)$ is close to 0, the experiment associated with X_R is superfluous: one of the two probabilities p_R or $p_{\overline{R}}$ is significantly greater than the other and we are almost sure of the issue. Similarly we define $h\left(S\mid R\right)=H\left(S\mid R\right)/\log_2 N$.

Definition 5. When $R\in S_\varphi$ is known, the rule $S\in S_\varphi$ is ε-superfluous if $r\left(S\mid R\right)=1-h\left(S\mid R\right)$ is greater than $1-\varepsilon$, where $\varepsilon\in\left[0,1\right]$.

By construction, $r\left(S\mid R\right)\in\left[0,1\right]$. With a value of ε small enough, the user can remove the superfluous rules in $S\varphi$.

It is easy to show that $H\left(S\right)-H\left(S\mid R\right)=H\left(R\right)-H\left(R\mid S\right)$: when X_R is known, the information growth in X_S is equivalent to the information growth in X_R when X_S is known. However, $r\left(R\right)$ and $r\left(S\right)$ are not necessarily equal. Hence, when comparing two rules for pruning, we eliminate the rule with the greatest r.

To compare the respective informational gains we introduce the ratio

$$G\left(S\mid R\right)=\frac{H\left(S\right)-H\left(S\mid R\right)}{H\left(S\right)}\text{ if }H\left(S\right)\neq 0\text{ and }G\left(S\mid R\right)=0\text{ otherwise}$$

Definition 5bis. When the rule $R\in S_{\varphi}$ is known, the rule $S\in S_{\varphi}$ is ε-redundant if $G\left(S\mid R\right)\geq 1-\varepsilon$.

Property 1. $G\left(S\mid R\right)\in[0,1]$.

As $H\left(S\mid R\right)\leq H\left(S\right)$ for any R, $G\left(S\mid R\right)\leq 1$. If $G\left(S\mid R\right)=1$ than $H\left(S\mid R\right)=0$. In this case the information quantity on X_S contained in X_R is maximal. And, X_R and X_S are probably closely linked; when R is already given, S is redundant. If $G\left(S\mid R\right)=0$ then knowing R brings no additional information on S; X_R and X_S are probably independent.

If X_R and X_S are independent then $H\left(S\mid R\right)=H\left(S\right)$ and $G\left(S\mid R\right)=0$. But the contrapositive is not necessarily true; nevertheless, in this case, the independence can be suspected.

If $H\left(S\right)=0$ then $H\left(S\mid R\right)=0$ and it is easy to prove that $H\left(R\mid S\right)=0$. This remark justifies the continuity of $G\left(S\mid R\right)$ on 0.

For illustration, let us compute the different coefficients for the rules $R=(v_3\rightarrow v_1)$ and $S=(v_5\rightarrow v_4)$. We obtain

$$H\left(S\right)=-\frac{1}{\ln 2}\left(\frac{29}{30}\ln\frac{29}{30}+\frac{1}{30}\ln\frac{1}{30}\right)=0.211=H\left(R\right)$$

and,

$$H\left(S\mid R\right)=\frac{1}{\ln 2}\left(\frac{28}{30}\ln\frac{28}{29}+\frac{1}{30}\ln\frac{1}{29}+\frac{1}{30}\ln 1\right)=0.209=H\left(R\mid S\right)$$

and $h\left(S\mid R\right)=0.209/\left(\ln 30/\ln 2\right)=0.043$. With the threshold $\varepsilon=0.05$ the rule S is ε-redundant. Moreover, as $G\left(S\mid R\right)=0.00948$, S is also ε-redundant when R is known. In this example, we do not distinguish the respective roles of S and R as they have a similar conditional entropy. But, in order to discover a possible implicative relationship between R and S we compute the cohesions $c\left(R\right)=0.94$ and $c\left(S\right)=0.877$ and the rule implication $\Psi\left(R\rightarrow S\right)=0.07$ and $\Psi\left(S\rightarrow R\right)=0$. Consequently, although S and R have the same conditional entropy, we can conclude that the tendency of implication of R on S is greater than the tendency of implication of S on R.

Application. Let us consider a series of R-rules extracted from a directed hierarchy. The previous definitions provide operational tools for reducing interactively (by the choice of ε) the rule set. The algorithm is incremental:

- Select the rule R with the greatest implication intensity and the greatest frequency in I.
- Sort the remaining rules by decreasing order of their implication intensities. For each rule S, compute $r\left(S \mid R\right)$ and $G\left(S \mid R\right)$ and reject S if $r\left(S \mid R\right)$ and $G\left(S \mid R\right)$ are greater than the fixed threshold.
- Start again the process with a rule R' with an implication intensity lower than R. And so on.

4 Mutual information with the Gini's coefficient

In this section we propose a different approach for rule reduction based on the Gini's coefficient. This coefficient is well-known to measure inequalities in a population. For our problem, it is interesting to quantify the dispersion of the distributions associated with rule realizations.

Let us first recall that the Gini's coefficient is a particular case of the Havrda and Charvat's α-entropy (Havrda and Charvat (1967)). Let X_R be a discrete random variable with a probability distribution for its k values defined by $(p_1, p_2, ..., p_k)$. The α-entropy is defined by

$$H_\alpha\left(R\right) = \frac{1}{1-\alpha}\left(\sum_{i=1}^{k} p_i^\alpha - 1\right)$$

The case $\alpha = 2$ corresponds to the Gini's coefficient: $Gini\left(R\right) = 1 - \sum_i p_i^2$.

Moreover, when α tends toward 1, the limit of the α-entropy is the Shannon's entropy. Thus, the semantic of these two coefficients are close.

Interpretation 1. The Gini's coefficient can be interpreted as a variance of Bernoulli independent variables of respective parameters $p_1, p_2, ..., p_k$. Indeed, it is easy to show that $1 - \sum_i p_i^2 = \sum_i p_i\left(1 - p_i\right)$.

Interpretation 2. The Gini's coefficient $1 - \sum_i p_i^2$ can be interpreted as a distance between the norms of two k-dimensional vectors: the components of the first vector are equal to 1 and those of the second one are equal to $p_1, p_2, ..., p_k$.

Proposition 2. The difference between the reduced entropy $H\left(R\right)$ and the Gini's coefficient $Gini\left(R\right)$ is positive on the interval $[0.5; 1]$.

In the binary case ($k = 2$) the Gini's coefficient is equal to $1 - p^2 - \left(1 - p^2\right) = 2p\left(1 - p\right)$. Then, the difference between $H\left(R\right)$ and $Gini\left(R\right)$ is a function of p defined by : $F(p) = -p\log_2 p - (1-p)\log_2(1-p) - 2p + 2p^2$. The function $F(p)$ decreases on $[0.5; 1]$and is equal to 0.5 and 0 when $p = 1$. Consequently, $H(R) - Gini(R) > 0$ on $[0.5; 1]$.

Let us precise the behavior of $F(p)$ in the vicinity of $p = 1$. The development of $H(R)$ gives : $H(R) \approx \frac{5}{2\ln 2}p(1-p)$. Hence, in the vicinity of $p = 1$, $H(R) - Gini(R) \approx 1.6p(1-p) > 0$. Consequently, the reduced entropy is always greater than the Gini's coefficient.

On the same way as for the Shannon's entropy, we now consider the conditional Gini's coefficient. Generally speaking, our approach is close to the proposition of Simovici and Jaroszewicz (2003). But it is adapted to R-rules.

Definition 6. Let R and S be two rules and X_R and X_S be their associated variables. Let us denote by p_i, $i = 1, 2$, the frequencies of X_R and p_{ij} the frequencies of (X_R, X_S). The conditional Gini's coefficient $Gini(S \mid R)$ is defined by

$$Gini(S \mid R) = 1 - \sum_{i=1}^{2}\sum_{j=1}^{2}\frac{p_{ij}^2}{p_j}$$

Let us remark that this sum is similar to a sum of generalized conditional variances, and it still can be interpreted as an information coefficient. Here, the modalities of X_S and X_R are true and false. Consequently,

$$Gini(S \mid R) = 1 - \left(\frac{p_{RS}^2}{p_R} + \frac{p_{R\overline{S}}^2}{p_R} + \frac{p_{\overline{R}S}^2}{p_{\overline{R}}} + \frac{p_{\overline{RS}}^2}{p_{\overline{R}}}\right)$$

Hence,

$$Gini(S \mid R) = \sum_{i=R,\overline{R}}\sum_{j=S,\overline{S}}\frac{p_{ij}}{\sqrt{p_i}}\left(1 - \frac{p_{ij}}{\sqrt{p_i}}\right)$$

We can deduce from this formula the information growth in X_S when X_R is known.

Definition 7. Let us consider two rules S and R. The Gini's gain for S knowing R is defined by $Gain_G(S \mid R) = Gini(S) - Gini(S \mid R)$.

This gain is a difference of variances; it measures the information quality brought by R on S. When X_R and X_S are independent, then $Gini(S \mid R) = Gini(S)$ similarly to the Shannon's gain. However, the converse is false.

For illustration, let us consider again the rules $R = (v_3 \rightarrow v_1)$ and $S = (v_5 \rightarrow v_4)$. Then, $Gini(S) = 0.0644 < H(S)$ and $Gini(S \mid R) = 0.0644$. There is no informational increasing on S when R is known. On this example, the Gini's coefficient is less discriminant than the conditional entropy.

5 Conclusion

In the SIA framework, we aim at improving the characterization of the redundant rules in rule set produced by automatical algorithms. In particular, we here focused on simple rules of the form $a \rightarrow b$ and R-rules of the form $R_1 \rightarrow R_2$ associated with a directed hierarchy. For a rule pair, the idea consists in measuring the gain of information brought by one rule on the other. Intuitively, one rule is *redundant* when the whole information associated with it is already known. In order to quantify this redundancy, we have proposed two different measures: the first one is based on the Shannon's entropy and the second one is based on the Gini's coefficient.

In the next future, we plan to make numerical simulations to experimentally confirm the complementarity of these two measures, and to show their efficiency for reducing rule sets. In particular, first experiments show that the algorithmic complexity of the rule pruning algorithm described in the application is significantly reduced thanks to the thresholds associated with the new measures.

References

AGRAWAL, R., IMIELINSKY, T., and SWANI, A. (1993): Mining association rules between sets of items in large databases. In: *Proc. of the ACM SIGMOD'93.* AAAI Press, 679–696.

BERNARD, J.-M. and POITRENAUD, S. (1999): L'analyse implicative bayesienne d'un questionnaire binaire: quasi-implications et treillis de Galois simplifié. *Mathématiques, Informatique et Sciences Humaines, 147, 25–46.*

BLANCHARD, J., GUILLET, F., and BRIAND, H. (2007): Interactive visual exploration of association rules with the rule focusing methodology. *Knowledge and Information Systems (to appear).*

GRAS, R. (1979): *Contribution à l'étude expérimentale et à l'analyse de certaines acquisitions cognitives et de certains objectifs didactiques en mathématiques.* PhD thesis, Université de Rennes I, France.

GRAS, R., ALMOULOUD, S.A., BAILLEUL, M., LARHER, A., POLO, M., RATSIMBA-RAJOHN, H., and TOTOHASINA, A. (1996): *L'implication statistique – Nouvelle méthode exploratoire de données.* La Pensée Sauvage editions, France.

GRAS, R. and KUNTZ, P. (2005): Discovering r-rules with a directed hierarchy. *Soft Computing, 1, 46–58.*

GRAS, R., KUNTZ, P., and BRIAND, H. (2001): The foundations of the implicative statistical analysis and some extensions for data mining (in french). *Mathématiques et Sciences Humaines, 154, 9–29.*

HAVRDA, J.-H. and CHARVAT, F. (1967): Quantification methods of classification processes. *Concepts of structural entropy – Kybernetica, 3, 30–37.*

HILDERMAN, R. and HAMILTON, H. (1999): *Knowledge discovery and interestingness measures: a survey.* Technical Report 99–04, University of Regina.

KLEMENTTINEN, M., MANNILA, H., RONKAINEN, P., TOIVONEN, H., and VERKAMO, A. (1994): Finding interesting rules from large sets of discovered association rules. In: *Proc. of the* 3^{rd} *Int. Conf. on Information and Knowledge Management.* ACM, 401–407.

KUNTZ, P., LEHN, R., GUILLET, F., and BRIAND, H. (2000): A user-driven process for mining association rules. In: *Proc. of Principles of Data Mining and Knowledge Discovery.* Springer Verlag, 483–489.

LENT, B., SWANI, A., and WIDOM, J. (1997): Clustering association rules. In: *Proc. of the* 13^{th} *Int. Conf. on Data Engineering.* 220–231.

LOEVINGER, J. (1947): A systemic approach to the construction and evaluation of tests of ability. *Psychological Monographs, 61 (4).*

SIMOVICI, D. and JAROSZEWICZ, S. (2003): Generalized conditional entropy and decision trees. *Revue d'Intelligence Artificielle, 17 (3), 369–380.*

VAILLANT, B. (2006): *Mesurer la qualité des règles d'association – Études formelles et expérimentales.* PhD thesis, Université de Bretagne Sud.

A Appendix

	v_1	v_2	v_3	v_4	v_5
1	1	1	1	0	0
2	1	1	1	0	0
3	1	1	1	0	0
4	1	1	1	0	0
5	1	1	1	0	0
6	1	1	1	1	0
7	1	1	0	1	1
8	1	1	0	1	1
9	1	1	1	1	1
10	1	1	1	1	1
11	1	0	1	1	0
12	1	0	1	1	0
13	1	0	1	1	0
14	1	0	1	1	1
15	1	0	1	1	0
16	1	0	1	1	1
17	1	0	1	1	1
18	1	0	1	1	0
19	1	0	1	1	0
20	1	0	1	0	1
21	1	0	1	1	0
22	1	0	1	0	0
23	1	0	0	1	1
24	1	0	0	0	0
25	0	0	0	0	0
26	0	0	0	0	0
27	0	0	0	0	0
28	0	0	0	1	0
29	0	0	1	1	1
30	0	0	0	1	0
Total	24	10	21	19	10

Mining Personal Banking Data to Detect Fraud

David J. Hand[1,2]

[1] Department of Mathematics, Imperial College London
South Kensington Campus, SW7 2AZ, UK
[2] Institute for Mathematical Sciences, Imperial College London
South Kensington Campus, London, SW7 2AZ, UK, *d.j.hand@imperial.ac.uk*

Abstract. Fraud detection in the retail banking sector poses some novel and challenging statistical problems. For example, the data sets are large, and yet each transaction must be examined and decisions must be made in real time, the transactions are often heterogeneous, differing substantially even within an individual account, and the data sets are typically very unbalanced, with only a tiny proportion of transactions belonging to the fraud class. We review the problem, its magnitude, and the various kinds of statistical tools have been developed for this application. The area is particularly unusual because the patterns to be detected change in response to the detection strategies which are developed: the very success of the statistical models leads to the need for new ones to be developed.

1 Background

The aim of this article is to review the application of statistical modelling ideas in the detection of fraud in the personal banking sector. The area poses some novel statistical challenges.

The *Concise Oxford Dictionary* defines fraud as ‘criminal deception; the use of false representations to gain an unjust advantage. ’As such, fraud must be as old as humanity itself. Indeed, one might go so far as to claim that it is older, since even animals are known to behave in ways which deceive others, although the notion of ‘criminal ’behaviour is uniquely human.

Banking fraud, in particular, has many faces. At one extreme, there is money laundering, in which one tries to pass off illegally gained funds and feed them into the legitimate banking system. At an intermediate level, there is fraud against organisations, such as commercial or public organisations. And at the far extreme there is fraud against an individual, such as through stolen or cloned credit cards. Banking fraud also covers a vast range of sizes, ranging from giant cases such as Enron and European Union fraud, to small personal cases such as selling forged tickets to soccer matches. No day passes without the national press mentioning cases of fraud - and, indeed, without countless frauds being perpetrated throughout the world.

This paper is chiefly concerned with banking fraud, and in particular fraud in the retail or personal banking sector. This covers credit cards, private residential mortgages, car finance, personal loans, current bank accounts, savings bank accounts, and so on. It is a natural application domain for statistics and related areas of data analysis, since it involves large numbers of individual units - people.

The personal banking sector has witnessed something of a revolution in recent decades. Instead of loans and other banking products being granted by the decisions of individual bank managers, there has been a shift towards the use of objective statistical models. Such models have many advantages over humans: they do not tire or suffer from irrational changes of mood, their performance can be monitored and improved in an evolutionary way by comparing the performance of slightly modified versions, they are very quick so that one does not have to wait for days for a decision, and above all, they are consistent and no subjective or illegal prejudices can accidentally creep in. These changes have been paralleled by other changes: nowadays huge databases summarising the transaction, purchasing, and payment history of individuals is stored in computer databases. Such data warehouses provide sources of information which can be mined to better understand how people behave, and to predict how they are likely to behave in the future. And systems to obtain credit, in particular, have changed completely. In the US at the end of 2005, outstanding consumer credit, *excluding mortgages*, exceeded two trillion dollars. This is in large part the result of technical innovation. As Alan Greenspan put it in Greenspan (2005): ‘Unquestionably, innovation and deregulation have vastly expanded credit availability to virtually all income classes. ’

2 Personal banking fraud

With such large sums of money involved, it would be surprising if fraudsters were not attracted. The scale of the problem is illustrated by the 2005 UK figures for plastic card fraud (one can find corresponding figures for any country). The largest category of fraud was ‘cardholder not present ’fraud, amounting to £183 million. This category includes phone, internet, and email fraud. The next largest was counterfeit fraud, amounting to £97 million. This includes skimming and cloning of cards, in which the electronic details are read and duplicated on another card. Close behind this was stolen or lost cards (£89 million), and this was followed by mail interception (£40 million), card identity theft from account takeover (£18 million), and card identity theft from fraudulent applications (£12 million). Of particular interest is that only the first of these, cardholder not present fraud, shows an increase over the 2004 figure. All the others show a decrease. This illustrates a particularly important point, to which I shall return below.

The figures above might be regarded as the tip of the iceberg. They represent clear direct fraud. In fact, the total loss due to fraud is much larger because of the additional indirect components. Overall, plastic card fraud can be regarded as being composed of several components:

1. immediate direct loss due to fraud - the figures given above;
2. cost of installing and running fraud prevention and detection systems;
3. cost of loss business, for example, while a stolen card is replaced;
4. the opportunity cost of fraud prevention and detection - the other, alternative, profitable things which the money might have been spent on;
5. the deterrent effect of public fraud on the spread of e-commerce.

Little wonder, then, that some estimates give total worldwide plastic card fraud in the many billions of dollars.

3 An arms race

I commented above that all types of plastic card fraud apart from cardholder not present had shown a decrease between 2004 and 2005. This is an important point, and one which characterises statistical research in this area and introduces novel challenges. When one develops a statistical model to understand nature - in physics or biology, for example - discoveries remain true, unless or until they are replaced by more elaborate descriptions of nature which explain the data in a superior way. In fraud detection, however, this is not the case. Fraud detection represents an ongoing arms race between the fraudsters and those tasked with detecting and preventing fraud, so that the problem is inherently non-stationary. Once systems are in place to prevent a particular type of fraud, the perpetrators do not abandon their lives of crime, but move onto some other approach. We have recently witnessed a nice example of this with chip and PIN technology in the UK. Chip and PIN technology replaces signatures and magnetic stripes on cards with Personal Identification Numbers and microchips on the cards. This system was launched in the UK on 14th February 2006. Some predicted that it would reduce credit card fraud by 90%. As a consequence, it was also predicted that it would lead to an increase in identity theft (in which full financial and personal details of the victim are stolen, so that loans and other financial products, including credit cards, can be taken out without the victim being aware of it) and in fraudulent credit card use in Europe, which still relied on the signature and magnetic stripe technology. And these predictions came true - Lloyds TSB, for example, observed an increased fraudulent use of UK credit cards abroad. There was also an increase in ATM theft and cardholder not present fraud. Worse than this, however, crooks also reverted to a new use of an old technology. They had long installed ‘skimmers ’in ATM machines, to record both the card details and the PIN numbers, and now they installed these in the machines used in chip and PIN systems. Over £1 million was stolen from Shell service stations before this scam was stopped.

Sleeper fraud provides another nice illustration of nonstationarity. In this scheme, fraudsters use the card in an apparently perfectly legitimate way, making transactions and repayments as if they were law-abiding users. Gradually, they ramp up their credit limit - until suddenly spending up to the limit and disappearing. It takes patience, of course, but can be lucrative, and it is very difficult to prevent.

At the time of writing, one of the newest technologies to be introduced in this arms war is the *one-time password.* There are several variants of this, but each involves using a unique password, different each time a transaction is made. This can be by using an algorithm which calculates the new password from the last one, or via time synchronised algorithms in the card and the authentication server, or in other ways. But how long will it be before fraudsters find a way round this?

4 Other challenges

If the plastic card fraud detection problem is unusual in that the characteristics of the fraud class of objects changes in response to the detection algorithms being installed, then it is also challenging in several other ways.

Generally, plastic card transaction data sets are large, often very large. If a bank has 10 million customers, making an average of 3 credit card transactions a week, then a year 's worth of transactions represents a lot of data. When one then recognises that between 70 and 80 items of information are recorded for each transaction (transaction type, date and time of transaction, amount, currency, local currency amount, merchant category, card issuer, ATM ID, POS type, number of times chip has been accessed, merchant city name, etc.) then it is easy to see that scalable and highly efficient algorithms are needed. In particular, unlike in statistical modelling, in which the aim is to produce a summary of the data which captures its distributional characteristics, so that one can use a sample of data, here it is absolutely necessary to examine each and every transaction. Dynamic updating to capture the intrinsic non-stationarity is a nice idea, but dynamic updating of millions of separate models, one for each account, as each new transaction is made, is likely to be impossible for advanced models such as support vector machines, random forests, or neural networks. Multilevel models may provide a partial answer here, in which the basic model form is the same for each customer, with just a few (easily updated) parameters being varied.

Raw fraud data sets are also typically unbalanced, having many more legitimate than fraudulent cases: an oft-quoted figure is that about 1 in a 1000 are fraudulent. This is crucial because of the familiar phenomenon, illustrated below, that high sensitivity and specificity in such cases do not translate into a high proportion of fraud cases amongst those predicted as fraudulent. The implication is that the two types of misclassification should be weighted very differently. Multi-stage procedures can be effective approaches in such cases, as outlined below.

There is also often a delay in learning the true class labels. In fact, this is a familiar problem in the banking sector, where these labels often do not become apparent until a later reconciliation or account checking stage. It can mean a lag in updating of distributions. It is compounded with the problem of incorrect labels. There is the obvious problems that account holders may not check their statements very rigorously, so that fraudulent transactions are mislabelled as legitimate. This is a one-way misclassification, and so may not be too serious in terms of classification accuracy (its primary impact being on the classification threshold). However, consider the case of an account holder making a series of legitimate transactions, and then deciding to get the cost reimbursed by claiming that the card had been stolen and the transactions were not theirs. Now the true labels become 'fraud ', even though the transaction pattern may be indistinguishable from a legitimate series of transactions. (Fortunately, in fact, such a series would typically be distinguishable, since normally the account holder sets out to maximise their gains, and so behaves differently from normal.)

5 Statistical tools

Various statistical approaches have been explored in the battle against fraud (Bolton and Hand, 2002). Here I am using 'statistics 'in the sense of Chamberss 'greater statistics '(see Chambers (1993) and the rejoinder to Bolton and Hand (2002)), to mean 'everything related to learning from data ', so that it includes machine learning, data mining, pattern recognition, and so on. Provost (2002) makes a nice analogy with the classic parable of the blind men and the elephant - each felt a different part of the creature and imagined an entirely different sort of animal. So it is with fraud detection: there are many different approaches. It is important to recognise that these approaches are not in competition. They can (subject to scalability and computational issues) be used simultaneously and in parallel. By this means, old weapons in the fraudsters armoury will be defeated even while new ones are being tackled.

The core approach is a rule-based or pattern matching approach. This is applied when a particular type of transaction or transaction pattern is known often to be indicative of fraud. For example, the pattern of two ATM withdrawal attempts in which the first takes out the maximum allowed and the second occurs within 24 hours is suspicious. It suggests that the second attempt was not aware of the first - and that two people are using the account. Another such intrinsically suspicious pattern is the credit card purchase of many small electrical items in quick succession, since these can easily be sold on the black market. We see from these examples that one cannot be certain, merely from the transaction pattern, that fraud has occurred. A human has to be in the loop. We shall return to this point when we consider measures for assessing the performance of fraud detection systems.

More generally, however, we will want to detect departures from normal behaviour for an individual, in unpredictable ways, as well as in predictably suspicious ways. This requires decisions about two aspects: what exactly is the unit of analysis, and what is the ‘norm ’relative to which behaviour is classified as ‘suspicious ’?

Superficially, the unit of analysis is simple enough: it is the transaction (lying in a space with 70-80 dimensions). Sometimes people use their cards in highly predictable ways (e.g. practicing ‘jamjarring ’, in which they use different cards for different categories of purchase), but in other cases the transactions are highly heterogeneous. Especially in the latter case, it can be advantageous to work with groups of transactions, rather than individual transactions. This can be done in various ways. We can, for example, summarise the transactions within a group (e.g. the last 5 transactions). This allows more flexibility of description and has the potential to capture more unusual patterns of behaviour. Of course, it sacrifices the immediacy of individual transaction analysis. It also requires tools for rapid updating of the summary descriptors.

Similarly, at a superficial level, the choice of norm is straightforward: we should compare the new behaviour of a customer with his or her previous behaviour. This requires sufficient data being available on that customer previously. It also enters the realm of scalability issues: if an entirely different model has to be built for each customer then updating may be expensive. A compromise may be the multilevel approach mentioned above.

A rather different approach is to compare the behaviour of a customer with that of other similar customers. In ‘peer group analysis ’(Bolton and Hand (2001), Ferdousi and Maeda (2006)), for example, we identify the k customers who have behaved most similarly to a target customer in the past, and then follow them to see if the behaviour of the target customers starts to deviate from their ‘peer group ’. In its simplest form, this is done for each customer separately.

In the above, modelling occurs at the level of the model of behaviour which we expect a legitimate account to follow, but there is no deeper conceptualisation possible. This is to regard the account as undergoing a state change when a fraudster hijacks it, from the legitimate to the fraud state. In the former, all transactions are taken to be legitimate, but in the latter there will be fraudulent transactions, perhaps with some legitimate ones mixed in. We can think of this as a latent variable model, this variable being the state, and our aim is to detect when the state change occurs: it is a change point problem. Such problems have been extensively explored, though most often in situations in which a single manifest variable is undergoing a level shift.

Various kinds of multilevel models are particularly valuable in fraud detection problems. A straightforward application of such models is multilevel screening. This can also help with the computation and scalability issues. In this approach, one applies a simple and quick method to eliminate the clearly non-fraudulent transactions: one computes a simple suspicion score and eliminates those with low values. Some frauds may get through, but one has to recognise that perfection is not achievable and if this initial screen can adjust the prior size of the fraud class from 0.001 to 0.01 or better then significant progress has been made. The second level may then use the same descriptive characteristics, combined in a much more elaborate and sophisticated way (e.g. using a random forest, treenet, support vector machine, or neural network) or may use additional data. The idea is analogous to the reject option, although it is one-sided.

Stolfo et al. (1997a, b) described an alternative use of multiple models, in which different fraud detection algorithms are used for different sectors, with the results being combined. Given that certain areas are more subject to fraud than others, this seems like a very sensible approach - why should one believe that the same sort of detection algorithm should apply in each area?

The aim is always to classify transactions or more general transaction groups into one of two classes: fraudulent or legitimate. Systems to achieve this can be based on supervised classification ideas, in which one uses samples of known frauds and known non-frauds to construct a rule which will allow one to assign new cases to a class (by comparing an estimated suspicion score with a threshold). But an alternative would be to estimate contours of the non-fraud class, classifying outlying points as potentially fraudulent. The contours here will most probably best be based on an individuals previous legitimate transactions. Breiman (2002) argues that the supervised approach is likely to be more effective.

So far, all of the discussion has been in terms of individual transactions, or groups of transactions within a given account, treating the accounts as independent. However, while accounts may indeed generally be independent, the ways fraudsters use accounts are not. Firstly, fraudsters tend to work in gangs, not individually (for example, stealing, recycling, and cloning multiple cards). And secondly, if a fraudster discovers a successful modus operandi, then they are likely to repeatedly use that until stopped. This can be made use of in detection systems. For example, if an account is known to have switched to the fraud state (that is, some of the transactions on an account are known to be fraudulent), one can look back at all of that accounts recent transactions and examine other accounts which made transactions at the same sites more carefully. Quite how effective this will be will depend on what data are stored about the transactions. If individual ATM identifiers are stored, it will be easy for ATM transactions, for example. If only high level merchant codes are stored for credit card transactions, however, then it would result in a much blunter instrument. The idea is a dynamic version of simple methods based on learning what merchant codes are intrinsically more likely to be associated with fraud.

6 Assessing performance of fraud detection tools

Although different kinds of techniques may be used to process a transaction or activity record, the aim in all cases is to assign them to one of two classes, fraud or non-fraud. This is even the case if the problem is viewed as one of detecting state change: one aims to classify those prior to the change as legitimate and those after the change as fraudulent. This means that an important class of performance assessment measures must be based on the two by two cross-classification of true class (fraud, non-fraud) by predicted class.

The classification community has developed many measures for such situations, tackling different aspects of performance. Simple ones include misclassification rate and specificity and sensitivity. As we have already mentioned, these are typically inappropriate in fraud detection problems because of the dramatically unbalanced class sizes: a very low misclassification rate (0.1% if only 0.1% of the transactions really are fraudulent) is achieved by assigning every transaction to the legitimate class. But this, of course, defeats the object. The point is that misclassifying a fraud case is much more serious than misclassifying a legitimate case. The former means a real financial loss, which could run into many thousands of pounds. The latter incurs only the cost of checking that the transaction is legitimate, plus also some customer irritation if the account is temporarily suspended. This irritation can be managed - after all, most customers like to know that the bank is looking out for them. If the true fraud rate is 0.1% then a detection rule which successfully classifies 99% of the fraud cases as fraudulent, and 99% of the legitimate cases as legitimate will in fact be correct in only 9% of the cases it predicts as fraudulent. This could mean substantial customer irritation, not to mention the cost 'wasted 'on the 91 in every 100 suspected frauds which are really legitimate.

There are also other aspects of fraud performance which one might want to take into account. Hand et al. (2006) point out that whenever a fraud is suspected, it incurs an investigation cost, regardless of whether a fraud has actually been committed or not. Thus a suitable measure might be based on minimising a suitably weighted combination of the total number of fraud alarms and the number of real frauds which evade detection. Even more elaborate measures may be based on the actual monetary losses incurred when a fraud does occur.

7 Conclusion

At a conference on banking fraud I attended not so long ago, a banker remarked to me that his bank ‘did not have any fraud ’. He was speaking tongue in cheek, of course, but some important points underlie his comment.

The first is that it is very important, for customer and shareholder confidence, to know that a bank is a reliable organisation, not subject to criminal attacks, and to the costs that that would imply. The contrary assertion (or, perhaps, admission) that the bank loses hundreds of millions of dollars per annum to fraud would hardly inspire confidence.

Secondly, at a superficial level there would appear to be an appropriate balance to be struck between the amount spent on detecting and preventing fraud and the amount of fraud prevented. One might decide that a break-even point was appropriate: it might be regarded as sensible to spend £x to prevent £x of fraud, but foolish to spend £y to prevent £x if $y > x$. This is all very well, but it ignores the deterrent effect: a fraud system costing £y may prevent substantially larger amounts of fraud merely because it is known to exist - merely because the bank is known to be able to detect fraud attacks.

In any case, while one might be able to quantify the amount spent on fraud detection and prevention systems, quantifying the amount saved by these systems is difficult. After all, if fraud is not attempted by virtue of a prevention strategy, how can its extent be measured? In general, quantifying the value of fraud detection systems is difficult.

I commented above that once a particular avenue of fraud has been prevented by an appropriate tool, fraudsters do not abandon their efforts, but change tacks. This means that a Pareto principle applies. 50% of fraud is easy to detect - the early methods used by those new to the game. But the next 25% is much harder, and the next 12% harder still. Indeed, it would be naive to suppose that all fraud is prevented or could be prevented, no matter how sophisticated the statistical models. Think of those previously law-abiding customers who suddenly realise that, if they claim their card has been stolen after a spending spree, they will be reimbursed. Think of sleeper fraud.

Other reviews of statistical approaches to fraud detection are given in Fawcett and Provost (2002), Bolton and Hand (2002) and Phua et al. (2005).

Acknowledgments

This work has been partially supported by EPSRC grant number EP/C532589/1 for the *Thinkcrime* project on *Statistical and machine learning tools for plastic card and other personal fraud detection.*

References

BOLTON, R.J. and HAND, D.J. (2001): Peer group analysis. Technical Report, Department of Mathematics, Imperial College, London.

BOLTON, R.J. and HAND, D.J. (2002): Statistical fraud detection: a review. *Statistical Science, 17, 235-255.*

BREIMAN, L. (2002): Comment on Bolton and Hand (2002). *Statistical Science, 17, 252-254.*

CHAMBERS, J.M. (1993): Greater or lesser statistics: a choice for future research. *Statistics and Computing, 3, 182-184.*

FAWCETT, T. and PROVOST, F. (2002): Fraud detection. In: W. Kloesgen and J. Zytkow (Eds.): *Handbook of Knowledge Discovery and Data Mining*, Oxford University Press, Oxford.

FERDOUSI, Z. and MAEDA, A (2006): Unsupervised outlier detection in time series data. In: *Proceedings of the 22nd International Conference on Data Engineering Workshops*, ICDEW06, IEEE, 51–56.

GREENSPAN, A. (2005): *Consumer finance.* Remarks presented at the Federal Reserve Systems Fourth Annual Community Affairs Research Conference, Washington DC, 8th April.

HAND, D.J., WHITROW, C., ADAMS, N.M., JUSZCZAK, P., and WESTON, D. (2006): Performance criteria for plastic card fraud detection tools. To appear in *Journal of the Operational Research Society.*

PHUA, C., LEE, V., SMITH, K, and GAYLER, R. (2005): A comprehensive survey of data mining-based fraud detection research. Technical Report, Monash University.

PROVOST, F. (2002): Comment on Statistical fraud detection: a review. *Statistical Science, 17, 249-251.*

STOLFO, S., FAN, W., LEE, W., PRODROMIDIS, A.L. and CHAN, P. (1997a): Credit card fraud detection using meta-learning: issues and initial results. In: *AAAI Workshop on AI Approaches to Fraud Detection and Risk Management*, AAAI Press, Menlo Park, CA, 83–90.

STOLFO, S.J., PRODROMIDIS, A. L., TSELEPIS, S., LEE, W., FAN, D.W., and CHANN, P.K. (1997b): JAM: Java agents for meta-learning over distributed databases. In: *AAAI Workshop on AI approaches to Fraud Detection and Risk Management*, AAAI Press, Menlo Park, CA, 91–98.

Finding Rules in Data

Tu-Bao Ho[1,2]

[1] Institute of Information Technology, Vietnamese Academy of Science and Technology, 18 Hoang Quoc Viet, Hanoi, Vietnam
[2] Japan Advanced Institute of Science and Technology (JAIST)
1-1 Asahidai, Nomi, Ishikawa, Japan, *bao@jaist.ac.jp*

Abstract. In the first year of my preparation for doctor thesis at INRIA in the group of Edwin, I worked on the construction of an inference engine and a knowledge base, by consulting various group members, for building an expert system guiding the data analysis package SICLA of the group. One day, Edwin asked me whether one can automatically generate rules for expert systems from data, and I started my new research direction. Since that time, my main work has been machine learning, especially finding rules in data. This paper briefly presents some learning methods we have developed.

1 Introduction

Twenty years ago, machine learning was in its infancy with few work and applications. The wave of artificial intelligence (AI) in early of years 1980s has fostered the development of machine learning. From the joined work on conceptual clustering with Michalski, Edwin found his interest in this young field of machine learning (Michalski et al., 1983). As a doctor candidate in his group at that time, he suggested if I can work on finding new ways to generate rules for expert systems from data, instead of working as knowledge engineers who try to acquire knowledge from human experts.

There have been a great progress in the field of machine learning. It has become an established area with sound foundation, rich techniques and various applications. Machine learning becomes one of the most active areas in computer science. This paper briefly presents some of our main work in machine learning since those days in the group of Edwin, from supervised learning (Ho et al., 1988), (Nguyen and Ho, 1999), (Ho and Nguyen, 2003) to unsupervised learning (Ho, 1997), (Ho and Luong, 1997), and some recent work on text clustering (Ho and Nguyen, 2002), (Ho et al., 2002), (Le and Ho, 2005), bioinformatics (Pham and Ho, 2007), kernel methods (Nguyen and Ho, 2007).

2 Rule induction from supervised data

In this section we briefly present three supervised learning methods of CABRO1 (Ho et al., 1988), CABRO2 (Nguyen and Ho, 1999), and LUPC (Ho and Nguyen, 2003).

2.1 CABRO1

CABRO1 (Construction Automatique à Base de Régles et à partir d'observations) is a method of rule induction from supervised data.

Let $D_1, D_2, ..., D_p$ be p finite domains and $D_1 \times D_2 \times ... \times D_p$. Elements of D are called *objects* and denoted by $\omega = (d_1, d_2, ..., d_p)$ where $d_j \in D_j$ for $j \in J = \{1, 2, ..., p\}$. A variable-value pair (X_j, d_j) defines an *elementary assertion* $A_{X_j = d_j}$ that determines the set $\omega_{X_j = d_j}$ of objects of D which have the value $d_j \in \{d_{j1}, d_{j2}, ..., d_{jq}\}$ for the variable X_j: $A_{X_j=d_j} : D \longrightarrow \{true, false\}$, $\omega = (d_1, ..., d_p) \mapsto A_{X_j=d_j}(\omega) = true,\ \ if\ X_j(\omega) = d_j$ and $\omega = (d_1, ..., d_p) \mapsto A_{X_j=d_j}(\omega) = false,\ \ if\ X_j(\omega) \neq d_j$.

We consider an *assertion* as a conjunction of elementary assertions: $A = \bigwedge(X_j, d_j),\ \ j \in J' \subseteq J\ \ and\ \ d_j \in D_j$, where $\bigwedge$ denotes the logical conjunction. An assertion A is a Boolean function from $D \longrightarrow \{true, false\}$, and it is also the identification function for the set: $\omega_A = \{\omega \in D \mid A(\omega) = true\}$.

Variables correspond to $j \in J'$ are said to be *tied to* the assertion. Variables correspond to $j \in J \setminus J'$ are said to be *free from* the assertion. Number of tied variables is called *length* of the assertion. One says also that assertion A *covers* the set ω_A. Assertion A is said to be *more general than* assertion B iff $\omega_B \subseteq \omega_A$. Assertion A is said to be *better than* assertion B iff $card(\omega_A) > card(\omega_B)$. A is a *representative assertion* generated from an object $\omega \in E$ if A is one of the best assertions formed by elementary assertions generated from ω.

Denote $\Re = \Re_C \cup \Re'_C$ the set of assertions to be found for C and C'. Naturally, assertions generated for each concept, for instance C, have to satisfy two following constraints: (1) *Covering*: Each observed object of the learning set E has to be recognized by an assertion of $\Re_C$: $E \subseteq \bigcup_{A \in \Re_C} \omega_A$, and (2) *Descriminating*: Assertions of C do not misrecognize members of $E' : \omega_A \cap E' = \emptyset$.

It is clear that the less general an assertion, the more discriminant it is. Depending on the data nature, one retains general but not perfect discriminant assertions or discriminant but not sufficient general assertions. The *belief measure* $\mu(A)$ for the assertion A of C is estimated as the ratio of the number of examples of C matched by A and the total number of examples of C and C' matched by A: $\mu(A) = card(\omega_A \cap E)/card(\omega_A \cap \Omega), (0 < \mu(A) \leq 1)$.

An assertion A is said *β-discriminant* if $\mu(A) \geq \beta$. In fact, instead of finding discriminant assertions one finds β-discriminant assertions depending on an acceptance threshold $\beta\ \ (0 < \beta \leq 1)$.

The main algorithm of CABRO1 is based on a *general-to-specific search*: one starts from an 'empty' assertion which is the 'most general' because all of its variables are free, then one ties the value $X_j(\omega)$ to this assertion so that the assertion covers approximately a maximum number of objects of E (the generality of the assertion will be diminished but it may remain non β- discriminant). This phase is repeated with the remaining values until one finds a β-discriminant assertion such that the next attempt does not improve the covering of the assertion. We propose a dual algorithm of the CABRO1 algorithm, based on a *specific-to-general* search strategy, in order to find a representative assertion A_ω from an object $\omega \in E$. On the contrary with CABRO1 algorithm, the dual algorithm starts from a 'full' assertion which is the 'most specific' and covers only the object ω. One tries to increase its generality and to diminish its speciality simultaneously in order to obtain a representative assertion.

2.2 CABRO2

The starting point of rough set theory (Pawlak, 1991) is the assumption that our "view" on elements of an object set O depends on an indiscernibility relation among them, that means an equivalence relation $E \subseteq O \times O$. Two objects $o_1, o_2 \in O$ are said to be *indiscernible* w.r.t E if $o_1 E o_2$. The *lower* and *upper* approximations of any $X \subseteq O$, w.r.t. an equivalence relation E, are defined as

$$E_*(X) = \{o \in O : [o]_E \subseteq X\}, \quad E^*(X) = \{o \in O : [o]_E \cap X \neq \emptyset\}$$

where $[o]_E$ denotes the equivalence class of objects which are indiscernible with o w.r.t the equivalence relation E. A subset P of the set of attributes used to describe objects of O determines an equivalence relation that divides O into equivalence classes each containing objects having the same values on all attributes of P. A key concept in the rough set theory is the *degree of dependency* of a set of attributes Q on a set of attributes P, denoted by $\mu_P(Q)$ $(0 \leq \mu_P(Q) \leq 1)$, defined as $\mu_P(Q) = |\bigcup_{[o]_Q} / P_*([o]_Q)|/|O|$.

If $\mu_P(Q) = 1$ then Q totally depends on P; if $0 < \mu_P(Q) < 1$ then Q partially depends on P; if $\mu_P(Q) = 0$ then Q is independent of P. The measure of dependency is fundamental in rough set theory as based on it important notions are defined, such as reducts and minimal sets of attributes, significance of attributes, etc.

This argument can be generalized and formulated for a measure of degree of dependency of an attribute set Q on an attribute set P

$$\mu'_P(Q) = \frac{1}{|O|} \sum_{[o]_P} max_{[o]_Q} |[o]_Q \bigcap [o]_P|$$

Theorem. *For every sets P and Q of attributes we have*

$$max_{[o]_Q} |[o]_Q|/|O| \leq \mu'_P(Q) \leq 1$$

We can define that Q totally depends on P iff $\mu'_P(Q) = 1$; Q partially depends on P iff $max_{[o]_Q}|[o]_Q|/|O| < \mu'_P(Q) < 1$; Q is independent of P iff $\mu'_P(Q) = max_{[o]_Q}|[o]_Q|/|O|$.

Given two arbitrary attribute sets P and Q, we define *R-measure* for the dependency of Q on P

$$\mu_P(Q) = \frac{1}{|O|} \sum_{[o]_P} max_{[o]_Q} \frac{|[o]_Q \bigcap [o]_P|^2}{|[o]_P|}$$

```
**Learn-Positive-Rule**($Pos, Neg, mina, minc$)

$RuleSet = \phi$
$\alpha, \beta \leftarrow$ **Initialize**($Pos, mina, minc$)
while ($Pos \neq \phi$ & $(\alpha, \beta) \neq (mina, minc)$)
    $NewRule \leftarrow$ **BestRule**($Pos, Neg, \alpha, \beta$)
    if ($NewRule \neq \phi$)
        $Pos \leftarrow Pos \setminus Cover^{+}(NewRule)$
        $RuleSet \leftarrow RuleSet \cup NewRule$
    else **Reduce**($\alpha, \beta$)
$RuleSet \leftarrow$ **PostProcess**($RuleSet$)
return($RuleSet$)

**BestRule**($Pos, Neg, \alpha, \beta$)

$CandRuleset = \phi$
**AttValPairs**($Pos, Neg, \alpha, \beta$)
while **StopCond**($Pos, Neg, \alpha, \beta$)
    **CandRules**($Pos, Neg, \alpha, \beta$)
$BestRule \leftarrow$
        First $CandidateRule$
         in $CandRuleset$
return($BestRule$)
```

Fig. 1. The scheme of algorithm LUPC

When consider Q as the class attribute and P a descriptive attribute, we can use $\mu_P(Q)$ as a measure for attribute selection in decision tree learning. CABRO2 is the decision tree induction using *R-measure* that has performance as high as state-of-the-art methods such C4.5 (Nguyen and Ho, 1999).

2.3 LUPC

LUPC (Learning Unbalanced Positive Class) is a separate-and-conquer rule induction method to *learn minority classes in unbalanced datasets*. LUPC consequently learns a rule set from Pos and Neg given user-specified minimum accuracy threshold ($mina$) and minimum cover ratio ($minc$). We can partially order the goodness of rules in terms of accuracy or support. Given two thresholds α and β, $0 \leq \alpha, \beta \leq 1$, on accuracy and support of rules, respectively. A rule R is $\alpha\beta$-strong if $acc(R) \geq \alpha$ and $sup(R) \geq \beta$. An $\alpha\beta$-strong rule R_i is said better than an $\alpha\beta$-strong rule R_j with respect to α if R_i has accuracy higher than that of R_j. An $\alpha\beta$-strong rule R_i is better than an $\alpha\beta$-strong rule R_j with respect to β if R_i has support higher than that of R_j. LUPC distinguishes three alternatives that occur in practice and that lead to the three corresponding types of search heuristics:

1. *Bias on rule accuracy*: It is to sequentially find rules with cover ratio equal and greater than $minc$ but accuracy is as large as possible.
2. *Bias on rule cover ratio*. It is to sequentially find rules with accuracy equal and greater than $mina$ but the cover ratio is as large as possible.
3. *Alternative bias on rule cover ratio and accuracy*. LUPC starts with highest values of α and β, and alternatively learns rules with bias on either accuracy or cover ratio, then reduces one of the corresponding α or β

while keeping the other. The search is done until reaching the stopping. condition.

Note that $cov^+(R)$ can be quickly determined because $|Pos| \ll |Neg|$. When searching for $\alpha\beta$-strong rules, a candidate rule will be eliminated without continuing to scan though large set Neg if this property holds during scanning.

Proposition 1. *Given a threshold α, a rule R is not $\alpha\beta$-strong for any arbitrary β if $cov^-(R) \geq ((1-\alpha)/\alpha) \times cov^+(R)$.*

Figure 1 presents the scheme of algorithm LUPC that consists of two main procedures *Learn-Positive-Rule* and *BestRule* (Ho and Nguyen, 2003). LUPC has been applied to study stomach cancer and hepatitis with successes.

3 Conceptual clustering

A theory of concept lattices has been studied under the name *formal concept analysis* (FCA) (Wille, 1982). Considers a *context* as a triple $(\mathcal{O}, \mathcal{D}, \mathcal{R})$ where $\mathcal{O}$ be a set of objects, $\mathcal{D}$ be a set of primitive descriptors and $\mathcal{R}$ be a binary relation between $\mathcal{O}$ and $\mathcal{D}$, i.e., $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{D}$ and $(o, d) \in \mathcal{R}$ is understood as the fact that object o has the descriptor d. For any object subset $X \subseteq \mathcal{O}$, the largest tuple common to all objects in X is denoted by $\lambda(X)$. For any tuple $S \in \mathcal{T}$, the set of all objects satisfying S is denoted by $\rho(S)$. A tuple S is *closed* if $\lambda(\rho(S)) = S$. Formally, a *concept* C in the classical view is a pair (X, S), $X \subseteq \mathcal{O}$ and $S \subseteq \mathcal{T}$, satisfying $\rho(S) = X$ and $\lambda(X) = S$. X and S are called *extent* and *intent* of C, respectively. Concept (X_2, S_2) is a *subconcept* of concept (X_1, S_1) if $X_2 \subseteq X_1$ which is equivalent to $S_2 \supseteq S_1$, and (X_1, S_1) is then a *superconcept* of (X_2, S_2).

It was shown that λ and ρ define a Galois connection between the power sets $\wp(\mathcal{O})$ and $\wp(\mathcal{D})$, i.e., they are two order-reversing one-to-one operators. As a consequence, the following properties hold which will be exploited in the learning process:

$$\begin{gathered}
\text{if} \quad S_1 \subseteq S_2 \quad \text{then} \quad \rho(S_1) \supseteq \rho(S_2) \quad \text{and} \quad \lambda\rho(S_1) \subseteq \lambda\rho(S_2) \\
\text{if} \quad X_1 \subseteq X_2 \quad \text{then} \quad \lambda(X_1) \supseteq \lambda(X_2) \quad \text{and} \quad \rho\lambda(X_1) \subseteq \rho\lambda(X_2) \\
S \subseteq \lambda\rho(S), \quad X \subseteq \rho\lambda(X) \\
\rho\lambda\rho = \rho, \quad \lambda\rho\lambda = \lambda, \quad \lambda\rho(\lambda\rho(S)) = \lambda\rho(S) \\
\rho(\textstyle\bigcup_j S_j) = \bigcap_j \rho(S_j), \quad \lambda(\bigcup_j X_j) = \bigcap_j \lambda(X_j)
\end{gathered}$$

The basic theorem in FCA states that the set of all possible concepts from a context $(\mathcal{O}, \mathcal{D}, \mathcal{R})$ is a *complete lattice*[1] $\mathcal{L}$, called Galois lattice, in which infimum and supremum can be described as follows:

$$\bigwedge_{t \in T} (X_t, S_t) = (\bigcap_{t \in T} X_t, \lambda\rho(\bigcup_{t \in T} S_t))$$

[1] A lattice $\mathcal{L}$ is complete when each of its subsetf X has a least upper bound and a greatest lower bound in $\mathcal{L}$.

Table 1. Scheme of OSHAM conceptual clustering

```
Input        concept hierarchy $H$ and an existing splittable concept $C_k$.
Result       $H$ formed gradually.
Top-level    call OSHAM(root concept, $\emptyset$).

1. While $C_k$ is still splittable, find a new subconcept of it that corresponds to the
   hypothesis minimizing the quality function $q(C_k)$ among $\eta$ hypotheses gener-
   ated by the following steps (a) Find a "good" attribute-value pair concerning
   the best cover of $C_k$.
   (b) Find a closed attribute-value subset $S$ containing this attribute-value pair.
   (c) Form a subconcept $C_{k_i}$ with the intent is $S$.
   (d) Evaluate the quality function with the new hypothesized subconcept.
   Form intersecting concepts corresponding to intersections of the extent of the
   new concept with the extent of existing concepts excluding its superconcepts.
2. If one of the following conditions holds then $C_k$ is considered as unsplittable
   (a) There exist not any closed proper feature subset.
   (b) The local instances set $C_k^r$ is too small.
   (c) The local instances set $C_k^r$ is homogeneous enough.
3. Apply recursively the procedure to concepts generated in step 1.
```

$$\bigvee_{t \in T} (X_t, S_t) = (\rho\lambda(\bigcup_{t \in T} X_t), \bigcap_{t \in T} S_t)$$

OSHAM (Making Automatically a Hierarchy of Structured Objects) is our proposed conceptual clustering method (Ho, 1997), (Ho and Luong, 1997). OSHAM allow generating descriptive rules from symbolic unsupervised datasets.

4 Tolerance rough set model and applications

The *tolerance rough set model* (TRSM) aims to enrich the document representation in terms of semantics relatedness by creating tolerance classes of terms in $\mathcal{T}$ and approximations of subsets of documents. The model has the root from rough set models and its extensions. The key idea is among three properties of an equivalence relation R in an universe U used in the original rough set model (reflexive: xRx; symmetric: $xRy \rightarrow yRx$; transitive: $xRy \wedge yRz \rightarrow xRz$ for $\forall x, y, z \in U$), the transitive property does not always hold in natural language processing, information retrieval, and consequently text data mining. In fact, words are better viewed as overlapping classes

Table 2. The TRSM nonhierarchical clustering algorithm

Input The set $\mathcal{D}$ of documents and the number K of clusters.
Result K clusters of $\mathcal{D}$ associated with cluster membership of each document.

1. Determine the initial representatives $R_1, R_2, ..., R_K$ of clusters $C_1, C_2, ..., C_K$ as K randomly selected documents in $\mathcal{D}$.
2. For each $d_j \in \mathcal{D}$, calculate the similarity $S(\mathcal{U}(\mathcal{R}, d_j), R_k)$ between its upper approximation $\mathcal{U}(\mathcal{R}, d_j)$ and the cluster representative $R_k, k = 1, ..., K$. If this similarity is greater than a given threshold, assign d_j to C_k and take this similarity value as the cluster membership $m(d_j)$ of d_j in C_k.
3. For each cluster C_k, re-determine its representative R_k.
4. Repeat steps 2 and 3 until there is little or no change in cluster membership during a pass through $\mathcal{D}$.
5. Denote by d_u an unclassified document after steps 2, 3, 4 and by $\mathrm{NN}(d_u)$ its nearest neighbor document (with non-zero similarity) in formed clusters. Assign d_u into the cluster that contains $\mathrm{NN}(d_u)$, and determine the cluster membership of d_u in this cluster as the product $m(d_u) = m(\mathrm{NN}(d_u)) \times S(\mathcal{U}(\mathcal{R}, d_u), \mathcal{U}(\mathcal{R}, \mathrm{NN}(d_u)))$. Re-determine the representatives R_k, for $k = 1, ..., K$.

which can be generated by *tolerance relations* (requiring only reflexive and symmetric properties).

The key issue in formulating a TRSM to represent documents is the identification of tolerance classes of index terms. We employ the co-occurrence of index terms in all documents from $\mathcal{D}$ to determine a tolerance relation and tolerance classes. Denote by $f_{\mathcal{D}}(t_i, t_j)$ the number of documents in $\mathcal{D}$ in which two index terms t_i and t_j co-occur. We define an uncertainty function I depending on a threshold θ as $I_\theta(t_i) = \{t_j \mid f_{\mathcal{D}}(t_i, t_j) \geq \theta\} \cup \{t_i\}$.

It is clear that the function I_θ defined above satisfies the condition of $t_i \in I_\theta(t_i)$ and $t_j \in I_\theta(t_i)$ iff $t_i \in I_\theta(t_j)$ for any $t_i, t_j \in \mathcal{T}$, and so I_θ is both reflexive and symmetric. This function corresponds to a tolerance relation $\mathcal{I} \subseteq \mathcal{T} \times \mathcal{T}$ that $t_i \mathcal{I} t_j$ iff $t_j \in I_\theta(t_i)$, and $I_\theta(t_i)$ is the tolerance class of index term t_i. A vague inclusion function ν, which determines how much X is included in Y, is defined as $\nu(X, Y) = |X \cap Y|/|X|$

This function is clearly monotonous with respect to the second argument. Using this function the membership function, a similar notion as that in fuzzy sets, μ for $t_i \in \mathcal{T}, X \subseteq \mathcal{T}$ can be defined as $\mu(t_i, X) = \nu(I_\theta(t_i), X) = |I_\theta(t_i) \cap X|/|I_\theta(t_i)|$.

Table 3. TRSM-based hierarchical agglomerative clustering algorithm

```
Input        A collection of $M$ documents $\mathcal{D} = \{d_1, d_2, \ldots, d_M\}$
Result       Hierarchical structure of $\mathcal{D}$
Given:    a collection of $M$ documents $\mathcal{D} = \{d_1, d_2, \ldots, d_M\}$
          a similarity measure $sim : \mathcal{P}(\mathcal{D}) \times \mathcal{P}(\mathcal{D}) \rightarrow R$
    for $j = 1$ to $M$ do
    $C_j = \{d_j\}$ end
    $H = \{C_1, C_2, , \ldots, C_M\}$
    $i = M + 1$
    while $|H| > 1$
        $(C_{n_1}, C_{n_2}) = \text{argmax}_{(C_u, C_v) \in H \times H} sim(\mathcal{U}(\mathcal{R}, C_u), \mathcal{U}(\mathcal{R}, C_v))$
        $C_i = C_{n_1} \cup C_{n_2}$
        $H = (H \setminus \{C_{n_1}, C_{n_2}\}) \cup \{C_i\}$
        $i = i + 1$
```

With these definitions we can define a tolerance space as $\mathcal{R} = (\mathcal{T}, I, \nu, P)$ in which the *lower approximation* $\mathcal{L}(\mathcal{R}, X)$ and the *upper approximation* $\mathcal{U}(\mathcal{R}, X)$ in $\mathcal{R}$ of any subset $X \subseteq \mathcal{T}$ can be defined as

$$\mathcal{L}(\mathcal{R}, X) = \{t_i \in \mathcal{T} \mid \nu(I_\theta(t_i), X) = 1\}$$

$$\mathcal{U}(\mathcal{R}, X) = \{t_i \in \mathcal{T} \mid \nu(I_\theta(t_i), X) > 0\}$$

The term-weighting method is extended to define weights for terms in the upper approximation $\mathcal{U}(\mathcal{R}, d_j)$ of d_j. It ensures that each term in the upper approximation of d_j but not in d_j has a weight smaller than the weight of any term in d_j.

$$w_{ij} = \begin{cases} (1 + \log(f_{d_j}(t_i))) \times \log \frac{M}{f_{\mathcal{D}}(t_i)} & \text{if } t_i \in d_j, \\ \min_{t_h \in d_j} w_{hj} \times \frac{\log(M/f_{\mathcal{D}}(t_i))}{1+\log(M/f_{\mathcal{D}}(t_i))} & \text{if } t_i \in \mathcal{U}(\mathcal{R}, d_j) \setminus d_j \\ 0 & \text{if } t_i \notin \mathcal{U}(\mathcal{R}, d_j) \end{cases}$$

The vector length normalization is then applied to the upper approximation $\mathcal{U}(\mathcal{R}, d_j)$ of d_j. Note that the normalization is done when considering a given set of index terms. Denote the document set by $\mathcal{D} = \{d_1, d_2, \ldots, d_M\}$ where $d_j = (t_{1j}, w_{1j};\ t_{2j}, w_{2j}; \ldots; t_{rj}, w_{rj})$ and $w_{ij} \in\ [0, 1]$. The set of all terms from $\mathcal{D}$ is denoted by $\mathcal{T} = \{t_1, t_2, \ldots, t_N\}$. In information retrieval, a query is given the form $Q = (q_1, w_{1q}; q_2, w_{2q}; \ldots; q_s, w_{sq})$ where $q_i \in \mathcal{T}$ and $w_{iq} \in [0, 1]$.

Table 2 and Table 3 describe two general TRSM-based nonhierarchical and hierarchical clustering algorithms. The TRSM-based nonhierarchical clustering algorithm can be considered as a reallocation clustering method to form K clusters of a collection $\mathcal{D}$ of M documents. The main point of the TRSM-based hierarchical clustering algorithm is at each merging step it uses upper approximations of documents in finding two closest clusters to merge.

In (Ho et al., 2002), we have applied TRSM and TRSM-based clustering algorithms to information retrieval and text analysis tasks. Interestingly, the TRSM cluster-based retrieval achieved higher recall than that of full retrieval in our experiments, especially the TRSM cluster-based retrieval usually offers precision higher than that of full retrieval in most experiments, and achieves recall and precision nearly as that of full search just after searching on one or two clusters.

5 Acknowledgments

I would like to express my deep thanks to Edwin Diday. His insight has been invaluable to me. I also wish to thanks people in his group for our long cooperation, especially Patrice Bertrand, Marc Csernel, Guy Cucumel, Eric Demonchaux, Roland Ducournau, Mireille Gettler-Summa, Yves Lechevallier, Joel Quinqueton, Henri Ralambondrainy, Myriam Touati.

References

HO, T.B., QUINQUETON, J., RALAMBONDRAINY, H., (1986): Using expert system techniques for interpretation of data analysis results. In: F. De Antoni, N. Laura, A. Rizzi (Eds.): *Proceedings of COMPSTAT'86.* Physica-Verlag Heidelberg Wien, 308–311.

HO, T.B. (1987): On the Design and Implementation of an expert system using the inference engine COTO. *Computers and Artificial Intelligence 6 (4), 297-310.*

HO, T.B., DIDAY, E., GETTLER-SUMMA, M. (1988): Generating rules for expert systems from observations. *Pattern Recognition Letters 7 (5), 265–271.*

HO, T.B. (1990): General-to-specific and specific-to-general algorithms in the CABRO concept learning method. *Proceedings of 1st Pacific Rim International Conference on Artificial Intelligence PRICAI'90*, 619–624.

HO, T.B. (1997): Discovering and using knowledge from unsupervised data. *Decision Support Systems 21(1), 27–41.*

HO, T.B., LUONG, C.M. (1997): Using case-based reasoning in interpreting unsupervised inductive learning results. *Proceedings of International Joint Conference on Artificial Intelligence IJCAI'97.* Morgan Kaufmann, 258–263.

NGUYEN, T.D., HO, T.B. (1999): An interactive-graphic system for decision tree induction. *Journal of Japanese Society for Artificial Intelligence 14(1), 131–138.*

HO, T.B., NGUYEN, N.B. (2002): Document clustering by tolerance rough set model. *International Journal of Intelligent Systems 17(2), 131–138.*

HO, T.B., KAWASAKI, S., NGUYEN, N.B. (2002): Cluster-based information retrieval with a tolerance rough set model. *International Journal of Fuzzy Logic and Intelligent Systems 2(1), 26–32.*

HO, T.B., NGUYEN, D.D. (2003): Learning minority classes and chance discovery. *Journal New Generation Computing 21(2), 149–161.*

LE, S.Q., HO, T.B. (2005): An association-based dissimilarity measure for categorical data. *Pattern Recognition Letters 26(6), 2549-2557.*

NGUYEN, C.H., HO, T.B. (2007): Kernel matrix evaluation. *Twentieth International Joint Conference on Artificial Intelligence IJCAI'07* (in press).

MICHALSKI, R., STEPP, R., DIDAY, E. (1983): Clustering objects into classes characterized by conjunctive concepts. In *Progress in Pattern Recognition, volume 1.* North Holland.

PAWLAK, Z. (1991): *Rough sets: Theoretical Aspects of Reasoning About Data*, Kluwer Academic Publishers.

PHAM, T.H., HO, T.B. (2007): A hyper-heuristic for descriptive rule induction. *International Journal of Data Warehousing and Mining 3(1), 54–66.*

WILLE, R. (1982): Restructuring lattice theory: An approach based on hierarchies of concepts, Rival, I. (Ed.) *Ordered Sets*, 445–470.

Mining Biological Data Using Pyramids

Géraldine Polaillon[1], Laure Vescovo[1], Magali Michaut[2], and Jean-Christophe Aude[2]

[1] Département Informatique, Supélec
Plateau de Moulon, 3 rue Joliot-Curie, 91192 Gif-sur-Yvette cedex, France,
geraldine.polaillon@supelec.fr, laure.vescovo@supelec.fr

[2] Service de Biologie Intégrative et de Génétique Moléculaire, CEA
CEA Saclay, 91191 Gif-sur-Yvette cedex, France, *magali.michaut@cea.fr, jean-christophe.aude@cea.fr*

Abstract. This paper is a review of promising applications of pyramidal classification to biological data. We show that overlapping and ordering properties can give new insights that can not be achieved using more classical methods. We examplify our point using three applications: (i) a genome scale sequence analysis, (ii) a new progressive multiple sequence alignment method, (iii) a cluster analysis of transcriptomic data.

1 Introduction

Biology has always benefited from advances in mathematics, more specifically in statistics and classification. Conversely, mathematical discoveries are interlinked with major challenges set down by biologists. Among the numerous examples of this "co-evolution" of sciences one can cite G.-L. Leclerc (1707-1788), known as *Comte de Buffon*, for his great work as both a naturalist and a mathematician. Recent technology breakthroughs have successively driven biology into the *genomic* and *post-genomic eras*. This quantum leap revealed the high complexity of biological organisms. Consequently the numerous and heterogeneous data produced every day require novel and efficient analysis methods for the biologists to investigate new hypotheses.

In 1984, Edwin Diday introduced the Pyramidal classification (Diday (1984)). It was one of the first methods that allowed determining and representing nested overlapping clusters. This approach became fully operational in 1990 with the publication of the complete ascending pyramidal classification algorithm (Bertrand (1990)).

The aim of this paper is to point out the potentiality of pyramids for the analysis of biological data. We present three applications dealing with genomic and transcriptomic data analysis. This examples illustrate that the inherent pyramid properties of overlapping and partial ordering can help with the interpretation of data.

This paper is organized as follows: first, two applications of pyramidal clustering are discussed on genomic data. One concerns genome scale sequence analysis, the other, the computation of multiple sequences alignments;

second, an application of pyramidal clustering is described with transcriptomic data obtained by DNA chips.

2 Genomic data

2.1 Genome scale sequence analysis

For several years, the success of numerous sequencing projects and their applications (*e.g.* transcriptom analysis) has led to the exponential increase of biological data. Thus, the availability of different genomes brought about the need for comparisons. For instance, by comparing the human genome with the genomes of different organisms, researchers can better grasp the structure and function of human genes and thereby develop new strategies in the battle against human diseases. In addition, comparative genomic (Konning et al. (1997), Park and Teichmann (1998)) provides a powerful new tool for the study of evolutionary changes among organisms, helping to identify genes that are conserved among species and genes giving each organism its own unique characteristics. In the context of comparative genomic, and among other methods, the pyramidal classification provided new interesting results (Codani et al. (1999), Aude et al. (1999)). More precisely, it allowed us to improve the representation and the analysis of the biological data. This point is fundamental: for example, it allowed to decipher the domain structure (functional subunit) of genes and to annotate genes (Louis et al. (2001)).

The following example deals with data from PHYTOPROT (Louis (2001)). This database is dedicated to the study of plants proteomes in order to elucidate functional relationships between genes of different species. All pairs of sequences have been compared and globally partitioned (Codani et al. (1999)); resulting clusters has been studied in details. Let study a family with the following proteins sequences: `APY_SOLTU` sequence of potato; `NTPA_PEA` sequence of garden pea; `004519`, `004520`, `022204`, `024091`, `023505`, `049676` sequences of *Arabidopsis thaliana*.

On figure 1.A, we have a dendogram obtained with the UPGMA clustering algorithm. We can observe two distinct clusters: the first one with the sequences `APY_SOLTU, NTPA_PEA, 049676`; the second one with all the others sequences. On figure 1.B, we have a pyramid computed on the same data. We rediscover both clusters, with an additional information. Indeed, the pyramid highlight the sequence `049676` as a link between both clusters.

Then the domains decomposition of the sequences is computed using MKDOM (see figure 2). In the first cluster, the sequences `APY_SOLTU, NTPA_PEA` have all their domains in common and share one of them with sequence `049676`. In addition `049676` shares two domains with the sequences of the second cluster. Therefore, domains decomposition confirm that sequence `049676` is a link between both clusters, as previously seen on the pyramid. The domain decomposition leads to the hypothesis that this sequence may be the

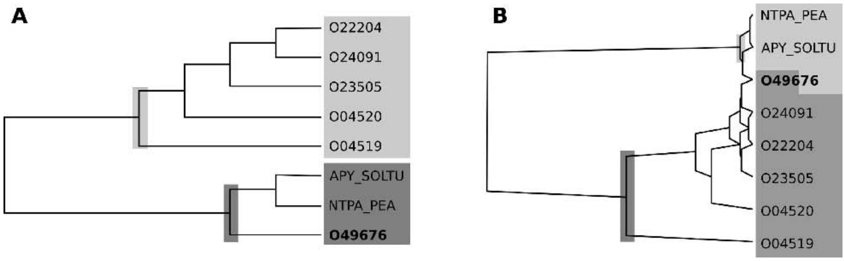


Fig. 1. A) The hierarchy obtained by the UPGMA method applied on a family of protein sequences from different plant organisms. One can unambiguously delineate two clusters, highlighted using grey boxes, from this hierarchy. **B)** The pyramidal representation obtained with the CAP algorithm on the same dataset. We observe two overlapping clusters, depicted by grey boxes. The intersection of both clusters is the sequence O49676. Thus, we can make the assumption that this sequence is the link between these two sets of sequences.

result of a gene fusion which is not detected by automatic syntactic annotation.

As a result, we can notice that the hierarchical representation is not able to determine links between two clusters. The pyramid with the properties of partial ordering and overlapping offers great interest for biological data. In this case, it permits to reconsider and correct the annotation of the sequence.

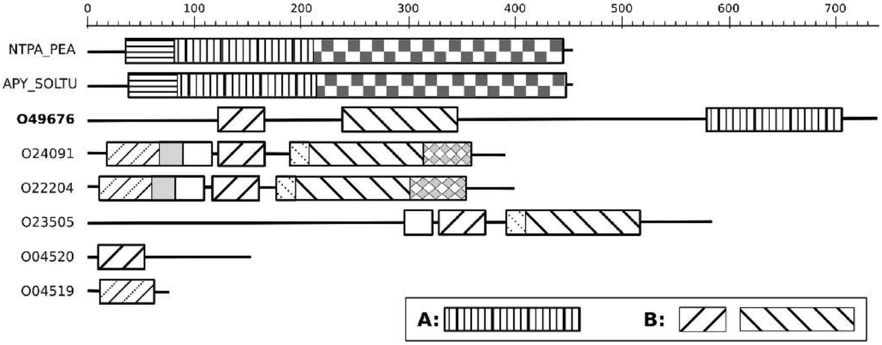


Fig. 2. This figure depicts the domains decomposition of protein sequences belonging to a PHYTOPROT family. They appear in the order given by the pyramid of figure 1.B. We observe that sequences of the first cluster (APY_SOLTU, NTPA_PEA, O49676) have the domain **A** in common, and the second cluster (all sequences from O49676 to O04519) the domains **B**. The intersection of both clusters is the sequence O49676, which possesses both domains **A** and **B**. It demonstrates that this sequence links both sets of sequences. Moreover it is revealed by a clear visual diagram.

2.2 Multiple sequence alignments computations

Using a set of nucleotidic or peptidic sequences, one can try to identify conserved sequence regions among them. The classic method to discover such patterns is to compute a multiple sequence alignment (Feng and Doolittle (1987)). A multiple alignment arranges the sequences in a scheme where positions believed to be homologous are written in a common column. Like in a pairwise alignment, when a given sequence does not possess a nucleotide or amino acid in a particular position an insertion (denoted by a dash) is added. Multiple sequence alignment is certainly one of the most used method in bioinformatic, and researches in this area are still undergoing development (Batzoglou (2005)). In practice, it is a key step in various sequence analysis and covers a wide field of applications,including: sequence annotation (Bulyk (2003)); function and structure (secondary or tertiary) prediction (Jones (1999)); phylogenetic studies (Phillips et al. (2000)).

Among the numerous algorithms available to compute such alignments, a common strategy, called progressive, emerged from the vast majority of these methods. This strategy is made of three steps: (i) a similarity matrix is calculated using the scores of a pairwise alignment method applied on all possible pairs of sequences; (ii) this matrix is used to compute a hierarchical tree, usually named guiding tree; (iii) finally, the bottom-up exploration of this tree is used to select the pair of sequences (or a previously aligned subset of sequences) to align. All published progressive algorithm alter or refine one or more of these steps (Lee et al. (2002), Edgar (2004), Do et al. (2005), Katah et al. (2005)). Recently, Vescovo et al. (2005) has undertaken a study to estimate the impact of selecting other guiding structure, using alternative algorithms and parameters (*e.g.* neighbor-joining, hierarchical tree build using different aggregation criteria...), on the resulting alignment. Indeed, until now we have little knowledge about the effect of this tree on the final alignment.

Progressive alignments methods also differ in the way they compute each pairwise alignments within steps (i) and (iii). Some of them use global alignments (*e.g.* ClustalW, Thompson (1994)) in which sequences are aligned on their whole length. Others use local alignments (*e.g.* PIMA, Smith and Smith (1992)) in which only subsequences are optimally aligned. Recently a third way, usually called mixed, has been investigated that combined both global and local alignments (*e.g.* M-Align, Van Walle et al. (2004)). This new approach seems to achieve better results using standard benchmark databases (see Van Walle (2004)). Hereafter we will describe a new mixed progressive alignment algorithm that uses pyramidal clustering as a key component (Vescovo et al. (2004)).

This new method introduces some modifications in step (ii) and (iii) of the progressive strategy described above. In step (ii) the modification is straightforward. The guiding tree, usually computed using the neighbor-joining algorithm (Saitou and Nei (1987)) is replaced by a pyramid computed using the CAP algorithm (Bertrand (1990)). The key idea is to use the overlapping properties of the pyramids to select the best alignment method (*i.e* global or local) in the step (iii). The principle of this algorithm is discussed with the example of sequences extracted from Thompson et al. (1994). The guiding structure is given in figure 3. Indeed, it makes sense to use local alignments when two set of sequences share a common pattern. This is precisely described by a cluster with a non empty intersection between its successors (*cf.* step 4 in figure 3). On the other hand, one can expect that successors with an empty intersection (*cf.* successors of step 5 in figure 3) don't reveal any shared pattern. In the latter case we would use global methods to align both sets of sequences. Moreover, the so-called local steps, such as step 4 in figure 3, require some adjustments in the definition of the two sets of sequences that are locally aligned. Basically, we have to deal with sequences that are present in each set, such as HBB_HUMAN in our example. We advocate to remove shared sequences from the largest set and to perform a local alignment. To preserve the key role of these shared sequences we also increase their weights, in the alignment procedure, to a significant extent.

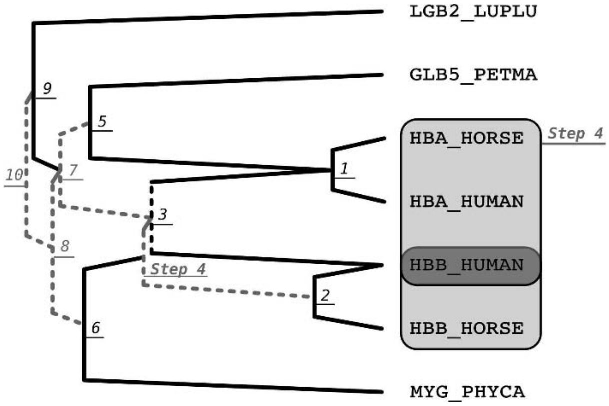


Fig. 3. This figure depicts a pyramid used as the guiding structure of a mixed progressive multiple sequence alignment algorithm. Labels on the right side of the figure are the swiss-prot accession numbers of the set of protein sequences to align. The pyramid is iteratively pruned by computing the consensus of the closest pair of sequences/consensus, according to the dissimilarity index (the steps numbers are indicated on each cluster). Solid lines indicate that the pair of sequences/consensus are aligned using a global method (steps 1, 2, 3, 5, 6, 9), whereas dashed lines indicate that sequences/consensus are aligned using a local algorithm (steps 4, 7, 8, 10).

We have successfully applied this new algorithm to the alignment of 11 homologous sequences from *Saccharomyces cerevisae* (*see* figure 4). All of them have been gathered in the same group using the genome scale analysis approach described in Codani et al. (1999) and previously explained. Querying the PFAM database (Bateman et al. (2004)), one can established that they have three domains in common: `Exo_endo_phos` (*exonuclease-endonuclease-phosphatase family*) depicted as black box on figure 4; `LRR` (*Leucine Rich Repeat*) as striped box; `PP2C` (*Protein Phosphatase 2C*) as grey box. But only `YAL021C` and `YJL005W` are composed of two distinct domains, respectively (`Exo_endo_phos`, `LRR`) and (`LRR`, `P2C`). Thus, a good multiple sequence alignment algorithm should not overlap these domains. To highlight the benefits of our strategy we have performed a comparison between three different softwares: ClustalW (Thompson et al. (1994)) the most used program to perform multiple sequence alignments that implements a global strategy; DiAlign (Morgenstern et al. (1996)) the standard local strategy method; PyrAlign (Vescovo et al. (2004)) the pyramid based mixed strategy described above.

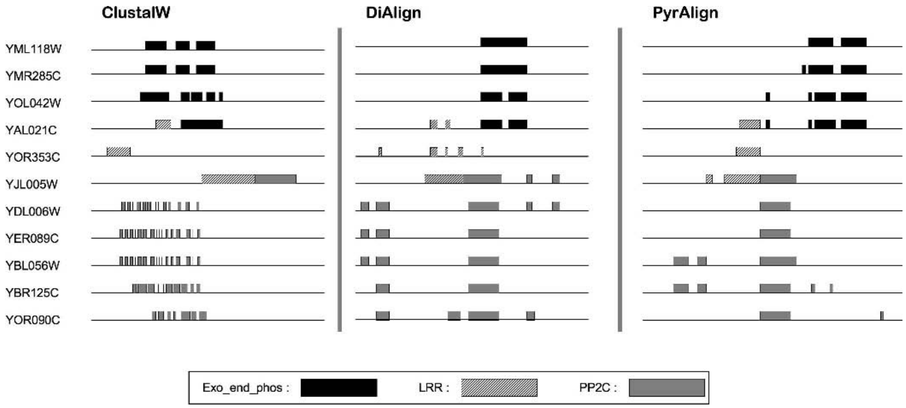


Fig. 4. This figure depicts the alignment of 11 sequences from *Saccharomyces cerevisae* computed by ClustalW, DiAlign and PyrAlign. The three domains, indicated within the box on the bottom of this figure, are used to benchmark these algorithms. ClustalW fails to correctly identify the domains: PP2C domain is split into many parts; LRR domain is not aligned across sequences; domains LRR and PP2C are overlapping. DiAlign also fails to build a correct alignment, domains are stacked and difficult to identify without supplementary knowledge. On the other hand, PyrAlign clearly delineates the domains and thus produces a better alignment.

On figure 4, one can easily notice that both ClustalW and DiAlign failed to split the three domains. Indeed, DiAlign stacks the `Exo_endo_phos` and `P2C` domains whereas ClustalW stacks all of them. PyrAlign is the only algorithm that clearly separate the domains, even if some of them are split in several parts such as `P2C`. Obviously all these programs fail to keep domains as continuous sequences of amino acids. For example ClustalW adds a lot of insertions within the `P2C` domain. DiAlign produces the same artifact when aligning the `LRR` domain. In one way, PyrAlign almost succeeds in keeping domains as single blocks, but is less effective in delineating their borders (*e.g.* the left side of the Exo_endo_phos domain). We can also argue that PyrAlign splits several domains, but domain borders are fuzzy and heavily depend on the underlying algorithmic used to inferred them. For instance the P2C domain depicted on gene YBR125C is defined as a single block in the PFAM database and as three blocks in the Panther database (Paul et al. (2003)). This example demonstrates the efficiency of the PyrAlign algorithm and its pyramidal guiding structure. However, due to the highest number of clusters in pyramids, this method has to compute more alignments than the others. Consequently the complexity of this algorithm is higher than any other progressive method. This could be an issue if one wants to compute multiple alignments of large sets of sequences.

3 Transcriptomic data

In the mid 90's, the DNA chip technology (Schena et al.(1995)) made a breakthrough in analyzing gene expression on a genomic scale (*i.e.* the transcriptom). It allowed to quantify the activity of hundred of thousands of genes under various conditions of given cell extracts. Nowadays, after many improvements, DNA chips are daily used by biologists around the world. As a consequence, large amounts of data have been produced by this technology. For instance, the GEO database already collected millions of expression profiles for over 100 organisms, submitted by over 600 researchers (Barrett et al. (2005)). A drawback of this technology is that measures are usually very noisy. Therefore, exhibiting significant variations is a challenging task (see (Speed (2003)) for details). Once these genes detected, one usually search for delineating co-expressed sets of genes. In this context, numerous clustering methods have been used (Eisen et al. (1998)). In this section we will detail the advantages of using pyramids to analyze DNA chips.

Our motivations were to investigate the partial order, induced by the pyramidal clustering, to delineate co-expressed and co-localized genes in prokaryotes. Indeed, such a set of genes, called *operon*, is regulated by the same promoter and transcribed as single mRNA transcript. Because of their unique operon structure, prokaryotes offer an additional feature to decipher the global regulatory network under various conditions (*e.g.* oxidative stress, inactivation of transcription factors...). Unfortunately, automatic discovery of operons from the genome sequence is a difficult task and no universal method has emerged yet. Thus, new approaches forged on the integration of other useful informations, such as gene expression data, have been tried (Sabatti et al (2002)). In the latter, authors have used a Bayesian classification scheme to predict whether the genes are in an operon or not. Since genes in operons are transcribed at the same level, Carpentier et al. (2004) have used this property to benchmark several micro-array clustering methods on their capability to gather such genes. In this section we will show that pyramidal classification is a very efficient method to discover sets of genes that are *potentially* transcribed as operon. Furthermore, pyramid graphs allow to easily identify co-expressed operon neighbors, providing a helpful tool to decipher regulatory mechanisms.

As part of the 2003-2006 French Nuclear Toxicology program (ToxNuc) we have been involved in the study of the effects of cadmium on several organisms. Cadmium and several cadmium-containing compounds are known carcinogens and can induce many types of cancer. This metal is used in many industrial processes such as metal plating and the production of nickel-cadmium batteries, pigments, plastics and other synthetics. Among the several organisms studied in this project, we focused our work on the cyanobacteria *Synechocystis*. *Synechocystis* is a unicellular non-nitrogen-fixing cyanobacterium and an inhabitant of fresh water. This organism has been one of the most popular organisms for genetic and physiological studies of photosynthesis. Our role in this project was to elucidate the molecular mechanisms involved in the cell response to cadmium toxicity. The transcriptom approach, using DNA chips, was used to characterize the kinetics of global changes in *Synechocystis* gene expression in response to continuous exposure to cadmium. Having processed all micro-arrays, we applied a linear model to exhibit significantly regulated genes. Then, we used a non-stringent p-value threshold ($p < 10^{-2}$), thus selecting ≈ 800 genes (*i.e.* the fourth of the entire genome). Finally a mixed hierarchical-pyramidal classification algorithm was designed to compare gene expression profiles based on their correlation. As a result we obtained a set of pyramids. The figure 5 is an excerpt of one of these pyramids that we will discuss hereafter.

Now we are able to check the ability of pyramidal clustering to efficiently report and predict operon genes. On figure 5 we have surrounded with grey box genes that are co-expressed and co-localized according to the pyramid. In addition we have checked that all genes of the same predicted operon are on the same DNA strain and oriented in the same direction. The first operon concerns genes involved in the motility of the cell. These proteins seem to be involved in bacteria fibrous proteins. These proteins are actually an operon (Yoshimura et al.(2002)) even if the gene `slr2018` is not annotated as a pilin-like protein. Furthermore these genes aren't neighbors within a hierarchy computed on the whole set of selected genes (data not shown). The second set of genes, predicted as an operon by the pyramid structure, reveal the efficiency of the method. On the figure we have manually annotated this operon as "hypothetical protein" because all corresponding genes have unknown functions according to Cyanobase (the cyanobacteria knowledge reference database). But mining other databases such as KEGG and the literature show that these genes are involved in the pilus assembly and required for mobility. Thus additionally to the fact that the method correctly predicts operon structures, it also gathers related operons. One more thing, the gene `sll1694` just below this operon is a known regulator of the pilus

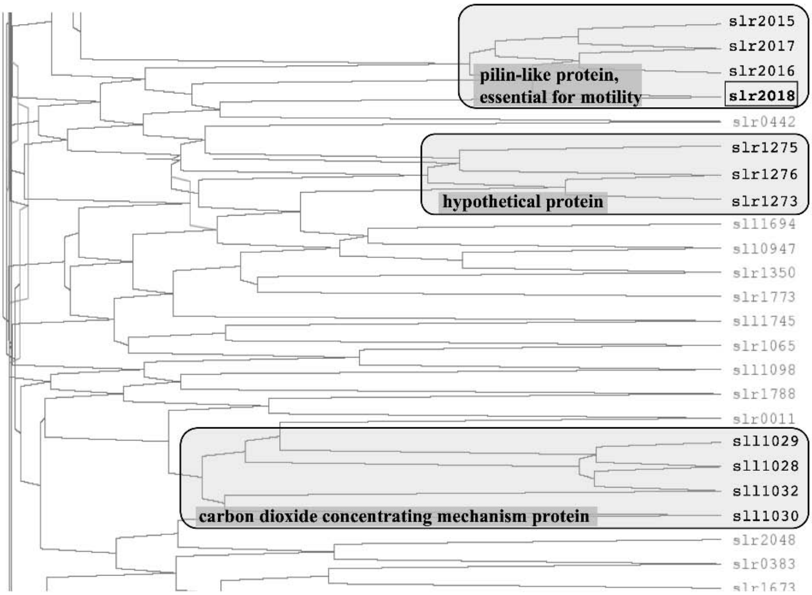


Fig. 5. This figure is an excerpt of the pyramidal classification of gene expression profiles from the cyanobacteria *Synechocystis*. Rounded boxes gather co-expressed and co-localized genes. These sets correspond to potential operons. Each of them is manually annotated using the Cyanobase database.

structure (Yoshihara et al.(2001)). Again this new element proved the accuracy of this approach. On the other hand `slr1274` has been missed, but one has to remember that these data are very noisy. The last predicted operon correctly gathered genes that are involved in the carbon dioxide concentrating mechanism. In this particular case our conclusions are motivated only by the similarity of genes annotations. Again, one gene `sll1031` is missing in this putative operon, for the same reasons as discussed previously.

In this section we have demonstrated the meaningful contribution of pyramidal clustering to the transcriptomic data analysis. This method should be considered with great interest for integrative approach of biological data analysis.

4 Discussion

In this article, we have shown the relevance of the pyramidal classification for biological data analysis. We illustrated this point through three different applications on *genomic* and *post-genomic* data. The first example discussed genome scale sequence analysis. It settled out the significance of clusters overlaps in deciphering links between families of proteins, thus improving sequences annotation. In the second example we used pyramids to specify a new algorithm for computing multiple alignment of sequences. This method implements a mixed progressive approach that is very promising compared to standard algorithms. The last example is about transcriptomic data clustering using pyramidal classification. Here we demonstrate the potential of the partial order, induced by the pyramid, to identify *operons*.

Thus, perspectives of using pyramids for the analysis of biological data are very encouraging. Besides the examples given in this article, there are still many fields, in biology, to investigate using pyramids. But some issues, like the poor readability of pyramidal graphs, complicate its adoption by researchers. This may be solved by both improving the mathematical framework (Bertrand and Janowitz (2002)), and developping new suitable visualization systems. Finally, we will have to overcome minds for considering overlapping.

However, the *pyramid* concept is largely adopted by the biologists community. Indeed, MEDLINE, the life science bibliographic information repository, already indexes more than 500 articles with the word *pyramid* found in the title. Futhermore, it is interesting to notice that one of the main *systems biology* article is titled **Life's complexity pyramid** (Oltvai and Barabasi (2002)).

5 Acknowledgements

Authors thank Peggy Baudouin-Cornu at the Integrative Biology Laboratory (CEA, France) for critical reading.

References

AUDE, J.-C., DIAZ-LAZCOZ, Y., CODANI, J.-J. and RISLER, J.-L. (1999): Application of the pyramidal clustering method to biological objects. *Computer and Chemistry 23(3-4), 303-315.*

BARRETT, T., SUZEK, T.O., TROUP, D.B., WILHITE, S.E., NGAU, W.-C., LEDOUX, P., RUDNEV, D., LASH, A.E., FUJIBUCHI, W. and EDGAR R. (2005): NCBI GEO: mining millions of expression profiles – database and tools. *Nucleic Acids Research, Database issue 33, D562-D566.*

BATEMAN, A., COIN, L., DURBIN, R., FINN, R.D., HOLLICH, V., GRIFFTHS-JONES, S., KHANNA, A., MARSHALL, M., MOXON, S., SONNHAMMER, E.L.L., STUDHOLME, D.J., YEATS, C. and EDDY, S.R. (2004): The Pfam protein families database. *Nucleic Acids Research 32, 138-141.*

BATZOGLOU, S. (2005): The many faces of sequence alignment. *Briefings in Bioinformatics 6(1), 6-22.*

BERTRAND, P. and DIDAY, E. (1990): Une généralisation des arbres hiérarchiques : les représentations pyramidales. *Rev. Statistique Appliquée 38(3), 53-78.*

BERTRAND, P. and JANOWITZ, M.F. (2002): Pyramids and Weak Hierarchies in The Ordinal Model for Clustering. *Discrete Appl. Math., 122, 55-81.*

BULYK, M.L. (2003): Computational prediction of transcription-factor binding site locations. *Genome Biol., 5(1), 201.*

CARPENTIER, A.-S., RIVA, A., TISSEUR, P., DIDIER, G. and HENAUT A. (2004): The operons, a criterion to compare the reliability of transcriptome analysis tools: ICA is more reliable than ANOVA, PLS and PCA. *Comput Biol Chem. 28(1), 3-10.*

CODANI, J.-J., COMET, J.-P., AUDE, J.-C., GLEMET, E., WOZNIAK, A., RISLER, J.-L., HENAUT, A. and SLONIMSKI, P.P. (1999): Automatic analysis of large scale pairwise alignments of protein sequences. In: A.G. Craig and J.D. Hoheisel(Eds.): *Methods in Microbiology: Automation, Genomic and Functional Analysis.* Academic Press, (28) 229-244.

DIDAY, E. (1984): Une représentation visuelle des classes empiétantes : les pyramides. *INRIA, Rapport de Recherche No. 291.*

DO, C.B. and MAHABHASYAM, M.SP. and BRODNO, M. and BATZOGLOU, S. (2005): ProbCons: Probabilistic consistency-based multiple sequence alignment. *Genome Research 15, 330-340.*

EDGAR, R.C. (2004): MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research 32(5), 1792-1797.*

EISEN, M.B. , SPELLMAN, P.T., BROWN, P.O. and BOTSTEIN, D. (1998): Cluster analysis and display of genome-wide expression patterns. *Proc Natl Acad Sci U S A. 95(25), 14863-14868.*

FENG, D.F. and DOOLITTLE, R.F. (1987): Progressive sequence alignment as a prerequisite to correct phylogenetic trees. *Journal of Molecular Evolution 25, 351-360.*

JONES, D.T. (1999): Protein Secondary Structure Prediction Based on position-specific Scoring Matrices. *J. Mol. Biol. 292, 195-202.*

KATOH, K., KUMA, K., TOH, H. and MIYATA, T. (2005): MAFFT version 5: improvement in accuracy of multiple sequence alignment. *Nucleic Acids Research 33(2), 511-518.*

KOONIN, E., MUSHEGIAN, A., GALPERIN M. and WALKER D. (1997): Comparison of archaeal and bacterial genomes: computer analysis of protein sequences predicts novel functions and suggests a chimeric origin for the archaea. *Mol Microbiol. 25, 619-637.*

LEE, C., GRASSO, C. and SHARLOW, M.F. (2002): Multiple sequence alignment using partial order graphs. *Bioinformatics 18(3), 452-464.*

LOUIS, A. (2001): La maitrise de l'information scientifique, clé de l'après séquencage *Thèse de l'Université Versailles Saint-Quentin.*

LOUIS, A., OLLIVIER, E., AUDE, J.-C. and RISLER, J.-L. (2001): Massive sequence comparisons as a help in annotating genomic sequences. *Genome Research 11, 1296-1303.*

MORGENSTERN, B., DRESS, A. and WERNER, T. (1996): DIALIGN: Multiple DNA and protein sequence alignment based on segment-to-segment comparison. *Proc. Nat. Acad. Sci. 32, 571-592.*

OLTVAI, Z.N. and BARABASI, A.L. (2002): Systems biology. Life's complexity pyramid. *Science 298(5594):763-4.*

PARK, J. and TEICHMANN, S. (1998): Divclus: an automatic method in the geanfammer package that finds homologous domains in single- and multi-domain proteins. *Bioinformatics 14, 144-150.*

PHILLIPS, A., JANIES, D. and WHEELER, W. (2000): Multiple sequence alignment in phylogenetic analysis. *Molecular Phylogenetics and Evolution 16(3), 317-330.*

SABATTI, C., ROHLIN, L., OH, M.K. and LIAO, J.C. (2002): Co-expression pattern from DNA microarray experiments as a tool for operon prediction. *Nucleic Acids Res. 30(13), 2886-93.*

SAITOU, N. and NEI, M. (1987): The Neighbor-Joining method: A new method for reconstructing phylogenetic trees. *Molecular Biology and Evolution 4, 406-425.*

SCHENA, M., SHALON, D., DAVIS, R.W. and BROWN, P.O. (1995): Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science. 270(5235), 368-371.*

SMITH, R.F. and SMITH, T.F. (1992): Pattern-Induced Multi-sequence Alignment (PIMA) algorithm employing secondary structure-dependent gap-penalties for comparative protein modelling. *Protein Engineering 5, 35-41.*

SPEED, T. (2003): *Statistical Analysis of Gene Expression Microarray Data.* Chapman & Hall / CRC, Boca Raton FL.

THOMAS,P.D., CAMPBELL,M.J., KEJARIWAL, A., MI, H., KARLAK, B., DAVERMAN, R., DIEMER, K., MURUGANUJAN, A. and NARECHANIA, A. (2003): PANTHER: a library of protein families and subfamilies indexed by function. *Genome Res. 13, 2129-2141 .Supplementary Materials.*

THOMPSON, J.D., HIGGINS, D.G. and GIBSON, T.J. (1994): Clustal W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Research 22(22), 4673-4680.*

VAN MALLE, I., LASTERS, I. and WYNS, L. (2004): Align-m - a new algorithm for multiple alignment of highly divergent sequences. *Bioinformatics 20(9), 1428-1435.*

VESCOVO, L., AUDE, J.–C., POLAILLON, G. and RISLER, J–L. (2004): Progressive multiple alignment based on pyramidal classification and applied to multi-domain proteins, *proceedings of the 12th International Conference on Intelligent Systems for Molecular Biology 2004, Glasgow, Scotland.*

VESCOVO, L., AUDE, J.–C. and POLAILLON, G. (2005): Guide structure calculation: a critical step for the accuracy of progressive multiple sequence alignment algorithms. *proceedings of the 4th European Conference of Computational Biology 2005, Madrid, Espagne.*

YOSHIHARA, S., GENG, X., OKAMOTO, S., YURA, K., MURATA, T., GO, M., OHMORI, M. and IKEUCHI M. (2001): Mutational analysis of genes involved in pilus structure, motility and transformation competency in the unicellular motile cyanobacterium Synechocystis sp. PCC 6803. *Plant Cell Physiol. 42(1),63-73.*

YOSHIMURA, H., YANAGISAWA, S., KANEHISA, M. and OHMORI, M. (2002): Screening for the target gene of cyanobacterial cAMP receptor protein SYCRP1. *Molecular microbiology 43(4), 843-853.*

Association Rules for Categorical and Tree Data

Henri Ralambondrainy and Jean Diatta

IREMIA, Université de la Réunion
15 av Cassin, 97715 Saint-Denis Message Cedex 9, France
ralambon@univ-reunion.fr, *jdiatta@univ-reunion.fr*

Abstract. The association rule mining problem is among the most popular data mining techniques. Association rules, whose significance is measured via quality indices, have been intensively studied for binary data. In this paper, we deal with association rules in the framework of categorical or tree-like-valued attributes.

1 Introduction

The association rule mining problem is among the most popular data mining techniques (Agrawal et al. (1993), Pasquier et al. (2000)). An association rule (AR) is an implication $U \to V$ that captures a certain relationship between binary attributes. In this paper, we deal with association rules in the framework of categorical or tree-like-valued attributes, using a meet-semilattice structure. We recall that a meet-semilattice is a poset that any two of whose elements have a greatest lower bound or ”meet”.

The paper is organized as follows:
Section 2 introduces AR and PQM in the context of characteristic functions lattice. Section 3 presents a way to define AR and PQM in a meet-semilattice. Section 4 an 5 respectively gives a language to represent categorical and tree data in a meet-semilattice structure. The paper is closed with a short conclusion and brief discussion about mining AR algorithms.

2 Association rules on binary data

2.1 The lattice of itemsets

Let U be a boolean map that represents a binary attribute, or a itemset that is a conjunction of binary attributes defined on a set of objects O. U takes its value in the domain $B = \{\mathbf{T}, \mathbf{F}\}$. The domain B is linearly ordered by $\mathbf{T} < \mathbf{F}$. We will consider U as a characteristic function of a subset U' of O, $U : O \longrightarrow B$. Let $\mathcal{P}(O)$ be the set of all subsets of O, and B^O the set of all characteristic functions, we will use the mapping $' : B^O \longrightarrow \mathcal{P}(O)$ to express that $U \in B^O$ is the characteristic function of the subset $U' = U^{-1}(\mathbf{T})$

(the "extension" of U). The set of all characteristic functions has a structure of boolean lattice

$$\mathcal{B} = (B^O; <, \vee, \wedge, \mathbf{1}, \mathbf{0}, \bar{\ })$$

isomorphic to the boolean lattice $\mathcal{P}(O)$. For all $o \in O$:

- $U \leq V \iff U(o) \leq V(o)$.
- $W = U \vee V \iff W(o) = U(o) \vee V(o)$. We have $W(o) = \mathbf{T}$ only if $U(o) = \mathbf{T}$ and $V(o) = \mathbf{T}$. It means concerning the extensions that

$$(U \vee V)' = U' \cap V'. \tag{1}$$

- $W = U \wedge V \iff W(o) = U(o) \wedge V(o)$. $W(o) = \mathbf{T}$ only if $U(o) = \mathbf{T}$ or $V(o) = \mathbf{T}$ then $(U \wedge V)' = U' \cup V'$.
- The smallest element of $\mathcal{B}$ is $\mathbf{0}$, the characteristic function of O i.e. $\forall o \in O, \mathbf{0}(o) = \mathbf{T}$.
- We denote by $\overline{U}$ the characteristic function complementary to U i.e. $\overline{U}(o) = \mathbf{T} \iff U(o) = \mathbf{F}$ and clearly $\mathbf{0} = \overline{U} \wedge U$.

An *association rule* (AR) is an ordered pair (U, V) of itemsets denoted $U \rightarrow V$. A rule quality measure is needed to capture relevant and interesting AR from the numerous number of potential candidates. We will be concerned with the so-called "Probabilistic Quality Measure" (PQM) (Diatta et al. (2007)).

2.2 Probabilistic quality measures

Let $U \rightarrow V$ an AR where U and V are itemsets. A *quality measure* is a real-valued function μ defined on $\mathcal{B} \times \mathcal{B}$. We will write $\mu(U \rightarrow V)$. A quality measure μ will be said to be probabilistic if it can be entirely expressed in terms of the probabilities $P(U')$, $P(V')$ and $P(U' \cap V')$. All these probabilities can be computed from the contingency table $K_{U \times V}$ (Table 1) defined from $n_U = |U'|, n_V = |V'|, n_{UV} = |U' \cap V'|$ et $n = |O|$. Some well-known probabilistic quality measures are given below:

- The support: $supp(U \rightarrow V) = \frac{|(U \vee V)'|}{|\mathbf{0}'|} = \frac{|U' \cap V'|}{|O|} = P(U', V')$
- The confidence: $conf(U \rightarrow V) = \frac{|(U \vee V)'|}{|U'|} = \frac{|U' \cap V'|}{|U'|} = P(V'|U')$
- The M_{GK} quality measure. The properties (i) and (ii) of the next remark may help to understand the definition of the PQM M_{GK}. Remark : Let U and V be two itemsets. Then, the following properties hold.
  (i) If U favors V, then $0 < P(V'|U') - P(V') \leq 1 - P(V')$.
  (ii) If U disfavors V, then $-P(V') \leq P(V'|U') - P(V') < 0$.
  (iii) "U disfavors V" is equivalent to "U favors $\overline{V}$"; indeed
  $1 - P(V') < 1 - P(V'|U')$ if and only if $P(\overline{V'}) < P(\overline{V'}|U')$.

Table 1. Contingency table K_{UV}

$V \diagdown V$	V	$\overline{V}$	
U	$n_{UV} = \lvert U' \cap V' \rvert$	$n_{U\overline{V}} = \lvert U' \cap \overline{V}' \rvert$	n_U
$\overline{U}$	$n_{\overline{U}V} = \lvert \overline{U}' \cap V' \rvert$	$n_{\overline{UV}} = \lvert \overline{U}' \cap \overline{V}' \rvert$	$n_{\overline{U}}$
	n_V	$n_{\overline{V}}$	n

Definition 1. The quality measure M_{GK} is defined by

$$M_{GK}(U \rightarrow V) = \begin{cases} \frac{P(V'|U')-P(V')}{1-P(V')}, & \text{if } U \text{ favors } V \\ & \text{or } U \text{ and } V \text{ are independent} \\ \frac{P(V'|U')-P(V')}{P(V')}, & \text{if } U \text{ disfavors } V \\ & \text{or } U \text{ and } V \text{ are independent} \end{cases}$$

It is easy to check that M_{GK} is a non symmetric PQM. Moreover, M_{GK} satisfies the three Piatetsky-Shapiro principles (Fayyad et al. (1996)). For more details, one can see (Diatta et al. (2007)). Now, we will define AR and PQM to categorical and tree data. For this task, we will consider meet-lattice structure.

3 Association rules on a meet-semilattice description context

Let $\mathcal{T}$ the objects description space and a map $\delta : O \longrightarrow \mathcal{T}$ which associates every element $o \in O$ with its description $\delta(o) \in \mathcal{T}$. We will notice that only the operator $\cap$ is used in the contingency table (Table 1) to compute a PQM. It means that only the structure of meet-semilattice $(\mathcal{P}(O); <, \cap)$ may be considered for AR and PQM. This remark motivates us to consider a space of description $\mathcal{T}$ having a structure of meet-semilattice. More precisely, we have

Proposition 1. *Let O be a set of objects and $\delta \in \mathcal{T}^O$ a description function. $(\mathcal{T}; <, \wedge)$ is a meet-semilattice, and define the map $\chi : \mathcal{T} \longrightarrow B^O$ that associates to each $u \in \mathcal{T}$ the characteristic function $U = \chi(u)$ of the subset $U' = \{o \in O | u \leq \delta(o)\}$, then*

1. *The map χ is a morphism between the semilattices $(\mathcal{T}; <, \wedge)$ and $(B^O; <, \vee)$ such that:*

$$\chi(u \wedge v) = \chi(u) \vee \chi(v). \tag{2}$$

2. *The map $' \circ \chi$ is a meet-preserving homomorphism between the meet-semilattices $(\mathcal{T}; <, \wedge)$ and $(\mathcal{P}(O); \subset, \cap)$ such that:*

$$\chi(u \wedge v)' = \chi(u)' \cap \chi(v)'. \tag{3}$$

Proof. We are going first to prove (3) before (2)

Observing that $U' = \{o \in O | u \leq \delta(o)\} = U^{-1}(\mathbf{T})$, we have

$$o \in U' = \chi(u)' \iff U(o) = \mathbf{T} \iff u \leq \delta(o).$$

By definition $\chi(u \wedge v)' = \{o \in O | u \wedge v \leq \delta(o)\}$, since $u \leq u \wedge v$ and $v \leq u \wedge v$, for each $o \in \chi(u \wedge v)'$, we have (3) $\chi(u \wedge v)' = \chi(u)' \cap \chi(v)'$ holds. $u \leq \delta(o)$ and $v \leq \delta(o)$, i.e. $o \in \chi(u)'$ and $o \in \chi(v)'$ then $o \in \chi(u)' \cap \chi(v)'$ and $\chi(u \wedge v)' \subset \chi(u)' \cap \chi(v)'$. Reciprocally, if $o \in \chi(u)' \cap \chi(v)'$, we have $u \leq \delta(o)\}$ and $v \leq \delta(o)$ and $u \wedge v \leq \delta(o)$ then $o \in \chi(u \wedge v)'$, and (3) $\chi(u \wedge v)' = \chi(u)' \cap \chi(v)'$ holds.

The equality (1) $(U \vee V)' = U' \cap V'$ is also written $(\chi(u) \vee \chi(v))' = \chi(u)' \cap \chi(v)'$ then from (3) it follows $\chi(u \wedge v)' = (\chi(u) \vee \chi(v))' \iff \chi(u \wedge v) = \chi(u) \vee \chi(v)$ (2) □

For $(u, v) \in \mathcal{T}^2$, we then identify a association rule $u \rightarrow v$, as the association rule $U = \chi(u) \rightarrow V = \chi(v)$. The set of itemsets is $\mathcal{T}$ to which is associated the binary itemsets: $\mathcal{M} = \{U = \chi(u) | u \in \mathcal{T}\} \subset \mathcal{B}$. The quality measure of $\mu(u \longrightarrow v)$ is $\mu(U \longrightarrow V)$. A PQM can be computed for any AR defined on $\mathcal{T}$ from the following indices

$$n_U = |U'| = |\chi(u)'|, n_V = |V'| = |\chi(v)'|, n = |O|.$$

and

$$n_{UV} = |U' \cap V'| = |\chi(u)' \cap \chi(v)'| = |\chi(u \wedge v)'|.$$

One important issue of the previous result is the following: it is not necessary to compute a binary context to find AR when the description space is a meet-semilattice. The research space for itemsets will be $\mathcal{T}$ and not the set of binary itemsets $\mathcal{M}$.

4 Association rules on categorical data

4.1 Representation of categorical data

An attribute A_q is said categorical, if its domain $dom(A_q)$ is a finite set of values. For example, the question "The Hot-Line (HL) service is ..." may be represented with a categorical attribute which domain is the following items {"No satisfactory", "Satisfactory", "Very satisfactory","No answer"}. The modality "No answer" may mean two things: the asked person did not give a response (the answer is "unknown") or the question has no sense because the asked person has not called the Hot-Line. To take into account these two possibilities we will denote by

- "$*$" an "unknown value", it also means that all the values are allowed
- "$\perp$" an "impossible value", when an answer has no sense. If necessary, this value may be added to the domain of an attribute.

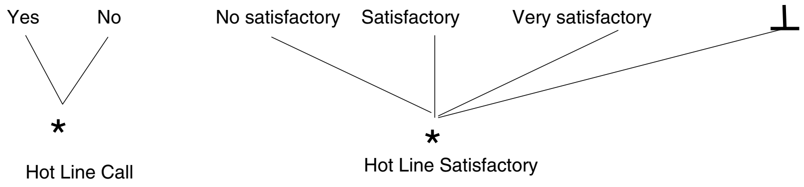


Fig. 1. Meet-semilattices on multivalued attributes.

The unknown value "$*$ " will be included to the domain of each categorical attribute: $D_q = dom(A_q) \cup \{*\}$. More precisely, each domain of any categorical attribute will be structured as a meet-semilattice $\mathcal{T}_q$ by considering $dom(A_q)$ as an antichain and $*$ as the smallest element of D_q (Figure 1)

$$\mathcal{T}_q = (D_q; <, \wedge, *).$$

The categorical data set is the description of the objects O with a set of categorical attributes $(A_q, \mathcal{T}_q)$ for $q \in Q$. Typically the data set that contains the answers of a poll conducted among a set of people is a good example of categorical data set. The objects description space is

$$\mathcal{T} = \Pi_q \mathcal{T}_q$$

which has a meet-semilattice structure as the product of the meet-semilattices $\mathcal{T}_q$. The description function is denoted $\delta : O \longrightarrow \mathcal{T}$ then

$$\delta(o) =< A_1 : w_1, \ldots, A_q : w_q, \ldots, A_p : w_p >\in \mathcal{T}$$

where $w_q \in \mathcal{T}_q$. An example of entity description is the following:

$$\delta(Jones) =< sex : "M", Region : *, HLCall : "no", HLSatisfaction : \perp >$$

For worth reading, we will not mention in a description, unknown values. For example, the previous description is written:

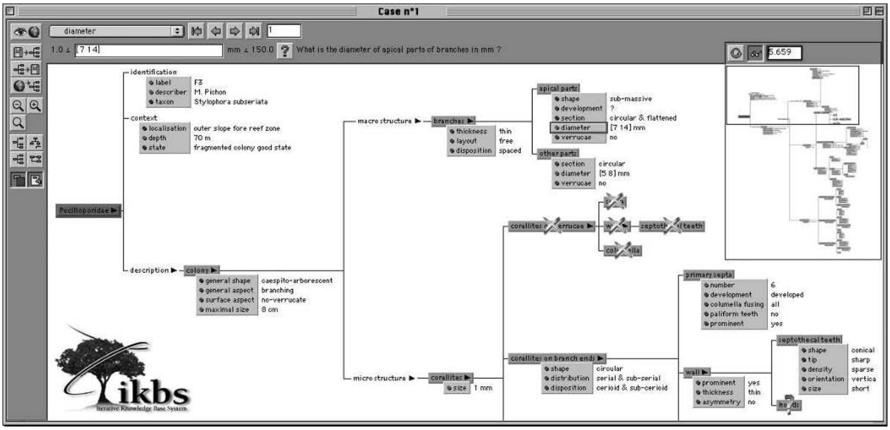
$$\delta(Jones) =< sex : "M", HLCall : "no", HLSatisfaction : \perp >$$

Let $u =< A_1 : u_1, \ldots, A_p : u_p >\in \mathcal{T}$ and $\delta(o) =< A_1 : w_1, \ldots, A_p : w_p >$. We will give the expression of the characteristic function $U = \chi(u)$. Denote by $(A_q : u_q)$ the characteristic function related to $< A_q : u_q >$:

$$(A_q : u_q) = \chi(< A_q : u_q >)$$

that is such as: $(A_q : u_q)(o) = \mathbf{T} \iff u_q \leq w_q$.
We have $U(o) = \mathbf{T} \iff u \leq \delta(o) \iff u_q \leq w_q$ for $q = 1, \ldots, p$, then it is obvious that

$$U = (A_1 : u_1) \wedge \ldots \wedge (A_p : u_p)$$



1.0 ≤ [7 14] mm ≤ 150.0 ? What is the diameter of apical parts of branches in mm ?

identification
- label F3
- describer M. Pichon
- taxon Stylophora subseriata

context
- localisation outer slope fore reef zone
- depth 70 m
- state fragmented colony good state

colony
- general shape caespito-arborescent
- general aspect branching
- surface aspect no-verrucate
- maximal size 8 cm

branches
- thickness thin
- layout free
- disposition spaced

apical parts
- shape sub-massive
- development ?
- section circular & flattened
- diameter [7 14] mm
- verrucae no

other parts
- section circular
- diameter [5 8] mm
- verrucae no

corallites
- size 1 mm

corallites on branch ends
- shape circular
- distribution serial & sub-serial
- disposition cerioid & sub-cerioid

primary septa
- number 6
- development developed
- columella fusing all
- paliform teeth no
- prominent yes

wall
- prominent yes
- thickness thin
- asymmetry no

septothecal teeth
- shape conical
- tip sharp
- density sparse
- orientation vertica
- size short

Fig. 2. A specimen of coral from *Pocilloporidae* family.

where ∧ is "the logical and". An association rule

$$u = < A_1 : u_1, \ldots, A_p : u_p > \rightarrow v = < A_1 : v_1, \ldots, A_p : v_p >$$

will be also written

$$U = (A_1 : u_1) \wedge \ldots \wedge (A_p : u_p) \rightarrow V = (A_1 : v_1) \wedge \ldots \wedge (A_p : v_p)$$

where $U = \chi(u)$ and $V = \chi(v)$. An example of AR on categorical data is given below:

$$(sex : "F") \wedge (age : "Old") \rightarrow (HLCall : "Many") \wedge (HLSatisfaction : "No")$$

5 Association rules on tree data

5.1 Representation of tree data

Many fields of real world applications, like biosystematics, deal with highly structured objects. For example, the Iterative Knowledge Base System (IKBS) (Conruyt et al. (1997)) has been design to extract knowledge from complex databases such as set of corals or sponges families. The Figure (2) displays a partial description of a specimen coral from the *Pocilloporidae* family.

We can notice its tree structure, missing components related to micro structure corallite, and some unknown values ("?"). To take into account all these features, we need to define a new type of attribute called "structured attribute", in contrast to categorical attributes that will be qualified as "simple". We assume that the domain of any attribute simple includes "unknown value" (∗) and "impossible value" (⊥).

Definition 2. Let (A_j, D_j), for $j \in J$, a set of simple attributes. A structured (or tuple) attribute tuple is a sequence

$$A :< A_1, \ldots, A_q, \ldots, A_p >$$

where A_q is a simple or structured attribute.

The domain $D = dom(A)$ of A is a set of structured values defined by induction, as following:

- If the type of A_q is simple, for $w_q \in D_q$ then $A_q : w_q \in D$
- Let $A_1, \ldots, A_p$ a set of simple or structured attributes.
  If $A_1 : w_1 \in D, \ldots, A_p : w_p \in D$ then $A :< A_1 : w_1, \ldots, A_p : w_p >\in D$ and is called "structured value".

Structured objects will be described using a tuple attribute $A :< A_1, \ldots, A_p >$ called data descriptive model (or schema in data base theory) and a map $\delta : O \longrightarrow D$ such as

$$\delta(o) =< A_1 : w_1, \ldots, A_q : w_q, \ldots, A_p : w_p >\in D$$

with : $w_q \in D_q$. The description δ may be graphically represented with a tree, where edges are the names of the attributes and the values of simple attributes are the leaves.

For example, the descriptive model of the *Pocilloporidae* family is the following tuple attribute

$$Pocilloporidae :< identification, context, description >$$

where $identification, context$ are simple attributes and $description$ is a tuple attribute:

$$description :< colony, macro - structure, micro - structure, ... >$$

A partial description of the entity "case1" is given below as illustration:

$\delta(case1) =< \ldots,$
$\quad micro - structure :< corallites :< size : 1, on - verrucae : \perp, \ldots >>,$
$\qquad \ldots >$

5.2 The meet-semilattice of tree data

The following proposition shows how to define a meet-semilattice on tree data.

Proposition 2. *Let $A :< A_1, \ldots, A_p >$ a tuple attribute. If each domain of the simple attributes A_j of A have a meet-semilattice structure $\mathcal{T}_j$ then the domain of A is meet-semilattice $\mathcal{T}$*

The proof is easy by induction. Let $A :< A_1 : v_1, \ldots, A_p : v_p >$ and $A :< A_1 : w_1, \ldots, A_p : w_p > \in D$ two structured values, the following properties hold.

- If A_j is a simple attribute, then in $\mathcal{T}_j$, we have:
  - $A_j : v_j < A_j : w_j \iff v_j < w_j$
  - $< A_j : v_j) \wedge (A_j : w_j) = A_j : v_j \wedge w_j$
- If $A_1, \ldots, A_p$ are simple attributes or structured attributes then
  - $A :< A_1 : v_1, \ldots, A_p : v_p > < \ A :< A_1 : w_1, \ldots, A_p : w_p >$ $\iff v_j < w_j$ pour $j = 1 \ldots p$
  - $A :< A_1 : v_1, \ldots, A_p : v_p > \wedge\ A :< A_1 : w_1, \ldots, A_p : w_p >$ $= A :< A_1 : v_1 \wedge w_1, \ldots, A_p : v_p \wedge w_p >$

In the meet-semilattice $\mathcal{T}$, RA can be defined and their significance measured with PQM.

An example of AR on corals from *Pocilloporidae* family is:

$$
\begin{array}{l}
(description :< colony :< general - aspect : branching >>) \\
\qquad\quad \wedge \\
(description :< macro - structure :< branches :< layout : free >>) \\
\qquad\quad \rightarrow \\
\quad (micro - structure :< corallites :< on - verrucae : \perp >>>)
\end{array}
$$

6 Conclusion

In this paper, we have dealt with association rules, mining with probabilistic quality measures, in the framework of categorical or tree-like-valued attributes. AR and PQM have been reformulated in the context of the lattice of characteristic functions. Then, we have shown how AR and PQM can be defined on a meet-semilattice. From this point of view, a way to structure categorical and tree data in a meet-semilattice has been introduced.

Next work will concern in developing algorithms to discover AR from categorical and tree data. Several approach has been proposed to extract AR from binary data sets. The A-priori (Agrawal and Srikant (1994)) or Close (Pasquier et al. (2000)) algorithm discovers relevant rules using support and confidence PQM constraints. Close extracts more efficiently frequent itemsets (itemsets which support are at least equal to user given minimum threshold) for AR using a Galois connection. As a huge number of redondeous rules may be extracted, methods for generating bases (Zaki and Ogihara (1998)) for association rules has been developed. A base is a minimal set of AR from which all rules, valid from a given quality measure, can be generated. When the objects description space has a meet-semilattice structure, the problem of discovering association rules and bases, will be addressed from theoretical results related to meet-semilattice: Galois connection (Brito (1994)) and conceptual weak hierarchy (Diatta and Ralambondrainy (2002)), multiway clustering (Diatta (2006), Diatta (2007)).

References

AGRAWAL, R., IMIALINSKI, T., SWAMI, A. (1993): Mining association rules between sets of items in large databases. In: P. Buneman and S. Jajodia (Eds.): *ACM SIGMOD International Conference on Management of Data.* ACM press, Washington,207–216.

AGRAWAL, R., SRIKANT, R. (1994): Fast algorithms for mining association rules. In: B. Jorge, Bocca, M.Jarke, and C. Zaniolo, (Eds.): *Proceed. of the 20 th VLDB Conference, 487-499.*

BRITO, P. (1994): Order Structure of Symbolic Assertion Objects. *IEEE Transactions on Knowledge and Data Engineering, 6 (5), 830–835.*

CONRUYT, N., GROSSER, D., RALAMBONDRAINY, H. (1997): IKBS: An Interative Knowledge Base System for improving description, Classification and identification of biological objects. *Proceedings of the Indo-French Workshop on Symbolic Data Analysis and its Applications 2, 212–224.*

DIATTA, J. (2006): Description-meet compatible multiway dissimilarities. *Discrete Applied Mathematics 154, 493–507.*

DIATTA, J. (2007): Galois closed entity sets and k-balls of quasi-ultrametric multiway dissimilarities, *Advances in Data Analysis and Classification 1, 53–65.*

DIATTA, J., RALAMBONDRAINY, H. (2002): The conceptual weak hierarchy associated with a dissimilarity measure. *Mathematical Social Sciences 44, 301–319.*

DIATTA, J., RALAMBONDRAINY, H., TOTOHASINA, A. (2007): Towards a unifying probabilistic implicative normalized quality measure for association rules. *Book Series Studies in Computational Intelligence* Springer Berlin/Heidelberg 43, 237-250.

FAYYAD, U.M., PIATETSKY-SHAPIRO, SMYTH, P. (1996): Knowledge discovery and data mining: towards a unifying framework. *Proceedings of the second International Conference on Knowledge Discovery and Data Mining.* Portland, OR, 82–88.

PASQUIER, N., BASTIDE, Y., TAOUIL, R., LAKHAL, L. (2000): Efficient mining of association rules using closed itemset lattices. *Information Systems 24, 25–46.*

ZAKI, M.J., OGIHARA, M. (1998): Theoretical foundations of association rules. *3rd SIGMOD'98 Workshop on Research Issues in Data Mining and Knowledge Discovery, 1–8.*

Induction Graphs for Data Mining

Djamel Abdelkader Zighed

ERIC Lab., University of Lyon 2
Campus Porte des Alpes, 5 av. P. Mendes-France, 69600 Bron, France,
abdelkader.zighed@univ-lyon2.fr

Abstract. Induction graphs, which are a generalization of decision trees, have a special place among the methods of Data Mining. Indeed, they generate lattice graphs instead of trees. They perform well, are capable of handling data in large volumes, are relatively easy for a non-specialist to interpret, and are applicable without restriction on data of any type (qualitative, quantitative). The explosion of softwares based on the paradigm of decision trees and more generally induction graphs is a rather strong evidence of their success. In this article, we present a complete method of induction graphs; the method SIPINA.

1 Introduction

In numerous domains such as Medicine, Sociology, Psychology, Meteorology, ... the specialists seek to predict a phenomenon. For this they use other phenomena which are easily accessible and supposed to be related to the phenomenon they aim to predict. In toxicology for example, the doctor attempts to identify the toxic agent absorbed by a patient in the coma by examining clinical-biological symptoms. He may wish to make available a prediction model that could help doctors to identify better and faster the cause of an intoxication at a patient. The objective of a modeling by induction graphs is to build a prediction model linking an attribute to be predicted, the toxic agent absorbed by the patient for example, to the explanatory attributes: pulse, temperature, state of consciousness, etc. Many approaches have been proposed so far. The nature (quantitative and/or qualitative) of the attributes used defines the choice of the mathematical framework in which one will place oneself. For example, if all attributes are quantitative, we can consider the techniques which are based on the linear algebra like the methods of linear regression or of discriminant analysis, whose detailed presentation can be found in various books such as Auray et al. (1991), Devijver and Kittler (1982) or Duda and Hart (1973). When the attributes are heterogeneous (certain are quantitative while others are qualitative) the decision trees such as CART (Breiman et al. (1984)) or with ID3 (Quinlan (1986)) and C4.5 (Quinlan (1993)) constitute suitable tool. The induction graphs generalize the decision tree concepts. All decision trees as well as induction graphs are based on very simple algorithms which lead to structures (tree or lattice) where each node corresponds to a subpopulation of individuals and where each branch corresponds to a value of a selected predictive attribute among all the others. The selection of the predictive attributes is based on a mathematical criterion. These selection criteria are based on information theory such as the entropy measures or statistic like the Chi-square.

The origin of research on induction graphs, certain authors, like Terrenoire (1970) and Tounissoux (1980), have based their research on Wald (1947) and Picard's works (1965). But the first algorithms which have led to decision trees as a particular structure of induction graphs, could be found in Morgan and Sonquist (1963). To select the attributes, these authors have used a statistical criterion. The consideration of a criterion resulting from information theory, to select the best attributes, has been proposed in the works of Picard (1965) and Terrenoire (1970). It is toward the end of the seventies that Quinlan (1986) began his work on the induction trees by publishing algorithm ID3. We have to mention that this algorithm was known and already used by Tounissoux (1974) and Routhier (1978). Some other approaches have been proposed among them we can mention the works of Ciampi et al. (1988).

The applications of induction graphs in the domains such as the diagnostic in medicine, sociology, marketing or archeology appeared at the begining of the seventies (Bertier and Bouroche (1981), Bouroche and Tenenhaus (1970), Laumon (1979), Ciampi (1989)).

In an induction graph, each path corresponds to a rule expressed in the form : If *condition*$_A$ then *conclusion*$_B$. The *condition*$_A$ represents a disjunction of a set of conjunctions. A conjunction is a set of logical propositions of type *attribute* $_X$=*value*$_y$. The whole number of rules constitute the prediction model, thus the rules are generally expressed in the formalism of the logic of propositions.

2 Framework and notations

Let Ω be a population of individuals or objects concerned with the problem of the prediction. To the members of this population is associated a particular attribute C, called class attribute. The determination of the forecast model ϕ is related to the assumption according to which the values taken by the statistical variable C are not due randomly but to particular situations that one can characterize. For that, the expert of the application domain draws up a list of statistical variables a priori called the exogenous variables which one notes $\mathbf{X} = (X_1, X_2, \ldots, X_p)$. The exogenous variables take their values on a representation space Σ that does not have any particular mathematical structure.

$$X : \Omega \longmapsto \Sigma \tag{1}$$
$$\omega \longmapsto \mathbf{X}(\omega) = (X_1(\omega), X_2(\omega), \ldots, X_p(\omega))$$

The objective is to seek a forecasting model ϕ such as, for an individual $\omega \in \Omega$ for which we do not know the class $C(\omega)$ but of which we know the state of all the exogenous variables $\mathbf{X}(\omega)$, we can predict his class of membership by means of ϕ. We expect to get for a large number of individuals of Ω:

$$\phi(\mathbf{X}(\omega)) = C(\omega) \tag{2}$$

3 Principle of induction graphs

The algorithm proposed by Zighed (1985), Zighed et al. (1992), Zighed and Rakotomalala (2000) we are going to describe, provides a set of successive partitions built on a learning sample Ω_l. These partitions are not necessarily hierarchical, but represent a lattice graph which acyclic. The construction algorithm is an heuristic which builds a succession of partitions by means of two operations : fusion and splitting. These operations are carried out on the terminal nodes of the induction graph. The terminal nodes define a partition on Ω_l. The objective of the algorithm of induction graphs is to optimize a criterion which evaluates the quality of the partition thus induced. This criterion will be defined later on.

In Figure 1, the main steps of construction are illustrated. Let us look at this graph as if it were a final result without being concerned with the details on how we exactly obtained it. We see that in the last partition we have 2 relatively contrasted terminal nodes. In s_7 we have 4 individuals out of 5 who

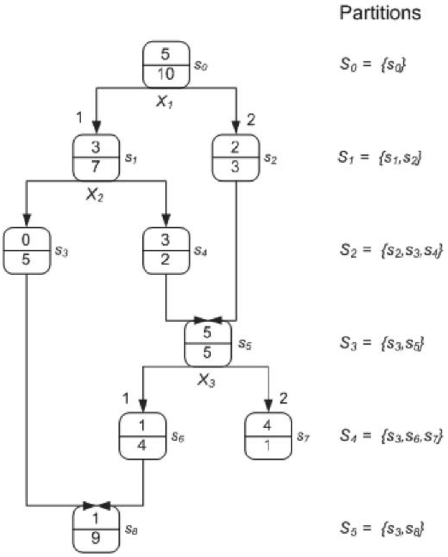


Fig. 1. Lattice graph.

belong to the class c_1, whereas in s_8 we have 9 individuals out of 10 who are in the class c_2. While following these two operations of splitting and fusion, we managed to build a partition in which almost all individuals of the same class are grouped in the same node. The process stops because none of the operations (splitting or fusion) generates a better partition. Provided that our sample is the representative of the original population, we can derive the prediction rules, R_1 and R_2, which are of the form :

If `condition` **Then** `conclusion` $(coef.)$

where `condition` is a logical expression composed of disjunction of conjunctions, and `conclusion` the majority class in the node described by the condition.

From the graph shown in Figure 1, we can naturally derive the two prediction rules that follow :

$$\begin{aligned} R_1 : & (X_1 = 1 \wedge X_2 = 1) \\ & \vee (X_1 = 1 \wedge X_2 = 2 \wedge X_3 = 1) \\ & \vee (X_1 = 2 \wedge X_3 = 1) \Rightarrow C = c_2(0.9) \end{aligned} \tag{3}$$

$$\begin{aligned} R_2 : & (X_1 = 1 \wedge X_2 = 2 \wedge X_3 = 2) \\ & \vee (X_1 = 2 \wedge X_3 = 2) \Rightarrow C = c_1(0.8) \end{aligned} \tag{4}$$

A coefficient *(coef.)* is generally associated to a rule. It reflects the degree of relevance of the rule. The proposed indicator is generally the proportion of individuals of the learning sample or the test sample associated to the majority class, but it could be any statistical coefficient such as τ of Goodman for example.

For instance, in the node s_8, the value of *coef* is $9/10 = 0.9$. That means that, 90% of the individuals of the learning sample verifying the condition of the rule, belong to the c_2 class. We hope naturally that such an assertion remains true for the general population.

4 Quality measure of partitions in the induction graph

Any partition S of the learning sample Ω_l, is perfectly described by a contingency table of m rows $(m > 1)$ that correspond to classes $c_i (i = 1 \ldots, m)$ and K columns $(K > 0)$ that correspond to its nodes $s_k (k = 1, \ldots, K)$. $K = 1$ means that all the individuals are in the same element (root node). In the example of Figure 1, the table T_2 associated to the partition S_2 is given by a contingency table :

We recall some notations and properties related to contingency tables.

- $n_{ij} \geq 0$ the size of population of the class c_i which is found at the node s_j
- $n_{i.}$ the total number of individuals belonging to the class c_i; $n_{i.} = \sum_{j=1}^{K} n_{ij}$

Classes (Rows) x Nodes (Col.)	s_2	s_3	s_4
c_1	2	0	3
c_2	3	5	2
Total per column	5	5	5

Table 1. Contingency table associated with the partition s_2 of Figure 1.

- $n_{.j}$ the total number of individuals belonging to the node s_j; $n_{.j} = \sum_{i=1}^{m} n_{ij}$
- n size of the whole sample $n = \sum_{i,j}^{m,K} n_{ij}$

In the process of construction of induction graph, we have found, in an iterative manner, a succession of partitions. We pass from the partition S_t to S_{t+1} if we improve the value of the criterion.

In other words, our criterion enables us to compare two partitions. Since to each partition S_t we associate a contingency table T_s with m rows and K columns, the criterion I that we seek to build must be a function of the contingency table T_s and takes its values in R^+ :

$$I : \bigcup_{k=1}^{\infty} R^{mk} \longmapsto R^+ \tag{5}$$

$$\forall T \in \bigcup_{k=1}^{\infty} R^{mk} \longmapsto I(T) \in R^+$$

Let S be a partition of Ω characterized by its contingency table T. The criterion I will have to verify the following properties:

- **Property 1 - Minimality** : The criterion I shall become minimal if in each node s_k of the partition S, all individuals belong to same class, i.e.

  $$\forall j \in \{1, \ldots, K\} \exists i \in \{1, \ldots, m\} : n_{ij} > 0 \texttt{ and } \forall q \neq i; n_{qj} = 0.$$

  On each column there is only one non null value.
- **Property 2 - Maximality** : The criterion I shall become maximal if in each node s_j of the partition S, all classes have same number of individuals.

  $$\forall j \in \{1, \ldots, K\} \forall (i, q) \in \{1, \ldots, m\}^2; n_{ij} = n_{qj}.$$

  It also means that all values of each column are equal.
- **Property 3 - Sensitivity to the size of the sample** : If we increase the size of the sample, for instance by multiplying the elements of contingency table T by a factor $\alpha > 1$, the value of the criterion I should decrease. For a given contingency table T relative to a partition S,

  $$\forall \alpha > 1; I(\alpha T) \leq I(T)$$

- **Property 4 - Symmetry** : For a given contingency table T, all permutations σ among the columns of T have no effect on the value of the criterion: $I(T_\sigma) = I(T)$. This constraint must be valid also for rows, i.e. all permutations δ among the rows of T have no effect on the value of the criterion : $I(T_\delta) = I(T)$.
- **Property 5 - Fusion** : If, among all columns of T, there exist two columns u and v which have the same probability distribution over the classes, then to merge these columns should lead to decrease of the value of the criterion. More formally, let's note by T_j the column j of the contingency table (distribution of the classes on the node s_j). The criterion $I(T)$ may be written as $I(T_1, \ldots, T_j, \ldots, T_K)$. For a given contingency table $T = (T_1, \ldots, T_j, \ldots, T_K)$ we require: if $\exists u, v \in \{1, \ldots, K\}$ and $\alpha > 0$ such that : $T_u = \alpha T_v$ then

$$I(T_1, \ldots, T_u, \ldots, T_v, \ldots, T_K) \geq I(T_1, \ldots, T_u + T_v, \ldots, T_K) \tag{6}$$

  This property enables us to reduce the complexity, number of nodes of the final partition, by merging nodes. this is the process which leads to a lattice structure.
- **Property 6 - Independence** :
  If we merge nodes which corresponds to adding the two respective columns in the contingency table, or if we create two columns by a split of one node, the change in the criterion must depend only on the merged columns or on the new nodes resulting from the segmentation.

$$I(T_1, \ldots, T_u, \ldots, T_v, \ldots, T_K) - I(T_1, \ldots, T_u + T_v, \ldots, T_K) = f(T_u, T_v) \tag{7}$$

On the basis of these six properties, we have built a family of evaluation criteria for a partition. The new criteria are derived from the classical measures of entropy such as the Shannons entropy or the Gini index.

Let's consider the parameter λ fixed by the user at a positive value. All following criteria check the six required properties.

- **Criterion based on Shannon's entropy** :

$$I(S) = \sum_{j=1}^{K} \frac{n_{.j}}{n} \left(- \sum_{i=1}^{m} \frac{n_{ij} + \lambda}{n_{.j+m\lambda}} \log_2 \frac{n_{ij} + \lambda}{n_{.j+m\lambda}} \right) \tag{8}$$

- **Criterion based on Gini's Index** :

$$I(S) = \sum_{j=1}^{K} \frac{n_{.j}}{n} \left(\sum_{i=1}^{m} \frac{n_{ij} + \lambda}{n_{.j+m\lambda}} \left(1 - \frac{n_{ij} + \lambda}{n_{.j+m\lambda}} \right) \right) \tag{9}$$

Many other similar functions may be built on the basis of the entropy measures.

The parameter $\lambda > 0$ controls the development of the graph and penalizes the nodes of weak size and so supports fusions between similar nodes. Many strategies could be adopted to fix it. The user may for instance fix this value arbitrarily , say 2. We have proposed an other more constructive procedure (cf. Zighed and Rakotomala (2000)).

5 The method SIPINA

5.1 Taking into account continuous attributes

There are many strategies for taking into account the continuous variables in the construction process of induction graphs. In the section below, we describe a very simple technique.

Let us consider the case where we wish to generate a partition from a node, say the root s_0 to simplify, using the continuous variable X_j. The only manner for reaching that point is to transform it into a discrete variable. Since all individuals of the learning sample Ω_l are in the root, the set of the values taken by the variable X_j is $X_j^{-1}(\Omega_l) = \{x_{j1}, \ldots, x_{j\alpha_j}\}$.

If we consider all these observed values on an axis, we note δ_{jk} the center of interval which has the boundaries x_{jk} and $x_{j,k+1}$

$$\delta_{jk} = \frac{x_{j,k+1} - x_{j,k}}{2} \tag{10}$$

We define $\alpha_j - 1$ medi-points in this way. Each one of them defines a bipartition of Ω_l that permits to transform the continuous variable X_j in an another binary variable X'_j in the following manner :

$$X'_j(\omega) = \{ \begin{matrix} 1 \ \texttt{if} \ X_j(\omega) \leq \delta_{jk} \\ 2 \ \texttt{if} \ X_j(\omega) > \delta_{jk} \end{matrix} \tag{11}$$

Thus, each bipartition defines two subpopulations $\Omega_1^k = \{\omega \in \Omega_l : X_j(\omega) \leq \delta_{jk}\}$ and $\Omega_2^k = \{\omega \in \Omega_l : X_j(\omega) > \delta_{jk}\}$, from which we can form a contingency table that shall serve to calculate the value of the criterion I.

The point of "optimal" discretization shall be the value δ_{jk} which leads to the minimum value of the criterion I. This point is sequentially searched among the all the possible values $\delta_{jk}; k = 1, \ldots, \alpha_j - 1$.

5.2 How to go from the partition S_i to the partition S_{i+1}

The Segmentation (Splitting) and the Fusion (Merging) of nodes will be the basic operations for building an induction graph. The algorithm aims therefore to seek, by mean of the two previous operations, the successive partitions such that at each iteration the uncertainty gain is maximized.

If we proceed systematically at splittings, we would take the risk of having a tree structure which might lead us to a too large number of nodes in the partition. More over, each node would have few individuals for being relevant from statistical point of view.

For this reason, we favor the merging. If no fusion enables us to obtain a new better partition, we proceed by fusion immediately followed by a segmentation.

Let's consider the example given by the Figure 2

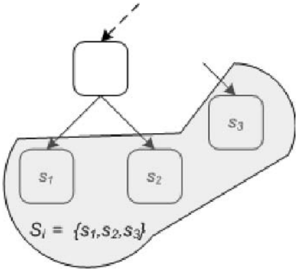


Fig. 2. Current partition.

The partition S_i has three nodes $\{s_1, s_2, s_3\}$. Let's assume that we have three binary exogenous variables X_1, X_2, X_3.

The passage of the partition S_i to partition S_{i+1} is carried out in three steps :

- Step 1- **Passage by Fusion** :
  On the example shown on Figure 3, one can note that from the partition S_i and by grouping pairs of nodes we can produce three different partitions :
  - $S^1_{i+1} = \{s_3, s_1 \cup s_2\}$ which gives an uncertainty gain of $\Delta(S^1_{i+1})$
  - $S^2_{i+1} = \{s_1, s_2 \cup s_3\}$ which gives an uncertainty gain of $\Delta(S^2_{i+1})$
  - $S^3_{i+1} = \{s_2, s_1 \cup s_3\}$ which gives an uncertainty gain of $\Delta(S^3_{i+1})$

  We denote by S^*_{i+1} the partition with:

$$\Delta I(S^*_{i+1}) = \max_{j=1,\ldots,3} \Delta(S^j_{i+1}) \tag{12}$$

  If the uncertainty gain is positive then $S_{i+1} = S^*_{i+1}$. The algorithm can then go back to the step 1 for generating one new partition. We will detail a bit more the general algorithm later; otherwise, i.e. if the uncertainty gain isn't positive ($\Delta(S^j_{i+1}) \leq 0$) then, one goes to step 2. Let's note, by the way, that the fusion is always carried out on a pair of nodes as shown in the Fig3

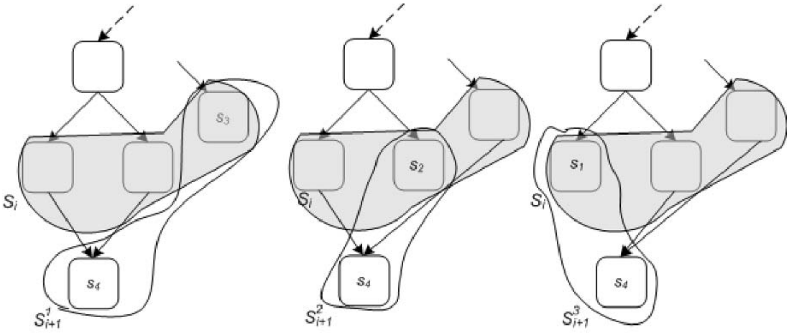


Fig. 3. Fusion.

- Step 2 - **Passage by Fusion / Segmentation** : Like in the step 1, we carry out all groupings between each pair of nodes. As shown on the Figure 4, we obtain three possibilities. On each node resulting from a merge, we look for the best variable X_j that leads, by segmentation, to the best partition which has the highest positive uncertainty gain. For example, on the same Figure, with three variables, we produce for each node, three partitions; That provides us nine partitions in all shown in the Figures 4.

  Then, among all the acceptable partitions, we will retain the one that leads to the positive highest value of uncertainty gain. Afterward, we can go back to step 1 to seek a new partition. If no partition resulting from this process has a positive value of uncertainty gain, then we will go at step 3.

- Step - 3 **Passage from S_i to S_{i+1} by segmentation** :

  On each terminal node S_i, we seek, by segmentation with all variables X_j, the best admissible partition. On the example shown by the Figure 5, with three variables, we derive, from each terminal node, three partitions, associated respectively to the three variables as indicated in the Fig5.

  Then, among all acceptable partitions, we will retain the one that has the highest positive value of uncertainty gain. Afterward, we can go back to step 1 to seek a new partition. If no partition resulting from this process has a positive value of uncertainty gain, then the algorithm stops and

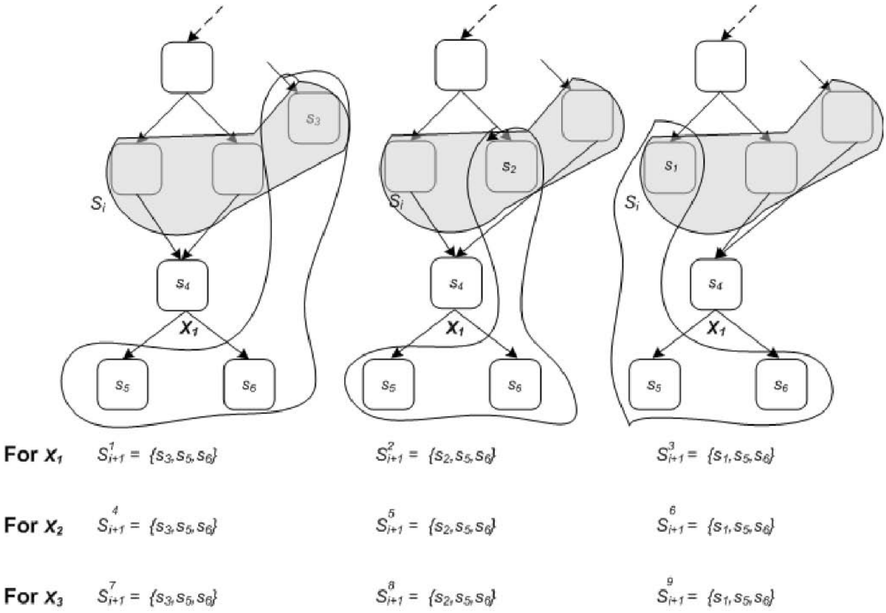

s3
S_i
s4
X1
s5
s6
s2
S_i
s4
X1
s5
s6
s1
S_i
s4
X1
s5
s6
For x1 S^1_{i+1} = {s3,s5,s6} S^2_{i+1} = {s2,s5,s6} S^3_{i+1} = {s1,s5,s6}
For x2 S^4_{i+1} = {s3,s5,s6} S^5_{i+1} = {s2,s5,s6} S^6_{i+1} = {s1,s5,s6}
For x3 S^7_{i+1} = {s3,s5,s6} S^8_{i+1} = {s2,s5,s6} S^9_{i+1} = {s1,s5,s6}

Fig. 4. Fusion followed by segmentation.

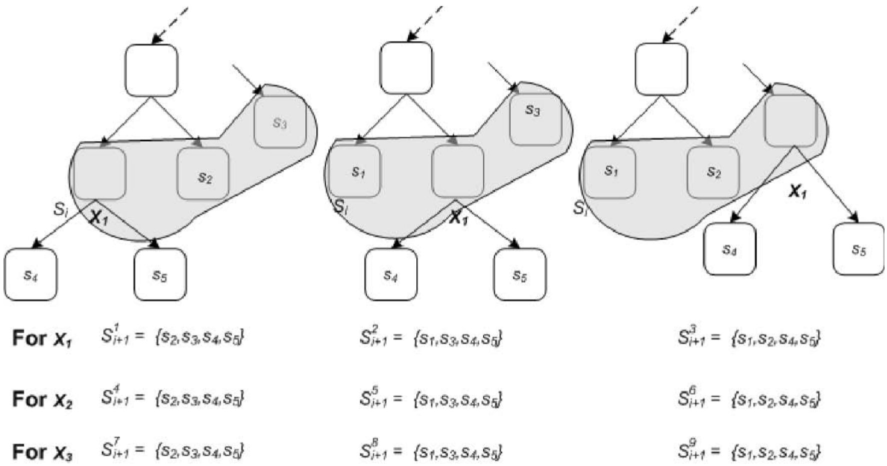

s3
s2
S_i
X1
s4
s5
s3
s1
S_i
X1
s4
s5
s1
s2
S_i
X1
s4
s5
For x1 S^1_{i+1} = {s2,s3,s4,s5} S^2_{i+1} = {s1,s3,s4,s5} S^3_{i+1} = {s1,s2,s4,s5}
For x2 S^4_{i+1} = {s2,s3,s4,s5} S^5_{i+1} = {s1,s3,s4,s5} S^6_{i+1} = {s1,s2,s4,s5}
For x3 S^7_{i+1} = {s2,s3,s4,s5} S^8_{i+1} = {s1,s3,s4,s5} S^9_{i+1} = {s1,s2,s4,s5}

Fig. 5. Segmentation.

that means that, among the possible operations described above, none can improve the criterion.

6 Conclusion

Method SIPINA belongs to the family of the methods largely exploited in the field of Data Mining. It provides a methodological framework which enables to generalize the concept of decision tree. It tries to bring rigorous answers to the concepts of minimal size of the nodes, of size of the tree, sensitivity to the size of the population. Let us say that SIPINA exploits the data much better than the other tree methods do and carries out some kind of pre-pruning. Even if a tree structure is simpler to read for a user, the graphs remain nevertheless an easy access. In book by Zighed and Rakotomalala (2000) we listed the totality of the methods containing graphs of induction and we provided many comparison tests among methods. We refer the reader to this book which exists unfortunately only in French for the moment.

References

AURAY, J.P., DURU, G. and ZIGHED D.A. (1991): *Analyse des données multidimensionnelles : les méthodes d'explication.* Editions A. Lacassagne, Lyon.

BERTIER, P. and BOUROCHE, J.M. (1981): *Analyse des données multidimensionnelles.* Presses Universitaires de France.

BOUROCHE, J.P. and TENENHAUS, M. (1970) Quelques méthodes de segmentation. *RAIRO 42, 29-42.*

BREIMAN, L., FRIEDMAN, J.H., OLSHEN, R.A. and STONE, C.J. (1984): *Classification and Regression Trees.* California: Wadsworth International.

CIAMPI, A., HOGG, S.A., McKINNEY, S. and THIFFAULT, J. (1988): RECPAM: a computer program for recursive partition and amalgamation for censored survival data and other situations frequently occurring in biostatistics I. Methods and program Features. *Computer Methods and Programs in Biomedicine 26, 239-256*

CIAMPI, A., THIFFAULT, J. and SAGMAN, U. (1989): RECPAM: a computer program for recursive partition and amalgamation for censored survival data and other situations frequently occurring in biostatistics II. Applications to data on small cell carcinoma of the lung. *Computer Methods and Programs in Biomedicine 30, 239-256*

DUDA, R.O. and HART, P.E. (1973): *Pattern Classification and Scene Analysis.* Wiley, N.Y.

DEVIJVER, P. and KITTLER, J. (1982): *Pattern Recognition: A Statistical Approach.* Prentice Hall.

LAUMON, B. (1979): *Une méthode de reconnaissance de formes pour l'estimation d'une variable continue : application à la docimologie.* PhD thesis, University of Lyon.

MORGAN, J. N. and SONQUIST, J. A. (1963): Problems in the analysis of survey data, and a proposal. *Journ. Amer. Stat. Assoc. 58, 415-434.*

PICARD, C. (1965): *Théorie des questionnaires. Les grands problèmes des sciences.* Gauthier-Villard.

QUINLAN, J.R. (1993): *C4.5: Programs for Machine Learning.* Morgan Kaufmann, San Mateo, CA.

QUINLAN, J.R. (1979): Discovering rules by induction from large collections of examples. In: D. Michie (Ed.): *Expert Systems in Micro Electronic Age.* Edinburgh University Press, 168-201.

QUINLAN, J.R. (1986): Induction of decision trees. *Machine Learning 1, 81-106.*

ROUTHIER, J.L. (1978): *Un processus d'interrogation latticiel : application á l'aide au diagnostic sur les nodules thyroidiens froids.* PhD thesis, University of Lyon 2.

TERRENOIRE, M. (1970): *Un modèle mathématique de processus d'interrogation : les pseudoquestionnaires.* PhD thesis, University of Grenoble.

TOUNISSOUX, D. (1974): *Pseudoquestionnaires et information.* Dissertation 3rd cycle, University of Lyon 1.

TOUNISSOUX, D. (1980): *Processus séquentiels adaptatifs de reconnaissance de Formes pour l'aide au diagnostic.* PhD thesis, University Claude Bernard - Lyon 1.

WALD, A. (1947): *Sequential Analysis.* Wiley.

ZIGHED, D.A. (1985): *Méthodes et outils pour les processus d'interrogation non arborescents.* PhD thesis, University Claude Bernard - Lyon 1.

ZIGHED, D.A., AURAY, J. P. and DURU G. (1992): *SIPINA : Méthode et logiciel.* Lacassagne.

ZIGHED, D.A. and RAKOTOMALALA R. (2000): *Graphes d'Induction : Apprentissage Automatique et Data Mining.* Hermès, Paris.

Part VI

Dissimilarities: Structures and Indices

Clustering of Molecules: Influence of the Similarity Measures

Samia Aci[1], Gilles Bisson[2], Sylvaine Roy[3], and Samuel Wieczorek[3]

[1] Centre de Criblage pour Molécules Bioactives
17, avenue des martyrs, 38054 Grenoble Cedex 9, France, *samia.aci@cea.fr*
[2] Laboratoire TIMC-IMAG, CNRS / UJF 5525
Domaine de la Merci, 38710 La Tronche, France, *gilles.bisson@imag.fr*
[3] Laboratoire Biologie, Informatique, Mathématiques, CEA-DSV-iRTSV
17, avenue des martyrs, 38054 Grenoble Cedex 9, France, {*samuel.wieczorek, sylvaine.roy@cea.fr*}

Abstract. In this paper, we present the results of an experimental study to analyze the effect of various similarity (or distance) measures on the clustering quality of a set of molecules. We mainly focused on the clustering approaches able to directly deal with the 2D representation of the molecules (*i.e.*, graphs). In such a context, we found that it seems relevant to use an approach based on asymmetrical measures of similarity. Our experiments are carried out on a dataset coming from the High Throughput Screening HTS domain.

1 Context

The discovery or the synthesis of molecules that activate or inhibit some biological systems is a central issue for biological research and health care. The High Throughput Screening (HTS) of a chemical library is a systematic approach to deal with this problem that has been used since twenty years by the pharmaceutical industry and more recently by academic researchers.

The objective of HTS is to rapidly evaluate, through automated approaches, the activity of a given collection of molecules on a given biological target that can be an enzyme or a whole cell. In practice, the results of a HTS test allow to highlight some tens of active molecules, named the "hits", representing a very small percentage of the initial collection. Indeed, the size of this collection, in academic laboratories, is typically between 10^3 and 10^6 molecules. However, these tests are just the beginning of the work since the identified molecules generally do not have some good characteristics in terms of *sensitivity* and *specificity* (a relevant molecule must by specific to the biological target and should be efficient with a small concentration). In addition, the results of HTS tests contain relatively high rates of false positives (molecules wrongly selected) and false negatives (molecules wrongly rejected).

In such a context, it is crucial to provide the chemists with some tools to explore the contents of the chemical libraries and especially to make easier the search for molecules that are structurally similar to the hits. A possible approach, given a relevant distance, is to seek the nearest neighbors of those hits. More broadly, chemists have a need for methods to automatically organize the collections of molecules in order to locate the active molecules within the chemical space. Above all, they would like to evaluate the real diversity of the chemical structures contained in a collection (this aspect is meaningful to decide the purchase of a set of molecules: the higher the diversity is, the smaller the collection to buy is). For all these constraints, the size of the collections makes manual approaches unfeasible.

Clustering methods (Berkhin (2002)) are well suited to carry out this type of task. However, with structurally complex objects such as molecules, it is obvious that the quality of the results depends on the capacity of the distance used by the clustering method to grasp the structural likeness and dissimilarities. Thus, in this article, we give an experimental study of the behavior of some classical structural distances proposed for the problem of molecule clustering. The outline of this paper is the following: in Section 2, we will present the distances we want to compare. In Section 3, we will detail the experimental material as well as the methodology employed to evaluate the clustering results. Finally, the results of our experiment will be described and commented in Section 4.

2 State of the art

2.1 Distances between molecules

The evaluation of a distance between two objects such as molecules (and more generally graphs) is a complex problem insofar as it requires, directly or indirectly, the search for partial isomorphic graphs. However, this difficulty can be overcome by using some alternative representations. For example, we can linearize the molecule as in the language SMILE of Weininger (1988) or, in an even more drastic way, we can turn the molecule into a collection of set of fragments (molecule subset), chosen by the system (Chemaxon) or specified by a set of criteria as in MolFea (Helma et al. (2003)). In the latter, the compounds can be represented by a vector of structural descriptors (named "structural keys"), each descriptor corresponding to one fragment of the molecule.

More recently, in the context of the Support Vector Machines, several kernel functions (comparable to distances) were proposed to deal with graph structures. These approaches obtain good performances in supervised learning to predict the activity of a molecule (Mahé et al. (2005)). Gartner et al. (2003) proposes a survey about the kernels that can be used with different kinds of structured representation and it is interesting to check if these kernels are directly usable within the framework of molecules clustering. Among recent works, we can also mention the marginalized kernels developed by Kashima et al. (2003) and extended by Mahé et al. (2004). In all of these approaches, the molecule representation is carried out globally by building an explicit or implicit collection of paths (linear fragments of the molecule) selected by the user or randomly drawn. The multiplication of the paths ensures a good sampling of the molecular structures.

However, it is also possible to assess the distance between molecules in a more dynamic way according to the actual possible matching between a given pair of molecules A and B (or graphs). Here, the idea is first to evaluate the quality of the local mapping between each pair of atoms in A and B and second, to find the best global matching. Such approach has been proposed by Frölich et al. (2005) who developed a kernel named "optimal matching" in order to predict the molecules activity on HTS tests. In the same way, we propose here an index, named Ipi, based on a close strategy (Wieczorek et al. (2006)), even though our motivations are different.

In the rest of the current section, we introduce the kernel function and how to derive a distance from them. First, we describe two kernels using a linear representation of the molecules, namely the Tanimoto kernel and the extension of the marginalized kernel. Then, we describe the optimal matching kernel and the Ipi index.

2.2 Kernel function and distance

Kernel functions are the basis of machine learning methods such as Support Vector Machines (SVMs). These functions map a set of objects from their input space (the space in which the objects are described) to a higher dimensional space, the so-called feature space F, where the inner product between the images of the objects is evaluated. To do so, one considers the set $X = x_1, ..., x_n$ of n objects and the feature map ϕ defined by: $\Phi : x \in X \mapsto \Phi(x) \in F$. A kernel is a function k such that for all $(x, y) \in X^2$ that satisfies: $k(x, y) = \langle \phi(x), \phi(y) \rangle$ where $\langle ..., ... \rangle$ is the inner product. One can derive the Euclidean distance between $\phi(x)$ and $\phi(y)$ in the feature space as follows:

$$\begin{aligned} \|\phi(x) - \phi(y)\|^2 &= \phi(x).\phi(y) - 2\phi(x).\phi(y) + \phi(x).\phi(y) \\ &= k(x, x) - 2k(x, y) + k(y, y) \end{aligned}$$

2.3 Tanimoto kernel

The Tanimoto kernel (Ralaivola et al. 2005) counts and compares walks in the underlying graphs. The idea is to represent each graph by a vector where each position is a boolean indicating the presence or not of a possible walk (a sequence of atoms) in the graph. Given two graphs x and y, the kernel $k_d(x, y)$ counts the number of common walks between x and y ($i.e$ the number of bits simultaneously equal to 1). The Tanimoto kernel k_d^t is defined as:

$$k_d^t(x, y) = \frac{k_d(x, y)}{k_d(x, x) + k_d(y, y) - k_d(x, y)}$$

The only parameter to set is the maximum length of walks in the graphs. In our experiments, we have set this value to 8 that is a classical value in chemoinformatics. This measure is similar to the Jaccard index.

2.4 Extension of the marginalized kernel

In the marginalized kernel (Kashima et al. (2003)), or MG-kernel, descriptors correspond to sequences randomly extracted from each molecule. Thus, the molecules are implicitly described by a vector of sequences as in the Tanimoto kernel, leading to some information loss when, for instance, some physicochemical properties are known about the atoms (charge, etc.). The marginalized kernel is defined as the sum of the similarities between all pairs of sequences extracted from each molecule. Each of these similarities is given by a kernel function, defined by the user: a simple definition consists in assigning a similarity of 1 if the sequences are equal and 0 if not.

We notice that Mahé et al. (2004) propose two extensions of this definition. First, the introduction of the Morgan index (that gives an additional knowledge about the atoms) helps to discriminate the atoms according to their environment. Second, in the new definition, the exploration of the graphs avoids to take into account any vertices (atoms) that have been previously visited; in this way, the extracted paths are more significant when the molecules contain many rings. We will use these extensions (implemented in the extMG-kernel) in our tests. This method has two main parameters:

- the number of iterations used to compute the Morgan index. We have used the value of 3 which gives the best results in the classification experiment in Mahé et al. (2004),
- the probability p_q of the search termination for the walks in the graphs. A value close to 1 (resp. close to 0) will generate short walks (resp. long walks). In our experiment, we tested three values: 0.1, 0.5 and 0.9.

2.5 Optimal assignment kernel

The Optimal Matching kernel (OA-kernel), introduced by Frölich et al. (2005), is based on a dynamic and local exploration of the molecular graphs. Contrary to the Tanimoto kernel and, to some extend, to the extMG-kernel, the representation language can take into account, in addition to the 2D structure, all the physicochemical knowledge concerning the properties of the atoms and their bonds. The kernel is computed in two steps that are conceptually close to those proposed in Bisson (1995). During the first one, the system assesses the value of the kernel k_nei between each pair of atoms in the two molecules. Once the matrix k_nei has been calculated, the second step consists of finding the best mapping between the atoms of molecules A and B in order to maximize the sum $k_nei(a_i, b_j)$ over the atomsa_i and b_j. This phase corresponds to the search of the maximum weight matching in a bipartite graph.

2.6 Structural similarity index Ipi

The index Ipi is based on the works of Bisson (1992), Bisson (1995) and Wieczorek et al. (2006) concerning the calculus of similarities between two any graphs. As in Frölich et al. (2005), the evaluation of the similarity relies on two steps, a local and a global one, but with two differences.

Local similarities between atoms. This step aims to compute a local similarity between each pair of atoms (a_i, b_j) belonging respectively to the molecules A and B. The key idea is to consider that two atoms a_i and b_j are more similar as they share common physicochemical properties, but also that the neighboring atoms to which they are bound by covalent bonds are themselves similar to each other. This recursive definition allows to express the problem in the form of an equation system (the algorithm is detailed in Wieczorek et al. (2006). During the resolution of this system, the structural similarities and dissimilarities A and B are automatically taken into account. The difference with the previously described approaches is that, here, we do not compute one local similarity but two: the (asymmetrical) similarity of a_i with respect to b_j and those of b_j with respect to a_i. The general motivation is the following. Let us consider two molecules A and B whose sizes (in terms of the number of atoms) are very different and such that A is included in B (in a representation by graphs, that means that A is a subgraph of B). From the point of view of A, the molecule B is very similar since it contains the same information. It is obviously not the case for B. In the context of a classical symmetrical similarity, the similarity between A and B should be mainly influenced by their difference of size. In our approach, using the mean of both inclusion values (those of A in B and vice versa) leads to a more realistic similarity allowing to break this size bias and to focus deeper on the existence of common substructures.

Global similarities between molecules. The goal of this step is to compute a global inclusion between two molecules A and B, denoted $Ipi(A, B)$. For that, we search, for each of the two inclusion matrices previously built (the one taking the atoms of A as reference and the other taking the atoms of B), a matching that maximizes the global inclusion. However again, we do not use the method of Frölich et al. (2005) because it leads to increase erroneously the value of inclusion between the molecules. Indeed, during the evaluation of the local inclusions, the values obtained between each pair of atoms (a_i, b_j) correspond to an optimum matching from a *local point of view.* Thus, the maximum weight matching can lead to choosing a set of matching decisions between atoms that are globally *pairwise exclusive* (see Subsection 4.1). Therefore, our search of the best matching is based on an heuristic parallel exploration of the two molecular graphs, guided by the similarity values between the atoms. That leads to identify the greatest common substructure between A and B. Finally, as we said earlier, the global similarity between the two molecules is then the mean of $Ipi(A, B)$ and $Ipi(B, A)$.

3 Experimental material and methodology

3.1 Chemical libraries

Two different chemical datasets have been used in the tests. They come from Sutherland et al. (2003) and contain the 2D structure of molecules and the main physicochemical properties of the atoms (using the SDF format). Those bases present the benefit of being already divided into well-defined chemical families, based on the molecular structures. The *Cox2* library contains a set of 467 molecules tested as inhibitors of the cyclo-oxygenase-2, divided into 13 families and the *Dhfr* library contains a set of 756 inhibitors of the dihydrofolate reductase, divided into 18 families.

3.2 Representation of the molecules

In the clustering, the quality of the discovered classes depends first on the significance of the features selected to represent the data and second on the adequacy between this representation and the current problem. The goal of this study is to carry out a clustering of molecules into chemical families based on the 2D structural properties of the molecules rather than on the interaction properties (for example, the topological distance between pharmacophores points).

The representation used is identical to those used in Frölich et al. (2005): one considers each molecule as a labeled graph whose nodes and edges correspond respectively to the atoms and the bonds; each atom is described by ten structural properties. In the case of the Tanimoto kernel and the extMG-kernel, the descriptions, in the form of walks in the graphs, are automatically collected by the methods.

3.3 Clustering methods

Many attempts have been done to use kernels in clustering algorithms and more generally to the domain of unsupervised learning. The main idea consists of mapping the instances in the feature space by means of kernels (functions) and to search clusters in this space. Ben-Hur et al. (2001) has modified the SVM algorithm to realize the clustering task by means of a Gaussian kernel. Another approach, proposed by Dhillon and Guan (2004), consists to use the k-means algorithm directly in the feature space. This is done by turning the Euclidean distance (usually used by k-means) to the distance between two objects in the feature space (i.e. the distance calculated from the kernel).

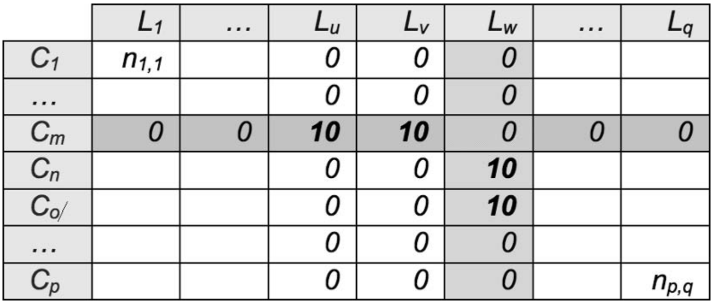
As for the latter, we have combined kernels (or similarity for Ipi) described in Section 3 with two standard clustering algorithms: the well-known Hierarchical Ascendant Classification (Berkhin (2002)) and Jarvis-Patrick (Jarvis and Patrick (1973)). The latter is largely used in chemoinformatics where it is the basis of commercial tools for clustering of molecules. Both methods need to adjust different parameters. As it is not a priori possible to identify the values leading to the best clustering, we have tested several combinations within ranges of values for each parameter.

3.4 Evaluation of the clustering

As emphasized in Candellier et al. (2006), it is difficult to evaluate the quality of clustering results without any validation criteria. That is fortunately not the case here since we know, for each dataset, the number of families (classes) that must be found by the system as well as the extensional description of these classes. We can then evaluate the results provided by the different distances and methods by measuring the difference between the original clusters and the learnt ones. The result of a categorization may be represented quantitatively in the form of a confusion matrix (Figure 1).

In this matrix, where the C_i represent the original classes and L_j the classes learnt by the system, each value $n_{i,j}$ represent the number of molecules that are simultaneously present in the classes C_i and L_j. There is a perfect match between the clustering when this matrix contains only one non-zero value for each line and each column.

A simple way to quantify the quality of a clustering is then to evaluate the mean of conditional entropies, which are associated to the lines and the columns of the matrix. It is necessary to distinguish between lines and columns since they contain different informations, as illustrated by Figure 1).



	L_1	...	L_u	L_v	L_w	...	L_q
C_1	$n_{1,1}$		0	0	0		
...			0	0	0		
C_m	0	0	**10**	**10**	0	0	0
C_n			0	0	**10**		
C_o/			0	0	**10**		
...			0	0	0		
C_p			0	0	0		$n_{p,q}$

Fig. 1. Confusion matrix to evaluate a clustering.

In the case of the class C_m, the categorization built by the system is not incorrect: this class has just been split in two new classes L_u and L_v, what is not open to criticism. However, in the case of L_w, the problem is quite different because the initial classes C_o and C_n have been merged. Thus, we have to consider two measures: the *Confusion Index* (CI) that quantifies the number of merged classes and the *Segmentation Index* (SI) which quantifies the number of split classes.

$$CI = -\sum_{i}^{p} \frac{\bar{C}_i}{N} \sum_{j}^{q} \frac{n_{i,j}}{\bar{C}_i} \log_2 \frac{n_{i,j}}{\bar{C}_i} \qquad \bar{C}_i = \sum_{j}^{q} n_{i,j}$$

$$\text{with} \quad \bar{L}_j = \sum_{i}^{q} n_{i,j}$$

$$SI = -\sum_{j}^{q} \frac{\bar{L}_j}{N} \sum_{i}^{p} \frac{n_{i,j}}{\bar{L}_j} \log_2 \frac{n_{i,j}}{\bar{L}_j} \qquad N = \sum_{i}^{p} \sum_{j}^{q} n_{i,j}$$

4 Experimental results

4.1 Evaluations on the basis of the CI and SI indexes

We experimentally observe that there is a clear consensus between the two clustering methods as for the quality of the clustering induced by the different distances. In other words, the qualitative ranking between the distances is independent of the tested algorithms. Thus, we only present here the results obtained with the Hierarchical Ascendant Classification. Figure 2 shows the evolution of the indexes CI and SI for the two datasets *Cox2* and *Dhfr*. The ranking between the distances obtained with the 2 datasets is the following: Ipi, Tanimoto kernel, OA-kernel and extMG-kernel. The advantage of Ipi is clear for *Dhfr*, at least for a few number of classes, but less for *Cox2*. However, in the latter, the confusion index CI becomes quickly low for both distances, indicating that the classification agrees with the one provided by the experts. In both tests, the CI and SI values of OA-kernel and extMG-kernel are always worst, OA-kernel being significantly better than extMG-kernel.

This ranking is interesting for two reasons. First, the good behavior of Tanimoto is surprising since it has been compared with new methods, well-suited to deal with graphs. Second, from the knowledge representation, there are two categories: both Tanimoto kernel and extMG-kernel represent the molecules by means of walks in the graphs contrary to OA-kernel and Ipi that take into account the whole structure of the graphs. When looking at the result, it seems that these representational bias is not determining since both Ipi and Tanimoto obtain similar scores. In fact, there is another point that could explain the current ranking.

Both extMG-kernel and OA-kernel only consider the similarities between atoms, bonds and more globally between molecules: it is a purely symmetrical measure. In Ipi, we have explained that the measure is a function of two asymmetrical indexes. For the Tanimoto kernel, the numerator corresponds to the common walks to the molecules A and B and the denominator corresponds to the sum of the walks present in one molecule but not in the other and vice versa. So, we retrieve here a notion of asymmetrical measure. Then, we could suggest that, independently of the representation of molecules, the

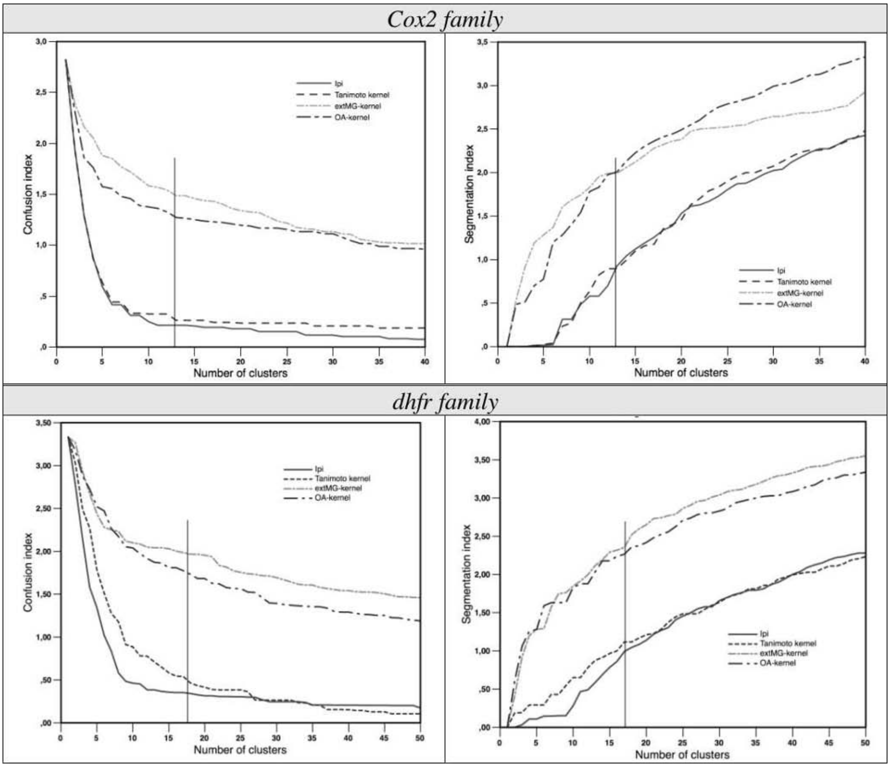


Fig. 2. Indexes SI and CI for the 4 distances used with the HAC on the datasets *Cox2* and *Dhfr*. The vertical line marks the original number of chemical families. The distance calculated from the extMG-kernel is made with $p_q = 0.1$ which gives the best results among the three values 0.1, 0.5, 0.9.

introduction of asymmetrical measures in the distance is very helpful to obtain some good clustering. Indeed, as discussed in Subsection 2.6, such kind of distance is less influenced by the difference of size of the molecules.

This hypothesis is confirmed by another experimental study (not described here) between OA-kernel and our measure Ipi. By replacing in our method the asymmetrical index by a symmetrical one, the behavior of the OA-kernel and Ipi becomes very close, Ipi remaining a little bit better. The two measures becomes quite identical, if we replace also the search of an optimal matching between the atoms by the maximum weighted matching calculus. Thus, the relevant factors seem to be the asymetrical aspect of the measure, and in some extend the better matching.

4.2 Quality of the classes learnt

By looking at the evolution of the values of CI and SI, it is quite difficult to have a reliable idea of the quality of the classes that have been built by the different approaches. For a given number of classes, one can represent CI (resp. SI) by a histogram showing the way the classes learnt (resp. the initial classes) organize the original classes (resp. the classes learnt).

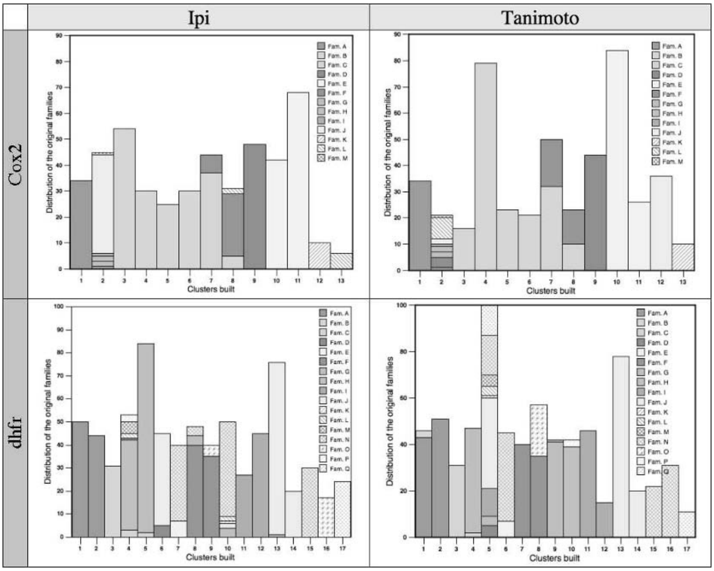


Fig. 3. Distribution of the initial families of molecules in the clusters learnt by the clustering algorithms.

Figure 3 shows this representation for the distances Ipi and Tanimoto for the expected number of families. Thus, each bar of the histogram represents the size of a class learned, and the color(s) correspond to the different families defined by the chemists (A, B, C, etc.). Ideally (for IC = 0), one should have only bars with one color. In the case of *Cox2*, the distance Ipi is able to retrieve most of the classes, the problems being restricted to the classes G, H, I, J which have very similar structures. Some classes such as B are split in several subclasses (2, 3, 4) that correspond to a clustering of the radicals (*i.e.* peripheral fragments), which are fixed on the general scaffold (the skeleton) of the family. Comparatively, the results obtained with the Tanimoto kernel are slightly worst with the emergence of the miscellaneous class 7 of great size (54 molecules). We can see also that the competencies of the two distances are a bit different: Ipi retrieves correctly the class D and Tanimoto retrieves the class J. In the case of *Dhfr*, one observes that the initial families are more fragmented, but in the case of Ipi, the families that have been retrieved correspond always in majority to one of the initial families. This result is good in the sense that i) as we said in the introduction, there is not a priori any universal distance and ii) the information used to compute the distances are principally extracted from a 2D structural representation of molecules.

Finally, we realized some experiments (not detailed here) in which the distance used to categorize is the mean of Ipi and Tanimoto. With *Cox2* and *Dhfr*, the results are always quite better that with each distance separately, what tends to suggest that the two distances catch different informations.

5 Conclusion

The analysis of the experimental results of High Throughput Screening (HTS) is a complex task. Thus, the chemists are eager of automatic methods of clustering that could put the light on structural analogs of hits and to evaluate the chemical diversity of their libraries. In this article, we have experimentally studied four distances (or index) on two well-known chemical datasets in order to evaluate the capacity of the algorithms to retrieve the families of molecules defined by the chemists.

The relatively disappointing results obtained with extMG-kernel and OA-kernel seems to indicate that the distances having some good performances in the supervised case are not always applicable with classical clustering algorithms. Moreover, by comparing the definition of the different distances, we have shown that the distance measures based on asymmetrical comparisons lead to better results than the one based on a plain symmetric definition.

It should be interesting to complete this study with the work that uses the SVM clustering methods, among others: (Finley and Joachims (2005), Ben-Hur et al. (2001)).

In this work, in spite of the simplicity of the approach, the good result of the Tanimoto kernel are somewhat surprising, since it is clearly less complete than the Ipi index. However, the latter present the advantage to take into account easier all the knowledge about the molecules and it is not necessary to manually choose the size of the structural keys to use. One still has to confirm with other data the good results obtained with *Cox2* and *Dhfr*.

References

BEN-HUR, A., HORN, D., SIEGELMANN, H.T. and VAPNIK, V. (2001): Support vector clustering. *Journal of Machine Learning Research, vol 2, 125-137.*

BISSON, G. (1992): Learning in FOL with a similarity measure. In: *Proceedings of 10th AAAI Conference.* San-Jose, 82–87.

BISSON, G. (1995): Why and how to define a similarity measure for object-based representation systems. In: *Proceedings of 2nd Int. Conf. on Building and Sharing Very Large-scale Knowledge Bases (KBKS).* IOS press, 236–246.

BERKHIN, P. (2002): Survey of Clustering Data Mining Techniques. Tech. rep., Accrue Software, San Jose, CA. http://citeseer.nj.nec.com/berkhin02survey.html.

CANDELLIER, L., TELLIER, I., TORRE, F. and BOUSQUET, O. (2006): Cascade evaluation of clustering algorithms, In: *Proceedings of ECML.* Berlin, 574–581.

CHEMAXON. http://www.chemaxon.com/

DHILLON, I.S. and GUAN, Y. (2004): Kernel k-means, spectral clustering and normalized cuts, In: *Proceedings of KDD.* Seattle, 551–556.

FINLEY, T. and JOACHIMS, T. (2005): Supervised clustering with support vector machines, In: *Proceedings of ICML.* Bonn, 217–224.

FRÖHLICH, H., WEGNER, J., SIEKER, F. and ZELL, A. (2005): A optimal assignment kernels for attributed molecular graphs, In: *Proceedings of ICML.* Bonn, 225–232.

GARTNER, T., FLACH, P. and WROBEL, S. (2003): On graph kernels: hardness results and efficient alternatives. In: *Proceedings of 16th Annual Conf. on Computational Learning Theory and 7th Annual Workshop on Kernel Machines.* Springer-Verlag, Berlin, 129–143.

HELMA, C., KRAMER, S. and De RAEDT, L. (2003): The molecular feature miner MolFea. In: *Proceedings of the Beilstein Workshop.* Bozen.

JARVIS, R.A. and PATRICK, E. A. (1973): Clustering using a similarity measure based on shared near neighbors. In: *IEEE Transactions on Computers.* C22: 1025–1034.

KASHIMA, H., KOJI, T. and AKIHIRO, I. (2003): Marginalized kernels between labeled graphs, In: *Proceedings of ICML.* Washington, DC, 321–328.

MAHE, P., UEDA, N., AKUTSU, T. and VERT, J.-P. (2004): Extensions of marginalized graph kernels, In: *Proceedings of the Twenty-First International Conference on Machine Learning (ICML).* ACM Press, 552–559.

MAHE, P., UEDA, N., AKUTSU, T., PERRET, J.-L. and VERT, J.-P. (2005): Graph kernels for molecular structure-activity relationship with support vector machines. *J. Chem. Inf. Model. 45(4), 939-951.*

RALAIVOLA, L., SWAMIDASS, S.J., SAIGO, H. and BALDI, P. (2005): Graph kernels for chemical informatics. *Neural Networks, Special Issue on Neural Networks and Kernel Methods for Structured Domains, 18:8, 1093-1110*

SUTHERLAND, J.J., O´BRIEN, L. A. and WEAVER, D. F. (2003): Spline-fitting with a genetic algorithm: a method for developing classification structure-activity relationships. *J. Chem. Inf. Comput. Sci. 43, 1906-1915*

WEININGER, D. (1988): SMILES 1. Introduction and encoding rules. *J. Chem. Inf. Comput. Sci., 28, 31.* See also: http://www.daylight.com/.

WIECZOREK, S., BISSON, G. and GORDON, MB. (2006): Guiding the search in the NO region of the phase transition problem with a partial subsumption test. In: *Proceedings of ECML 2006.* LNCS 4212, Berlin, 817–824.

Group Average Representations in Euclidean Distance Cones

Casper J. Albers[1], Frank Critchley[1], and John C. Gower[1,2]

[1] Department of Statistics, The Open University
Walton Hall, Milton Keynes MK7 6AA, United Kingdom
[2] Corresponding author: *j.c.gower@open.ac.uk*

Abstract. The set of Euclidean distance matrices has a well-known representation as a convex cone. The problems of representing the group averages of K distance matrices are discussed, but not fully resolved, in the context of SMACOF, Generalized Orthogonal Procrustes Analysis and Individual Differences Scaling. The polar (or dual) cone representation, corresponding to inner-products around a centroid, is also discussed. Some new characterisations of distance cones in terms of circumhyperspheres are presented.

1 Introduction

An $n \times n$ matrix $\mathbf{D} = \{d_{ij}^2\}$ of squared Euclidean distances is symmetric, with zero diagonal and $m = \binom{n}{2}$ essentially different non-negative off-diagonal values and hence may be represented by a point with coordinates $\mathbf{d} = vec(\mathbf{D})$ in m-dimensional Euclidean space. $\mathcal{D}$ denotes the set of all such m-vectors $\mathbf{d}$. Here, vec denotes stringing out the subdiagonal values of $\mathbf{D}$ as a vector. It is well known that $\mathcal{D}$ forms a convex cone. Writing $\mathbf{N} = \mathbf{11}'/n$, the centered distance matrix $\mathbf{B} = -\frac{1}{2}(\mathbf{I} - \mathbf{N})\mathbf{D}(\mathbf{I} - \mathbf{N})$ is symmetric with zero row-sums, and is positive semi-definite (p.s.d.) (Schoenberg (1935)). Thus, $\mathbf{B}$ is a member of $\mathcal{B}$, a sub-cone of the convex cone of all p.s.d. matrices; this result is sometimes stated as $-\mathbf{D}$ is p.s.d. on $\mathbf{x}'\mathbf{1} = 0$. Because $\mathbf{B}$ has zero row and column sums, $\mathcal{B}$ also has dimensionality m, a property that allows the coordinates $\mathbf{b} = vec(\mathbf{B})$ to be represented, without loss of information, by only the m elements below the diagonal, with a corresponding redefinition of $\mathcal{B}$. $\mathcal{B}$ consists of all the vectors making an obtuse angle with everything in $\mathcal{D}$, and conversely, so that the smallest angle between $\mathbf{b} \in \mathcal{B}$ and $\mathbf{d} \in \mathcal{D}$ is 90 degrees. This representation has been found useful for demonstrating some basic least-squares properties of minimising Sstress in multidimensional scaling (Critchley (1980)) and in developing MDS algorithms (Haydn et al. (1991)), whose terminology of referring to EDMs (Euclidean Distance Matrices) we adopt. Recently, Dattorro (2006) has given a masterly account of the properties of convex squared-distance cones. One line of development expresses the cone of EDMs as the intersection of two simpler convex cones, e.g. the cone of matrices p.s.d. on $\mathbf{x}'\mathbf{1} = 0$ and the cone of symmetric matrices with zero diagonal. Then, efficient algorithms such as that of Dykstra

(1983) may be used to find the best EDM $\mathbf{D}$ that approximates any observed symmetric $\mathbf{\Delta}$. Similarly, Critchley (1980) noted that $\mathbf{D}$ is characterized by the properties (i) that $\delta - \mathbf{d} \in \mathcal{B}$, where $\delta = vec(\mathbf{\Delta})$ and (ii) $\mathbf{d}'(\delta - \mathbf{d}) = 0$. However, if, as is usual, one is interested in r-dimensional (r small) approximations, we encounter difficulties because r-dimensional EDMs, although occurring as extremal rays of $\mathcal{D}$, do not themselves form a convex cone.

We shall not explore approximation here but shall be concerned with some problems arising from the simultaneous representation of K EDMs, $\mathbf{D}_1, \mathbf{D}_2, \ldots, \mathbf{D}_K$. The analysis of K EDMs is common in statistics where each $\mathbf{D}_k$ is modeled as a simple function of a common matrix $\mathbf{D}$, i.e. $\mathbf{D}_k = f_k(\mathbf{D})$ $k = 1, 2, \ldots, K$. The precise forms of the functions and of $\mathbf{D}$ vary among statistical methods but the central idea is that there is some group-average, represented by $\mathbf{D}$, to which each $\mathbf{D}_k$ is simply related. It would be interesting to know the location in the cones $\mathcal{D}$ and $\mathcal{B}$ of the group average relative to the K EDMs. The methods which we examine are (i) SMACOF (Heiser and De Leeuw (1979)) (ii) Generalised Procrustes Analysis (e.g. Gower and Dijksterhuis (2005)) and Individual Differences Scaling, INDSCAL (Carroll and Chang (1972)). All these methods are well-established with supporting software. Computation of the group-averages pertaining to the different methods is not a major difficulty but our hope is that their cone representations may give further insight into the properties of such methods. Thus, this paper is concerned with the positioning of the group-average in $\mathcal{D}$ and $\mathcal{B}$; we shall see, the problem is far from trivial and much remains to be done.

Not only are EDMs, with their cones $\mathcal{D}$ and $\mathcal{B}$, important but also configurations of n points that generate the EDMs. Thus, any decomposition $\mathbf{B} = \mathbf{X}\mathbf{X}'$ gives a matrix $\mathbf{X}$ whose rows generate the distances comprising the elements of $\mathbf{D}$. Of course, $\mathbf{X}$ is determined only up to an arbitrary orthogonal transformation. How then can these configuration matrices be fitted into the cone representations? Using the singular value decomposition $\mathbf{X} = \mathbf{U\Sigma V}'$, we note that the orthogonal transformation $\mathbf{Y} = \mathbf{XVU}' = \mathbf{U\Sigma U}'$ is symmetric and, because singular values are non-negative, $\mathbf{Y}$ is p.s.d.. Furthermore, because $\mathbf{B}$ is centred, so is $\mathbf{Y}$. Thus $\mathbf{Y}$ is a member of $\mathcal{B}$. It follows that any centred configuration matrix $\mathbf{X}$ may be represented uniquely in the same cone $\mathcal{B}$ as its inner-product $\mathbf{XX}'$. Indeed because $\mathbf{XX}' = \mathbf{U\Sigma}^2\mathbf{U}'$ and $\mathbf{Y} = \mathbf{U\Sigma U}'$ the points representing these matrices in $\mathcal{B}$ are different weighted sums of the same elementary inner-product matrices $\mathbf{u}_r\mathbf{u}_r'$, represented by points U_r ($r = 1, 2, \ldots, R$), say, where R is the dimensionality of $\mathbf{X}$. Being of deficient rank, these points are necessarily on the surface of the cone, whose interior only contains full rank matrices. The geometry is shown in Figure 1.

The linkage between the point D representing an EDM $\mathbf{D}$ in $\mathcal{D}$ and its inner-product counterpart $\mathbf{B}$, represented by B in $\mathcal{B}$, is suggested in Figure 1. The algebraic expression of the linear transformation $\mathbf{b} = \mathbf{Td}$ and its inverse $\mathbf{d} = \mathbf{Kt}$ have been given by Critchley (1988) and Gower and Groenen (1991) and it is not difficult to interpret them in terms of geometric orthogonal

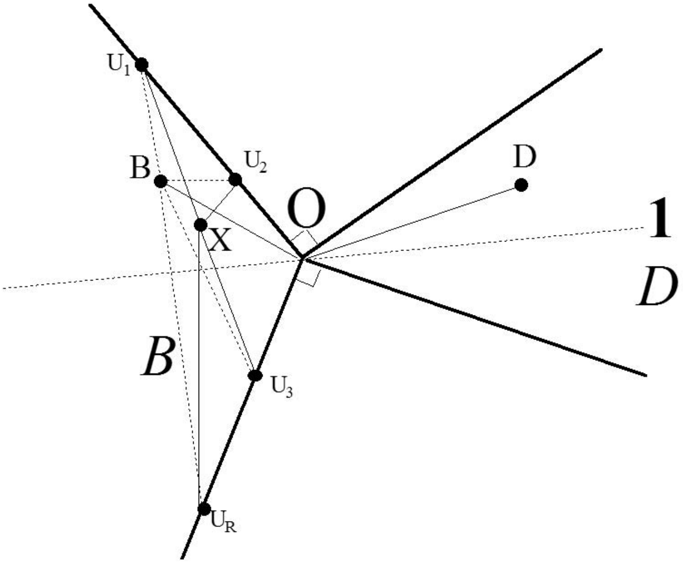


Fig. 1. The point D represents an EDM with inner-product matrix represented by the point B. X represents the symmetricised configuration matrix that generates **B** and hence **D**. B and X are different weighted sums of the same elementary inner-product matrices represented by $\mathrm{U}_1, \mathrm{U}_2, \ldots, \mathrm{U}_R$.

projections onto the direction **1**, representing the unit ray of both cones, and onto an n-dimensional subspace orthogonal to **1** that generates the symmetric matrix $\mathbf{D11}' + \mathbf{11}'\mathbf{D}$ which may be recognized from the expansion $-2\mathbf{B} = (\mathbf{I} - \mathbf{N})\mathbf{D}(\mathbf{I} - \mathbf{N}) = \mathbf{D} - \mathbf{ND} - \mathbf{DN} + (\mathbf{1}'\mathbf{D1}/n)\mathbf{N}$. The term $\frac{1}{2}(\mathbf{1}'\mathbf{D1}/n)$ is the total sum-of-squares about their centroid of all configurations that generate **D**, a quantity that is proportional to the length of the projection of **d** onto **1**. Apart from this simple representation, the remainder of the geometry of projections does not seem to lend itself to elegance. Nevertheless, corresponding to every set of points in $\mathcal{B}$ is another set of points in $\mathcal{D}$ which, in principle, allows any geometry in the one space to be transformed into a dual geometry on the other space.

2 Special cases

We now look at some of the detailed geometry of the statistical methods under discussion.

2.1 SMACOF

Here, we present a variant of the method that minimises $\sum_{k=1}^{K} ||\mathbf{D}_k - \mathbf{D}||^2$ which is, of course, given by $\mathbf{D} = \frac{1}{K}\sum_{k=1}^{K}(\mathbf{D}_k)$. Actually, SMACOF (Scal-

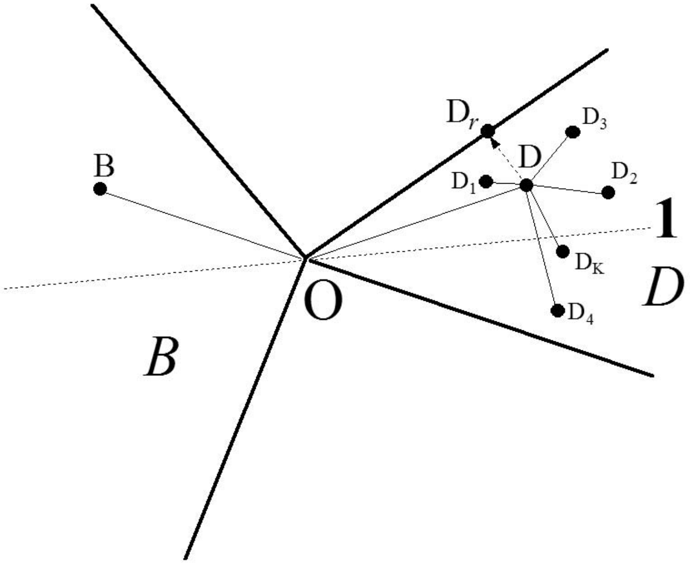


Fig. 2. SMACOF. The cone $\mathcal{D}$ of EDMs with its polar cone $\mathcal{B}$ of centred inner-product matrices $\mathcal{B}$. The EDM $\mathbf{D}$ is shown at the centroid of K EDMs. Also shown on the surface of the cone is the nearest r-dimensional approximation $\mathbf{D}_r$ to $\mathbf{D}$ together with the linkage of $\mathbf{B}$ to $\mathbf{D}$.

ing by Majorizing a Complicated Function) operates on distances and not squared distances but the mean of K matrices of (unsquared) distances is not necessarily another distance matrix, implying that matrices of unsquared distances do not define a convex cone. In terms of the EDM cone $\mathcal{D}$, $\mathbf{D}$ is simply at the centroid of the points representing the K matrices of squared distances. This is probably the simplest representation of a group-average. Of course, SMACOF, being an MDS method, is interested in r-dimensional configurations $\mathbf{X}$ that approximate $\mathbf{D}$ where r is small. These can be found by a variety of MDS algorithms; in the case of true SMACOF by using a majorisation algorithm to minimise the Stress criterion and in our variant by minimizing Sstress. Geometrically, this means finding the nearest r-dimensional EDM to $\mathbf{D}$ and this lies on the surface of $\mathcal{D}$. However, corresponding to the group-averages in $\mathcal{D}$ there is a complementary set of points $\mathrm{B}_1, \mathrm{B}_2, \ldots, \mathrm{B}_K$ with their group-average in $\mathcal{B}$. The geometry is shown in Figure 2.

2.2 Generalised Procrustes Analysis (GOPA)

Gower and Dijksterhuis (2005) discuss many variants of GPA but here we are concerned with the most popular method, Generalised Orthogonal Procrustes Analysis (GOPA) which is confined to orthogonal transformations $\mathbf{Q}_k$ ($k = 1, 2, \ldots, K$), so preserving distances among the configurations. Specifically, we are given centred configurations $\mathbf{X}_k$ $(k = 1, 2, \ldots, K)$, all assumed to have the same number of columns, and we require to find the $\mathbf{Q}_k$ $(k = 1, 2, \ldots, K)$ that minimize:

$$\sum_{k=1}^{K} ||\mathbf{X}_k \mathbf{Q}_k - \mathbf{G}||^2 \quad \text{where} \quad \mathbf{G} = \frac{1}{K} \sum_{k=1}^{K} (\mathbf{X}_k \mathbf{Q}_k)\,, \text{ the group average.}$$

As explained above, every centered configuration may be regarded as a point in $\mathcal{B}$. However, the symmetricising orientations of the configurations given by $\mathbf{VU}'$ derived from the singular value decomposition are very unlikely to coincide with those given by the optimal estimates of $\mathbf{Q}_k$ derived by GOPA. If the two orientations do coincide, $\mathbf{G}$ would be at the centroid as in SMACOF. We are thus led to consider where the GOPA group average lies relative to the individual configurations in the cones $\mathcal{D}$ and $\mathcal{B}$. Every configuration $\mathbf{X}_k$ defines a unique EDM $\mathbf{D}_k$, so it would be interesting to know how the EDM of the group average generated by the GOPA $\mathbf{G}$ relates to the centroid derived from SMACOF. Because the symmetricised configurations of $\mathcal{B}$ are not optimally oriented they are not likely to lead to anything useful. However, they do have a centroid which is also a member of $\mathcal{B}$. Is this configuration and the EMD it generates of any interest?

A further property of GOPA is more encouraging. It is known that necessary and sufficient conditions for the optimal GOPA fit is that $\mathbf{G}'\mathbf{X}_k\mathbf{Q}_k$ $(k = 1, 2, \ldots, K)$ is symmetric and p.s.d. (see e.g. Gower and Dijksterhuis (2005)). It follows that the matrices $\mathbf{G}'\mathbf{X}_k\mathbf{Q}_k$ are members of $\mathcal{B}$ that do incorporate their optimal GOPA rotations; these points have a centroid $\mathbf{G}'\mathbf{G}$. All these points have their complementary EDMs in $\mathcal{D}$. Furthermore, the residual sum-of-squares arising from the kth configuration is:

$$\mathit{trace}\,(\mathbf{X}_k'\mathbf{X}_k + \mathbf{G}'\mathbf{G} - 2\mathbf{G}'\mathbf{X}_k\mathbf{Q}_k)\,.$$

It is usual to pre-scale the data so that $\mathit{trace}(\mathbf{X}_k'\mathbf{X}_k) = 1$, so that the first two terms are constant for all settings of k. The third term is the projection of $\mathbf{G}'\mathbf{X}_k\mathbf{Q}_k$ onto the unit ray, so giving a neat geometrical representation of the residual sum-of-squares.

2.3 INDSCAL

This method defines a group-average configuration matrix $\mathbf{X}$ specified in a few dimensions R that are weighted in such a way that $\mathbf{X}$ generates an approximation to the individual centred inner product matrices $\mathbf{B}_k$ $(k = 1, 2, \ldots, K)$. Specifically, we require that approximately:

$$\mathbf{B}_k = \mathbf{X}\mathbf{W}_k\mathbf{X}'$$

where each $\mathbf{W}_k$ is a diagonal matrix with positive elements. Usually, the approximations are found by minimizing $\sum_{k=1}^{K} ||\mathbf{B}_k - \mathbf{X}\mathbf{W}_k\mathbf{X}'||^2$ using an

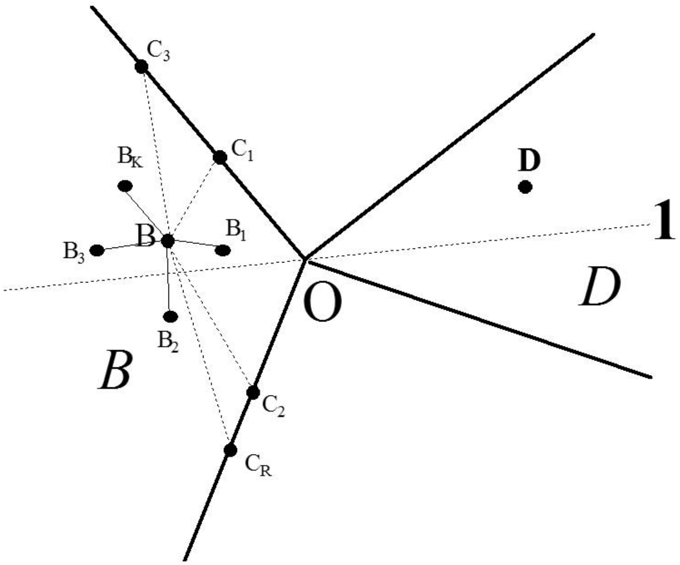


Fig. 3. The inner-product matrices $\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_K$ are different weighted means of the elementary group-average inner-product matrices $\mathbf{C}_1, \mathbf{C}_2, \ldots, \mathbf{C}_R$; the group-average $\mathbf{B}$ is itself an unweighted mean of these points. A similar geometry exists in $\mathcal{D}$ but for simplicity is not shown in detail.

ALS algorithm called CANDECOMP (Carroll and Chang, 1972). So far as the cone $\mathcal{B}$ is concerned, each column of $\mathbf{X}$ defines an elementary inner-product matrix $\mathbf{C}_r = \mathbf{x}_r\mathbf{x}'_r$ represented by a point C_r ($r = 1, 2, \ldots, \mathrm{R}$). Thus, each $\mathbf{B}_k$ is at a weighted mean $\sum_{r=1}^{\mathrm{R}} w_{kr}\mathbf{C}_r$ while unit weights generate a group-average inner product matrix $\mathbf{B} = \mathbf{X}\mathbf{X}'$. Figure 3 illustrates the geometry, where dotted lines indicate the group-average $\mathbf{B}$. Similar lines may be thought of as joining each $\mathbf{B}_k$ to $\mathrm{C}_1, \mathrm{C}_2, \ldots, \mathrm{C}_K$.

3 Other representations

In the above, we have concentrated on the relationship between $\mathcal{D}$ and $\mathcal{B}$ defined by $\mathbf{B} = -\frac{1}{2}(\mathbf{I}-\mathbf{N})\mathbf{D}(\mathbf{I}-\mathbf{N})$ or, equivalently, by $\mathbf{B} = -\frac{1}{2}(\mathbf{I}-\mathbf{1}\mathbf{s}')\mathbf{D}(\mathbf{I}-\mathbf{s}\mathbf{1}')$ where $\mathbf{s} = \mathbf{1}/n$. It is this choice of $\mathbf{s}$ that ensures that the row and columns sums of $\mathbf{B}$ are zero and that generating coordinates of $\mathbf{X}$ are centered at the centroid. This choice also defines a linear transformation of $\mathbf{D}$ and ensures that the cones $\mathcal{D}$ and $\mathcal{B}$ are orthogonal. Despite these nice properties it may be worth considering other choices of $\mathbf{s}$ that give different centrings (Gower (1982)). Of special interest is to choose $\mathbf{s}$ to centre at the circumcentre. Gower (1985) showed that: for every EDM $\mathbf{D}$, there exists a circumhypersphere iff $\mathbf{1}'\mathbf{D}^{-}\mathbf{1} \neq 0$ given by $\mathbf{s} = \mathbf{D}^{-}\mathbf{1}/(\mathbf{1}'\mathbf{D}^{-}\mathbf{1})$. This has radius $\mathrm{R}^2 = -(\mathbf{1}'\mathbf{D}^{-}\mathbf{1})^{-1}$.

Furthermore, if $\mathbf{1}'\mathbf{D}^{-}\mathbf{1} = 0$ there exists a g-circumhypersphere 0 given by $\mathbf{s} = (\mathbf{D}^2)^{-}\mathbf{1}/(\mathbf{1}'(\mathbf{D}^2)^{-}\mathbf{1})$. This has radius $\mathrm{R}^2 = (\mathbf{1}'(\mathbf{D}^3)^{-}\mathbf{1})/(\mathbf{1}'(\mathbf{D}^2)^{-}\mathbf{1})^2$.

Here, $\mathbf{D}^{-}$ represents any g-inverse of $\mathbf{D}$, and a g-circumhypersphere is one whose radius minimizes the sum-of-squares of the differences between a sphere and the actual, unequal, squared radii. Thus, we have three situations, (i) $\mathbf{D}$ is non-singular, has a circumhypersphere, and, as usual lies in the interior of the cone $\mathcal{D}$, (ii) $\mathbf{D}$ is singular so lies on the exterior of $\mathcal{D}$ but $\mathbf{1}'\mathbf{D}^{-}\mathbf{1} \neq 0$, and so there continues to be a true circumhypersphere and (iii) $\mathbf{1}'\mathbf{D}^{-}\mathbf{1} = 0$ so $\mathbf{D}$ lies on the exterior of $\mathcal{D}$ and there is no circumhypersphere, although there is a g-circumhypersphere. One may say that when $\mathbf{1}'\mathbf{D}^{-}\mathbf{1} = 0$, the circumhypersphere has infinite radius, so that the generating coordinates lie on a flat. An important property is that circumcentres are defined for all points in the interior of $\mathcal{D}$ and for some points on extremal rays.

Substituting for $\mathbf{s}$ the transformation is found to have the particularly simple form $\mathbf{C} = -\frac{1}{2}\mathbf{D} + \mathrm{R}^2\mathbf{1}\mathbf{1}'$ but this does not represent a linear function because R is the above-mentioned function of $\mathbf{D}$. The dimensionality of $\mathbf{C}$ remains m and, as with $\mathbf{B}$, we may work with $\mathbf{c} = vec(\mathbf{C})$. Next, we show how to recover the complete form of $\mathbf{D}$, at least for interior points of $\mathcal{D}$, from the sub-diagonal elements, $\mathbf{C}_0$, of $\mathbf{C}$. We know that $\mathbf{C} = -\frac{1}{2}(\mathbf{I} - \mathbf{1}\mathbf{s}')\mathbf{D}(\mathbf{I} - \mathbf{s}\mathbf{1}')$ for some $\mathbf{s} = \mathbf{D}^{-}\mathbf{1}/(\mathbf{1}'\mathbf{D}^{-}\mathbf{1})$ and some unknown $\mathbf{D}$, which we require to construct. Thus, $\mathbf{C} = \mathbf{C}_0 + \mathrm{K}\mathbf{I}$ for some K equal to the unknown R^2. Because $\mathbf{C}\mathbf{s} = 0$, it follows that $\mathbf{C}_0\mathbf{s} = -K\mathbf{s}$ and we may set $\mathbf{K} = -\lambda$ where λ is the smallest (necessarily negative) eigenvalue of $\mathbf{C}_0$. Note that this setting ensures that $\mathbf{C}$ is p.s.d., which it would not be if any other negative eigenvalue were chosen. Having identified R^2 we may construct $\mathbf{C} = \mathbf{C}_0 + \mathrm{R}^2\mathbf{I}$ and $\mathbf{D} = 2(\mathrm{R}^2\mathbf{1}\mathbf{1}' - \mathbf{C}) = 2\{\mathrm{R}^2(\mathbf{1}\mathbf{1}' - \mathbf{I}) - \mathbf{C}_0\}$.

The above shows that every $\mathbf{C}_0$ corresponds to a unique EDM $\mathbf{D}$. To increase understanding of this geometry, we investigate contours of constant R^2 in $\mathcal{D}$. A full study is not possible here and we content ourselves with examining the cross-section of $\mathcal{D}$ that contains the unit ray $\mathbf{1}$ and the fundamental rank-2 EDM which consists of two sets of p and q $(n = p + q)$ points coincident at points P and Q, say, respectively. We assume that P and Q are unit distance apart, defining a vector $\mathbf{r}$ giving the squared distances (all equal to unity or zero). Although $\mathbf{D}$ is only of rank-2, P and Q lie on a 'circle' centred at their midpoint and so have a circumcircle with $\mathrm{R}^2 = \frac{1}{4}$. Other points $\lambda\mathbf{r}$ on the same ray will replace R^2 by $\lambda\mathrm{R}^2$. The unit ray $\mathbf{1}$ corresponds to an EDM of a regular simplex and has $\mathbf{R}^2 = (n-1)/2n$. Also, $\mathbf{r}$ has pq unit values so that $\mathbf{r}'(\mathbf{r} - \mathbf{1}) = 0$, showing that $\mathbf{r}$ is the orthogonal projection of $\mathbf{1}$ onto the ray through $\mathbf{r}$. This is shown in Figure 4(a), which gives the basic geometry of the cross-section of the EDM cone. We wish to find the circumradius of any EDM in the plane defined by $\mathbf{1}$ and $\mathbf{r}$. Writing $\mathbf{D}_1$ and $\mathbf{D}_2$ for the matrix forms of these EDMs, we have:

$$-2\mathbf{D} = -2(\lambda\mathbf{D}_1 + \mu\mathbf{D}_2) = \lambda(\mathbf{I} - \mathbf{11}') - \mu\begin{pmatrix} \mathbf{0} & \mathbf{11}' \\ \mathbf{11}' & \mathbf{0} \end{pmatrix}$$
$$= \begin{pmatrix} \lambda(\mathbf{I} - \mathbf{11}') & -(\lambda+\mu)\mathbf{11}' \\ -(\lambda+\mu)\mathbf{11}' & \lambda(\mathbf{I} - \mathbf{11}') \end{pmatrix}$$

where the lengths of the vectors $\mathbf{1}$ are assumed defined by context and where, without loss of generality, we have assumed that the p and q points are labeled consecutively. After detailed algebraic manipulations we find that the circumradius $R^2_{\lambda\mu}$ of $(\lambda\mathbf{D}_1 + \mu\mathbf{D}_2)$ is given by:

$$2R^2_{\lambda\mu} = \lambda + \frac{\mu^2 pq - \lambda^2}{2\mu pq + n\lambda} = \mu\left(\frac{\lambda}{\mu} + \frac{pq - \lambda^2/\mu^2}{2pq + n\lambda/\mu}\right).$$

The right-hand form is valid only when $\mu \neq 0$, when it shows that for constant λ/μ the value of R^2 increases proportionally to μ. This result merely confirms that R^2 increases with μ as one proceeds along the ray defined by the ratio λ/μ. When $\lambda = 0$ and $\mu = 1$ we define $\mathbf{r}$ and correctly obtain $\mathrm{R}^2 = \frac{1}{4}$; when $\lambda = 1$ and $\mu = 0$ we define $\mathbf{1}$ and correctly obtain $\mathrm{R}^2 = n(n-1)/2$. When $2pq + n\lambda/\mu = 0$ the circumradius becomes infinite, so $\lambda/\mu = -2pq/n$ defines the extremal ray other than $\mathbf{r}$ the cross-section under consideration, as is shown in Figure 4(a).

We can derive the contours for constant $\mathrm{R}^2 = \frac{1}{4}$, other contours are easily obtained by proportion. The values of p and q affect these contours. Generally the contour is hyperbolic as is shown in Figure 4(d) for $p = 2$ and $q = 6$. One branch of the hyperbola is outside the cone so is irrelevant. For emphasis we have high-lighted the part of the relevant branch that is inside the cone. This state of affairs is modified when $p = 1$ or $p = q$. When $p = 1$ the formula simplifies to:

$$2R^2_{\lambda\mu} = \frac{(\lambda+\mu)^2(n-1)}{n\lambda + 2(n-1)\mu}$$

which represents a parabola, shown in Figure 4(b) for $p = 1$ and $q = 7$. When n is even and $p = q$, $\mu p + \lambda$ is a common factor and provide this is not zero we have:

$$2R^2_{\lambda\mu} = \frac{(2p-1)\lambda + \mu p}{2p}$$

which represents a single straight line; the contours are then a set of parallel lines which may be shown to be orthogonal to the unit ray; the extremal ray $\mu p + \lambda$, R^2_∞, provides a further, pathological, contour. This is shown in Figure 4(c) for $p = q = 4$.

Although Figure 4 is based on the case $n = 8$, the results are completely general. Of course, different labeling of the n points will give different rays but their geometry is identical. Also, different settings of p and q will change scales but not the more fundamental geometry of hyperbolas, parabolas and pairs of

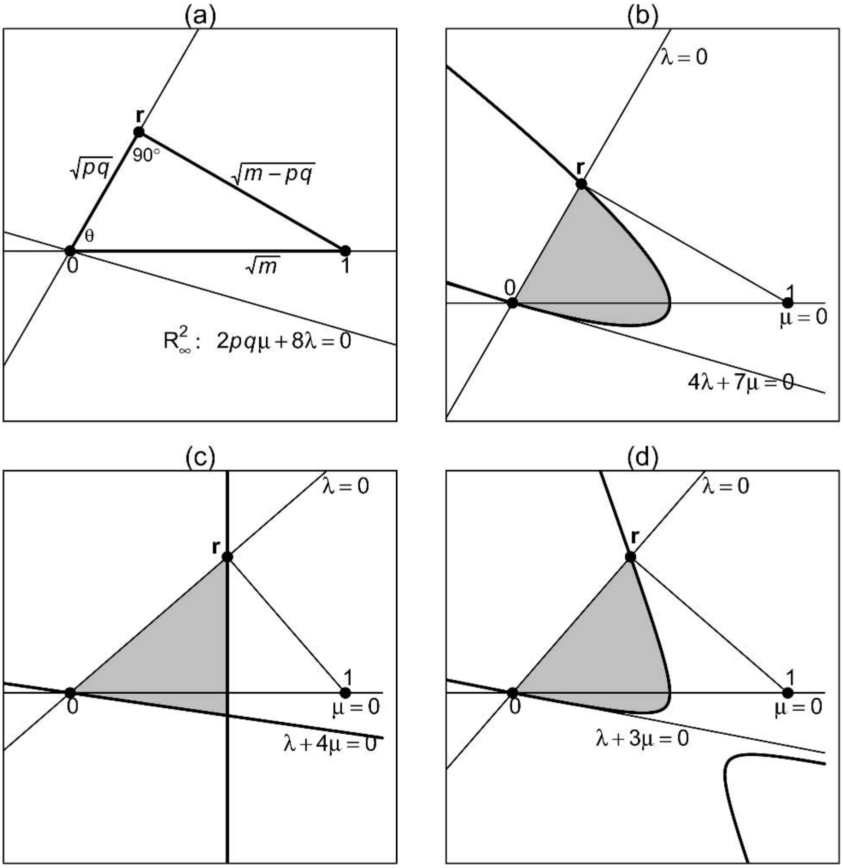


Fig. 4. Figure (a) shows the basic geometry of the cross-section of the EDM cone containing the vectors **r** and **1**. The remaining figures give contours of constant circumradius $\mathrm{R}^2 = \frac{1}{4}$ for various choices of p and q where $p + q = n = 8$. We introduce $m = n(n + 1)/2 = 28$. The part of the contour in the EDM cone is highlighted in grey. Figure (d) shows the usual hyperbolic contour found (here $p = 2$ and $q = 6$). Figures (b) and (c) show the special parabolic and linear solutions found when $p = 1$ and $p = q$, respectively.

lines. These results confirm the complexity of the geometry of the EDM cone, especially in the vicinity of the extremal rays. The formula $\mathbf{C} = -\frac{1}{2}\mathbf{D}+\mathrm{R}^2\mathbf{11}'$ readily allows the contours in $\mathcal{D}$ to be transformed into contours in a space $\mathcal{C}$, analogous to the cone $\mathcal{B}$. One merely has a reflection of an EDM $\frac{1}{2}\mathcal{D}$ in the origin, together with a translation R^2 in the direction of the unit ray. Unfortunately, R^2 varies with $\mathbf{D}$ but it is constant along the contour for constant R^2, so scaled versions of the same contours persist. It seems that although $\mathcal{C}$, like $\mathcal{B}$ is part of the cone of p.s.d. matrices, it is not itself a cone.

However, in the context of group average representations things are similar to the representations in $\mathcal{B}$. The main change is that configuration matrices $\mathbf{X}_k$ first have to be centred at their circumcentre before being symmetricised by rotating their singular value forms. There is then a special difficulty when a circumcentre does not exist when the translation term becomes infinite. This situation will be common when $\mathbf{X}_k$ is a data-matrix but not when $\mathbf{X}_k$ is derived from similarities. The spaces $\mathcal{C}$ and $\mathcal{D}$ are not orthogonal and, indeed, may intersect.

References

CARROLL, J.D. and CHANG, J.J. (1972): Analysis of individual differences in multidimensional scaling via an n-way generalization of Eckart-Young decomposition. *Psychometrika 35, 283-319.*

CRITCHLEY, F. (1980): Optimal norm characterisations of multidimensional scaling methods and some related data analysis problems. In: E. Diday et al. (Eds.): *Data Analysis and Informatics.* North Holland, Amsterdam, 209–229.

CRITCHLEY, F. (1988): On certain linear mappings between inner-products and squared-distance matrices. *Linear Algebra and its Applications 105, 91-107.*

DATTORRO, J. (2006): *Convex optimization and Euclidean distance geometry.* Meboo Publishing, Palo Alto, California, USA.

DYKSTRA, R.L. (1983): An algorithm for restricted least squares regression. *Jounrnal of the American Statistical Association 78, 837-842.*

GOWER, J.C. (1982): Euclidean distance geometry. *The Mathematical Scientist 7, 1-14.*

GOWER, J.C. (1985): Properties of Euclidean and non-Euclidean distance matrices. *Linear Algebra and its Applications 67, 81-97.*

GOWER, J.C. and DIJKSTERHUIS, G.B. (2005): *Procrustes Problems.* Oxford University Press, Oxford, UK.

GOWER, J.C. and GROENEN, P.J.F. (1991): Applications of the modified Leverrier-Faddeev algorithm for the construction of explicit matrix spectral decompositions and inverses. *Utilitas Mathematica 40, 51-64.*

HEISER, W. and DE LEEUW, J. (1979): *How to use SMACOF-1, A program for metric muktidimensional scaling.* Department of Datatheory, Faculty of Social Sciences, University of Leiden, Wassenaarseweg 80, Leiden, The Netherlands, 1-63.

HAYDN, T.L., WELLS, J. LIU, W-M and TARAZAGA, P. (1991): The cone of distance matrices. *Linear Algebra and its Applications 144, 153-169.*

SCHOENBERG, I.J. (1935): Remarks to Maurice Frechet's article "Sur la définition axiomatique d'une classe d'espaces vectoriels distanciés applicables vectoriellement sur l'espace de Hilbert". *Annals of Mathematics 36, 724-732.*

On Lower-Maximal Paired-Ultrametrics

Patrice Bertrand and François Brucker

GET-ENST Bretagne, Dpt Lussi, 2 rue de la Châtaigneraie, CS 17607,
35576 Cesson Sévigné Cedex, France
{*patrice.bertrand,francois.brucker*}*@enst-bretagne.fr*

Abstract. The weakly indexed paired-hierarchies (shortly, p-hierarchies) provide a clustering model that allows overlapping clusters and extends the hierarchical model. There exists a bijection between weakly indexed p-hierarchies and the so-called paired-ultrametrics (shortly, p-ultrametrics), this correspondence being a restriction of the bijection between weakly indexed pyramids and Robinsonian dissimilarities. This paper proposes a generalization of the well-known HAC clustering method to compute a weakly indexed p-hierarchy from a given dissimilarity d. Moreover, the p-ultrametric associated to such a weakly indexed p-hierarchy is proved to be lower-maximal for d and larger than the sub-dominant ultrametric of d.

1 Introduction

Clustering models do not generally allow overlapping clusters. The widely used hierarchical clustering model is a typical example: if A and B are any two clusters, then $A \cap B \in \{\phi, A, B\}$. Some clustering models have been developed to allow overlapping clusters like weak-hierarchies (Bandelt and Dress (1989)) or pyramids (Diday (1969); Fichet (1986)) but they may produce a lot of clusters. More precisely, a hierarchy has at most $2n-1$ clusters if n is the number of objects, whereas a weak hierarchy or a pyramid may have at most $n(n+1)/2$ clusters. The p-hierarchies (Bertrand (2002)) can be seen as an intermediary model. Indeed, they are a generalization of hierarchies and a particularization of pyramids. They allow overlapping clusters, but admit at most the integer part of $5(n-1)/2$ clusters.

This paper shows an algorithm that generalizes the HAC (Hierarchical Agglomerative Clustering) algorithm in order to produce a p-hierarchy from a given dissimilarity. Moreover, the dissimilarity associated to this p-hierarchy is lower-maximal with respect to the original dissimilarity, generalizing the relationship between the single-linkage algorithm and the sub-dominant ultrametric.

The rest of the text is organized as follows. We will first introduce most of the definitions we will need, and define p-hierarchies and p-ultrametrics. Then, after having characterized p-ultrametrics through a kind of intersection graph, we will describe an algorithm which aim is to approximate a given dissimilarity by a p-ultrametric. Finally, an example shows the differences between the ultrametric sub-dominant and the computed p-ultrametric.

2 Basic definitions and properties

In what follows, E will designate a finite ground set, and as usual, $\mathcal{P}(E)$ will be the set of all subsets of E. A *collection* $\mathcal{F}$ *on* E will be defined as any subset of $\mathcal{P}(E)$. To work with collections, we will implicitly endow each collection with the inclusion order, thus $\max \mathcal{F}$ will denote the maximal elements of $\mathcal{F}$ for the inclusion order. Moreover, $\widehat{\mathcal{F}}$ will denote the closure under intersection of $\mathcal{F}$, and $\mathcal{F}$ will be said to be *closed* whenever $\widehat{\mathcal{F}} = \mathcal{F}$. For all $A \subseteq E$, we will set:

$$\mathcal{F} \downarrow A = \{B \in \mathcal{F} \mid B \subseteq A\}.$$

A *clustering system on* E will designate any closed collection on E that contains E and its singletons. This paper is concerned with a particular kind of clustering systems on E called *p-hierarchies.* In order to define them, we have to introduce some definitions and notations. Let $A, B \in \mathcal{P}(E)$. The set A is said to be *hierarchical with* B if $A \cap B \in \{\phi, A, B\}$, and *properly intersects* B ($A \cap B \notin \{\phi, A, B\}$) otherwise. By extension, a collection $\mathcal{F}_1$ on E is said to be hierarchical with another collection $\mathcal{F}_2$ on E if A is hierarchical with B for all $A \in \mathcal{F}_1$ and all $B \in \mathcal{F}_2$. The set A is called hierarchical with a collection $\mathcal{F}$ on E if $\{A\}$ is hierarchical with $\mathcal{F}$. Then, a collection $\mathcal{F}$ on E is said to be hierarchical if $\mathcal{F}$ is hierarchical with itself. Moreover, a collection $\mathcal{F}$ on E is called *paired-hierarchical* (shortly *p-hierarchical*) if each element of $\mathcal{F}$ properly intersects at most one element of $\mathcal{F}$ (Bertrand (2002)).

Within this terminology, note that a *hierarchy* is any hierarchical clustering system. Similarly, a *paired-hierarchy* (shortly, *p-hierarchy*) is defined as any p-hierarchical clustering system.

A main feature in classification is the equivalence between *dissimilarity* models and clustering system models. Let us detail this equivalence. A dissimilarity d on E is an application from E^2 to $\Re^+$ such that $d(x,y) = d(y,x) \geq d(x,x) = 0$ for all $x, y \in E$. Moreover, a dissimilarity d on E is said to be *proper* if and only if $d(x,y) = 0$ implies $x = y$. In what follows, all dissimilarities will be assumed to be proper, so that "dissimilarity" will must be understood as synonymous with "proper dissimilarity".

To state the equivalence we have to associate clustering systems with dissimilarities. First recall that the *diameter* of a subset A, denoted $\mathrm{diam}_d(A)$, is defined by $\mathrm{diam}_d(A) = \max\{d(x,y) \mid x, y \in A\}$. A *maximal linked set* of d is then a subset A of E such that it does not exist A' with $A \subset A'$ and $\mathrm{diam}_d(A) = \mathrm{diam}_d(A')$. In other words, a maximal linked set of d is a subset of E that is maximal at a given diameter.

Let's denote by $\mathcal{L}_d(h)$ (with $h \in \Re^+$) the set of all maximal linked sets of d with diameter equal to h, and note that $\bigcup \mathcal{L}_d(\Re^+)$ is the set of all maximal linked sets of d. In addition, for all $h \in \Re^+$, let $\mathcal{M}_d(h) = \max \bigcup \mathcal{L}_d([0,h])$ and, when h is strictly positive, let $\mathcal{M}_d(h^-) = \max \bigcup \mathcal{L}_d([0,h[)$.

It can be noted that $\mathcal{L}_d(h) = \mathcal{M}_d(h) \setminus \mathcal{M}_d(h^-)$. Moreover, since $d(x,x) = 0$ for all $x \in E$, the collection $\mathcal{M}_d(h)$ is always non empty for all $h \geq 0$.

Then the above-mentioned equivalence between dissimilarities and clustering systems consists of the correspondence $\varPhi$ defined by $\varPhi(d) = (\widehat{\mathcal{C}}, \mathrm{diam}_d)$, with $\mathcal{C} = \bigcup \mathcal{L}_d(\Re^+)$. More precisely, the map $\varPhi$ is injective from the set of all dissimilarities on E into the set of all weakly indexed clustering systems on E (Batbedat (1988); Bertrand (2000)), where a clustering system $\mathcal{F}$ is said to be weakly indexed if it is equipped with a map $f : \mathcal{F} \to \Re^+$ that satisfies:

- $f(A) \leq f(B)$ if $A \subseteq B$,
- $f(A) = f(B)$ and $A \subseteq B$ implies that $A = \bigcap\{C \in \mathcal{F} \,|\, A \subset C\}$.

Since in this paper we consider only proper dissimilarities, we must also consider only weakly indexed clustering systems $(\mathcal{F}, f)$ on E such that $f^{-1}(0)$ coincides with the collection of singletons of E (*cf.* Bertrand (2000)). Moreover, note also that an *indexed* clustering system designates any weakly indexed clustering system $(\mathcal{F}, f)$ such that f is strictly monotone, *i.e.*, $f(A) < f(B)$ if $A \subset B$.

The map $\varPhi$ induces various bijections, in particular the well-known bijection between the set of *ultrametrics* (an ultrametric is any dissimilarity d on E such that $\max\{d(x,y), d(y,z)\} \leq d(x,z)$ for all $x, y, z \in E$) and the set of indexed hierarchies (*cf.* Johnson (1967); Benzécri (1973)). A bijection of interest in this paper is the bijection (also induced by $\varPhi$) between the set of *paired-ultrametrics* (shortly, *p-ultrametrics*) and the set of all weakly indexed closed p-hierarchies (*cf.* Bertrand (2002)). A dissimilarity d on E is a p-ultrametric if and only if for all 4-element subset A of E, it exists a non trivial part B of A such that $d(b, b') \leq d(a, b) = d(a, b')$ for all $a \in A$ and all $b, b' \in B$.

Note that when $\mathcal{F}$ is a p-hierarchy, so are $\widehat{\mathcal{F}}$ and $\mathcal{F} \cup \{\{x\} \,|\, x \in E\} \cup \{E\}$. It follows that a dissimilarity d is a p-ultrametric if and only if $\bigcup \mathcal{L}_d(\Re^+)$ is p-hierarchical.

Let us now focus again on dissimilarities. The aim of this paper is to approximate a given dissimilarity d by a p-ultrametric. To do this, we will mainly use properties of $\bigcup \mathcal{L}_d(\Re^+)$, and this involves some new notations.

Let d be a dissimilarity on E and A and B two subsets of E. We define $d(A, B)$ by $d(A, B) = \min\{d(a, b) \,|\, a \in A \setminus B, b \in B \setminus A\}$.

We denote by $\mathbf{I}_p(\mathcal{F}) = (\mathcal{F}, F)$ the graph with the collection $\mathcal{F}$ as vertex set and $AB \in F$ if and only if $A \cap B \notin \{\phi, A, B\}$ (this graph is sometimes known as the *overlap graph* of $\mathcal{F}$).

If $\mathbf{G}$ is a graph whose vertex set is a collection on E, we will write:

$$\mathrm{Part}\,(\mathbf{G}) = \{\bigcup \mathbf{C} \mid \mathbf{C} \text{ is a connected component of } \mathbf{G}\}$$

It is clear that the collection $\mathrm{Part}\,(\mathbf{G})$ is a partition of the vertex set of $\mathbf{G}$. Since $\mathrm{Part}\,(\;)$ will be mainly used for graph $\mathbf{G} = \mathbf{I}_p(\mathcal{F})$, we will write $\mathrm{Part}\,(\mathcal{F})$ instead of $\mathrm{Part}\,(\mathbf{I}_p(\mathcal{F}))$. Finally, for a collection $\mathcal{F}$ on E, we denote also $\mathbf{G}_d[\mathcal{F}](h)$ the graph $(\mathcal{F}, F)$ defined by $AB \in F$ if and only if $d(A, B) = h$, and we write $\mathbf{G}_d[h]$ instead of $\mathbf{G}_d[\mathrm{Part}\,(\mathcal{M}_d(h^-))](h)$.

The following lemma, which uses many previous definitions, is the basement of the hereafter development.

Lemma 1. *Let d be a dissimilarity on E and $h > 0$. If C_1 and C_2 are two distinct elements of* $\mathrm{Part}\,(\mathbf{G}_d[h])$*, then $d(C_1, C_2) > h$.*

Proof. If $d(C_1, C_2) < h$, it exists $x \in C_1$ and $y \in C_2$ such that $d(x, y) < h$. Since $\{x, y\}$ can be extended into a maximal linked set with diameter $d(x, y)$, x and y must be in the same set in $\mathrm{Part}\,(\mathcal{M}_d(h^-))$, thus $C_1 = C_2$, which is not possible.

If $d(C_1, C_2) = h$, it exists an element A of $\mathrm{Part}\,(\mathbf{G_d}[\mathbf{h}])$ such that $C_1 \subseteq A$ and $C_2 \subseteq A$. Thus again $C_1 = C_2$, which is not possible. ∎

3 A characterization of p-ultrametrics and ultrametrics

We will characterize, in this section, p-ultrametrics and ultrametrics through conditions involving $\mathbf{G}_d[h]$ graphs, where $h > 0$ and d designates an arbitrary dissimilarity on E.

We will write $\mathrm{child}_{\mathcal{F}} A = \max\{\, B \in \mathcal{F} \mid B \subset A \,\}$ for any collection $\mathcal{F}$ on E and any subset A of E. Moreover, we will write $\mathrm{child}_d(A)$ instead of $\mathrm{child}_{\bigcup \mathcal{L}_d(\Re^+)}(A)$. First, one can remark that:

Remark 1. Let A be a subset of E such that $\mathrm{diam}_d(A) = h$. We have that $\mathrm{child}_d(A) = \mathrm{child}_{\mathcal{F}}(A)$ where $\mathcal{F} = \bigcup \mathcal{L}_d([0, h[)$.

Lemma 2. *Let d be a p-ultrametric on E and $h > 0$. If $C \in$* $\mathrm{Part}\,(\mathbf{G}_d[h])$ *then $\mid \mathcal{M}_d(h) \downarrow C \mid \in \{1, 2\}$ and $\mid \mathcal{L}_d(h) \downarrow C \mid \in \{0, 1, 2\}$.*

Proof. Let us prove that $\mid \mathcal{M}_d(h) \downarrow C \mid \in \{1, 2\}$. Since $C \neq \phi$ it is clear that $1 \leq |\mathcal{M}_d(h) \downarrow C|$. Suppose that $|\mathcal{M}_d(h) \downarrow C| > 2$ and let $A \in \mathcal{M}_d(h) \downarrow C$. We must have $A \subset C$, and thus $B \in \mathcal{M}_d(h) \downarrow C$ exists such that $A \cap B \neq \phi$, by definition of $C \in \mathrm{Part}\,(\mathbf{G}_d[h])$. Since $|\mathcal{M}_d(h) \downarrow C| > 2$ and d is p-ultrametric, $A \cup B \subset C$. Therefore, it must exist $D \in \mathcal{M}_d(h) \downarrow C$ such that either $A \cap D \neq \phi$ or $B \cap D \neq \phi$ because $C \in \mathrm{Part}\,(\mathbf{G}_d[h])$. This cannot be possible because $\bigcup \mathcal{L}_d(\Re^+)$ is a p-hierarchy.

The second assertion derives then from $\mathcal{L}_d(h) = \mathcal{M}_d(h) \setminus \mathcal{M}_d(h^-)$. ∎

To investigate properties of the connected components of $\mathbf{G}_d[h]$ when d is a p-ultrametric (proposition 1) we introduce two more notations. Let $\mathbf{C}$ be a connected component of $\mathbf{G}_d[h]$ (with $h \geq 0$), and $C = \bigcup \mathbf{C}$. We denote:

- $\mathcal{B}_d(A, h) = \{B \in \mathrm{Part}\,(\mathrm{child}_d(A)) \mid d(C \setminus A, B) > h\}$, for $A \in \mathbf{C}$;
- $\mathcal{K}_d(\mathbf{C}, h) = \{(\mathcal{A}, \mathcal{B}) \mid \mathcal{A}, \mathcal{B} \in \mathcal{P}(\mathbf{C}),\ \mathcal{A} \cap \mathcal{B} = \phi,\ d(\bigcup \mathcal{A}, \bigcup \mathcal{B}) > h\}$.

Proposition 1. *Let d be a p-ultrametric on E, $h > 0$ a value taken by d, $\mathbf{C}$ a connected component of $\mathbf{G}_d[h]$ and $C = \bigcup \mathbf{C}$. One and only one of the following five assertions is satisfied:*

(i) $\mid \mathbf{C} \mid = 1$ *and* $\mathcal{L}_d(h) \downarrow C = \phi$,
(ii) $\mid \mathbf{C} \mid = 1$ *and* $\mathcal{L}_d(h) \downarrow C = \{C\}$,
(iii) $\mid \mathbf{C} \mid > 1$, $\mathcal{K}_d(\mathbf{C}, h) = \phi$, $\bigcup_{A \in \mathbf{C}} \mathcal{B}_d(A, h) = \phi$, *and* $\mathcal{L}_d(h) \downarrow C = \{C\}$
(iv) $\mid \mathbf{C} \mid > 1$, $\mathcal{K}_d(\mathbf{C}, h) = \phi$, $\bigcup_{A \in \mathbf{C}} \mathcal{B}_d(A, h) \neq \phi$,
and $\mathcal{L}_d(h) \downarrow C = \{C \setminus (\bigcup_{A \in \mathbf{C}} \mathcal{B}_d(A, h))\}$,
(v) $\mid \mathbf{C} \mid > 1$, $\mathcal{K}_d(\mathbf{C}, h) \neq \phi$, $\mathcal{L}_d(h) \downarrow C = \{C \setminus (\bigcup \mathcal{A}^*), C \setminus (\bigcup \mathcal{B}^*)\}$
with $\{(\mathcal{A}^*, \mathcal{B}^*)\} = \max \mathcal{K}_d(\mathbf{C}, h)$, *and* $\bigcup_{A \in \mathbf{C}} \mathcal{B}_d(A, h) = \phi$.

Proof. 1 By definition, $\mathbf{C} = \{A_1, \ldots, A_r\}$ where each $A_j \in \text{Part}\,(\mathcal{M}_d(h^-))$. Since d is p-ultrametric, A_j is the union of one or two elements of $\mathcal{M}_d(h^-)$. Assume first that $r = |\mathbf{C}| = 1$, thus $C = A_1$. If $A_1 \in \mathcal{M}_d(h^-)$, then we have $\mathcal{L}_d(h) \downarrow C = \phi$. Otherwise, it must exist $M_1, M_2 \in \mathcal{M}_d(h^-)$ such that $A_1 = M_1 \cup M_2$. In this case, if $d(M_1, M_2) > h$ then $\mathcal{L}_d(h) \downarrow C = \phi$, and if $d(M_1, M_2) = h$ then $\mathcal{L}_d(h) \downarrow C = \{C\}$ since each element of $\mathcal{L}_d(h) \downarrow C$ cannot properly intersects both M_1 and M_2 which form a non hierarchical pair. Thus (i) or (ii) must hold when $\mid \mathbf{C} \mid = 1$.

We now assume that $r = \mid \mathbf{C} \mid > 1$. Since $d(x, y) = h$ for some $x \in A_1$ and $y \in A_2$, we have $\mathcal{L}_d(h) \downarrow C \neq \phi$, and so $\mid \mathcal{L}_d(h) \downarrow C \mid \in \{1, 2\}$ by Lemma 2.

Assume first that $\mid \mathcal{L}_d(h) \downarrow C \mid = 1$ and denote as N the unique element of $\mathcal{L}_d(h) \downarrow C$. Observe that N intersects each A_j in $\mathbf{C}$ and thus $\mathcal{K}_d(\mathbf{C}, h) = \phi$. Moreover, since d is p-ultrametric, N contains at least all but one of the A_j's.

Suppose first that N contains all the A_j's. Then $\mathcal{L}_d(h) \downarrow C = \{C\}$ and thus $d(C \setminus A_j, B) \leq h$ for all $B \in \text{Part}\,(\text{child}_d(A_j))$ and all $j \leq r$. Consequently, $\bigcup_{A \in \mathbf{C}} \mathcal{B}_d(A, h) = \phi$ and (iii) holds. Suppose now that N contains all the A_j's except A_1. Consider $B \in \text{Part}\,(\text{child}_d(A_1))$. Since d is p-ultrametric, we have either $B \subseteq A_1 \setminus N$ and then $d(C \setminus A_1, B) > h$, or $B \subseteq A_1 \cap N$ and then $d(C \setminus A_1, B) \leq h$. It follows that $\bigcup_{A \in \mathbf{C}} \mathcal{B}_d(A, h) = A_1 \setminus N \neq \phi$. Since $A_1 \setminus N = C \setminus N$ and thus $N = C \setminus (A_1 \setminus N) = C \setminus \bigcup_{A \in \mathbf{C}} \mathcal{B}_d(A, h)$, it follows that (iv) holds.

Finally, assume that $\mid \mathcal{L}_d(h) \downarrow C \mid = 2$ and let $\mathcal{L}_d(h) \downarrow C = \{N_1, N_2\}$. Note that $N_1 \cap N_2 \neq \phi$ and $N_1 \cup N_2 = C$, otherwise $\mid \mathcal{L}_d(h) \downarrow C \mid > 2$. Therefore, each A_j is hierarchical with $\{N_1, N_2\}$ and $\bigcup_{A_j \in \mathbf{C}} \mathcal{B}_d(A_j, h) = \phi$. Moreover, denoting $\mathcal{N}_j^\star = \{A \in \mathbf{C} \,|\, A \subseteq N_j \setminus N_{3-j}$ for $j \in \{1, 2\}$, it is easily checked that $(\mathcal{N}_1^\star, \mathcal{N}_2^\star) = \max \mathcal{K}_d(\mathbf{C}, h)$. In addition, $N_1 = C \setminus (N_2 \setminus N_1) = C \setminus (\bigcup \mathcal{N}_2^\star)$, and similarly $N_2 = C \setminus (\bigcup \mathcal{N}_1^\star)$, which proves (v). ∎

Figure 1 depicts the five possible cases of proposition 1.

Since a dissimilarity u is a p-ultrametric if and only if $\bigcup \mathcal{L}_d(\Re^+)$ is p-hierarchical, the following characterization is clear.

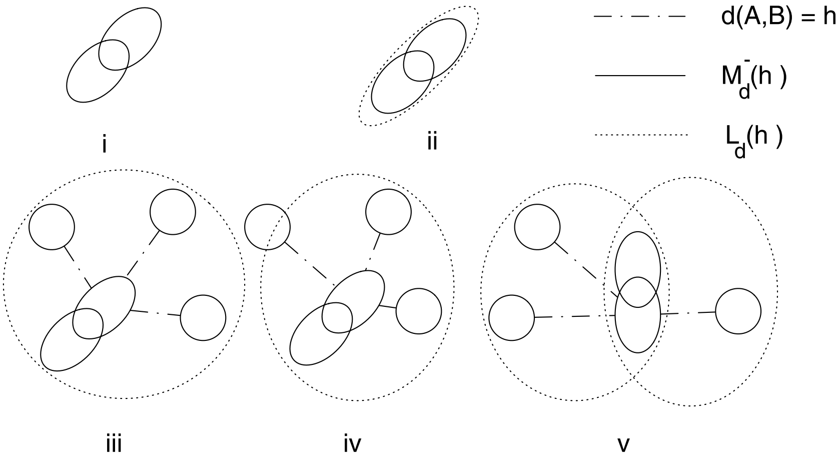


Fig. 1. The five cases of proposition 1.

Corollary 1. *A dissimilarity u an E is a p-ultrametric if and only if for all $h \in u(E^2) \setminus \{0\}$ and all connected component $\mathbf{C}$ of $\mathbf{G}_u[h]$ one and only one of the five assertions of proposition 1 is satisfied.*

Corollary 1 may be particularized to characterize ultrametrics, as shown in corollary 2.

Corollary 2. *A dissimilarity u on E is an ultrametric if and only if for all $h \in u(E^2) \setminus \{0\}$ and all connected component $\mathbf{C}$ of graph $\mathbf{G}_u[h]$, we have $\mathcal{L}_u(h) \downarrow C \in \{\phi, C\}$, with $C = \bigcup \mathbf{C}$.*

4 Construction of a lower-maximal p-ultrametric

We will here propose an algorithm which constructs a p-ultrametric u that is lower-maximal for a given dissimilarity d.

Recall that a dissimilarity d on E is *smaller* than a dissimilarity d' on E ($d' \preceq d$) if and only if $d(x,y) \leq d'(x,y)$ for all $x, y \in E$. Given a set $\mathcal{D}$ of dissimilarities on E, a dissimilarity d' in $\mathcal{D}$ is *lower-maximal* for a given dissimilarity d if and only if $d' \in \max\{d'' \mid d'' \preceq d,\ d'' \in \mathcal{D}\}$. It is well known that the set of ultrametrics admits a unique lower-maximal ultrametric for any given dissimilarity d and that, in addition, this lower-maximal ultrametric $u[d]$ is indeed the maximum of all ultrametrics u such that $u \preceq d$.

The hereunder algorithm will show that it exists a lower-maximal p-ultrametric for all given dissimilarity d that is also larger than $u[d]$. It can be shown that this lower-maximal p-ultrametric is never a maximum. The algorithm constructs a weakly indexed p-hierarchy associated to a p-ultrametric through the map Φ^{-1}. One can show that for a given weakly indexed p-hierarchy $(\mathcal{F}, f)$ and for any $x, y \in E$ we have (*cf.* Batbedat (1988); Bertrand (2000)): $[\Phi^{-1}(\mathcal{F}, f)(x, y) = \min\{f(A) \mid A \in \mathcal{F},\, x, y \in A\}$.

Algorithm. Construction of a lower-maximal p-ultrametric (u_p) for dissimilarity d on E.

```
begin
  $i \leftarrow 0$
  $\mathcal{F}_i \leftarrow \{\{x\} \mid x \in E\}$            (elements of $\mathcal{F}_i$ will be called *clusters*)
  $f_i(\{x\}) \leftarrow \{0\}$ for all $x \in E$    (values taken by $f_i$ will be called *levels*)
  $d_i \leftarrow d$
  while $|\,\mathrm{Part}\,(\max \mathcal{F}_i)\,| > 1$
    $\mathcal{F}_{i+1} \leftarrow \mathcal{F}_i$ ; $f_{i+1} \leftarrow f_i$ ; $d_{i+1} \leftarrow d_i$
    $i \leftarrow i + 1$
    $h_i \leftarrow \min\{d(A, B) \mid A \neq B,\, A, B \in \max \mathcal{F}_i\}$
    let $\mathbf{C}_1, \ldots, \mathbf{C}_{r_i}$ be the connected components of $\mathbf{G}_d[\mathrm{Part}\,(\max \mathcal{F}_i)](h_i)$
    for $1 \leq j \leq r_i$
      $C \leftarrow \bigcup \mathbf{C}_j$ ; $M_1 \leftarrow C$ ; $M_2 \leftarrow C$
      if $(\max \mathcal{F}_i) \downarrow C$ contains $A, B$ such that $d(A, B) = h_i$ then
        $\mathcal{J} \leftarrow \{A \in \mathbf{C} \cap \mathcal{F}_i \mid \mathcal{B}_d(A, h_i) \neq \phi\}$
        $\mathcal{K} \leftarrow \mathcal{K}_d(\mathbf{C}_j, h_i)$
(1)     if $\mathcal{J}$ or $\mathcal{K}$ is not empty then
          choose one which is not empty.
(2)       if $\mathcal{J}$ was chosen
            then let $M_1 \in \mathcal{J}$ ; $M_2 \leftarrow C \setminus \bigcup \mathcal{B}_d(M_1, h_i)$
(3)       if $\mathcal{K}$ was chosen
            then let $(\mathcal{A}^\star, \mathcal{B}^\star) \in \max \mathcal{K}$ ; $M_1 \leftarrow C \setminus \bigcup \mathcal{A}^\star$; $M_2 \leftarrow C \setminus \bigcup \mathcal{B}^\star$
        end (if)
(4)     for $k = 1, 2$: if $M_k \notin \mathcal{F}_i$ then
          $\mathcal{F}_i \leftarrow \mathcal{F}_i \cup \{M_k\}$; $f_i(M_k) \leftarrow h_i$
          for $x, y \in M_k$ : $d_i(x, y) \leftarrow \min\{d_i(x, y), h_i\}$
        end (if) ; end (for)
(5)     if $M_1 \cap M_2 \notin \mathcal{F}_i$ then
          $\mathcal{F}_i \leftarrow \mathcal{F}_i \cup \{M_1 \cap M_2\}$; $f_i(M_1 \cap M_2) \leftarrow \min\{f_i(M_1), f_i(M_2)\}$
        end (if)
      end (if)
    end (for)
  end (while)
  $N \leftarrow i$ ; $u_p \leftarrow d_N$
end
```

Lets show an example. On right part of Figure 2, the depicted p-hierarchy represents (via the bijection Φ) a lower-maximal p-ultrametric obtained by the algorithm from the dissimilarity of Table 1. On left part, the single link hierarchy represents the ultrametric sub-dominant of this dissimilarity. Showing overlapping clusters, the p-hierarchy clearly refines the single link hierarchy.

Table 1. A dissimilarity (taken from Bandelt and Dress (1989)).

	1	2	3	4	5	6	7
1. H. Sapiens	0	0.19	0.18	0.24	0.36	0.52	0.77
2. P. Paniscus		0	0.07	0.23	0.37	0.56	0.80
3. P. Troglodytes			0	0.21	0.37	0.51	0.77
4. G. Gorilla				0	0.38	0.54	0.75
5. P. Pygmæus					0	0.51	0.76
6. H. Lar						0	0.74
7. Cercopithecids							0

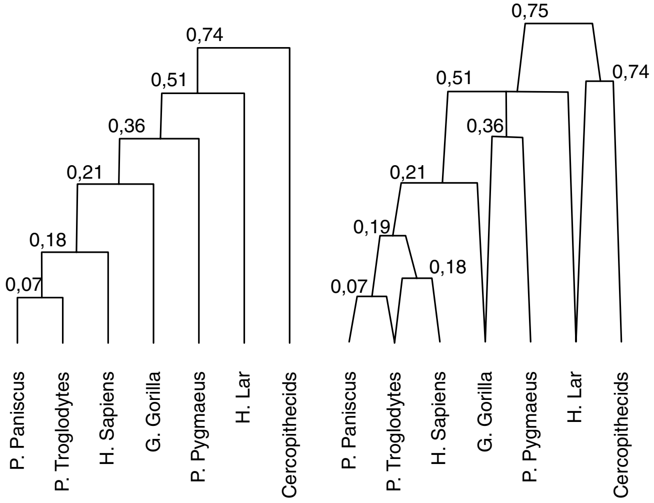


Fig. 2. Pyramids associated with the ultrametric sub-dominant (left) and a lower-maximal p-ultrametric (right) for the dissimilarity given in table 1.

Theorem 1. *For any dissimilarity d, the hereabove algorithm computes in a finite number N of iterations both* $(\mathcal{F}_N, f_N)$*, which is a weakly indexed p-hierarchy, and its associated p-ultrametric* $u_p = d_N$*. Moreover,*

(i) u_p *is greater than or equal to the sub-dominant ultrametric* $u[d]$ *of* d*,*
(ii) u_p *is a lower-maximal p-ultrametric of* d*.*

Proof. First, we show by induction on $i \geq 0$ that $\mathcal{F}_i$ is closed and p-hierarchical. It is obviously true for $i = 0$. Suppose it true for $i-1$ with $i \geq 1$, and consider the step i.

Indeed, at step i at most two maximal clusters and their intersection are added to $\mathcal{F}_{i-1}$ for each connected component $\mathbf{C}_j$ $(1 \leq j \leq r_i)$. More precisely, denoting $C_j = \bigcup \mathbf{C}_j$:

- C_j is added if $\mid \mathbf{C}_j \mid = 1$, $\max \mathcal{F}_{i-1} \downarrow C_j = \{A, B\}$ and $d(A, B) = h_i$,
- C_j is added if $\mid \mathbf{C}_j \mid > 1$ and $\mathcal{J}$ and $\mathcal{K}$ are empty (note that it then exists $\{A, B\} \subseteq \max \mathcal{F}_{i-1} \downarrow C_j$ such that $d(A, B) = h_i$),
- A, $C_j \setminus \bigcup B_d(A, h_i)$ (with $A \in \mathcal{J}$) and their intersection are added if $\mathcal{J}$ is not empty and chosen,
- $C_j \setminus \bigcup \mathcal{A}$, $C_j \setminus \bigcup \mathcal{B}$ (with $(\mathcal{A}, \mathcal{B}) \in \max \mathcal{K}$) and their intersection are added if $\mathcal{K}$ is not empty and chosen.

In all these cases, the added clusters satisfy the p-hierarchical condition, and are such that $\mathcal{F}_i$ is closed whenever $\mathcal{F}_{i-1}$ is closed. Thus $\mathcal{F}_i$ is closed and p-hierarchical by the hypothesis of induction, as required.

Let us now consider the end of step i. According to the clusters added during step i, for all A, B distinct in $\max \mathcal{F}_i$, $d(A, B) = d_i(A, B) > h_i$. This implies that for all $i \geq 0$, we have $h_{i+1} > h_i$ and $d_{i+1} \preceq d_i$ (by definition of d_{i+1}). Therefore, the series (h_i) is strictly growing, the series (d_i) is decreasing and both take their values in those taken by d. It results that the algorithm will stop in a finite number N of iterations. For the last iteration N, we will then have $\mid \max \mathcal{F}_N \mid = 1$, and thus $E \in \mathcal{F}_N$, which proves that $\mathcal{F}_N$ is p-hierarchy on E. Using again a proof by induction on $i \geq 0$, it is easily checked that $(\mathcal{F}_N, f_N)$ is a weakly indexed p-hierarchy. Now, by definition of d_N, it it is clear that $d_N = \Phi^{-1}(\mathcal{F}_N, f_N)$, and thus d_N is a p-ultrametric.

(i). We have to prove that $u[d] \preceq u_p = d_N$. Proceeding by induction on $i \geq 0$, we will prove that $u[d] \preceq d_i$ for all $i \geq 0$.

First, it is clear that $u[d] \preceq d_0$ for $d_0 = d$. Now, suppose that $u[d] \preceq d_{i-1}$ with $i \geq 1$ and let us prove that $u[d] \preceq d_i$. Since $d_i \preceq d_{i-1}$, it suffices to prove that $u[d](x, y) \leq d_i(x, y)$ only when $d_i(x, y) < d_{i-1}(x, y)$. Let us then assume $d_i(x, y) < d_{i-1}(x, y)$. By definition of d_i, it must exist some connected component $\mathbf{C}_j$ such that $x, y \in \bigcup \mathbf{C}_j$. Thus it exists a path $x = u_1, \ldots, u_p = y$ $(p \geq 2)$ where $u_k \in A_k \in \mathbf{C}_j$ for every $1 \leq k \leq p$, and two consecutive vertices A_k and A_{k+1} are either equal or linked. Therefore $d_{i-1}(u_k, u_{k+1}) \leq h_i$ for all $1 \leq k < p$ because for all $A \in \mathbf{C}_j$ and all $x', y' \in A$, $d_{i-1}(x', y') \leq h_{i-1} < h_i$ and $\mathbf{C}_j$ is a connected component of $\mathbf{G}_d[\mathrm{Part}\,(\max \mathcal{F}_{i-1})](h_i)$. Using the hypothesis of induction, we then obtain the required inequality:

$$u[d](x, y) \leq \max_{1 \leq k < p} u[d](u_k, u_{k+1}) \leq \max_{1 \leq k < p} d_{i-1}(u_k, u_{k+1}) \leq h_i \leq d_i(x, y).$$

(ii). Let d' be a p-ultrametric such that $d' \preceq d$. It suffices to set that $d' \preceq d_i$ for all $i \geq 0$. Clearly $d' \preceq d_0$. Now suppose that it exists $i \geq 1$ such that

$d' \preceq d_{i-1}$ and $d' \not\preceq d_i$. Thus x, y exist in E such that $d_i(x,y) < d'(x,y)$. At step i, let $\mathbf{C}_{j_0}$ be the connected component such that $x, y \in \bigcup \mathbf{C}_{j_0}$. Let us prove that there exists $x', y' \in E$ such that $d_N(x', y') > d'(x', y')$, which will prove that d_N and d' are not $\preceq$-comparable, and thus that (ii) holds. We will distinguish three cases.

Case 1. Assume it exists $x' \in \bigcup \mathbf{C}_j$ and $y' \notin \bigcup \mathbf{C}_j$ such that $d'(x', y') \leq h_i$. Since $d_i(x', y') > h_i$ and $h_k > h_i$ for all $k > i$ we have $d_N(x', y') = u_p(x', y') > d'(x', y')$, as required.

Case 2. Assume it exists x', y' in some $A \in \mathcal{F}_{i-1}$ such that $d'(x', y') \neq d_{i-1}(x', y')$. Since $u_p(x', y') = d_i(x', y') = d_{i-1}(x', y')$ for all $x', y' \in A$ (because $A \in \mathcal{F}_{i-1}$) and $d' \preceq d_{i-1}$, we must have $d'(x', y') < d_{i-1}(x', y') = u_p(x', y')$, as required.

Case 3. Assume cases 1 and 2 do not hold. We give the main lines of the proof in this case. First, the connected components of $\mathbf{G}_{d_{i-1}}[h_i]$ and $\mathbf{G}_{d'}[h_i]$ coincide. Since $\mathbf{G}_{d_{i-1}}[h_i] = \mathbf{G}_d[\text{Part}\,(\max \mathcal{F}_{i-1})](h_i)$, all the connected components $\mathbf{C}_j$'s at step i coincide with the connected components of $\mathbf{G}_{d'}[h_i]$. Moreover, case 2 does not hold, thus for all $z, t \in \bigcup \mathbf{C}_{j_0}$, $d_i(z,t) < h_i \Rightarrow d'(z,t) = d_i(z,t)$. Then, using the definition of the algorithm, a routine check shows that the set $d_i^{-1}(]h_i, +\infty[) \cap (\bigcup \mathbf{C}_{j_0})^2$ either contains or is uncomparable with the set $d'^{-1}(]h_i, +\infty[) \cap (\bigcup \mathbf{C}_{j_0})^2$. Since $d'(x,y) > d_i(x,y) = h_i$, there exists $x', y' \in \bigcup \mathbf{C}_{j_0}$ such that $d_i(x', y') > d'(x', y')$. Thus we have $u_p(x', y') > d'(x', y')$, as required. ∎

References

BANDELT, H.-J., DRESS, A.W.M. (1989): Weak Hierarchies Associated with Similarity Measures – an Additive Clustering Technique. *Bulletin of Mathematical Biology 51, 133-166.*

BATBEDAT, A. (1988): Les isomorphismes HTS et HTE (après la bijection de Benzécri-Johnson). *Metron, 46, 47–59.*

BENZÉCRI, J.-P. (1973): *L'Analyse des données : la Taxinomie*, vol. 1. Dunod, Paris.

BERTRAND, P. (2000): Set Systems and Dissimilarities. *Europ. J. Combinatorics 21, 727–743.*

BERTRAND, P. (2002): Set Systems for which Each Set Properly Intersects at Most One Other Set - Application to Pyramidal Clustering, cahier du Ceremade 0202, Univ. Paris-Dauphine, France.

DIDAY, E. (1986): Orders and Overlapping Clusters in Pyramids. In: J. De Leeuw, W. Heiser, J. Meulman, and F. Critchley (Eds.): *Multidimentional Data Analysis Proceedings.* DSWO Press, Leiden, 201–234.

FICHET, B. (1986): Data Analysis: Geometric and Algebric Structures. In: Y.A. Prohorov, and V.U. Sasonov (Eds.): *First world congress of the Bernoulli Society proceedings.* VNU Science Press, Utrecht, 123–132.

JOHNSON, S.C. (1967): Hierarchical Clustering Schemes. *Psychometrika 32, 241-254.*

A Note on Three-Way Dissimilarities and Their Relationship with Two-Way Dissimilarities

Victor Chepoi[1] and Bernard Fichet[2]

[1] Laboratoire d'Informatique Fondamentale, Université de la Méditerranée
163 avenue de Luminy, 13288 Marseille Cedex 9, France, *chepoi@lif.univ-mrs.fr*

[2] Laboratoire d'Informatique Fondamentale, Faculté de Médecine
Université de la Méditerranée, 27, Bd. Jean Moulin, 13385 Marseille cedex 5, France, *bernard.fichet@medecine.univ-mrs.fr*

Abstract. This note is devoted to three-way dissimilarities defined on unordered triples. Some of them are derived from two-way dissimilarities via an L_p-transformation ($1 \leq p \leq \infty$). For $p < \infty$, a six-point condition of Menger type is established. Based on the definitions of Joly-Le Calvé and Heiser-Bennani Dosse, the concepts of three-way distances are also discussed. A particular attention is paid to three-way ultrametrics and three-way tree distances.

1 Introduction

During the last two decades, we have witnessed a growing interest in three-way data analysis. Many results and methods have been established. See, for instance, the book of Coppi-Bolasco (1989). In this note, we pay attention to three-way dissimilarities, following the axiomatization given in the basic articles of Joly-Le Calvé (1985), Heiser-Bennani Dosse (1997) and Deza-Rosenberg (2000). We also refer to the dissertation thesis of Bennani Dosse (1993) and the talk of Joly-Le Calvé on ternary distances at IFCS meeting, Charlottesville, 1989.

A three-way dissimilarity t on a finite set I of size n indicates the (common) lack of resemblance between all of triples and is a natural extension of the usual (two-way) dissimilarities. As for two-way dissimilarities, symmetry is required. However, the following main question is raised: are we concerned with triples of the type $\{i, i, j\}$? The answer is "yes" in the models proposed by the above-mentioned authors, except the model in a paragraph of Bennani Dosse (1993). Here we deal with three-way dissimilarities only defined on unordered triples, as in the pioneering work of Hayashi (1972).

We denote by $\mathcal{P}_3(I)$ (resp. $\mathcal{P}_2(I)$) all subsets of I with three (resp. two) elements. Then a two-way predissimilarity d may be regarded as a mapping from $\mathcal{P}_2(I)$ into $\mathbb{R}$. We similarly define a three-way predissimilarity t as a mapping from $\mathcal{P}_3(I)$ into $\mathbb{R}$. Those are dissimilarities if they are nonnegative. For brevity, the value of t on $\{i, j, k\}$ is noted t_{ijk}, and the value of d on $\{i, j\}$ is noted d_{ij}. Of course, three-way dissimilarities t may be constructed from a two-way dissimilarity d via some transformations. For instance, Hayashi (1972) proposes to use the area of triangles derived from two-way Euclidean distances. In the present note, we pay attention to the L_p-*transformation* $(1 \leq p \leq \infty)$:

$$t_{ijk} = [d_{ij}^p + d_{jk}^p + d_{ki}^p]^{1/p} \ \text{ for every } \ \{i, j, k\} \in \mathcal{P}_3(I).$$

When $p = 1$, the formula remains valid for predissimilarities, and in that case we say that t is of perimeter type. The following six-point condition will be established.

Theorem 1. *A three-way predissimilarity t is of perimeter type if and only if for every subset $\{i_1, i_2, i_3, j_1, j_2, j_3\}$ of six distinct elements the following equality (1) holds*

$$3(t_{i_1 i_2 i_3} - t_{j_1 j_2 j_3}) = \sum_k [(t_{i_1 i_2 j_k} + t_{i_2 i_3 j_k} + t_{i_3 i_1 j_k}) - (t_{i_k j_1 j_2} + t_{i_k j_2 j_3} + t_{i_k j_3 j_1})].$$

The authors referred in this text have introduced different types of three-way metricity, employing several conditions on distinct or non-distinct elements. In our context, we keep two main definitions and exhibit some properties. Ultrametricity and tree-metricity are evoked in the same sense. In particular, there is a six-point condition for tree-metricity and a counter-example shows that such a condition is sharp.

Most of the results given here, have been presented by the authors at different meetings, namely these of the International Federation of Classification Societies, held in Rome, 1998, and those of the Ordinal and Symbolic Data Analysis, held in Darmstadt, 1997, and in Amherst, Massachusetts, 1998.

2 The perimeter model

We note by $\mathcal{D}$ and $\mathcal{T}$ the sets of two-way and three-way predissimilarities on n points, respectively. Clearly, those are real vector spaces with respective dimension $n(n-1)/2$ and $n(n-1)(n-2)/6$. A function t in $\mathcal{T}$ is of *perimeter type* if there is d in $\mathcal{D}$ such that

$$\text{for every } \ \{i, j, k\} \in \mathcal{P}_3(I), \ \ t_{ijk} = d_{ij} + d_{ik} + d_{jk} \ . \tag{2}$$

Denote by $f : \mathcal{D} \rightarrow \mathcal{T}$ the mapping associated with the perimeter model: namely, $t = f(d)$ if and only if (2) holds. Clearly, f is linear.

A simple example is given by star predissimilarities. Recall that a two-way star predissimilarity d is defined by real coefficients $\{w_i : i \in I\}$. We have $d_{ij} = w_i + w_j$. Similarly, we define a *three-way star* predissimilarity t by setting $t_{ijk} = a_i + a_j + a_k$ for given coefficients $\{a_i : i \in I\}$. Then, for every two-way star predissimilarity d, $f(d)$ is a star one and, conversely, for every three-way star predissimilarity t, there exists a star one d such that $t = f(d)$. It is well-known that for $n = 3$, every d in $\mathcal{D}$ is a star predissimilarity. Analogously, we have:

Proposition 1. *For $n = 4$, every $t \in \mathcal{T}$ is a star predissimilarity.*

Proof. A simple calculation shows that $3a_1 = t_{123} + t_{124} + t_{134} - 2t_{234}$, and that similar equalities hold for a_2, a_3, a_4. □

Thus for $n = 4$, and hence for $n = 3$, the mapping f is surjective.

The following equations lead us to a necessary condition for the perimeter model. They have been established by Bennani Dosse (1993) and rediscovered by the authors of this note. Dots mean summation over all possible units, unordered pairs or triples.

Summing in (2) for fixed i, j over all $k \neq i, j$ gives:

$$t_{ij.} = (n-4)d_{ij} + d_{i.} + d_{j.} \tag{3}$$

Now, for fixed i, summing over all $j \neq i$ give $t_{i..} = (n-3)d_{i.} + d_{..}$, whence summing up over all i gives $t_{...} = (n-2)d_{...}$ Notice that the previous equations are obvious for $n = 3, 4$. As a result, we obtain the following necessary condition for a t of perimeter type:

$$(n-3)(n-4)d_{ij} = (n-3)t_{ij.} - (t_{i..} + t_{j..}) + 2t_{...}/(n-2). \tag{4}$$

The following proposition is similar to the one of Joly-Le Calvé (1995). This is an immediate consequence of (4).

Proposition 2. *For $n > 4$, $t \in \mathcal{T}$ is of perimeter type if and only if for all $\{i, j, k\} \in \mathcal{P}_3(I)$ we have*

$$(n-4)t_{ijk} = (t_{ij.} + t_{ik.} + t_{jk.}) - 2(t_{i..} + t_{j..} + t_{k..})/(n-3) + 6t_{...}/(n-2)(n-3).$$

Checking if t is of perimeter type can be done by solving a linear system with $O(n^3)$ rows and $O(n^2)$ columns, with a $(0,1)$-matrix. Let us observe that (4) provides an $O(n^3)$-time algorithm for answering this question.

We can complete now the proof of Theorem 1. First, Proposition 1 shows that the statement is true for $n \leq 4$. So, suppose $n > 4$. Condition (4) shows that f is injective. Consequently, f is a bijection for $n = 5$, since $\mathcal{D}$ and $\mathcal{T}$ have the same dimension (equal to ten). Thus, the statement of the theorem is true for $n = 5$, and we may suppose that $n > 5$.

Now, we prove that t is of perimeter type if and only if its restriction to every subset of six points is. Necessity is obvious. Conversely, suppose that t obeys the announced six-point criterion. Let $i_1, ..., i_5$ be five distinct elements of I and i_6 and i'_6 be two other elements different from $i_1, ..., i_5$. Let $J = \{i_1, ..., i_5, i_6\}$, $J' = \{i_1, ..., i_5, i'_6\}$, and let u, u' be the restrictions of t to J and J', respectively. By hypothesis, there exist (unique) two-way predissimilarities d and d' on J and J', associated with u and u' via the perimeter model. Since f is a bijection for $n = 5$, u and u', hence d and d', share a common restriction on $\{i_1, ..., i_5\}$. Thus $d_{i_1 i_2}$ does not depend of the choice of i_6 in J and, by finite induction, $d_{i_1 i_2}$ does not depend of the choice of i_3, i_4, i_5, and i_6 either. So, a two-way predissimilarity d has been (well-) defined for every pair $\{i_1, i_2\}$ and clearly t obeys the perimeter condition for every triple.

Now, for every system of six distinct elements $i_1, i_2, i_3, j_1, j_2, j_3$, the condition of Proposition 2 can be written as

$$12t_{i_1i_2i_3} = 6(t_{i_1i_2.} + t_{i_1i_3.} + t_{i_2i_3.}) - 4(t_{i_1..} + t_{i_2..} + t_{i_3..}) + 3t_{...}$$

Checked on all triples, this condition appears to be equivalent to the one of the theorem. This concludes the proof of Theorem 1.

Now, we specify scalar products on $\mathcal{D}$ and $\mathcal{T}$, by considering the corresponding canonical bases as orthonormal. Then we may introduce the Moore-Penrose generalized inverse f^+ of f ($f^+ = f^{-1}$ when $n = 5$). For $n > 5$, f^+ defines a least squares approximation of perimeter type.

Proposition 3. *For $n > 5$ and for every $t \in \mathcal{T}$, $d = f^+(t)$ is characterized by (4).*

Proof. Let t' be the orthogonal projection of t into $f(\mathcal{D})$. Then, by definition, $f(d) = t'$. The projection t' is the solution of the minimization problem

$$\min \sum \{(t_{ijk} - t'_{ijk})^2 : t' \in f(\mathcal{D})\},$$

or equivalently

$$\min \sum \{(t_{ijk} - d_{ij} - d_{ik} - d_{jk})^2 : d \in \mathcal{D}\}.$$

For every pair $\{i, j\}$, taking the partial derivative with respect to d_{ij}, gives:

$$\sum_{k \neq i,j} [t_{ijk} - d_{ij} - d_{ik} - d_{jk}] = 0.$$

This is (3), which also yields (4). □

In fact, for $n \leq 4$, f^+ characterizes the unique solution in the subspace orthogonal to $\mathrm{Ker}(f)$. The following lemma will be used. We denote by $\mathcal{D}_{st}$ the subspace of $\mathcal{D}$ of all star predissimilarities and by $\mathcal{D}^*$ the subspace of $\mathcal{D}$ of all d obeying $d_{i.} = 0$ for every $i \in I$.

Lemma 1. *For every $n \geq 3$, the orthogonal decomposition $\mathcal{D} = \mathcal{D}_{st} \oplus \mathcal{D}^*$ holds.*

Proof. It is well-known that $\mathcal{D}_{st}$ is n-dimensional, a basis of which is given by the 1-dichotomies δ^i, $i \in I$. All coefficients w_j defining δ^i are null, except w_i, which is equal to 1. Moreover, it is easy to see that the 1-dichotomies are orthogonal to $\mathcal{D}^*$. Thus $\mathcal{D}_{st}$ and $\mathcal{D}^*$ are orthogonal. In order to prove the direct decomposition, define $g : \mathcal{D} \to \mathbb{R}^I$, where $g(d)$ has the components $d_{i.}$ for all $i \in I$. Then $\mathcal{D}^*$ =Ker(g), so that $\dim(\mathcal{D}^*) \geq n(n-1)/2 - n$. □

Note that this lemma remains valid when the basis of $\mathcal{D}$ is only orthogonal, with a precise norm on the basis vectors. Precisely, the basis vector, associated with the pair $\{i,j\}$ has a squared norm equal to $m_i m_j$, for fixed positive weights $\{m_k : k \in I\}$. In that case, the quantities $d_{i.}$ are weighted sums.

Proposition 4. *For $n = 4$ and for every t in $\mathcal{T}$, $d = f^+(t)$ is the two-way star predissimilarity (associated with the three-way star one t).*

Proof. By Proposition 1, t is a three-way star predissimilarity associated with a two-way star one d. But equations (3) and (4) show that $\mathrm{Ker}(f) \subseteq \mathcal{D}^*$. Thus d is orthogonal to $\mathrm{Ker}(f)$ and $d = f^+(t)$. □

For $n = 3$, $\mathrm{Ker}(f) = \{d : d_{..} = 0\}$. Consequently, f^+ maps onto the line defined by all d with equal coordinates. If $d = f^+(t)$, then $d_{ij} = d_{ik} = d_{jk} = t_{ijk}/3$.

We end this section with a few remarks. First, observe that $t = f(d)$ is a three-way dissimilarity when d is a (two-way) dissimilarity, but the converse is no longer true whenever $n \geq 4$. For $n = 4$, the three-way star dissimilarity defined by the coefficients $(-2, 1, 1, 1)$ does not admit any dissimilarity d such that $f(d) = t$; in contrast, the dissimilarity defined by the coefficients $(-1, -1, 4, 4)$ admits a (non-star) dissimilarity d such that $f(d) = t$.

Regarding the L_p-transformation ($1 \leq p < \infty$) in terms of p^{th}-power, the previous results apply. A three-way dissimilarity t is the L_p-transformation of a two-way dissimilarity if and only if t^p can be derived from a dissimilarity via the perimeter model. For $n > 5$, t obeys this condition if and only if its restriction to every subset of six points does. Notice that most of the preceding results do not hold for $p = \infty$. For example, the L_∞-transformation $d \to t$ is never injective.

3 Three-way metrics

In this section we deal with some extensions of two-way metrics (semi-distances), i.e. dissimilarities obeying the well-known triangular inequality. In the above-mentioned papers, an axiomatization has been discussed in details. In particular, Heiser-Bennani Dosse (1997) develop the relationship with a parametrized triangle inequality sensu Andreae-Bandelt (1995). In the context of three-way dissimilarities only defined on triples of distinct elements, axiomatization is simpler.

3.1 Definitions

We keep two main definitions:
- the *weak metricity* (Joly-Le Calvé (1995)): for distinct $i, j, k, l \in I$,

$$t_{ijk} \leq t_{ikl} + t_{jkl}, \text{ or equivalently, } \max[t_{ijk}, t_{jkl}] \leq t_{ijl} + t_{ikl}.$$

-the *strong metricity* (Heiser-Bennani Dosse (1997)):

$$2t_{ijk} \leq t_{ijl} + t_{ikl} + t_{jkl} \quad (\textit{tetrahedral inequality}).$$

Let us note that Deza-Rosenberg (2000) define, in the multiway case, a so called $n-$semimetricity. Using the same terminology, their simplex inequality for $n = 2$ (three-way case) is the tetrahedral inequality, except a coefficient 1 in the left-hand inequality. So, their definition turns out to be weaker than our weak-metricity. If we denote $\beta_1 \leq \beta_2 \leq \beta_3 \leq \beta_4$ the values of t on the four triples of a given quadruple, weak-metricity is equivalent to $\beta_4 \leq \beta_1 + \beta_2$ and strong metricity to $2\beta_4 \leq \beta_1 + \beta_2 + \beta_3$ ($\beta_4 \leq \beta_1 + \beta_2 + \beta_3$ sensu Deza-Rosenberg). Thus strong metricity implies weak metricity and both imply nonnegativity, like the usual triangle inequality.

Proposition 5. *Every three-way dissimilarity derived from a two-way metric via an L_p-transformation $(1 \leq p \leq \infty)$ is a strong metric.*

The proof of Heiser-Bennani Dosse (1997), established in a different model, holds here. In contrast, in our context, the converse is not true, even for $p = 1$: consider a two-way dissimilarity with the constant value 2 on every pair, except one pair, the value of which is 5.

Of course, there are three-way weak/strong metrics which are not of perimeter type. Consider the strong metric t with constant values and t' in a neighborhood of t and not in the subspace $f(\mathcal{D})$. However, for $n > 5$, Theorem 1 shows that a three-way (strong) metric derives from a two-way metric via the perimeter model if and only if its restriction to every subset of six points does. More precisely, using (4) and Theorem 1, one may prove after some computations the following proposition.

Proposition 6. *A three-way dissimilarity derives from a two-way metric via the perimeter model if and only if for every system $(i_1, i_2, i_3, j_1, j_2, j_3)$ of six distinct elements, the condition (1) of Theorem 1 and the following condition are fulfilled:*

$$\sum_k (t_{i_1 i_3 j_k} + t_{i_2 i_3 j_k} - t_{i_1 i_2 j_k}) \geq (t_{i_3 j_1 j_2} + t_{i_3 j_1 j_3} + t_{i_3 j_2 j_3}) - t_{j_1 j_2 j_3}.$$

An example of metrics is given by a star three-way predissimilarity t, with coefficients $a_1 \leq \ldots \leq a_n$. Then t is a strong metric if $a_1 \geq 0$, a weak metric if $2a_1 + a_2 \geq 0$, and a dissimilarity if $a_1 + a_2 + a_3 \geq 0$.

The inequalities occurring in the definitions of metricity show that the sets of weak and strong metrics are polyhedral cones. Consequently the sets of those metrics of perimeter type are too. Thus, different types of least squares approximations may be solved by some usual procedures, such as the algorithms of Lawson and Hanson (1974) or Dykstra (1983).

3.2 Three-way ultrametrics

Ultrametricity is treated in the same spirit. Recall that a two-way ultrametric is a dissimilarity fulfilling the ultrametric inequality $d_{ij} \leq \max[d_{ik}, d_{jk}]$ for every triple. For a three-way dissimilarity t, we here define:

-*weak ultrametricity* (Joly-Le Calvé (1995)): for distinct $i, j, k, l \in I$,

$$t_{ijk} \leq \max[t_{ijl}, t_{ikl}, t_{jkl}].$$

-*strong ultrametricity* (Bennani Dosse (1993)): for distinct $i, j, k, l \in I$,

$$t_{ijk} \leq \max[t_{ijl}, t_{ikl}], \text{ or equivalently, } \max[t_{ijk}, t_{jkl}] \leq \max[t_{ijl}, t_{ikl}].$$

Denoting by $\beta_1 \leq \beta_2 \leq \beta_3 \leq \beta_4$ the four values of t on the four triples of a given quadruple, weak ultrametricity is equivalent to $\beta_3 = \beta_4$ and strong ultrametricity is equivalent to $\beta_2 = \beta_3 = \beta_4$. Thus a strong ultrametric is a weak ultrametric. This is also a strong metric. However a weak ultrametric is not always a weak metric (it is a metric sensu Deza-Rosenberg). For this reason, we pay more attention to the strong condition. Notice that some strong ultrametrics are not of perimeter type: with Theorem 1, consider t verifying $t_{i_1 i_2 i_3} < t_{j_1 j_2 j_3} < c$ and $t_{uvw} = c$ otherwise.

Proposition 7. *If d is a two-way ultrametric, then the L_∞-transformation of d is a strong ultrametric.*

Proof. Using the well-known indexed hierarchy associated with (I, d), check all configurations on four points. □

It is easy to see that for $n = 4$, every strong ultrametric is the L_∞-transformation of an ultrametric. For $n > 4$, we have the following characterization (we recall that the *subdominant* of a two-way dissimilarity d is the greatest ultrametric less than d).

Proposition 8. *A three-way strong ultrametric t is the L_∞-transformation of some ultrametric if and only if it coincides with the L_∞-transformation of δ_*, where δ_* stands for the subdominant of δ defined by $\delta_{ij} = \min_k t_{ijk}$ for every pair.*

Proof. Observe that δ is the greatest dissimilarity with L_∞-transformation less than t, and δ_* is the greatest ultrametric with an L_∞-transformation less than t. □

The following counter-example shows that the condition is not always fulfilled for $n > 4$. Namely, choose t verifying $t_{ikl} = t_{jkm} = 1$ and $t_{uvw} = 2$ otherwise, for five units $\{i, j, k, l, m\}$.

It is quite easy to show that every 3-way dissimilarity t admits a subdominant weak and a subdominant strong ultrametrics. Both may be computed by recursively shrinking some values of t over all quadruples, in order to fulfil the constraints of the respective definitions. This is a pure extension of the algorithm developed by Roux (1968) for the 2-way ultrametric subdominant. See, also, Benzécri (1973). Clearly, there also exists a subdominant ultrametric of t, L_∞-transformation of some ultrametric, namely the L_∞-transformation of δ_* defined in Proposition 8.

The previous three subdominant approximations yield, by a simple translation, three approximations according to the supremum norm on $\mathcal{T}$, in their own context. For this, apply the general result linking subdominants and L_∞-approximations, as established by Chepoi-Fichet (2000).

3.3 Three-way tree-metrics

Recall that a two-way metric d is said to be of tree-type, if the metric space (I, d) embeds isometrically in some weighted tree, endowed with the usual distance. A necessary and sufficient condition is given by the famous four-point condition:

$$d_{ij} + d_{kl} \leq \max[d_{ik} + d_{jl}, d_{il} + d_{jk}],$$

for every quadruple of distinct elements; see, for instance, Buneman (1974). We define here a *three-way strong metric* t of *tree-type* as a three-way strong metric derived from a metric of tree-type via the perimeter model. Equivalently t is of tree-type if and only if it derives from a metric via the perimeter model and it obeys the following five-point condition:

$$t_{ijm} + t_{klm} \leq \max[t_{ikm} + t_{jlm}, t_{ilm} + t_{jkm}] \tag{5}$$

for every subset of five distinct elements.

Theorem 1 shows that for $n > 5$, t is of tree-type if and only if its restriction to every subset of six points is too. The following counter-example shows that the condition cannot be weakened. It exhibits a three-way strong metric t on six points i, j, k, l, u, v, which is not of perimeter type, hence not of tree-type, whose restrictions to all subsets of five points are of tree-type. Figure 1 and symmetrical ones define the restrictions.

It is well-known that every two-way ultrametric obeys the four-point condition. Similarly, one may prove:

Proposition 9. *Every three-way strong ultrametric obeys condition (5).*

Proof. Without loss of generality one may suppose that t_{ijk} is the smallest value among the ten values taken by t on the five points i, j, k, l, m. Again, one may suppose that $t_{ijl} \leq t_{ijm}$. Thus, by ultrametricity we obtain:

$$t_{ijk} \leq t_{ijl} = t_{ikl} = t_{jkl} =: A \leq t_{ijm} = t_{ikm} = t_{jkm} =: B.$$

To establish the five-point condition, we must prove, for every specific point, that among the three possible sums of two values, two are equal and greater than the third one. We distinguish two cases.

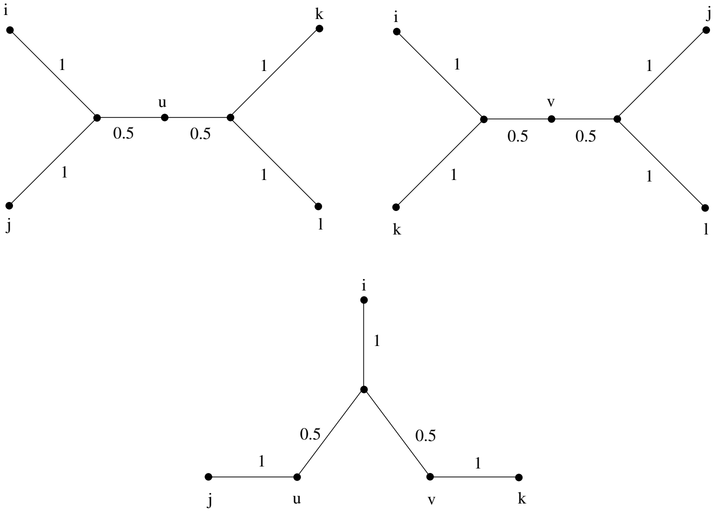


Fig. 1. A three-way strong metric which is not of tree-type, but whose restrictions to all subsets of five points are of tree-type.

First suppose $A < B$. By ultrametricity on $\{i, j, l, m\}$, $t_{ilm} = t_{jlm} = B$. Similarly, by symmetry, $t_{klm} = B$. Using symmetry between i, j, k, we observe that the condition is satisfied, for m, l or i, as specific point. Now suppose $A = B$. By ultrametricity on $\{i, j, l, m\}$, we obtain $\max[t_{ilm}, t_{jlm}] = B$. Similarly, by symmetry, $\max[t_{ilm}, t_{klm}] = \max[t_{jlm}, t_{klm}] = B$. If $t_{ilm} = t_{jlm} = t_{klm} = B$, the five-point condition is satisfied (using symmetry), for every specific point i or l. If say $t_{ilm} < t_{jlm} = t_{klm} = B$, the five-point condition is still fulfilled (using symmetry) whatever the specific point is i, j or l. □

As an immediate consequence, we have:

Corollary 1. *Every three-way strong ultrametric derived from a metric via the perimeter model, is of tree-type.*

3.4 r-Way dissimilarities

All concepts introduced above extend to the r-way case, by defining r-way dissimilarities or r-way metrics, even if a vast variety of definitions may be suggested for metricity or ultrametricity. In particular, denoting by $\mathcal{P}_r\,(I)$ the set of all subsets of size r, we define an *r-way predissimilarity,* as a mapping t from $\mathcal{P}_r(I)$ into $\mathbb{R}$. The value t on $A \in \mathcal{P}_r(I)$, is noted t_A.

As a basic example, one has the r-way star predissimilarity related to a family $\{w_i : i \in I\}$: for all $A \in \mathcal{P}_r(I)$, $t_A = \Sigma\{w_i : i \in A\}$. From an s-way predissimilarity d, $(s < r)$, we generalize the perimeter model by introducing the L_1-transformation: for all $A \in \mathcal{P}_r(I)$, $t_A = \Sigma\{d_B : B \in \mathcal{P}_s(A)\}$. We have the following immediate proposition.

Proposition 10. *If an r-way predissimilarity is the L_1-transformation of an s-way predissimilarity, then it is the L_1-transformation of an s'-way one, for every $s < s' < r$.*

In particular, every r-way star-predissimilarity is the L_1-transformation of an s-way one, for every $2 \leq s < r$.

Let us note that strange as it may seem, our definition applies for $r = 1$. In that case, any (1-way) predissimilarity is defined by the quantities $t(\{i\}) = w_i$ (say), $i \in I$. This is a star one, and any (two-way) star predissimilarity is the L_1-transformation of such a (1-way) one.

Many results may be extended. For instance, we have the following proposition.

Proposition 11. *Let f be the (linear) function, mapping the set (vector space) of r-way predissimilarities into the set (vector space) of $(r+1)$-way ones, via the L_1-transformation. Then:*
(i) f is injective if $n > 2r+1$;
(ii) f is a bijection if $n = 2r+1$;
(iii) f is surjective if $n < 2r+1$.

Acknowledgment. We are grateful to Patrice Bertrand for a careful reading of the manuscript and for several useful remarks.

References

ANDREAE, T. and BANDELT, H.J. (1995): Performance guarantees for approximation algorithms depending on parametrized triangle inequalities. *SIAM Journal on Discrete Mathematics 8, 1-16.*

BENNANI DOSSE, M. (1993): Analyses métriques à trois voies. Dissertation thesis, Université de Haute-Bretagne, France.

BENZÉCRI, J.P. (1973): *L'analyse des Données, 1 La Taxinomie.* Dunod, Paris.

BUNEMAN, P. (1974): A note on metric properties of trees. *J. Combin. Theory, Ser. B 17, 48-50.*

CHEPOI, V. and FICHET, B. (2000): l_∞-Approximation via subdominants. *Journal of Mathematical Psychology 44, 600-616.*

COPPI, R. and BOLASCO, S. (1989): *Multiway Data Analysis.* North-Holland, Amsterdam.

DEZA, M.-M. and ROSENBERG, I.G. (2000): n-Semimetrics. *Europ. J. Combinatorics 21, 797-806.*

DYKSTRA, R.L. (1983): An algorithm for restricted least squares regressions. *Journal of American Statistical Association 78, 837-842.*

HAYASHI, C. (1972): Two dimensional quantification based on the measure of dissimilarity among three elements. *Annals of the Institute of Statistical Mathematics 24, 251-257.*

HEISER,W.J. and BENNANI, M. (1997): Triadic distance models: axiomatization and least squares representation. *Journal of Mathematical Psychology 41, 189-206.*

JOLY, S. and LE CALVÉ, G. (1995): Three-way distances. *Journal of Classification 12, 191-205.*

LAWSON, C. and HANSON, R.J. (1974): *Solving Least Squares Problems.* Prentice-Hall.

ROUX, M. (1968): *Un Algorithme pour Construire une Hiérarchie Particulière.* Dissertation thesis, ISUP, University of Paris VI.

One-to-One Correspondence Between Indexed Cluster Structures and Weakly Indexed Closed Cluster Structures

Jean Diatta

IREMIA, Université de la Réunion
15 avenue René Cassin, BP 7151, 97 715 Saint-Denis Messag. Cedex 9, France,
jean.diatta@univ-reunion.fr

Abstract. We place ourselves in a setting where singletons are not all required to be clusters, and we show that the resulting cluster structures and their corresponding closure under finite nonempty intersections still have the same minimal members. Moreover, we show that indexed cluster structures and weakly indexed closed cluster structures correspond in a one-to-one way.

1 Introduction

The most known cluster structure is certainly the hierarchical one whose specificity lies in the absence of overlapping clusters, which makes it suitable for data visualization. However, the absence of overlap prevents the hierarchical cluster structure from being able to figure out situations where an entity shares features with entities from different clusters. To cope with this, overlapping cluster structures have been introduced or considered by several authors (Diday (1884), Batbedat (1988), Durand and Fichet (1988), Bandelt and Dress (1989), Diatta and Fichet (1994)). Some of these cluster structures are closed under finite nonempty intersections.

In this note, we place ourselves in a setting where singletons are not all required to be clusters, and we show that the resulting cluster structures and their corresponding closure under finite nonempty intersections still have the same minimal members. Moreover, we show that indexed cluster structures and weakly indexed closed cluster structures correspond in a one-to-one way. This result generalizes the one obtained by Batbedat (1988) in the particular case where each singleton is assumed to be a cluster. The paper is organized as follows.

Cluster structures are introduced in Section 2 within the general setting where singletons are not all required to be clusters. In Section 3, we consider closed cluster structures and show that they have the same minimal members as their corresponding cluster structures. Finally, the bijection between weakly indexed closed cluster structures and indexed cluster structures is given in Section 4 before a short conclusion.

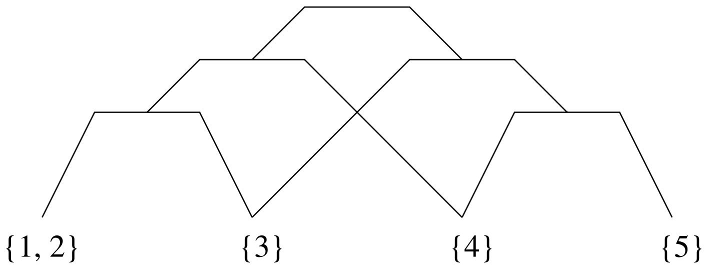


Fig. 1. A Hasse diagram representing a cluster structure.

2 Cluster structures

Let E be a finite nonempty set. A *cluster structure* on E is a collection $\mathcal{C}$ of subsets of E, satisfying conditions (CS1), (CS2) and (CS2'), where:

(CS1) the empty set is not a member of $\mathcal{C}$ whereas the ground set E is, i.e., $\emptyset \notin \mathcal{C}$ and $E \in \mathcal{C}$;

(CS2) the set of minimal members of $\mathcal{C}$ (w.r.t. set inclusion) partitions E; in other words, these minimal members are non-empty, pairwise disjoint, and they cover E (i.e. their union equals E);

(CS2') every non-minimal member of $\mathcal{C}$ is the union of members of $\mathcal{C}$ it properly contains, i.e., for all $X \in \mathcal{C}$: $\cup\{Y \in \mathcal{C} : Y \subset X\} \in \{\emptyset, X\}$.

The pair of conditions (CS2) and (CS2') is often replaced by a stronger condition requiring each singleton to be a member of $\mathcal{C}$. Actually, a cluster structure satisfying this strong requirement is said to be total or definite. Figure 1 represents a cluster structure $\mathcal{C}_1$ on the 7-element set $E_1 := \{1, 2, 3, 4, 5, 6, 7\}$.

3 Closed cluster structures

To every subset collection can be associated its closure consisting of arbitrary intersections of its members. As we are concerned with collections of nonempty subsets of finite sets, we will consider only finite nonempty intersections. The closure of a subset collection $\mathcal{C}$ under finite nonempty intersections will be denoted by $\overline{\mathcal{C}}$, and $\mathcal{C}$ will be said to be *closed* when it satisfies the condition (CS3) below:

(CS3) the intersection of two members of $\mathcal{C}$ is either empty or a member of $\mathcal{C}$, i.e., $X, Y \in \mathcal{C}$ implies $X \cap Y \in \mathcal{C} \cup \{\emptyset\}$.

It may be noted that conditions (CS2) and (CS2') are equivalent under conditions (CS1) and (CS3). Moreover, for any subset X of E, conditions

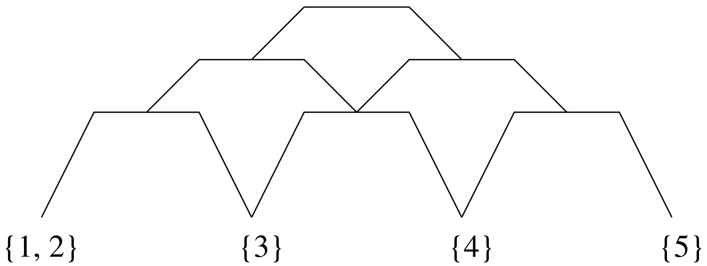


Fig. 2. A Hasse diagram representing a closed cluster structure.

(CS1) and (CS3) guarantee the existence of the least member of $\mathcal{C}$ containing X.

A *closed cluster structure* is a cluster structure satisfying Condition (CS3) above. As an example, Figure 2 represents a closed cluster structure which is the closure of the cluster structure presented in Figure 1; incidentally, it can be noticed that if the pair $\{1, 2\}$ is considered as a singleton, then the above closed cluster structure is nothing else than a pyramid in the sense defined by Diday (1884).

To show that $\overline{\mathcal{C}}$ is a closed cluster structure when $\mathcal{C}$ is a cluster structure, we just need to prove that every minimal member of $\mathcal{C}$ is also a minimal member of $\overline{\mathcal{C}}$. This will result from the two following lemmas.

Lemma 1. *The following conditions are equivalent for a collection $\mathcal{C}$ of nonempty subsets of E.*

(a) $C, C' \in \mathcal{C}$ and C minimal in $\mathcal{C}$ imply $C \cap C' \in \{\emptyset, C\}$.
(b) Every minimal member of $\mathcal{C}$ is minimal in $\overline{\mathcal{C}}$.

Proof. (a) implies (b). Let C be a minimal member of $\mathcal{C}$ and let $C_1, \ldots, C_p$ be members of $\mathcal{C}$ such that $(C_1 \cap \cdots \cap C_p) \cap C \neq \emptyset$. Then for each $i = 1, \ldots, p$, $C \cap C_i \neq \emptyset$, so that, by (a), $C \subseteq C_1 \cap \cdots \cap C_p$. Then C is minimal in $\overline{\mathcal{C}}$ since we cannot have $C_1 \cap \cdots \cap C_p \subset C$.
(b) implies (a). Let C be a minimal member of $\mathcal{C}$ and let $C' \in \mathcal{C}$ such that $C \cap C' \neq \emptyset$. Then $C \cap C' \in \overline{\mathcal{C}}$, so that $C \cap C' = C$ since, by (b), C is minimal in $\overline{\mathcal{C}}$.
□

Lemma 2. *The conjunction of conditions (CS2) and (CS2') is equivalent to the conjunction of conditions (a) and (c), where:*

(a) $C, C' \in \mathcal{C}$ and C minimal in $\mathcal{C}$ imply $C \cap C' \in \{\emptyset, C\}$;
(c) minimal members of $\mathcal{C}$ cover E.

Proof. Let $\mathcal{C}$ satisfy conditions (CS2) and (CS2') and let C be one of its minimal members. We just need to derive condition (a). Let $C' \in \mathcal{C}$ such that $C \cap C' \neq \emptyset$. If C' is minimal in $\mathcal{C}$, then, by (CS2), $C = C'$, proving (a). If C' is not minimal, let $x \in C \cap C'$. Then, by condition (CS2'), there exists a finite maximal sequence $(C'_i)_{1 \leq i \leq p}$ of members of $\mathcal{C}$ such that $x \in C'_1 \subset \cdots \subset C'_p \subset C'$. Then C'_1 is minimal, so that, by condition (CS2), $C = C'_1$, proving (a).

Conversely, let C be a non-minimal member of $\mathcal{C}$ and let $x \in C$. Let C' be a minimal member of $\mathcal{C}$ containing x (C' exists by condition (c)). Then, by condition (a), $C' \subset C$ since C is not minimal, proving (CS2'). To complete the proof we just have to show that minimal members of $\mathcal{C}$ are pairwise disjoint. This follows from condition (a).

□

Let a *reducible* member of a cluster structure $\mathcal{C}$ be a non-minimal member which can be obtained as the intersection of other members of $\mathcal{C}$. Let $\mathcal{C}^o$ be the subset collection obtained from $\mathcal{C}$ by removing the reducible members of $\mathcal{C}$. Then, the result below shows that the closure of a cluster structure is a closed cluster structure.

Proposition 1. *The following hold for any cluster structure $\mathcal{C}$:*

(a1) $\overline{\mathcal{C}}$ *is a closed cluster structure;*
(a2) $\mathcal{C}$ *and* $\overline{\mathcal{C}}$ *have the same minimal members;*
(a3) $(\overline{\mathcal{C}})^o \subseteq \mathcal{C}$.

Conversely, the following hold for any closed cluster structure $\mathcal{C}$:

(b1) $\mathcal{C}^o$ *is a cluster structure;*
(b2) $\mathcal{C}$ *and* $\mathcal{C}^o$ *have the same minimal members;*
(b3) $\overline{\mathcal{C}^o} = \mathcal{C}$.

Proof. According to Lemmas 1 and 2, to prove assertions (a1) and (a2), it is sufficient to show that minimal members of $\overline{\mathcal{C}}$ are minimal in $\mathcal{C}$. Now, this follows from the fact that minimal members of $\mathcal{C}$ are minimal in $\overline{\mathcal{C}}$ as well as they partition the ground set. The other assertions are immediate.

□

It may be noticed that, for a cluster structure $\mathcal{C}$, $(\overline{\mathcal{C}})^o = \mathcal{C}$ if and only if $\mathcal{C}$ has no reducible member.

4 Indexed cluster structures and weakly indexed closed cluster structures

Let $\mathcal{C}$ be a cluster structure on E. A *pre-index* on $\mathcal{C}$ is an order preserving map $f : (\mathcal{C}, \subseteq) \to (\mathbb{R}_+, \leq)$ taking the zero value on minimal members of $\mathcal{C}$, i.e.,

(i) C minimal implies $f(C) = 0$;

(ii) $C \subseteq C'$ implies $f(C) \leq f(C')$.

In the sequel, we will assume that a pre-index f takes the value zero only on minimal members, hence, $f(C) = 0$ if and only if C is minimal. A canonical pre-index f_c can be obtained by letting $f_c(C)$ be the number of elements of the union of members of $\mathcal{C}$ properly contained in C. An *index* on $\mathcal{C}$ is a strict pre-index, i.e., a pre-index f such that $C \subset C'$ implies $f(C) < f(C')$. A *weak index* (Bertrand, 2000) on $\mathcal{C}$ is a pre-index f such that

$$C \subset C' \text{ and } f(C) = f(C') \text{ imply } C = \cap\{C'' \in \mathcal{C} : C \subset C''\}.$$

When f is a pre-index (resp. an index, a weak index) on a cluster structure $\mathcal{C}$, the pair $(\mathcal{C}, f)$ is called a *pre-indexed* (resp. an indexed, a *weakly indexed*) cluster structure. Let $(\mathcal{C}, f)$ be a pre-indexed cluster structure on E. Let $\text{Inter}(\mathcal{C}, f)$ denote the pair $(\overline{\mathcal{C}}, \overline{f})$, where $\overline{f}$ is defined on $\overline{\mathcal{C}}$ by

$$\overline{f}(C) = \min\{f(C') : C' \in \mathcal{C} \text{ and } C \subseteq C'\}.$$

On the other hand, define an *f-maximal* member of $\mathcal{C}$ to be a non-minimal member $C \in \mathcal{C}$ such that there is no member $C' \in \mathcal{C}$ such that $C \subset C'$ and $f(C) = f(C')$. Let $\text{Strict}(\mathcal{C}, f)$ denote the pair $(\underline{\mathcal{C}}, \underline{f})$, where $\underline{\mathcal{C}}$ is composed of minimal and f-maximal members of $\mathcal{C}$, and $\underline{f}$ the restriction of f on $\underline{\mathcal{C}}$. Then $\text{Strict}(\mathcal{C}, f)$ is clearly an indexed cluster structure. Moreover, Proposition 2 below, proven by Batbedat (1988) in the particular case of definite cluster structures, still holds in the setting adopted in the present paper, since each of the maps Strict and Inter preserves minimal members.

Proposition 2.

(i) *If* $(\mathcal{C}, f)$ *is an indexed cluster structure, then* $\text{Strict}(\text{Inter}(\mathcal{C}, f)) = (\mathcal{C}, f)$.

(ii) *If* $(\mathcal{C}, f)$ *is a pre-indexed closed cluster structure, then* $\text{Inter}(\text{Strict}(\mathcal{C}, f)) = (\mathcal{C}, f)$ *if and only if any irreducible member of* $\mathcal{C}$ *is* f*-maximal.*

The next proposition shows that indexed cluster structures and weakly indexed closed cluster structures correspond in a one-to-one way.

Proposition 3.

(i) *If* $(\mathcal{C}, f)$ *is an indexed cluster structure on* E*, then* $\text{Inter}(\mathcal{C}, f)$ *is a weakly indexed closed cluster structure on* E*. Moreover,* $\text{Strict}(\text{Inter}(\mathcal{C}, f)) = (\mathcal{C}, f)$.

(ii) *Conversely, if* $(\mathcal{C}, f)$ *is a weakly indexed closed cluster structure on* E*, then* $\text{Strict}(\mathcal{C}, f)$ *is an indexed cluster structure on* E*. Moreover,* $\text{Inter}(\text{Strict}(\mathcal{C}, f)) = (\mathcal{C}, f)$.

Proof. (i). We just have to prove that $\overline{f}$ is a weak index on $\overline{\mathcal{C}}$. Indeed, $f(C) = 0$ if and only if C is minimal in $\mathcal{C}$. Now, as minimal members of $\mathcal{C}$ coincide

with those of $\overline{\mathcal{C}}$, $\overline{f}(C) = 0$ if and only if C is minimal in $\overline{\mathcal{C}}$. On the other hand, let $C_1, C_2 \in \overline{\mathcal{C}}$ such that $C_1 \subset C_2$. Then clearly $\overline{f}(C_1) \leq \overline{f}(C_2)$. If, in addition, $\overline{f}(C_1) = \overline{f}(C_2)$, then $C_1 \notin \mathcal{C}$ since, otherwise, there would be $C \in \mathcal{C}$ such that $C_1 \subset C_2 \subseteq C$ with $f(C_1) = \overline{f}(C_1) = \overline{f}(C_2) = f(C)$, which is impossible since $(\mathcal{C}, f)$ is indexed. Hence $C_1 = \cap\{C' \in \mathcal{C} : C_1 \subset C'\}$, as required. The second assertion derives from Proposition 2 (i).
(ii). Here again, we only have to prove that $\underline{f}$ is an index on $\underline{\mathcal{C}}$. Now, this derives from the following: (1) $\mathcal{C}$ and $\underline{\mathcal{C}}$ have the same minimal members, and (2) there are no two members C_1, C_2 of $\underline{\mathcal{C}}$ such that $C_1 \subset C_2$ and $f(C_1) = \underline{f}(C_1) = \underline{f}(C_2) = f(C_2)$. The second assertion follows from Lemma 2 (ii) because, by definition of a weak index, a non-minimal cluster C_1 belonging to $\mathcal{C}$ is necessarily reducible if it is not f-maximal.
□

5 Conclusion

We discussed the relationships between cluster structures and closed cluster structures within a setting where singletons are not all required to be clusters. Moreover, we proved a bijection between indexed cluster structures and weakly indexed closed cluster structures, generalizing the result obtained by Batbedat (1988) in the particular case where each singleton is assumed to be a cluster.

References

BANDELT, H.-J. and DRESS, A.W.M. (1989): Weak hierarchies associated with similarity measures: an additive clustering technique. *Bull. Math. Biology 51, 113-166.*

BATBEDAT, A. (1988): Les isomorphismes HTS et HTE (après la bijection de Benzécri/Johnson) (première partie). *Metron 46, 47-59.*

BERTRAND, P. (2000): Set Systems and Dissimilarities. *Europ. J. Combinatorics 21, 727-743.*

DIATTA, J. and FICHET, B. (1994): From Apresjan hierarchies and Bandelt-Dress weak hierarchies to quasi-hierarchies. In: E. Diday, Y. Lechevalier, M. Schader, P. Bertrand and B. Burtschy (Eds.): *New Approaches in Classification and Data Analysis.* Springer-Verlag, 111-118.

DIDAY, E. (1984): Une représentation visuelle des classes empiétantes : les pyramides. Technical Report 291, INRIA, France.

DURAND, C. and FICHET, B. (1988): One-to-one correspondences in pyramidal representation: a unified approach. In H. H. Bock (Ed.): *Classification and Related Methods of Data Analysis.* North-Holland, Amsterdam, 85-90.

Adaptive Dissimilarity Index for Gene Expression Profiles Classification

Ahlame Douzal Chouakria[1], Alpha Diallo[2], and Françoise Giroud[3]

[1] TIMC-IMAG TIMB (CNRS UMR 5525), Université Joseph Fourier Grenoble 1,
F-38706 LA TRONCHE Cedex, France, *Ahlame.Douzal@imag.fr*

[2] TIMC-IMAG RFMQ (CNRS UMR 5525), Université Joseph Fourier Grenoble 1,
F-38706 LA TRONCHE Cedex, France, *Alpha.Diallo@imag.fr*

[3] TIMC-IMAG RFMQ (CNRS UMR 5525), Université Joseph Fourier Grenoble 1,
F-38706 LA TRONCHE Cedex, France, *Francoise.Giroud@imag.fr*

Abstract. DNA microarray technology allows to monitor simultaneously the expression levels of thousands of genes during important biological processes and across collections of related experiments. Clustering and classification techniques have proved to be helpful to understand gene function, gene regulation, and cellular processes. However the conventional proximity measures between genes expression data, used for clustering or classification purpose, do not fit gene expression specifications as they are based on the closeness of the expression magnitudes regardless of the overall gene expression profile (shape). We propose in this paper an adaptive dissimilarity index which would cover both values and behavior proximity. The effectiveness of the adaptive dissimilarity index is illustrated through a classification process for identification of genes cell cycle phases.

1 Introduction to microarray technology

Though most cells in our bodies contain the same genes, not all of the genes are used in each cell. Some genes are turned on, or "expressed" when needed. Such specific genes define the "molecular pattern" related to a specific function of a cell and in most cases appear as organized in a molecular regulation network. To know how cells achieve such specialization, scientists need a way to identify which genes each type of cell expresses. Microarray technology now allows us to look at many genes at once and determine which are expressed in a particular cell type (Eisen and Brown (1999)). DNA molecules representing many genes are placed in discrete spots regularly organized in a line/column matrix on a microscope slide. This is called a DNA microarray. Thousands of individual genes (clones) can be spotted on a single square inch slide surface. Next, total messenger RNA (the working copies of genes within cells, indicators of which genes are being used) is purified from cells. The RNA molecules are then labeled by attaching a fluorescent dye and spread over the DNA dots on the microarray. Due to a phenomenon termed base-pairing, RNA will stick to the gene it came from (this is the hybridization process). After washing away all of the unstuck RNA, we can look at the microarray under a microscope and see which RNA remains stuck to the DNA spots. Fluorescent mea-

surements are performed using specific scanners and related spot fluorescent values are extracted from images (http://genomewww.stanford.edu/HumanCellCycle/HeLa/). Since we know which gene each spot represents, we can determine which genes are turned on in the cells. Some researchers are using this powerful technology to learn which genes are turned on or off in diseased versus healthy human tissues for example. The genes that are expressed differently in the two tissues may be involved in causing the disease. In other experiments time-course DNA microarray analysis are necessary to determine temporal genomic expression profiles relative to the dynamic progression of a specific biological process or to response at stimulation or treatment. In this paper we will be interested in the dynamic progression of cell division cycle. Additionally, in order to take in account systematic biases in the measured expression levels related to experimental factors, two-channel array experiments are usually performed. It consists in using a reference material in parallel to the tested material. For example: normal cells used as references versus pathological ones being the tested cells. Both materials are labeled using two different colors (green and red) and are mixed in equal proportion prior to hybridization. The final expression measured is given as log(base2)ratio between the tested material against the reference one.
The purpose of clustering or classification tasks is to determine co-expressed genes which indicate co-function and co-regulation. Because different genes are usually functionally implied in a same regulation network, users of microarrays data may not only be interested in clustering or classifying genes, but also be interested in the relationship between these clusters (e.g. which clusters are most close to each other), and the relationship between the genes within the same cluster (e.g. which gene can be considered as the representative of the cluster and which ones are at the boundary area of the cluster).

2 Proximity measure between genes expression data

For clustering or classifying a set of genes expression profiles evolving over time, the commonly used proximity measures are the euclidean distance or the person's correlation coefficient. Let $g_1 = (u_1, ..., u_p)$ and $g_2 = (v_1, ..., v_p)$ be the expressions levels of two genes g_1, g_2 observed at the instant of times $(t_1, ..., t_p)$. On the one hand, the Euclidean distance δ_E between g_1 and g_2 is defined as: $\delta_E(g_1, g_2) = \left(\sum_{i=1}^{p}(u_i - v_i)^2\right)^{\frac{1}{2}}$. It stems directly from the above definition that the closeness between two expression profiles depends on the closeness of the values observed at corresponding points of time. δ_E ignores the information of interdependence among the observed values. However, for genes expression data, the overall shapes of gene expression patterns are of greater interest than the individual magnitudes at corresponding instants of time.

2.1 Shape proximity measures

The alternate conventional measure to estimate the similarity between gene expression shapes is Pearson's coefficient correlation (called classical correlation). Unfortunately, we will illustrate in the following that the classical correlation do not score well for proximity between shapes either. For shape proximity measure, we propose the temporal correlation coefficient introduced in Chouakria Douzal (2003), Chouakria Douzal and Nagabhushan (2006) and defined as follows:

$$\textsc{cort}(g_1, g_2) = \frac{\sum_{i=1}^{p-1}(u_{(i+1)} - u_i)(v_{(i+1)} - v_i)}{\sqrt{\sum_{i=1}^{p-1}(u_{(i+1)} - u_i)^2}\sqrt{\sum_{i=1}^{p-1}(v_{(i+1)} - v_i)^2}}$$

The temporal correlation coefficient $\textsc{cort} \in [-1, 1]$ presents an interesting property, it allows to estimate the linear dependency between the growths of two gene expression profiles, observed at corresponding times. A value of $\textsc{cort} = 1$ means that the growths (positive or negative) observed on both expression profiles, at any corresponding instant of time, are similar in direction and rate (similar behavior). On the contrary a value of -1 means that the growths observed on both expression profiles, at any corresponding instant of time, are similar in rate but opposite in direction (opposite behavior). Finally, a value of 0 expresses that the growths observed on both expression patterns are stochastically linearly independent (different behaviors).

2.2 Adaptive dissimilarity index for gene expression proximity

Our aim is to provide a new dissimilarity index model D which would cover both proximity on values $\delta_E(g_1, g_2)$ and on behavior $cort(g_1, g_2)$. The model would allow to adjust the weights of behavior (shape) or values components. The proposed model is based on an adaptive tuning function which modulates the proximity on values according to the proximity on behavior. The modulating function will increases the proximity on values if the proximity on behavior (i.e the temporal correlation) decreases from 0 to -1. The resultant dissimilarity D approaches the proximity on values if the temporal correlation is zero (different behaviors). Finally, the modulating function will decreases the proximity on values if the proximity on behavior (i.e temporal correlation) increases from 0 to +1. The formulation to compute the resultant dissimilarity index D is:

$$D_(S_1, S_2) = f(cort(S_1, S_2)).\delta_E(S_1, S_2)$$

where $f(x)$ is an exponential adaptive tuning function fitting the above properties:

$$f(x) = \frac{2}{1 + exp(k\ x)} \quad k \in 0, 1, ...$$

Figure 1 shows the adaptive tuning function effect for several values of k. The parameter k defines the weights, in the dissimilarity index D, of both behavior and values components as summarized in the Table 1. For instance for k=5, $D \rightsquigarrow 2\ \delta_E$ when $cort \rightsquigarrow -1$ and decreases until $D \rightsquigarrow 0$ when $cort \rightsquigarrow 1$, finally when $cort \rightsquigarrow 0$ $D \rightsquigarrow \delta_E$. Figure 1 illustrates that higher is the value of k, higher will be the temporal correlation weight and lower will be δ_E weight.

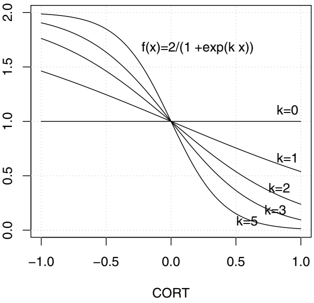


Fig. 1. The adaptive tuning effect.

	Behavior weight (%)	Values weight (%)
k=0	0%	100 %
k=1	50%	50%
k=2	80%	20%
k=3	90%	10%
$k \geq 5$	$\rightsquigarrow$ 100%	$\rightsquigarrow$ 0%

Table 1. Behavior (*cort*) and Values (δ_E) weights according to k.

3 Classification for genes expression profiles

We propose to compare the adaptive dissimilarity index with the classical correlation through a genes classification (assignment) approach. For the genes classification purpose, we define first two conventionally used genes assignment approaches: a supervised and an unsupervised approaches. Two genes samples are considered: a learning sample based on a set of well-studied genes, and a test sample based on a set of published genes compiled from the literature. Let's give briefly the algorithmic details of these assignment approaches, first in the case of the classical correlation Cor as a genes proximity measure, then in the case of the adaptive dissimilarity index D.

3.1 Supervised and unsupervised assignment approaches based on the classical correlation Cor

The supervised assignment approach based on the classical correlation noted ($SupAss - Cor$) consists to assign each gene to the most similar prior class (Average-Link, Centroid-Link,...) of the well-studied genes. The assessment step consists to evaluate the rand index between the obtained and the prior classes of the published genes. The unsupervised assignment approach based on $1 - Cor$ noted ($UnsupAss - cor$) consists first to perform an hierarchical clustering of the whole genes to classify, then each obtained cluster is assigned to the most similar prior class as detailed in the following:

```
Begin UnsupAss-Cor
 % assignment part %
- Perform an Hierarchical clustering (Average-Link) of the whole genes
- Extract the Nb clusters partition,
- Estimate the proximity  between each obtained cluster
  and the Nb prior classes of the well-studied genes,
- Assign each cluster to the most similar class,
- Assign each gene to the cluster's class it belongs in.
 %assessement part%
- Evaluate the rand index between the obtained and prior classes
  of the published genes
End
```

3.2 Supervised and unsupervised assignment approaches based on the adaptive dissimilarity index D

The main idea of the assignment approach based on the adaptive D, is to learn the weights of both values and behavior components of D to fit best the prior partition of the well-studied genes. Let's give the algorithmic steps of the supervised assignment approaches based on D and noted $SupAss - D$.

```
Begin SupAss-D based on the adaptive D
% assignment part %
  - For each value of k from 0 to  6 per 0.1   %(61 values)
      - Assign each gene to the most similar class
        (Average-Link, Centroid-link,...)
      - Evaluate the rand index between the obtained
        and the prior classes of the well-studied genes.
  - End For
  - Let k* be the value of k maximizing the rand index
    and Pk* the corresponding obtained assignments
  % assessement part%
  - Evaluate the rand index between the obtained and prior
   classes of the published genes
 End
```

Similarly the unsupervised assignment approach based on D noted $UnSupAss-D$ algorithmic steps are:

```
Begin UnsupAss-D
 % assignment part %
 - For each value ok k from 0 to  6 per 0.1    %(61 values)
    - Perform an Hierarchical clustering (Average-Link) of the whole genes
   - Extract the Nb clusters partition,
    - Estimate the proximity  between each obtained cluster
      and the Nb prior classes of the well-studied genes,
    - Assign each cluster to the most similar class,
    - Assign each gene to the cluster's class it belongs in.
 - End For
 - Let k* be the value of k maximizing the obtained rand
    index and Pk* the corresponding obtained assignments
 %assessement part%
 - Evaluate the rand index between the obtained and prior classes
   of the published genes 19 End
```

4 Application and results

4.1 Data description

In this paper we will focus on the specific biological events occurring during cell proliferation, this process insuring the multiplication or reproduction of cells and which is drastically aberrant in cancer cells. The cell cycle, or cell-division cycle, is the series of events between one cell division and the next one. The cell cycle consists of progression along four distinct phases: G1 phase, S phase (DNA synthesis or DNA replication), G2 phase and M phase. A molecular surveillance system monitors the cell's progress through the cell cycle and checkpoints help to ensure that a cell divides only when it has completed all of the molecular prerequisites for producing healthy daughter cells. These restriction points mark the transition from one phase to another : the transition from G1 to S phase is the first such transition (G1/S). According to that, we will focus on the G1/S, S, G2, G2/M and M/G1 phases and transitions we will short cut named "cell cycle phases" in the text. The genome-wide program of gene expression during the cell division cycle has been investigated in a wide range of organisms Spellman et al. (1998), Cho et al. (2001), Oliva et al. (2005), using DNA microarrays. In this paper we will focus on a set of genes expression data recorded in the third experimentation of Whitfield et al. published data Whitfield et al. (2002) (http://genome-www.standford.edu/Human-CellCycle/Hela/). The dataset describes 1099 genes, periodically expressed in the human cell cycle. RNA was isolated from Hela cells et 1 hour intervals after release from a synchronous arrest in S phase. Two lists of genes are considered respectively for learning and assessment steps. On the one hand a list of 20 well-studied genes composed of 4 referenced genes for each of the 5 phases is used for learning step (Table 2, Figure 4). On the other hand, and for assessment step, a list of 39 genes was compiled Whitfield et al. (2002) from the literature that had been shown to be cell cycle regulated by traditional bio-molecular methods (Table 5).

4.2 Identification of genes cell cycle phases results

To illustrate the efficiency of the adaptive dissimilarity index D against the classical correlation, we compare their effectiveness to identify the cell cycle phases of the 39 published genes, through the supervised and unsupervised

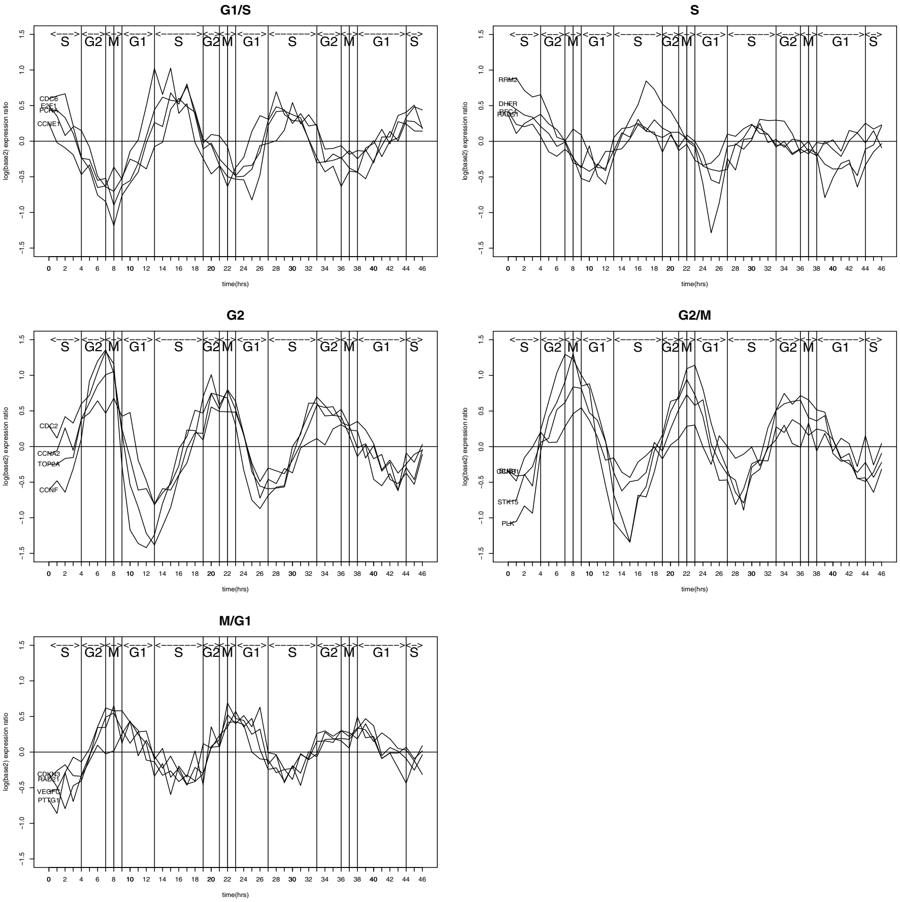


Fig. 2. Gene expression profiles for the 20 well-characterized cell cycle genes whose expression peaks in each phase of the cell cycle : G1/S, S, G2, G2/M and M/G1. The double arrowed lines delimit the time duration for each cell cycle phase : G1, S, G2 and M.

Phase	G1/S	S	G2	G2/M	M/G1
Name	CCNE1,E2F1 CDC6,PCNA	RFC4,DHFR RRM2, RAD51	CDC2, TOP2A CCNF, CCNA2	STK15,BUB1 CCNB1, PLK	PTTG1, RAD21 VEGFC, CDKN3

Table 2. List of the 20 genes assigned in Whitfield et al. (2002) to the 5 cell cycle phases.

assignment approaches. On the one hand, we have performed a supervised assignment *SupAss* for centroid and average link. We have then compared the obtained assignments when the supervised approach is based on the classical correlation and on the adaptive D. The obtained results of the assignments of the 39 published genes and the corresponding corrected rand index are reported in the Table 5 at the columns 3-4 for average-link and 5-6 for centroid-link. On the other hand, we have performed the unsupervised assignment approach $UnsupAss$ based respectively on the classical correlation and on the proposed dissimilarity index D. First an hierarchical clustering is performed on the whole 1099 genes based respectively on $1 - Cor$ and Dk. A 5 clusters partition is then extracted. For each extracted cluster we estimate it's dissimilarity to each of the well-referenced phases. The dissimilarity values between the 5 clusters and the 5 phases are reported in the Tables 3 and 4. The obtained dendrograms illustrating the 5 obtained clusters and the identified cell cycle phases are given in the Figure 3. Each gene is then assigned to the cluster's phase it belongs in. The assignments of the 39 published genes obtained through $UnsupAss - Cor$ and $UnsupAss - Dk$ are reported in the last two columns of the Table 5.

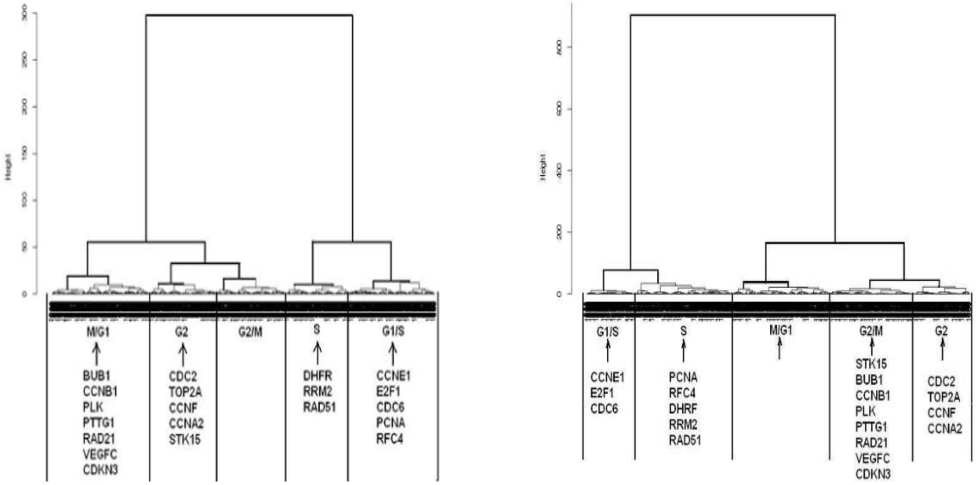


Fig. 3. The unsupervised approach: the dendrograms of the 1099 genes and their phases identification based on Cor(left) and D(right).

5 Discussion and future scope

5.1 Comparative analysis

Let's note that the assignments obtained in Whitfield et al. (2002) corresponds to the centroid-linkage $SupAss-Cor$ (3rd column). We can first show,

	G1/S	S	G2	G2/M	M/G1
Cluster 1	0.755	**0.416**	0.806	1.236	1.461
Cluster 2	**0.404**	0.632	1.314	1.589	1.512
Cluster 3	1.461	0.976	**0.345**	0.451	0.663
Cluster 4	1.500	1.056	0.540	**0.475**	0.694
Cluster 5	1.494	1.411	0.737	0.457	**0.426**

Table 3. Unsupervised approach based on Cor: similarity between the 5 extracted clusters and the 5 well-referenced phases.

	G1/S	S	G2	G2/M	M/G1
Cluster 1	1.520	**0.762**	3.284	5.016	4.123
Cluster 2	**0.502**	1.180	6.374	7.238	4.936
Cluster 3	5.709	2.194	**0.761**	0.907	1.161
Cluster 4	6.688	3.565	0.989	**0.464**	0.521
Cluster 5	4.264	3.025	2.873	1.755	**1.158**

Table 4. Unsupervised approach based on Dk* (k*=3.9): similarity between the 5 extracted clusters and the 5 well-referenced phases.

that whatever is the considered variant of the supervised approach, the rand index of $SupAss - D$ is greater than the one obtained through $SupAss - Cor$, as illustrated at the last row of the Table 5. Hence, the genes cell cycle phases of the 39 published genes are better identified through the adaptive dissimilarity index D than through the classical correlation. Through the both assignment approaches $UnsupAss - Cor$ and $UnsupAss - Dk$, each cluster is assigned to a distinguish phases. However the 20 referenced genes are not well distributed through the 5 extracted clusters. Indeed, through $UnsupAss - Cor$, 7 referenced genes 4 from M/G1 and 3 from G2/M are merged in a same cluster labeled as M/G1, with one cluster including no referenced genes (Figure 3 on left) and labeled as G2/M. A nearly similar distribution is obtained through $UnsupAss - D$, 8 referenced genes 4 from M/G1 and 4 from G2/M are merged in a same cluster labeled as G2/M, with one cluster including non referenced genes (Figure 3 on right) and labeled as M/G1. Finally, all the obtained assignment results show that whatever is the assignment approach (supervised or unsupervised) the identification of the genes cell cycle phases is better through the adaptive dissimilarity index than through th classical correlation.

5.2 The unsupervised classification: a promising tool for better understanding of dynamic cell cycle events

Considering the actual fast progression in the acquisition of new biological data, mainly due to recent developments in high throughoutput experimental methods (such as DNA microarrays), biological concepts and knowledge are undergoing drastic and rapid evolution. Keeping this in mind it appears quite reasonable to expect some invaluable assistance from unsupervised classification methods rather than supervised ones to help in understanding the complexity of life. The results obtained in this specific study, dedicated to

Name	Published Phase	Supervised Average-Link		Supervised Centroid-Link		UnSupervised	
		COR	D_{k*}	COR	D_{k*}	COR	D_{k*}
E2F5	G1	G2/M	M/G1	G2/M	M/G1	G2/M	G2/M
CCNE1	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S
CCNE2	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S
CDC25A	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S
CDC6	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S
CDKN3	G1/S	M/G1	M/G1	M/G1	M/G1	M/G1	G2/M
E2F1	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S
MCM2	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S	S
MCM6	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S
NPAT	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S
PCNA	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S	S
SLBP	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S
BRCA1	S	S	S	S	S	G1/S	S
CDKN2C	S	G2	S	G2	S	G2	G2
DHFR	S	S	S	S	S	S	S
MSH2	S	G1/S	S	G1/S	S	G1/S	S
NASP	S	G1/S	G1/S	G1/S	G1/S	G1/S	G1/S
RRM1	S	S	S	S	S	S	S
RRM2	S	S	S	S	S	S	S
TYMS	S	S	S	S	S	S	S
CCNA2	G2	G2	G2	G2	G2	G2	G2
CCNF	G2	G2	G2	G2	G2	G2	G2
CENPF	G2	G2/M	G2/M	G2/M	G2/M	M/G1	G2M
TOP2A	G2	G2	G2	G2	G2	G2	G2
BIRC5	G2/M	G2/M	M/G1	G2/M	M/G1	M/G1	G2/M
BUB1	G2/M	G2/M	G2/M	G2/M	G2/M	M/G1	G2/M
BUB1B	G2/M	G2/M	G2/M	G2/M	G2/M	G2/M	G2/M
CCNB1	G2/M	G2/M	M/G1	G2/M	M/G1	M/G1	G2/M
CCNB2	G2/M	G2/M	M/G1	G2/M	G2/M	M/G1	G2/M
CDC2	G2/M	G2	G2	G2	G2	G2	G2
CDC20	G2/M	G2/M	M/G1	G2/M	M/G1	M/G1	G2/M
CDC25B	G2/M	G2/M	M/G1	G2/M	M/G1	M/G1	G2/M
CDC25C	G2/M	G2	M/G1	G2	M/G1	G2/M	G2
CDKN2D	G2/M	M/G1	M/G1	G2/M	M/G1	M/G1	M/G1
CENPA	G2/M	G2	G2	G2/M	G2	G2	G2/M
CKS1	G2/M	G2	G2	G2	G2	G2	G2
CKS2	G2/M	G2/M	G2/M	G2/M	G2/M	M/G1	G2/M
PLK	G2/M	G2/M	G2/M	G2/M	G2/M	M/G1	G2/M
STK15	G2/M	G2/M	G2/M	G2/M	G2/M	G2	G2/M
Rand Index		0.760	0.830	0.790	0.818	0.757	0.771

Table 5. The assignment cell cycle phases of the 39 published genes.

better understanding cell cycle progression and regulation, bring some support to such an expectation. For example, considering results obtained by the unsupervised classification associated to D (Fig.5 right) it's possible to drawback three interesting and encouraging remarks. Note first, the classification process lead to the 5 expected cell cycle phases, then the PCNA gene which has been chosen by Whitfield et al. (2002) as representative of G1/S phase has been classified by the $UnsupASS-D$ approach in the S phase. And effectively it's quite well established that PCNA is a DNA polymerase expressed at the highest levels in the S-phase. Indeed if PCNA is first expressed in mid-G1, PCNA expression peaks in S phase and continues to be weakly expressed in G2 and M phases of the cell cycle. Finally, among the four misclassified M/G1 genes as G2/M by the $UnsupASS-D$ approach we will just discuss, as an example, on the PTTG1 gene. It has been recently demonstrated, by classical molecular biology methods, that the PTTG1 expression peaks at the S-G2 transition and declined thereafter Vlotides et al. (2006). According to that, it makes sense to work out PTTG1 gene as classified in the G2/M cluster rather than in the M/G1 one. On the basis of all these encouraging remarks our future works will focus on the biological processes related to the different genes obtained in the 5 clusters including new biological knowledge and the genes implication in regulation cell cycle molecular network.

6 Conclusion

This paper focuses on a new application domain of the microarrays and genes expression profile analysis. We introduce the microarrays technology, discuss main challenges of genes expression profile analysis and the great need of clustering and classification techniques. For genes expression profile classification, we propose an adaptive dissimilarity index which would cover both values and behavior proximity. We show it's effectiveness for genes identification cell cycle phases , whatever is the considered assignment approach.

References

CHO, R.J., HUANG, M., CAMPBELL, M.J., DONG, H., STEINMETZ, L., SAPINOSO, L., HAMPTON, G. , ELLEDGE, S.J., DAVIS, R.W. and LOCKHART, D.J. (2001): Transcriptional regulation and function during the human cell cycle. *Nature Genetics 27(1), 48-54.*

CHOUAKRIA DOUZAL, A. (2003): Compression technique preserving correlations of a multivariate temporal sequence. In: Berthold M R, Lenz H J, Bradley E, Kruse R, Borgelt C (Eds): *Advances in Intelligent Data Analysis.* Springer, Berlin Heidelberg, 566-577.

CHOUAKRIA DOUZAL, A. and NAGABHUSHAN, P. (2006): Improved Fréchet distance for time series. In: V. Batagelj, H.-H. Bock, A. Ferligoj, A. Ziberna (Eds): *Data Science and Classification*, Springer, 29-38.

EISEN, M.B. and BROWN, P.O. (1999): DNA arrays for analysis of gene expression. *Methods Enzymol 303, 179-205.*

JAVIER, A., BATA-CSORGO, Z., ELLIS, C.N., KANG, S., VOORHEES, J.J. and COOPER, K.D. (1997): Rapamycin (Sirolimus) inhibits proliferating cell nuclear antigen expression and blocks cell cycle in the G1 phase in human keratinocyte stem cells. *J. Clin. Invest 99(9), 2094-2099.*

OLIVA, A., ROSEBROCK, A., FERREZUELO, F., PYNE, S. , CHEN, H., SKIENA, S., FUTCHER, B. and LEATHERWOOD, J. (2005): The cell cycle-regulated genes of Schizosaccharomyces pombe. *PLoS Biol, 3(7):e225.*

SPELLMAN, P.T., SHERLOCK, G., ZHANG, M.Q., IYER, V.R., ANDERS, K., EISEN, M.B., BROWN, P.O., BOTSTEIN, D. and FUTCHER, B. (1998): Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization.*Mol. Biol. Cell 9, 3273-3297.*

VLOTIDES, G. , CRUZ-SOTO, M., RUBINEK, T., EIGLER, T., AUERNHAMMER, C.J. and MELMED, S. (2006): Mechanisms for growth factor-induced Pituitary Tumor Transforming Gene-1 (PTTG1) expression in pituitary folliculostellate TtT/GF cells, Molecular Endocrinology. First published ahead of print September 7, as doi:10.1210/me.2006-0280.

WHITFIELD, M.L., SHERLOCK, G., MURRAY, J. I., BALL, C.A., ALEXANDER, K.A., MATESE, J.C., PEROU, C.M., HURT, M.M., BROWN, P.O. and BOTSTEIN, D. (2002): Identification of genes periodically expressed in the human cell cycle and their expression in tumors. *Molecular Biology of the Cell 13 (6), 1977-2000.*

Lower (Anti-)Robinson Rank Representations for Symmetric Proximity Matrices

Lawrence J. Hubert[1] and Hans-Friedrich Köhn[2]

[1] Department of Psychology, University of Illinois
603 East Daniel Street, Champaign, Illinois 61820, USA
lhubert@cyrus.psych.uiuc.edu

[2] Department of Psychology, University of Illinois
603 East Daniel Street, Champaign, Illinois 61820, USA
hkoehn@cyrus.psych.uiuc.edu

Abstract. Edwin Diday, some two decades ago, was among the first few individuals to recognize the importance of the (anti-)Robinson form for representing a proximity matrix, and was the leader in suggesting how such matrices might be depicted graphically (as pyramids). We characterize the notions of an anti-Robinson (AR) and strongly anti-Robinson (SAR) matrix, and provide open-source M-files within a MATLAB environment to effect additive decompositions of a given proximity matrix into sums of AR (or SAR) matrices. We briefly introduce how the AR (or SAR) rank of a matrix might be specified.

1 Introduction

Various methods have been developed in the classification literature for representing the structure that may be present in a symmetric proximity matrix. The motivating bases for these strategies have been diverse, and include the reliance on spatial analogues (e.g., in multidimensional scaling), graph-theoretic concepts (e.g., in hierarchical clustering and the construction of additive trees), and order-constrained approximation matrices (e.g., matrices that satisfy the set of (anti-)Robinson (AR) order restrictions, characterized by a pattern of entries within each row and column never decreasing when moving away from the main diagonal in any direction; for historical precedents, see Robinson (1951)). It is within this last category of approximating a given proximity matrix by another that is order-constrained (and where, for convenience, proximity is now assumed keyed as a dissimilarity, so smaller values reflect more similar objects) in which Diday's contributions loam large. In the early 1980's and culminating in Diday (1986), he introduced the field to how (anti-)Robinson matrices may generally be represented through what are called pyramidal indices and their associated graphical display, or more broadly, to the relevance of the (graph-theoretic) literature on object seriation and its relation to the notion of an (anti-)Robinson form. We briefly review in this short paper a few of the advances in the last two decades, emphasizing, in particular, how sums of AR matrices might be identified and fitted through the minimization of a least-squares loss criterion. For a very comprehensive and current review of the whole area of hierarchical representations and their various extensions, the reader is referred to Barthélemy, Brucker, and Osswald (2004).

2 Some definitions

Given an arbitrary symmetric $n \times n$ matrix, $\mathbf{A} = \{a_{ij}\}$, where the main diagonal entries are considered irrelevant and assumed to be zero (i.e., $a_{ii} = 0$ for $1 \leq i \leq n$), $\mathbf{A}$ is said to have an anti-Robinson (AR) form if after some reordering of the rows and columns of $\mathbf{A}$, the entries within each row and column have a distinctive pattern: moving away from the zero main diagonal entry within any row or any column, the entries never decrease. The entries in any AR matrix $\mathbf{A}$ can be reconstructed exactly through a collection of M subsets of the original object set $S = \{O_1, \ldots, O_n\}$, denoted by $S_1, \ldots, S_M$, and where M is determined by the particular pattern of tied entries, if any, in $\mathbf{A}$. These M subsets have the following characteristics:

(i) each S_m, $1 \leq m \leq M$, consists of a sequence of (two or more) consecutive integers so that $M \leq n(n-1)/2$. (This bound holds because the number of different subsets having consecutive integers for any given fixed ordering is $n(n-1)/2$, and will be achieved if all the entries in the AR matrix $\mathbf{A}$ are distinct).

(ii) each S_m, $1 \leq m \leq M$, has a diameter, denoted by $d(S_m)$, so that for all object pairs within S_m, the corresponding entries in $\mathbf{A}$ are less than or equal to the diameter. The subsets, $S_1, \ldots, S_M$, can be assumed ordered as $d(S_1) \leq d(S_2) \leq \cdots \leq d(S_M)$, and if $S_m \subseteq S_{m'}$, $d(S_m) \leq d(S_{m'})$.

(iii) each entry in $\mathbf{A}$ can be reconstructed from $d(S_1), \ldots, d(S_M)$, i.e., for $1 \leq i, j \leq n$,

$$a_{ij} = \min_{1 \leq m \leq M} \{d(S_m) \mid O_i, O_j \in S_m\},$$

so that the minimum diameter for subsets containing an object pair $O_i, O_j \in S$ is equal to a_{ij}. Given $\mathbf{A}$, the collection of subsets $S_1, \ldots, S_M$ and their diameters can be identified by inspection through the use of an increasing threshold that starts from the smallest entry in $\mathbf{A}$, and observing which subsets containing contiguous objects emerge from this process. The substantive interpretation of what $\mathbf{A}$ is depicting reduces to explaining why those subsets with the smallest diameters are so homogenous.

If the matrix $\mathbf{A}$ has a somewhat more restrictive form than just being AR, and is also *strongly* anti-Robinson (SAR), a convenient graphical representation can be given to the collection of AR reconstructive subsets $S_1, \ldots, S_M$ and their diameters, and how they can serve to retrieve $\mathbf{A}$. Specifically, $\mathbf{A}$ is said to be strongly anti-Robinson (SAR) if (considering the above-diagonal entries of $\mathbf{A}$) whenever two entries in adjacent columns are equal

$(a_{ij} = a_{i(j+1)})$, those in the same two adjacent columns in the previous row are also equal $(a_{(i-1)j} = a_{(i-1)(j+1)}$ for $1 \leq i-1 < j \leq n-1)$; also, whenever two entries in adjacent rows are equal $(a_{ij} = a_{(i+1)j})$, those in the same two adjacent rows in the succeeding column are also equal $(a_{i(j+1)} = a_{(i+1)(j+1)}$ for $2 \leq i+1 < j \leq n-1)$.

The reconstruction of an SAR matrix through the collection of consecutively defined object subsets, $S_1, \ldots, S_M$, and their diameters, and how these serve to reconstruct $\mathbf{A}$ can be modeled graphically (see Figure 1). Internal nodes would be at a height equal to the diameter of the respective subset; the consecutive objects forming that subset are identifiable by downward paths from the internal nodes to the terminal nodes corresponding to the objects in $S = \{O_1, \ldots, O_n\}$. An entry a_{ij} in $\mathbf{A}$ can be reconstructed as the minimum node height of a subset for which a path can be constructed from O_i up to that internal node and then back down to O_j.

As a few final introductory historical notes, there is now a rather extensive literature on graphically representing a matrix having an AR or SAR form. The reader interested in pursuing some of the relevant literature might begin with the earlier cited reference by Diday (1986) and his introduction to graphically representing an AR matrix by a 'pyramid', and then continue with the review by Durand and Fichet (1988), who point out the necessity of strengthening the AR condition to one that is SAR if a consistent graphical (pyramidal) representation is to be possible with no unresolvable graphical anomalies. For further discussion and development of some of these representations issues, the reader is referred to Diatta and Fichet (1998), Critchley (1994), Critchley and Fichet (1994), and Mirkin (1996, Chapter 7).

2.1 An illustrative numerical example

The proximity matrix given in Table 1 was published by *The New York Times* (July 2, 2005), and contains the percentages of non-unanimous cases in which the U.S. Supreme Court Justices *dis*agreed from the 1994/95 term through 2003/04 (known as the Rehnquist Court). The (upper-triangular portion of the) dissimilarity matrix is given in the same row and column order as the *Times* data set, with the justices ordered from "liberal" to "conservative":

1: John Paul Stevens (St)
2: Stephen G. Breyer (Br)
3: Ruth Bader Ginsberg (Gi)
4: David Souter (So)
5: Sandra Day O'Connor (Oc)
6: Anthony M. Kennedy (Ke)
7: William H. Rehnquist (Re)
8: Antonin Scalia (Sc)
9: Clarence Thomas (Th)

The lower-triangular portion of Table 1 is a best-fitting (least-squares) SAR matrix obtained with the MATLAB M-file `sarobfnd.m` mentioned in the next section. The variance-accounted-for is 98.62%, so there is little residual variability left. A graphical representation is given in Figure 1; the 'pyramidal' structure would be more apparent if the vertical lines were tilted slightly inward toward the internal nodes.

	St	Br	Gi	So	Oc	Ke	Re	Sc	Th
1 St	.00	.38	.34	.37	.67	.64	.75	.86	.85
2 Br	.36	.00	.28	.29	.45	.53	.57	.75	.76
3 Gi	.36	.28	.00	.22	.53	.51	.57	.72	.74
4 So	.37	.29	.22	.00	.45	.50	.56	.69	.71
5 Oc	.66	.49	.49	.45	.00	.33	.29	.46	.46
6 Ke	.70	.55	.55	.53	.31	.00	.23	.42	.41
7 Re	.70	.55	.55	.53	.31	.23	.00	.34	.32
8 Sc	.86	.74	.74	.70	.46	.42	.33	.00	.21
9 Th	.86	.74	.74	.70	.46	.42	.33	.21	.00

Table 1. Dissimilarities among the nine Supreme Court justices above the diagonal; best-fitting SAR values below the diagonal.

3 Computational procedures within MATLAB

The recent monograph by Hubert, Arabie, and Meulman (2006) provides a collection of open-source M-files (i.e., the code is freely available) within a MATLAB environment to effect a variety of least-squares structural representations for a proximity matrix. Among these are strategies to search for good-fitting AR and SAR forms, including additive decompositions of up to two such structures for a single given proximity matrix. We do not give the algorithmic details here on how these M-files are built, and instead, refer the reader to the Hubert et. al (2006) monograph. We have collected all the relevant M-files together at `http://cda.psych.uiuc.edu/diday_mfiles`. The three M-files, `arobfnd.m`, `biarobfnd.m`, `triarobfnd.m`, fit respectively, one, two, and three AR matrices to a given input proximity matrix; the three M-files, `sarobfnd.m`, `bisarobfnd.m`, `trisarobfnd.m`, are for the strengthened SAR forms. The two files, `triarobfnd.m` and `trisarobfnd.m`, are unique to this site, and should provide a programming template to extend easily, when needed, the additive decomposition to four or more matrices.

We give the help header for the representative file `triarobfnd.m` below, along with an application to a randomly constructed 10 × 10 proximity matrix (obtained from the contributed M-file `randprox.m`). As can be seen, the (random) matrix is perfectly reconstructed by the three AR matrices (a variance-accounted-for of 1.0 is achieved). For example, the (4,6) entry in `prox` of .7948 is reconstructed based on the given output permutations, `outpermone`, `outpermtwo`, and `outpermthree`; explicitly, we use the (4,10) entry in `targone` (.8290), the (8,9) entry in `targtwo` (−.0515), and the (3,9) entry in `targthree` (.0173): .7948 = .8290 + (−.0515) + (.0173).

```
>> help triarobfnd

  TRIAROBFND finds and fits the sum of three anti-Robinson
  matrices using iterative projection to a symmetric
  proximity matrix in the $L_{2}$-norm based on permutations
  identified through the use of iterative quadratic assignment.

  syntax: [find,vaf,targone,targtwo,targthree,outpermone, ...
      outpermtwo,outpermthree] = triarobfnd(prox,inperm,kblock)

  PROX is the input proximity matrix ($n \times n$ with a zero
  main diagonal and a dissimilarity interpretation);
  INPERM is a given starting permutation of the first $n$
  integers; FIND is the least-squares optimal matrix (with
  variance-accounted-for of VAF to PROX and is the sum of the
  three anti-Robinson matrices TARGONE, TARGTWO, and TARGTHREE
  based on the three row and column object orderings given by
  the ending permutations OUTPERMONE, OUTPERMTWO, and
  OUTPERMTHREE. KBLOCK defines the block size in the use of
  the iterative quadratic assignment routine.
```

```
>> prox = randprox(10)

prox =

         0    0.6979    0.3784    0.8600    0.8537    0.5936    0.4966    0.8998    0.8216    0.6449
    0.6979         0    0.8180    0.6602    0.3420    0.2897    0.3412    0.5341    0.7271    0.3093
    0.3784    0.8180         0    0.8385    0.5681    0.3704    0.7027    0.5466    0.4449    0.6946
    0.8600    0.6602    0.8385         0    0.6213    0.7948    0.9568    0.5226    0.8801    0.1730
    0.8537    0.3420    0.5681    0.6213         0    0.9797    0.2714    0.2523    0.8757    0.7373
    0.5936    0.2897    0.3704    0.7948    0.9797         0    0.1365    0.0118    0.8939    0.1991
    0.4966    0.3412    0.7027    0.9568    0.2714    0.1365         0    0.2987    0.6614    0.2844
    0.8998    0.5341    0.5466    0.5226    0.2523    0.0118    0.2987         0    0.4692    0.0648
    0.8216    0.7271    0.4449    0.8801    0.8757    0.8939    0.6614    0.4692         0    0.9883
    0.6449    0.3093    0.6946    0.1730    0.7373    0.1991    0.2844    0.0648    0.9883         0

>> [find,vaf,targone,targtwo,targthree, ...
   outpermone,outpermtwo,outpermthree] = ...
   triarobfnd(prox,randperm(10),2)

find =

         0    0.6979    0.3784    0.8600    0.8536    0.5936    0.4966    0.8998    0.8216    0.6449
    0.6979         0    0.8180    0.6602    0.3420    0.2897    0.3412    0.5341    0.7271    0.3093
    0.3784    0.8180         0    0.8385    0.5681    0.3704    0.7027    0.5466    0.4449    0.6946
    0.8600    0.6602    0.8385         0    0.6213    0.7948    0.9568    0.5226    0.8801    0.1730
    0.8536    0.3420    0.5681    0.6213         0    0.9797    0.2714    0.2523    0.8757    0.7373
    0.5936    0.2897    0.3704    0.7948    0.9797         0    0.1365    0.0118    0.8939    0.1991
    0.4966    0.3412    0.7027    0.9568    0.2714    0.1365         0    0.2987    0.6614    0.2844
    0.8998    0.5341    0.5466    0.5226    0.2523    0.0118    0.2987         0    0.4692    0.0648
    0.8216    0.7271    0.4449    0.8801    0.8757    0.8939    0.6614    0.4692         0    0.9883
    0.6449    0.3093    0.6946    0.1730    0.7373    0.1991    0.2844    0.0648    0.9883         0


vaf =

    1.0000


targone =

         0    0.6591    0.6591    0.6601    0.6601    0.7509    0.7754    0.7755    0.8757    0.8801
    0.6591         0    0.3569    0.5849    0.6601    0.7509    0.7509    0.7755    0.8290    0.8290
    0.6591    0.3569         0    0.3704    0.6601    0.6720    0.6851    0.7755    0.7840    0.8290
    0.6601    0.5849    0.3704         0    0.1030    0.2063    0.2661    0.3883    0.7840    0.8290
    0.6601    0.6601    0.6601    0.1030         0    0.2063    0.2418    0.3883    0.4269    0.8290
    0.7509    0.7509    0.6720    0.2063    0.2063         0    0.0283    0.3290    0.3290    0.6651
    0.7754    0.7509    0.6851    0.2661    0.2418    0.0283         0    0.2702    0.3290    0.5290
    0.7755    0.7755    0.7755    0.3883    0.3883    0.3290    0.2702         0    0.2963    0.5263
    0.8757    0.8290    0.7840    0.7840    0.4269    0.3290    0.3290    0.2963         0    0.5263
    0.8801    0.8290    0.8290    0.8290    0.8290    0.6651    0.5290    0.5263    0.5263         0


targtwo =

         0   -0.1489    0.0312    0.0312    0.0312    0.0492    0.0578    0.1813    0.2296    0.4148
   -0.1489         0   -0.1392   -0.0471   -0.0333    0.0492    0.0578    0.0578    0.1344    0.1344
    0.0312   -0.1392         0   -0.0537   -0.0333    0.0281    0.0376    0.0376    0.0376    0.0620
    0.0312   -0.0471   -0.0537         0   -0.2446    0.0281    0.0376    0.0376    0.0376    0.0620
    0.0312   -0.0333   -0.0333   -0.2446         0   -0.2488   -0.1600    0.0376    0.0376    0.0620
    0.0492    0.0492    0.0281    0.0281   -0.2488         0   -0.1600   -0.0080    0.0160    0.0160
    0.0578    0.0578    0.0376    0.0376   -0.1600   -0.1600         0   -0.3058   -0.0080         0
    0.1813    0.0578    0.0376    0.0376    0.0376   -0.0080   -0.3058         0   -0.0515   -0.0426
    0.2296    0.1344    0.0376    0.0376    0.0376    0.0160   -0.0080   -0.0515         0   -0.3495
    0.4148    0.1344    0.0620    0.0620    0.0620    0.0160         0   -0.0426   -0.3495         0


targthree =

         0   -0.1217   -0.0376   -0.0312    0.0346    0.0346    0.1510    0.1958    0.1962    0.1962
   -0.1217         0   -0.1345   -0.1345    0.0346    0.0346    0.0364    0.1113    0.1113    0.1675
   -0.0376   -0.1345         0   -0.1345   -0.0065   -0.0065   -0.0065   -0.0065    0.0173    0.0964
   -0.0312   -0.1345   -0.1345         0   -0.2651   -0.0065   -0.0065   -0.0065    0.0145    0.0145
    0.0346    0.0346   -0.0065   -0.2651         0   -0.0065   -0.0065   -0.0065    0.0080    0.0145
    0.0346    0.0346   -0.0065   -0.0065   -0.0065         0   -0.0917   -0.0243   -0.0243         0
    0.1510    0.0364   -0.0065    0.0065   -0.0065   -0.0917         0   -0.1680   -0.0243   -0.0229
    0.1958    0.1113   -0.0065   -0.0065   -0.0065   -0.0243   -0.1680         0    0.0289   -0.0239
    0.1962    0.1113    0.0173    0.0145    0.0080   -0.0243   -0.0243   -0.0289         0   -0.1362
    0.1962    0.1675    0.0964    0.0145    0.0145         0   -0.0229   -0.0239   -0.1362         0


outpermone =

     9     1     3     6     7     8    10     2     5     4


outpermtwo =

     5     7     1     2     9     3     8     6     4    10


outpermthree =

     9     8     4     5     3     7    10     1     6     2
```

4 The concept of minimum AR (or SAR) matrix rank

Based on the type of M-file (`triarobfnd.m`) illustrated in the previous section, a rather natural question arises as to the number of AR (or SAR) components necessary to exhaust perfectly any given proximity matrix. The minimum such number will be referred to as the AR (or SAR) rank of a symmetric proximity matrix. As we saw for the random 10 × 10 matrix in the example of the last section, we usually can do quite well with many fewer components than the order of the matrix. Although we might expect this to be true for a data matrix that is well-structured (and where two or three AR or SAR components are all that is needed to effectively exhaust the given proximity matrix), the same also appears to hold for merely randomly structured matrices.

To make this last point even more clear, a small Monte Carlo analysis was carried out in which 1000 random proximity matrices (with entries uniform on (0,1)), of sizes 10, 20, 30, 40, and 50, were approximated by sums of AR matrices to the point where at least a VAF of 99% was achieved. The frequency results (out of 1000 such randomly generated matrices) are tabulated below:

	Number AR Components Needed								
Matrix Size	2	3	4	5	6	7	8	9	10
10	37	959	4						
20			316	684					
30					994	6			
40						205	795		
50								995	5

Figure 2 illustrates, by means of box-and-whisker plots, the incremental gain in VAF as a function of the number of fitted AR components.

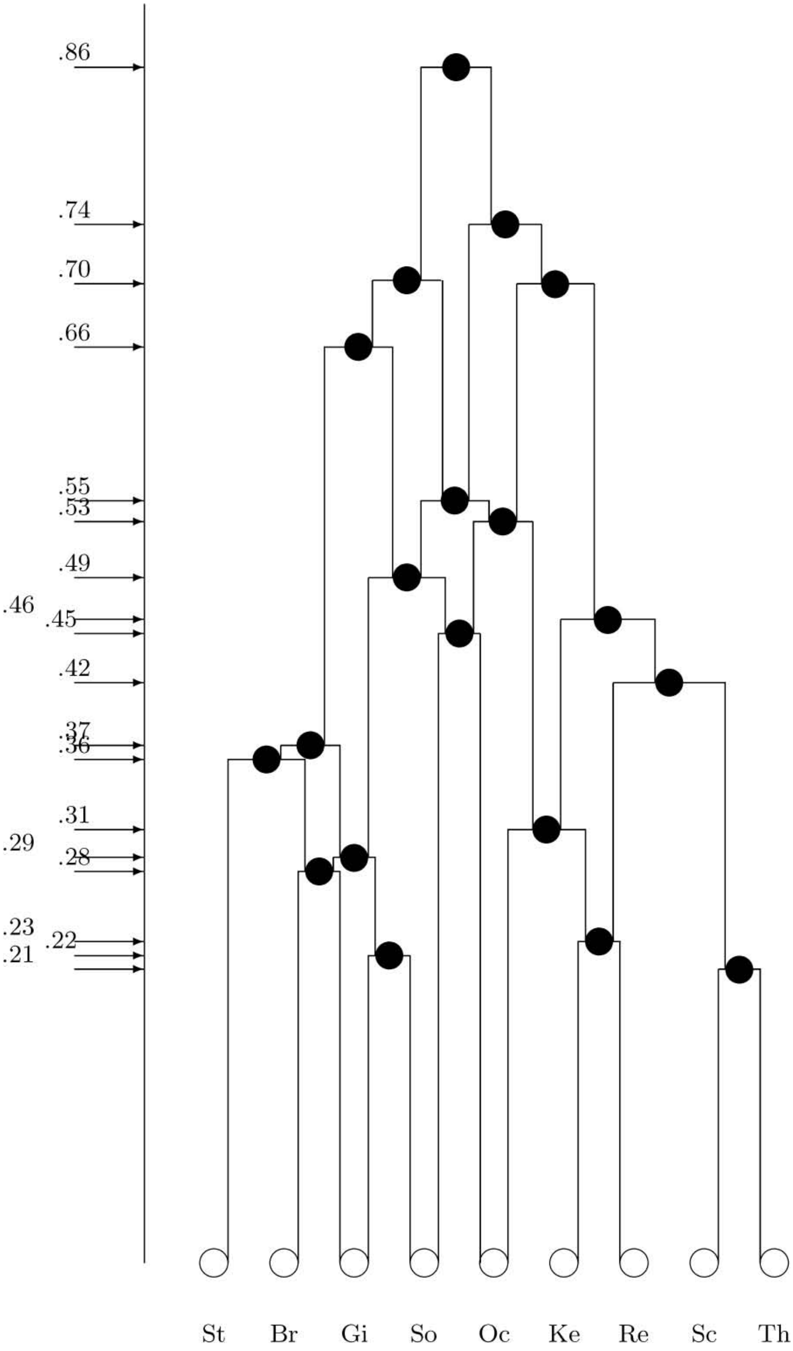


Fig. 1. A 'pyramidal' representation for the SAR matrix given in Table 1 having VAF of 98.62%.

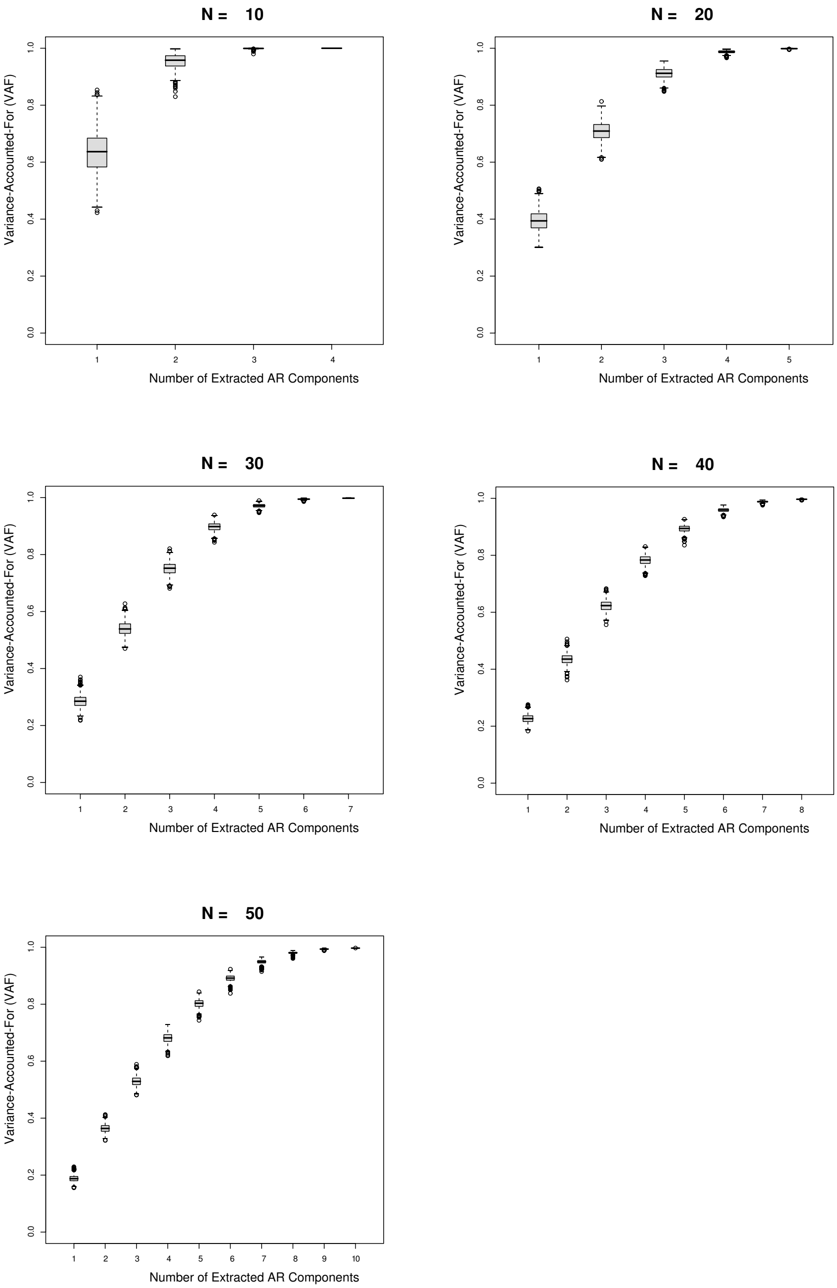


Fig. 2. Incremental VAF Gains for Differing Numbers of AR Components.

References

BARTHÉLEMY, J.-P., BRUCKER, F. and OSSWALD, C. (2004): Combinatorial optimization and hierarchical classifications. *4OR: A Quarterly Journal of Operations Research 2 (3), 179–219.*

CRITCHLEY, R. (1994): On exchangeability-based equivalence relations induced by strongly Robinson and, in particular, by quadripolar Robinson dissimilarity matrices. In: B. van Cutsem (Ed.): *Classification and Dissimilarity Analysis.* Springer-Verlag, New York, 173–199.

CRITCHLEY, R. and FICHET, B. (1994): The partial order by inclusion of the principal classes of dissimilarity on a finite set, and some of their basic properties. In: B. van Cutsem (Ed.): *Classification and Dissimilarity Analysis.* Springer-Verlag, New York, 5–65.

DIATTA, J. and FICHET, B. (1998): Quasi-ultrametrics and their 2-ball hypergraphs. *Discrete Mathematics 192 (1-3), 87–102.*

DIDAY, E. (1986): Orders and overlapping clusters by pyramids. In: J. De Leeuw, W. Heiser, J. Meulman and F. Critchley (Eds.): *Multidimensional Data Analysis.* DSWO Press, Leiden, 201–234.

DURAND, C. and FICHET, B. (1988): One-to-one correspondences in pyramidal representations: A unified approach. In: H.-H. Bock (Ed.): *Classification and Related Methods of Data Analysis.* North-Holland, Amsterdam, 85–90.

HUBERT, L., ARABIE, P. and MEULMAN, J. (2006): *The Structural Representation of Proximity Matrices with MATLAB.* SIAM, Philadelphia.

MIRKIN, B. (1996): *Mathematical Classification and Clustering.* Kluwer, Dordrecht.

ROBINSON, W.S. (1951): A method for chronologically ordering archaeological deposits. *American Antiquity 19 (4), 293–301.*

Density-Based Distances: a New Approach for Evaluating Proximities Between Objects. Applications in Clustering and Discriminant Analysis

Jean-Paul Rasson and François Roland

Statistical Unit, University of Namur,
8 Rempart de la Vierge, B-5000 Namur, Belgium, *jean-paul.rasson@fundp.ac.be*

Abstract. The aim of this paper is twofold. First it is shown that taking densities between objects into account to define proximities between them is intuitively a right way to process. Secondly, some new distances based on density estimates are defined and some properties are presented. Many algorithms in clustering or discriminant analysis require the choice of a dissimilarity: two applications are presented, one in clustering and the other in discriminant analysis, and illustrate the benefits of using these new distances.

1 Introduction

In many statistical studies, a common first step consists in determining how two objects are *close*. This is the case for several cluster analysis techniques or for nearest-neighbor classification. Similarity and dissimilarity coefficients are numbers that measure and summarize the proximity between two objects described by the same set of variables. In some situations, the set of similarity or dissimilarity coefficients between the objects under investigation is available (such an example is given in Kruskal and Wish (1978), page 30) and can be use. But most of the time, the objects are described only by a pattern matrix (or profile matrix) and the similarity or dissimilarity coefficients have to be computed. Many different ways have been proposed: see for example Gower and Legendre (1986) for a list and discussion of their properties. But it appears that none of them takes into account the density functions from which objects are issued. In this paper, we propose to study density estimation contributions in defining dissimilarity coefficients.

2 Similarity, dissimilarity and distance

Let us fix some notations that will be used throughout the paper. The n objects under investigation are denoted by $\mathbf{x}_1, \cdots, \mathbf{x}_n$; they are described by p variables $Y_1, \cdots, Y_p$ such that $x_{ij} = Y_j(\mathbf{x}_i)$ is the value observed for the variable Y_j on the object $\mathbf{x}_i$. The set of all possible values for the variable Y_j

(called the observation domain) is denoted by $\mathcal{Y}_j$ ($j \in \{1, \cdots, p\}$). With these notations, let us define the concepts of similarity, dissimilarity and distance.

If $E = \{\mathbf{x}_1, \cdots, \mathbf{x}_n\}$, a similarity coefficient (or simply a similarity) on the set E is an application s from $E \times E$ into $I\!R^+$ satisfying:

P1 $\forall(\mathbf{x}_i, \mathbf{x}_j) \in E \times E,\ s(\mathbf{x}_i, \mathbf{x}_j) = s(\mathbf{x}_j, \mathbf{x}_i);$
P2 $\forall(\mathbf{x}_i, \mathbf{x}_j) \in E \times E,\ \mathbf{x}_i \neq \mathbf{x}_j) :\ s(\mathbf{x}_i, \mathbf{x}_i) = s(\mathbf{x}_j, \mathbf{x}_j) > s(\mathbf{x}_i, \mathbf{x}_j).$

A dissimilarity coefficient (or simply a dissimilarity) on the set E is an application d from $E \times E$ into $I\!R^+$ satisfying **P1** and the following condition:

P2' $\forall \mathbf{x}_i \in E,\ d(\mathbf{x}_i, \mathbf{x}_i) = 0.$

A metric dissimilarity (or simply a distance) on the set E is a dissimilarity such that the triangle inegality holds:

P3 $\forall \mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k \in E,\ d(\mathbf{x}_i, \mathbf{x}_j) \leq d(\mathbf{x}_i, \mathbf{x}_k) + d(\mathbf{x}_k, \mathbf{x}_j).$

When the p variables are quantitative (i.e. $\mathcal{Y}_j \subseteq I\!R$, $j \in \{1, \cdots, p\}$), Minkowski metrics offer a convenient way for measuring proximities:

$$d(\mathbf{x}_i, \mathbf{x}_j) = \left(\sum_{k=1}^{p} |x_{ik} - x_{jk}|^r \right)^{\frac{1}{r}}. \tag{1}$$

When $r = 2$, the Equation (1) reduces to Euclidean distance that is certainly the most popular choice. Other common alternatives are the *City-block* distance (obtained when $r = 1$ in Equation (1)), the Chebychev distance:

$$d(\mathbf{x}_i, \mathbf{x}_j) = \max_{1 \leq k \leq p} |x_{ik} - x_{jk}|. \tag{2}$$

or the Mahalanobis distance[1] that takes account of the variance-covariance matrix S:

$$d(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)' S^{-1} (\mathbf{x}_i - \mathbf{x}_j). \tag{3}$$

As written in the introduction, there exists many different measures of dissimilarity. Most of them work as follow: they evaluate the difference d_{ijk} between objets $\mathbf{x}_i$ and $\mathbf{x}_j$ along the k^{th} variable ($k \in \{1, \cdots, p\}$), and combine these differences to obtain the dissimilarity. To our knowledge, no dissimilarity measure use the distribution of the values $x_{1k}, \cdots, x_{nk}$ to compute the d_{ijk} ($k \in \{1, \cdots, p\}$) but we think that it could solve many problems. In the next section, we present one situation amongst others that argue in that way.

[1] We should prefer the term statistic because Equation (3) does not satisfy the triangle inequality.

3 Motivations

The problem is illustrated in Figure 1. Obviously, there are two "clouds" of objects: a dense one on the left and a sparse one on the right. What is less obvious is the answer to the question: does the object **x** belong to the lefthandside or to the righthandside group ? If the Euclidean distance and a k nearest neighbor approach are used to answer this question, the object **x** will be assigned to the lefthandside group. However, it seems more intuitive to assign **x** to the righthandside group because it is sparse whereas the left one is dense.

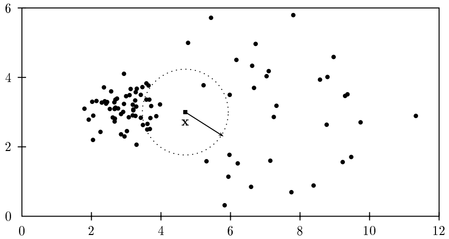


Fig. 1. Necessity to use the density estimates to assign an object.

To solve the problem, the distance between two objects must take into account not only their positions but also the local density where they are located: if they belongs to the same high density region of the space, the distance between them should be smaller than if it exist a low density region between them. Using the eyes, one of the best *classifier* in dimension two, a *hole* in a high density appears more important than a *hole* of the same size in a sparse density group. A similar argument was used by Wong and Lane (1983) to develop a hierchical clustering algorithm referred in litterature by the terms *density linkage clustering.* Let us also note that Hartigan (1975) uses the notion of density to define the term *natural cluster.* Because most of the time the densities are unknown, they have to be estimated: this is the subject of the next section.

4 Density estimation by the kernel method: the univariate case

To estimate the p univariate densities, we have decided to use a non parametric method. We have chosen the kernel method because it is very popular and well studied (Silverman (1986), Scott (1992)). But others methods are imaginable such as histograms or wavelets.

If X is a real random variable whose density function is f, given n realizations $x_1, \cdots, x_n$ drawn from f, the kernel estimator $\hat{f}$ is defined $\forall x \in I\!R$

by

$$\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{h} K(\frac{x - x_i}{h}). \tag{4}$$

The function $K(\cdot)$ is a *kernel function*. It means that $K(\cdot)$ is a continuous, positive and symetric function and satisfies $\int_{-\infty}^{+\infty} K(t)dt = 1$. Common choices for $K(\cdot)$ are given in the Figure 2.

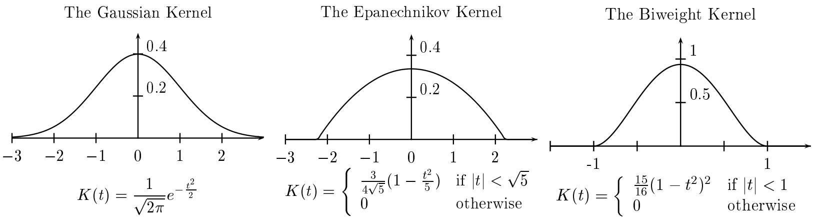


Fig. 2. Expression and shape of three different popular kernels.

The parameter h is called the *bandwidth* or the *smoothing parameter*. The kernel estimator corresponds to a sum of *bumps* centered on the different values $x_1, \cdots, x_n$ (the lefthand side graphic in Figure 3); the shape of the bumps is determined by the kernel function $K(\cdot)$ and their width by the smoothing parameter. The correct estimation of the value of h is crucial: if it is too small, the estimator is unstable and presents modes that are inexistent in reality; if it is too large, the main features of the density (such as bimodality) are obscured. The three righthand side graphics in Figure 3 illustrate that behaviour. The optimal general bandwith has still not been found. For a detailed discussion on the estimation of the smoothing parameter h, see for example Silverman (1986), pages 43-61.

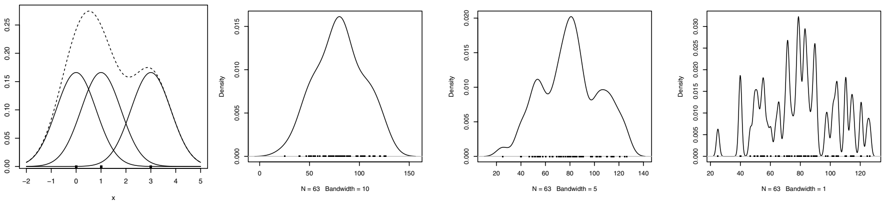


Fig. 3. Lefthandside: interpretation of the kernel estimator as a sum of bumps centred on the observations. Righthandside: importance of a correct choice for the value of the smoothing parameter.

5 Density-based distances

Given a set E of n objects $\mathbf{x}_1, \cdots, \mathbf{x}_n$ described by p quantitative variables $Y_1, \cdots, Y_p$, we propose the two new following measures of dissimilarity:

$$d(\mathbf{x}_i, \mathbf{x}_i) = \left(\sum_{k=1}^{p} \left| \int_{x_{ik}}^{x_{jk}} \hat{f}_{Y_k}(t)dt \right|^r \right)^{\frac{1}{r}} \tag{5}$$

and

$$d(\mathbf{x}_i, \mathbf{x}_i) = \max_{1 \leq k \leq p} \left| \int_{x_{ik}}^{x_{jk}} \hat{f}_{Y_k}(t)dt| \right|. \tag{6}$$

The functions $\hat{f}_{Y_k}(\cdot)$ ($k \in \{1, \cdots, p\}$) are the univariate kernel estimators along each variable Y_k. The kernel function is the Gaussian kernel but other choices are possible and do not change much the results. For each estimator $\hat{f}_{Y_k}(\cdot)$, the value of its smoothing parameter h_k is given by:

$$h_k = 1.06 \min(s_k, R_k/1.34) n^{-0.2} \tag{7}$$

where s_k is the empirical variance and R_k the interquartil range of the values $x_{1k}, \cdots, x_{nk}$. See Silverman (1986) page 47 for a justification. When the number of objects n is large, the integrals in Equations (5) and (6) can be replaced by

$$\hat{F}_{Y_k}(x_{ik}) - \hat{F}_{Y_k}(x_{jk}). \tag{8}$$

$\hat{F}_{Y_k}(\cdot)$ is the empirical cumulative distribution function defined as follow:

$$\hat{F}_{Y_k}(x) = \frac{1}{n} \sum_{i=1}^{n} 1\!\!I_{]-\infty, x]}(x_{ik}) \tag{9}$$

where $1\!\!I_D(\cdot)$ denotes the indicatrice function.

It is easy to demonstrate that the properties **P1** and **P2'** are satisfied. The demonstration that the triangular inequality **P3** also holds, and thus that these new dissimilarities are distances, is nearly immediate if we notice that for all $k \in \{1, \cdots, p\}$ and for all $\mathbf{x}_i$, $\mathbf{x}_j$, $\mathbf{x}_l$:

$$\left| \int_{x_{ik}}^{x_{jk}} \hat{f}_{Y_k}(t)dt \right| \leq \left| \int_{x_{ik}}^{x_{lk}} \hat{f}_{Y_k}(t)dt \right| + \left| \int_{x_{lk}}^{x_{jk}} \hat{f}_{Y_k}(t)dt \right|. \tag{10}$$

Theses distances can be related to a statistical model based on non homogeneous Poisson processes, and developed by Rasson and Granville (1995, 1996) in clustering. If the n objects $\mathbf{x}_1, \cdots, \mathbf{x}_n$ to be clustered are a realization of a non homogeneous Poisson process with intensity[2] $f(\cdot)$ on a convex domain $D \subset I\!\!R^p$ that is the union of g unknown convex disjoint domains

[2] If the intensity $f(\cdot)$ is unknown, it is estimated by the kernel method.

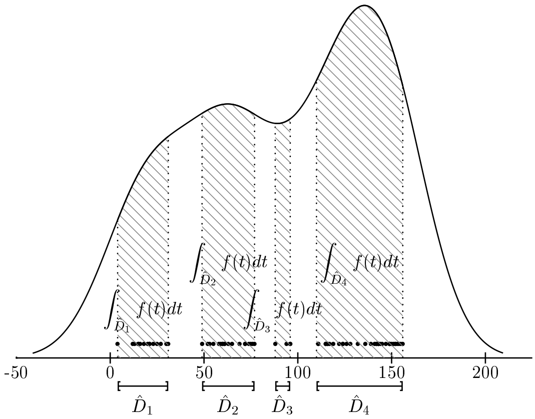


Fig. 4. The hypervolums criterion when $p = 1$. The number of clusters g is fixed to be four.

$D_1, \cdots, D_g$ $(D = \cup_{l=1}^{g} D_l)$, the maximum likelihood solution indicates to find a partition into g clusters $C_1 \cdots, C_g$ such that

$$\sum_{l=1}^{g} \int_{H(C_l)} f(\mathbf{t})d\mathbf{t} \tag{11}$$

is minimal, where $H(C_l)$ denotes the convex hull of the objects belonging to the cluster C_l $(l \in \{1, \cdots, g\})$. This criterion is known as the *generalized hypervolums criterion.* If p equals one, the criterion reduces to the determination of g disjoints intervals $\hat{D}_1, \cdots, \hat{D}_g$ containing all objects such that

$$\sum_{l=1}^{g} \int_{\hat{D}_l} f(t)dt \tag{12}$$

is minimal (Figure 4).

When p equals one, the distance defined in Equations (5) and (6) between x_i and x_j reduce to

$$d(x_i, x_j) = \left| \int_{x_i}^{x_j} \hat{f}(t)dt \right| \tag{13}$$

and the intervals $\hat{D}'_1, \cdots, \hat{D}'_g$ obtained by minimising

$$\sum_{l=1}^{g} \sum_{x_i, x_j \in \hat{D}'_l} d(x_i, x_j) \tag{14}$$

are the same than the intervals $\hat{D}_1, \cdots, \hat{D}_g$ obtained from Equation (12). Similar results have still to be found when p is greater than two and all our attention is devoted to this question.

6 Applications

The dataset under study in the first application consists of protein consumption measurements in twenty-five European country for nine food groups (Weber and Weber (1974)). If the usual Euclidean distance is used to compute the dissimilarity matrix, no hierarchical clustering algorithm provides easily interpretable results. For example, the Ward method provides the dendrogram on the lefthandside in Figure 5. But, if the distance given either by Equation (5) ($r = 2$) or by Equation (6) is used, the dendrograms provided by the Ward method (respectively center and righthandside of Figure 5) reflects very well the geographical and political situation of these twenty-five countries at that time. The main difference between these two dendrograms is the level where Eastern Europe is merged with Western Europe.

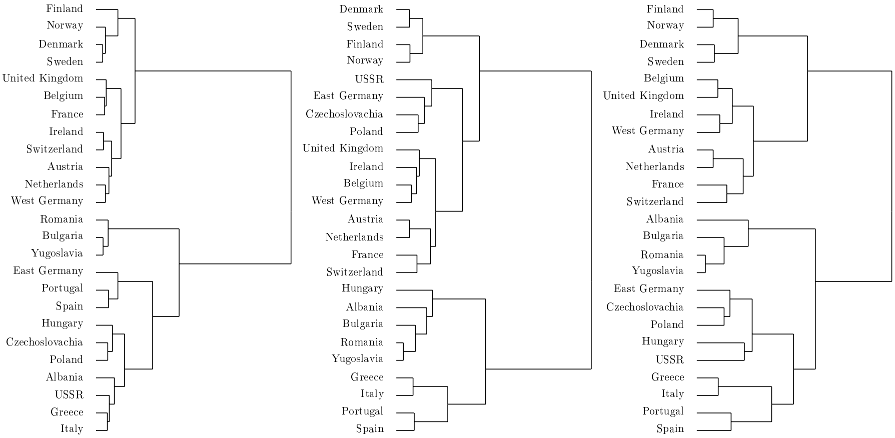


Fig. 5. Dendrogram obtained by the Ward method. Lefthandside: the Euclidean distance. Center: the distance defined in Equation (5), $r = 2$. Righthandside: the distance defined in Equation (6).

For example, in the righthandside dendrogram in Figure 5, the groups, from top to bottom, are: Scandinavia (Finland, Norway, Denmark and Sweden), Eastern Europe (Belgium, United Kingdom, Ireland, West Germany, Austria, Netherlands, France and Switzerland), the Balkans (Albania, Romania, Bulgaria and Yugoslavia), Eastern Europe (East Germany, Czechoslovakia, Poland, Hungary and USSR), the Mediterranean (Greece and Italy) and finally the Iberians (Spain and Portugal). A two-dimensional representation of the twenty-five coutries is given in Figure 6 by performing ordinal multidimensional scaling on the dissimilarity matrix computed from the distance defined in Equation (6).

The second application relates to early enterprises bankruptcy detection. The dataset contains 2727 Belgian enterprises that have sent their statement

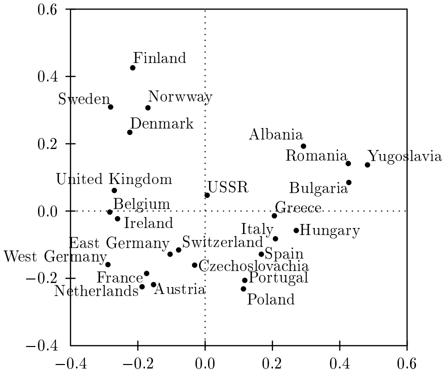


Fig. 6. A two-dimensional representation of the twenty-five country using ordinal multidimensional scaling on the dissimilarity matrix computed from the distance defined in Equation (6).

of account to the SPF Finance (*Service Public Fédéral of Finance*) in 1997. Each enterprise is described by 28 financial ratios (quantitative variables) and by a binary variable: it equals 1 if the enterprise was declared failed during the next three year (from 1997 to 2000) and equals 0 otherwise. The objective is to find a discriminant rule to distinguish the failed enterprises from the healthy ones. The major problem is that among the 2727 enterprises, only a small number (175 enterprises) went to bankruptcy (the *risk group*): a discriminant rule declaring all enterprises as healthy presents a good classifications rate of 93.58% but no loaner will accept it. The discriminant rule has to maximize the good classifications rates in the risk group while offering a acceptable good classifications rate in the *normal group.* Using a k nearest neighbor approach with our density-based distances encounters that requirement whereas the use of the Euclidean distance fails completely. The Fisher linear discriminant analysis performs better but the good classifications rate for the risk group is still 10% lower than ours. All results are given in Table 1.

The last application is still in clustering but the dataset in this case consists of eight objects described by four interval variables (Bock and Diday (2000)): the well-known Ichino oils dataset (Ichino and Yagushi (1994)). From the viewpoint of chemists, it is known that linseed and perilla oils are used for paint, cottonseed, sesame, camellia and olive oils are used for foods and cosmetics and endly beef-tallow and hog fat are fats. To be able to handle with interval variables, our density-based distances have to be adapted: in the Equations (5) and (6), the integrals $\left| \int_{x_{ik}}^{x_{jk}} f_{Y_k}(t)dt \right|$ expressing the difference between objects $\mathbf{x}_i$ and $\mathbf{x}_j$ along the kth variable are replaced by

$$\left(\left| \int_{m_{ik}}^{m_{jk}} f_{M_k}(t)dt \right|^2 + \left| \int_{l_{ik}}^{l_{jk}} f_{L_k}(t)dt \right|^2 \right)^{\frac{1}{2}} \qquad (15)$$

k **nearest neighbor** ($k = 5$) **Distance defined in Equation (5)**		
	Declared as	
	Healthy	Bankruptcy
Healthy	1769	783
Bankruptcy	36	139
Bankruptcy: 79.42% Healthy: 69.31%		

k **nearest neighbor** ($k = 5$) **Distance defined in Equation (6)**		
	Declared as	
	Healthy	Bankruptcy
Healthy	1724	828
Bankruptcy	30	145
Bankruptcy: 82.85% Healty: 67.59%		

k **nearest neighbor** ($k = 5$) **Euclidean distance**		
	Declared as	
	Healthy	Bankruptcy
Healthy	2119	433
Bankruptcy	96	79
Bankruptcy: 45.14% Healthy: 83.03%		

Fisher **linear discriminant analysis**		
	Declared as	
	Healthy	Bankruptcy
Healthy	1852	700
Bankruptcy	51	124
Bankruptcy: 70.86% Healty: 72.57%		

Table 1. Results for the k nearest neighbor algorithm using the density-based distances defined in Equations (5) and (6), the Euclidean distance and for the Fisher linear discriminant analysis on the financial dataset. Good classification rates are estimated by cross-validation.

Each interval variable Y_k with $Y_k(\mathbf{x_i}) = [a_{ik}, b_{ik}]$ is transformed into two classical variables M_k and L_k such that $M_k(\mathbf{x}_i) = (a_{ik} + b_{ik})/2 = m_{ik}$ and $L_k(\mathbf{x}_i) = (b_{ik} - a_{ik})/2 = l_{ik}$; M_k corresponds to the midpoint of the interval and L_k to its the half-length. A other alternative sould have been to use some kind of density-based Hausdorff distance (De Carvalho et al. (2006)). Computing dissimilarities between oils thanks to the symbolic version of Equation (6) and performing a complete linkage algorithm provides the dendrogram shown in Figure 7. A two-dimensional representation of the objects obtained by ordinal multidimensional scaling is also given in Figure 7. These results are very interesting. For example a three clusters partition corresponds to the chemical classification.

7 Discussion

With a two-dimensional example, we have motivated the fact that densities must be taken into account to define proximity between objects. We propose some new distances based on density estimates. We use the non parametric kernel method to estimate the densities. It means that the new distances are appropriate no matter of the distribution of the objects under study. The theoretical properties of these new distances have still to be studied more deeply but in practice, we can already affirm that they perform better than the Euclidean distance in many situations.

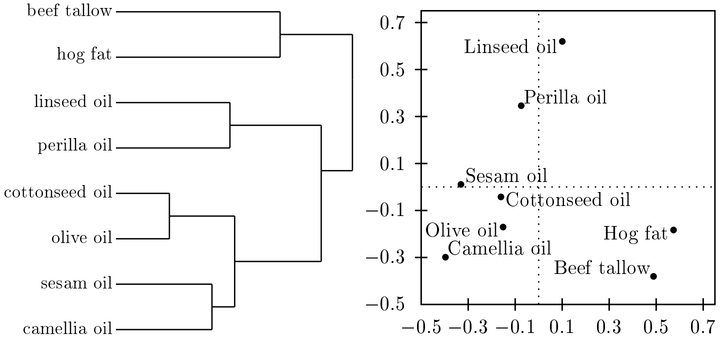


Fig. 7. Left: dendrogram obtained with the complete linkage algorithm and the symbolic version of the dissimilarity defined in Equation (6). Right: two-dimensional representation of the eight oils by ordinal multidimensional scaling.

References

BOCK, H.-H., and DIDAY, E. (2000): *Analysis of Symbolic Data. Exploratory methods for extracting statistical information from complex data.* Springer Verlag, Heidelberg.

DE CARVALHO, F. de A.T., and SOUZA, R.M.C.R., and CHAVENT, M. and LECHEVALLIER, Y. (2006): Adaptative Hausdorff distances and dynamic clustering of symbolic interval data. *Pattern Recognition Letters 27(3) , 167-179.*

GOWER, J.C., and LEGENDRE, P. (1986): Metric and Euclidean properties of dissimilarity coefficients. *Journal of Classification 3 , 5-48.*

HARTIGAN, J.A. (1975): *Clustering Algorithms.* John Wiley & Sons.

ICHINO, M., and YAGUSHI, H. (1994): Generalize Minkowski metrics for mixed feature-type data analysis. *IEEE Transactions on System, Man, and Cybernetics 24 (4), 698-708.*

KRUSKAL, J.B., and WISH, M. (1978): *Multidimensional Scaling.* London: Sage University Paper Series in Quantitative Applications in th Social Sciences.

RASSON, J.P., and GRANVILLE, V. (1995): Multivariate discriminant analysis and maximum penalized likelihood density estimation. *Journal of the Royal Statistical Society B(57), 501-517.*

RASSON, J.P., and GRANVILLE, V. (1996): Geometrical tools in classification. *Computational Statistics and Data Analysis 23, 105-123.*

SCOTT, D.W. (1992): *Multivariate Density Estimation. Theory, Practice, and Visualization.* John Wiley & Sons.

SILVERMAN, B.W. (1986): *Density Estimation for Statistics and Data Analysis.* Chapman & Hall.

WEBER, A., and WEBER, E. (1974): The structure of world protein consumption and future nitrogen requirement. *European Review of Agricultural Economics, 2 (2), 169-192.*

WONG, M.A., and LANE, A. (1983): A *k*th nearest neighbor clustering procedure. *Journal of the Royal Statistical Society 45 , 362-368.*

Robinson Cubes

Matthijs J. Warrens and Willem J. Heiser

Psychometrics and Research Methodology Group, Leiden University Institute for Psychological Research, Leiden University, Wassenaarseweg 52, P.O. Box 9555, 2300 RB Leiden, The Netherlands
Warrens@fsw.leidenuniv.nl; Heiser@fsw.leidenuniv.nl

Abstract. A square similarity matrix is called a Robinson matrix if the highest entries within each row and column are on the main diagonal and if, when moving away from this diagonal, the entries never increase. This paper formulates Robinson cubes as three-way generalizations of Robinson matrices. The first definition involves only those entries that are in a row, column or tube with an entry of the main diagonal. A stronger definition, called a regular Robinson cube, involves all entries. Several examples of the definitions are presented.

1 Introduction

Let $\mathbf{A} = \{a_{ij}\}$ be a $m \times m$ matrix. A similarity matrix $\mathbf{A}$ is called a *Robinson matrix* if the highest entries within each row and column of $\mathbf{A}$ are on the main diagonal (elements a_{ii}) and if the entries never increase when moving away from the diagonal. If $\mathbf{A}$ is a dissimilarity matrix, then $\mathbf{A}$ is called a Robinson matrix if the lowest entries are on the main diagonal and if the entries never increase when moving away from this diagonal in any direction (in this case Hubert et al. (1998) speak of an anti-Robinson matrix). Since an object i has usually 0 dissimilarity with itself, this main diagonal consists of 0s in the latter case. If the $\mathbf{A}$ is symmetric, that is, $a_{ij} = a_{ji}$, then $\mathbf{A}$ is a Robinson matrix if we have

$$1 \leq i < j \leq m \Rightarrow a_{ij} \leq a_{i+1j} \quad \text{and} \quad 1 \leq j \leq i < m \Rightarrow a_{ij} \geq a_{i+1j}$$

for similarities, and

$$1 \leq i < j \leq m \Rightarrow a_{ij} \geq a_{i+1j} \quad \text{and} \quad 1 \leq j \leq i < m \Rightarrow a_{ij} \leq a_{i+1j}$$

for dissimilarities.

The Robinson property of a (dis)similarity matrix reflects an ordering of the objects, but also constitutes a clustering system with overlapping clusters. Such ordered clustering systems were introduced under the name *pyramids* by Diday (1984, 1986) and under the name *pseudo-hierarchies* by Fichet (1984, 1986). The CAP algorithm to find an ordered clustering structure was described in Diday (1986) and Diday and Bertrand (1986), and later extended to deal with symbolic data by Brito (1991) and with missing data by Gaul

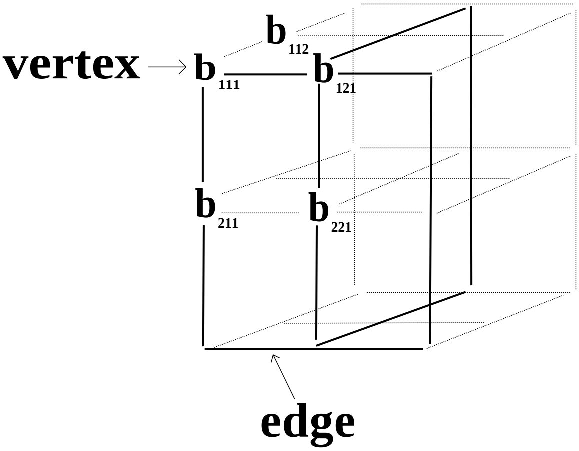


Fig. 1. Some aspects of a cube.

and Schader (1994). Chepoi and Fichet (1997) describe several circumstances in which Robinson matrices are encountered. For an in-depth review of overlapping clustering systems we refer to Barthélemy et al. (2004).

Let $\mathbf{B} = \{b_{ijk}\}$ be a $m \times m \times m$ array. In the present paper the concept of a Robinson matrix is extended to a three-way (dis)similarity cube $\mathbf{B}$, which will be called a *Robinson cube*. Whereas a matrix is characterized by rows and columns, a cube consists of rows, columns and *tubes*. The eight elements of $\mathbf{B}$ characterized by

$$b_{ijk} \quad \text{for } i, j, k = 1 \text{ or } m$$

are called the *vertices* of the cube. The twelve rows, columns and tubes containing two vertices are called the *edges* of $\mathbf{B}$. Some aspects of a cube are demonstrated in Figure 1.

The remainder of the paper looks as follows. Several definitions and some properties of a Robinson cube are presented in the next section. Various examples are presented in Section 3. Section 4 contains the discussion.

2 Definitions and properties

Before defining a Robinson cube we turn our attention to two natural requirements for cubes. Similar to a matrix $\mathbf{A}$ a cube $\mathbf{B}$ may satisfy *three-way symmetry*:

$$b_{ijk} = b_{ikj} = b_{jik} = b_{jki} = b_{kij} = b_{kji} \quad \text{for all } i,\ j \text{ and } k.$$

Another natural requirement for a cube **B** is the restriction

$$b_{iji} = b_{ijj} \quad \text{for all } i \text{ and } j.$$

The latter requirement is called *diagonal-plane equality* (Heiser and Bennani, 1997, p. 191) because it requires equality of the three matrices $\{b_{iij}\}$, $\{b_{iji}\}$ and $\{b_{ijj}\}$, which are formed by cutting the cube diagonally, starting at one of the three edges joining at the vertex b_{111}. A weak extension of the Robinson matrix is the following definition.

Definition 1. A (dis)similarity cube **B** is called a *Robinson cube* if the highest (lowest) entries within each row, column and tube of **B** are on the main diagonal (elements b_{iii}) and moving away from this diagonal, the entries never increase (decrease).

From Definition 1 it follows that a similarity cube **B** is a Robinson cube if we have

$$1 \le i < j \le m \Rightarrow \begin{cases} b_{ijj} \le b_{i+1jj} \\ b_{jij} \le b_{ji+1j} \\ b_{jji} \le b_{jji+1} \end{cases} \quad \text{and} \quad 1 \le j \le i < m \Rightarrow \begin{cases} b_{ijj} \ge b_{i+1jj} \\ b_{jij} \ge b_{ji+1j} \\ b_{jji} \ge b_{jji+1}. \end{cases}$$

The inequalities for a dissimilarity cube are obtained by interchanging $\le$ and $\ge$ in the right parts of both equations. If the similarity cube **B** satisfies the diagonal-plane equality, then **B** is a Robinson cube if we have

$$1 \le i < j \le m \Rightarrow \begin{cases} b_{ijj} \le b_{i+1jj} \\ b_{jij} \le b_{ji+1j} \end{cases} \quad \text{and} \quad 1 \le j \le i < m \Rightarrow \begin{cases} b_{ijj} \ge b_{i+1jj} \\ b_{jij} \ge b_{ji+1j}. \end{cases}$$

Moreover, if the similarity cube **B** satisfies three-way symmetry, then **B** is a Robinson cube if we have

$$1 \le i < j \le m \Rightarrow b_{ijj} \le b_{i+1jj} \quad \text{and} \quad 1 \le j \le i < m \Rightarrow b_{ijj} \ge b_{i+1jj}.$$

Note that, although this is perhaps suggested in the above argument, a Robinson cube that satisfies three-way symmetry does not necessarily satisfy the diagonal-plane equality.

Note that not all entries of **B** are involved in Definition 1. More precisely, only those entries that are in a row, column or tube with an entry of the main diagonal are involved in Definition 1. A stronger definition compared to Definition 1 is the following.

Definition 2. A cube **B** is called a *regular Robinson cube* if

1. **B** is a Robinson cube (Definition 1)

2. all matrices, which are formed by cutting the cube perpendicularly, where for each matrix $\mathbf{A}$ entry a_{11} is an element of one of the three edges joining at the vertex b_{111} (with $a_{11} = b_{111}$ if $\mathbf{A}$ is one of the three faces joining at the vertex b_{111}), are Robinson matrices.

An example of a regular Robinson cube is the bottom cube in Figure 2. A regular Robinson cube has some interesting features. For example, if $\mathbf{B}$ is a regular Robinson cube then it satisfies both three-way symmetry and the diagonal-plane equality. These properties become clear from the following result on the composition of a regular Robinson cube.

Proposition 1. Let $q = \min(i, j, k)$ and $r = \max(i, j, k)$. If $\mathbf{B}$ is a regular Robinson cube, then its entries b_{ijk} equal

$$b_{qrs} = b_{rqs} = b_{qsr} = b_{rsq} = b_{sqr} = b_{srq} \quad \text{for } s = q, ..., r.$$

Proof. The idea for the proof is depicted in Figure 1. First, let $\mathbf{A}$ be the front face of the cube, where $a_{11} = b_{111}$. Since b_{221} is a diagonal element of $\mathbf{A}$, $\mathbf{A}$ is a Robinson matrix if $b_{121} \leq b_{221}$. Next, let $\mathbf{A}$ be the cutting perpendicular on the front face of the cube, with $a_{11} = b_{121}$. Since b_{121} is a diagonal element of $\mathbf{A}$, the latter is a Robinson matrix if $b_{121} \geq b_{221}$. Thus, if $\mathbf{B}$ is a regular Robinson cube, then $b_{121} = b_{221}$ $(= b_{211} = b_{212} = b_{112} = b_{122})$. □

3 Examples

The most popular functions for triadic dissimilarities used in classification literature are the symmetric L_p-transforms:

$$b_{ijk} = (a_{ij}^p + a_{ik}^p + a_{jk}^p)^{1/p}.$$

For example, for $p = 1$ we have the perimeter function, for $p = 2$ the generalized Euclidean function, and for $p = \infty$ the generalized dominance function, that is, $b_{ijk} = \max(a_{ij}, a_{ik}, a_{jk})$. An alternative function for dissimilarities is the variance function, defined by

$$b_{ijk}^2 = \text{var}(a_{ij}, a_{ik}, a_{jk}) \; = \; (a_{ij}^2 + a_{ik}^2 + a_{jk}^2) - \frac{1}{3}(a_{ij} + a_{ik} + a_{jk})^2$$

which is also symmetric in i, j and k (De Rooij and Gower, 2003, p. 188).

Proposition 2. Let $\mathbf{A}$ and $\mathbf{B}$ be respectively a dissimilarity matrix and cube. Suppose b_{ijk} is defined as a L_p-transform or the variance function. Then $\mathbf{B}$ is a Robinson cube if and only if $\mathbf{A}$ is a Robinson matrix.

Proof. For $1 \leq i < j \leq m$ we have

$$b_{ijj} = (2a_{ij}^p)^{1/p} \geq (2a_{i+1j}^p)^{1/p} = b_{i+1jj} \quad \text{if and only if} \quad a_{ij} \geq a_{i+1j}$$

for a L_p-transform of a_{ij}, a_{ik} and a_{jk}, and

$$b_{ijj}^2 = 2a_{ij}^2 - \frac{1}{3}(2a_{ij})^2 \geq 2a_{i+1j}^2 - \frac{1}{3}(2a_{i+1j})^2 = b_{i+1jj}^2$$

if and only if

$$\frac{2}{3}a_{ij}^2 \geq \frac{2}{3}a_{i+1j}^2 \quad \text{if and only if} \quad a_{ij} \geq a_{i+1j}$$

for the variance function of a_{ij}, a_{ik} and a_{jk}. A similar property holds for $b_{ijj} \leq b_{i+1jj}$ for $1 \leq j \leq i < m$. □

A stronger property holds for the dominance function for dissimilarities, or equivalently the minimum function $b_{ijk} = \min(a_{ij}, a_{ik}, a_{jk})$ for similarities.

Proposition 3. Let **A** and **B** be respectively a similarity matrix and cube. If $b_{ijk} = \min(a_{ij}, a_{ik}, a_{jk})$, then **B** is a regular Robinson cube if and only if **A** is a Robinson matrix.

Proof. If **A** is a Robinson matrix then the minimum function has the property

$$1 \leq i \leq j \leq k \leq m \quad \Rightarrow \quad b_{ijk} = \min(a_{ij}, a_{ik}, a_{jk}) = a_{ik}$$

which fulfills the second requirement in Definition 2. Moreover, we have

$$\begin{aligned} 1 \leq i < j \leq m \Rightarrow b_{ijj} = a_{ij} \leq a_{i+1j} = b_{i+1jj} \quad \text{and} \\ 1 \leq j \leq i < m \Rightarrow b_{ijj} = a_{ij} \geq a_{i+1j} = b_{i+1jj} \end{aligned}$$

which shows the first requirement of Definition 2. □

Suppose the data at hand are binary (0/1) scores and that there are n records of i, j and k. Denote by

$$\begin{aligned} n_i &= \text{the number of 1s in } i \\ n_{ij} &= \text{the number of 1s common in } i \text{ and } j \\ n_{ijk} &= \text{the number of 1s common in } i,\ j \text{ and } k. \end{aligned}$$

In the remainder of this paper we assume that all matrices and cubes are of the similarity kind. However, the properties below could also have been formulated for dissimilarities.

Proposition 4. Let the Jaccard similarity coefficient be defined as

$$a_{ij} = \frac{n_{ij}}{n_i + n_j - n_{ij}} \qquad \text{for pairs of objects, and}$$

$$b_{ijk} = \frac{n_{ijk}}{n_i + n_j + n_k - (n_{ij} + n_{ik} + n_{jk}) + n_{ijk}} \quad \text{for triples of objects.}$$

(The latter definition comes from Heiser and Bennani, 1997, p. 196). Then **B** is a Robinson cube if and only if **A** is a Robinson matrix.

Proof. The result follows from the fact that

$$a_{ij} = \frac{n_{ij}}{n_i + n_j - n_{ij}} = b_{ijj}. \quad \square$$

Proposition 5. If $a_{ij} = n_{ij}$ and $b_{ijk} = n_{ijk}$, then the following statements are equivalent:

1. **A** is a Robinson matrix
2. **B** is a regular Robinson cube
3. $b_{ijk} = \min(a_{ij}, a_{ik}, a_{jk})$.

Proof. The result follows from the fact that $n_{ijj} = n_{ij}$, and if **A** is a Robinson matrix, then n_{ijk} has the property

$$1 \leq i \leq j \leq k \leq m \quad \Rightarrow \quad n_{ijk} = \min(n_{ij}, n_{ik}, n_{jk}) = n_{ik}. \quad \square$$

The result in Proposition 5 applies to the Russel-Rao similarity coefficient which is defined as n_{ij}/n for pairs of objects and n_{ijk}/n for triples of objects (Heiser and Bennani, 1997, p. 197). A sufficient condition for **A** with elements $a_{ij} = n_{ij}$ to be a Robinson matrix can be found in Hodson et al. (1971, p. 279). Let the binary scores be in a $n \times m$ table **X**, for example

$$\mathbf{X} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

where the objects i, j and k identify the columns of **X**. Suppose that the columns of **X** are ordered such that in each row the 1s are bunched together: **X** is said to posses the *consecutive 1s* property (see, for example, Hubert, 1974, p. 977 or Heiser, 1981, p. 73). If the rows of **X** contain consecutive 1s, then **A** with elements $a_{ij} = n_{ij}$ is a Robinson matrix. It follows from Proposition 5 that this condition is then also sufficient for **B** with elements $b_{ijk} = n_{ijk}$ to be a Robinson cube. Alternatively, it is also possible to generalize the original proof in Hodson et al. (1971) for a matrix to a cube.

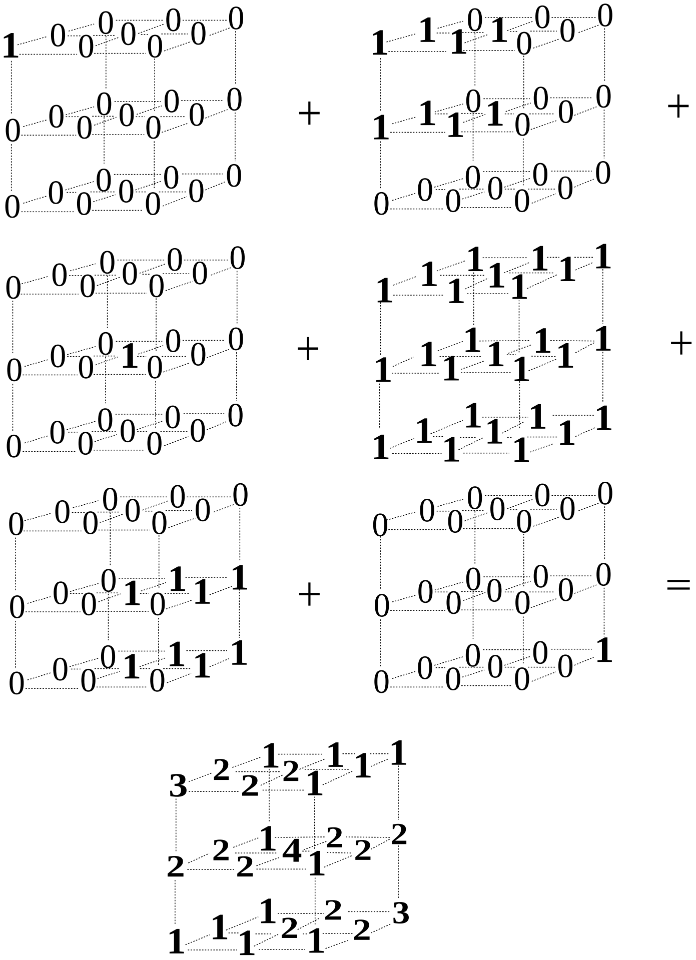


Fig. 2. The sum of regular Robinson cubes is a regular Robinson cube.

Proposition 6. If the columns of a binary table are ordered such that the rows contain consecutive 1s, then **B** with elements $b_{ijk} = n_{ijk}$ is a regular Robinson cube.

Proof. For the sake of an example consider the binary table **X**. The proof is further depicted in Figure 2. The first six cubes are the similarity cubes with elements n_{ijk} corresponding to the six rows of **X**. If a row has consecutive 1s, the similarity cube corresponding to this row, is a Robinson cube. The seventh and last cube in Figure 2, is the cube with elements n_{ijk} for the complete table

X. Figure 2 visualizes an interesting property of regular Robinson cubes: the sum of regular Robinson cubes is again a regular Robinson cube. □

4 Discussion

A data array arranged in a cube in which rows, columns and tubes refer to the same objects has been called three-way one-mode, or triadic data. Such data have been studied in attempts to identify higher order interactions among objects (Heiser and Bennani, 1997). In this paper, we have shown that we can recognize a simple order among the objects in triadic data, by a generalization of the Robinson property for dyadic data. We have discussed a general version of the Robinson cube, and a more specific one. Studying several definitions of triadic (dis)similarities, we found that in most cases, if a dyadic (dis)similarity is Robinsonian, then the triadic (dis)similarity is Robinsonian, too. A regular Robinson cube occurs only with the Russel-Rao coefficient calculated on an attribute matrix with the consecutive 1s property, and with the dominance metric for dissimilarities.

References

BARTHÉLEMY, J.-P., BRUCKER, F. and OSSWALD, C. (2004): Combinatorial optimization and hierarchical classifications. *4OR 2, 179-219.*

BRITO, P. (1991): *Analyse de données symboliques: pyramides d'heritage.* Thèse de doctorat, Université Paris 9.

CHEPOI, V. and FICHET, B. (1997): Recognition of Robinsonian dissimilarities. *Journal of Classification 14, 311-325.*

DE ROOIJ, M. and GOWER, J.C. (2003): The geometry of triadic distances. *Journal of Classification 20, 181-220.*

DIDAY, E. (1984): *Une représentation visuelle des classes empiétantes: les pyramides.* Research report 291, INRIA.

DIDAY, E. (1986): Orders and overlapping clusters in pyramids. In: J. de Leeuw, W.J. Heiser, J.J. Meulman and F. Critchley (Eds.): *Multidimensional Data Analysis.* DSWO Press, Leiden, 201-234.

DIDAY, E. and BERTRAND, P. (1986): An extension of hierarchical clustering: the pyramidal representation. In: E.S. Gelsema and L.N. Kanal (Eds.): *Pattern Recognition in Practice II.* North-Holland, Amsterdam, 411-424.

FICHET, B. (1984): Sur une extension de la notion de hiérarchie et son équivalence avec quelques matrices de Robinson. *Actes des "Journées de statistique de la Grande Motte", 12-12.*

FICHET, B. (1986): Data analysis: geometric and algebraic structures. In: Y.A. Prohorov et al. (Eds.): *First World Congress of the Bernoulli Society Proceedings.* V.N.U. Science Press, 123-132.

GAUL, W. and SCHADER, M. (1994): Pyramidal classification based on incomplete dissimilarity data. *Journal of Classification 11, 171-193.*

HEISER, W.J. (1981): *Unfolding Analysis of Proximity Data.* Leiden University, Leiden.

HEISER, W.J. and BENNANI, M. (1997): Triadic distance models: axiomatization and least squares representation. *Journal of Mathematical Psychology 41, 189-206.*

HODSON, F.R., KENDALL, D.G. and TAUTU, P. (1971): *Mathematics in the Archaeological and Historical Sciences.* University Press, Edinburgh.

HUBERT, L.J. (1974): Problems of seriation using a subject by item response matrix. *Psychological Bulletin 81 (12), 976-983.*

HUBERT, L.J., ARABIE, P. and MEULMAN, J.J. (1998): Graph-theoretic representations for proximity matrices through strongly-anti-Robinson or circular strongly-anti-Robinson matrices. *Psychometrika 43, 81-91.*

Part VII

Multivariate Statistics

Relative and Absolute Contributions to Aid Strata Interpretation

M. Carmen Bravo[1] and José M. García-Santesmases[2]

[1] Universidad Complutense de Madrid, Servicio Informático de Apoyo a Docencia e Investigación, Edificio Real Jardín Botánico Alfonso XIII, 28040 Madrid, Spain, *mcbravo@pas.ucm.es*

[2] Universidad Complutense de Madrid, Departamento de Estadística e Investigación Operativa, Facultad de Ciencias Matemáticas, 28040 Madrid, Spain

Abstract. Strata generalisation by symbolic objects is presented when there is a class variable to be explained simultaneously in all strata. This is attained by a generalised recursive tree-building algorithm for populations partitioned into strata and described by symbolic data, that is, more complex data structures than classical data. Symbolic objects describe decisional nodes and strata. This paper presents some measures to interpret strata and nodes. The method is integrated into the SODAS Software (Symbolic Official Data Analysis System), partially supported by ESPRIT-20821 SODAS and IST-25161 ASSO.

1 Introduction

Generalisation of strata is obtained by a generalised recursive tree-building algorithm (Breiman et al. (1984)) for a population partitioned into strata, such as individuals of a country divided into regions. Common predictors (including modal probabilistic variables and variables presenting hierarchical dependencies) and a class variable describe population in all strata. A modal variable associates to input data units a probability distribution over a set of categories. Hierarchical dependence between two variables occurs when a variable is non applicable for specific values of the other one.

The algorithm considers the strata structure in all its steps. Symbolic objects describe decisional nodes and strata. A stratum is described by a set of symbolic objects that represent rules for prediction of the class variable, obtaining a conjoint interpretation of strata in the context of all strata. We define and show how relative and absolute contributions help to strata interpretation pointing out strata with common and different rules and the importance or distribution of rules for a stratum. Node identification detects antagonistic rules, that is, the same antecedents of a rule predict a set of classes or its complementary, depending on the stratum. General formalisation can be extended to other symbolic data.

The method incorporates some advantages of Symbolic Data Analysis: treatment of complex data structures and aggregated data, symbolic data can be derived from data bases or given by an expert, confidentiality of individuals is guaranteed, input and output language are the same understable language to the user.

2 Algorithm

In this section, input and output data of the algorithm are presented, together to a brief summary of the algorithm. The main measures used by the algorithm are also presented.

Input data. Let Ω be a set of individuals, $E = \{S_1, ..., S_m\} \subset \mathcal{P}(\Omega)$ a partition of Ω. Thus, each element of E, $S_i \subset \Omega$ is a group of individuals, called a stratum (for $\omega \in \Omega$, $M(\omega) = i \iff \omega \in S_i$). Let individuals $\omega \in \Omega$ be described by the predictors Y_j, $j = 1, ..., p$ and the class variable Z. Different input variable types are considered: (1) *Ω a set of monoevaluated data*: Variables Y_j are categorical single-valued mappings from Ω to $\mathcal{Y}_j = \{1, ..., l_j\}$; (2) *$\Omega$ a set of multievaluated data*: Variables Y_j are mappings from Ω to $\mathcal{P}(\mathcal{Y}_j)$; (3) *$\Omega$ a set of probabilistic modal data*: Variables Y_j are modal variables with finite domain $\mathcal{Y}_j$, that is, for $\omega \in \Omega$, $Y_j(\omega) = q_j^\omega$ is a probability distribution over $\mathcal{Y}_j$, identified with $(1\, q_j^\omega(1), ..., l_j\, q_j^\omega(l_j))$. The symbolic data description $Y_j(\omega)$ can represent either the uncertainty for an individual or the variation for a group of individuals regarding categories in $\mathcal{Y}_j$. Case (2) is considered a particular case of probabilistic data, defining the uniform distribution for the categories given by $Y_j(\omega)$. In all cases, Z is a categorical single-valued mapping from Ω to $\mathcal{Z} = \{1, \ldots, s\}$.

The objectives are to explain the class variable by the predictors, affected by stratum membership; obtain sets of strata where this explanation is the same; describe a stratum by these class variable explanations together with their importance.

Output data. A decision tree can be represented by an *organised* set of *assertions* (Ciampi et al. (1996), Bravo and García-Santesmases (1997)). In our case, each decisional node described by the assertion $t_k = \beta_k \wedge \alpha_k \wedge \mu_k$, represents a set of strata for which the same rule for prediction of the class variable can be applied. The tree is represented by:

$$T = \{\beta_k \wedge \alpha_k \wedge \mu_k\}_{k=1,\ldots,K} \tag{1}$$

where K is the number of decisional nodes; β_k is a conjunction of events (each of them belonging to $B = \{b = [Y_j \in D_j], b^c = [Y_j \in \mathcal{Y}_j - D_j] | D_j \subset \mathcal{Y}_j\}$; D_j is a subset of the space of categories $\mathcal{Y}_j$; for modal data, $\sim$ replaces $\in$) defined in the predictors Y_j; α_k is a modal symbolic event describing the prediction for Z; and $\mu_k = [M \in S^k]$ with $S^k \subseteq \{1, \ldots, m\}$ is a Boolean event in the variable M. The μ_k is true for all individuals $\omega \in \Omega$ that belong to a stratum indicated in S^k. Stratum indicators in S^k are identified in *steps*

3 to 5 of the algorithm (see below). Function $\wedge$ is the product. Assertion $\beta_k \wedge \mu_k$ describes the population (its extension) for which the prediction of Z is described by α_k. For monoevaluated data, $\beta_k \wedge \mu_k(\omega)$ takes value 1 when ω belongs to the node k and value 0 otherwise ; for probabilistic data, $\beta_k \wedge \mu_k(\omega)$ is the probability of node k given ω, that is, for stratum indicators in S^k, the probability of descriptions D_j (in β_k), given ω. For example, given the assertion $\beta_k \wedge \mu_k$, with $\beta_k = [sex \sim f] \wedge [salh25 \sim yes]$, $\mu_k = [M \in S^k]$, and an individual $\omega \in \Omega$, defined by probabilistic data $sex(\omega) = (m(.3), f(.7))$, $salh25(\omega) = (yes(.2), no(.8))$, $M(\omega) = i$, then $(\beta_k \wedge \mu_k)(\omega) = [sex(\omega) \sim f] \wedge [salh25(\omega) \sim yes] \wedge [M(\omega) \in S^k] = .3 \cdot .2 \cdot 1 = .06$, when the stratum $i \in S^k$ and zero otherwise. This value gives the probability, given ω, of being $female$ and with $salh25$ for individuals belonging to stratum S_i, with $i \in S^k$.

An example of a decisional node in the case of input monoevaluated data, is $[sex = f] \wedge [salh25 = yes] \wedge [clerk \sim (no(0.10), yes(0.90))] \wedge [NACE \in \{services, electric\}]$. This assertion gives for individuals in services and electricity $NACE$ sectors the rule *if sex is female and mean gross hourly earnings is below the first quartile then the estimated probability to be clerk is* 0.9.

Each stratum is also described by an *organised* set of *weighted assertions*, the decisional tree node descriptions where the stratum belongs to (see Section 3 and Bravo and García-Santesmases (2000b), Bravo (2004a)).

Algorithm. The aim is to build recursively an *organised* set of assertions $T = \{t_k\}_{k=1\ldots K}$ (see (1)), by binary partitioning the population and combining at each step maximisation of an extended information content (EIC) measure of the tree with respect to Ω and selection of new decisional nodes. The EIC criterion measures the quality of prediction for the class variable in a new partition, taking into account stratum membership in the cut. For modal predictors, we obtain uncertainty partitions, that is, an individual $\omega \in \Omega$ does not belong to an element of the partition with certainty but it has a probability of belonging to it. The quality of prediction is tested for subsets of strata in order to build decisional nodes. A decisional node is a leaf for some strata, while the other strata follow the recursive method. For these strata that follow the recursive method a stopping criterion is also checked. In each step of the algorithm, T is composed of exploratory (obtained from the recursive partition, they can be binary split further on) and decisional nodes (split in a previous step from an exploratory node, they are terminal). The quality of prediction for the class variable by the predictors and strata is given by the information content measure (IC) of the tree with respect to Ω (Bravo and García-Santesmases (2000a), Bravo (2004a)).

The IC measure is defined as:

$$IC\{T, \Omega\} := -\sum_{k=1}^{K} P(\beta_k \wedge \mu_k) Ent(Z|\beta_k \wedge \mu_k) \tag{2}$$

The EIC measure when node r is split by b, b^c is defined as:

$$\begin{aligned} EIC\{T,r,b,\mathit{\Omega}\} &:= IC\{T(r),\mathit{\Omega}\} \\ -P(\beta_r \wedge b \wedge \mu_r)&\sum\nolimits_{i\in S^r} P([M=i]|\beta_r \wedge b \wedge \mu_r)Ent(Z|\beta_r \wedge b \wedge [M=i]) \\ -P(\beta_r \wedge b^c \wedge \mu_r)&\sum\nolimits_{i\in S^r} P([M=i]|\beta_r \wedge b^c \wedge \mu_r)Ent(Z|\beta_r \wedge b^c \wedge [M=i]) \end{aligned} \tag{3}$$

where $T(r) = T - \{\beta_r \wedge \alpha_r \wedge \mu_r\}$ is the tree that results from T when the node r is removed. The value $P([M=i]|a)$ is the estimated conditional probability of the stratum S_i to the node described by a and $Ent(Z|.)$ is the entropy (Quinlan (1990)) for Z in the corresponding node.

Given an assertion $\beta \wedge \mu$ ($\mu = [M \in S]$), defining a node in Y_j, Z, for monoevaluated data $\beta \wedge \mu$ is a Boolean assertion and $P(\beta \wedge \mu)$ and $P([M = i]|\beta \wedge \mu)$ are estimated in a frequentist way as:

$$P(\beta \wedge \mu) = \frac{Card(Ext_{\mathit{\Omega}}(\beta \wedge \mu))}{Card(\mathit{\Omega})};\; P([M=i]|\beta \wedge \mu) = \frac{Card(Ext_{S_i}(\beta \wedge \mu))}{Card(Ext_{\mathit{\Omega}}(\beta \wedge \mu))}$$

For probabilistic data, probabilities are estimated by:

$$P(\beta \wedge \mu) = \frac{\sum_{\omega\in\mathit{\Omega}}(\beta \wedge \mu)(\omega)}{Card(\mathit{\Omega})};\; P([M=i]|\beta \wedge \mu) = \begin{cases} \frac{\sum_{\omega\in S_i}\beta(\omega)}{\sum_{\omega\in\mathit{\Omega}}(\beta\wedge\mu)(\omega)} & i \in S \\ 0 & \text{otherwise} \end{cases}$$

Descriptions for Z in α_k are given by probability distributions over $\{1,...,s\}$. For monoevaluated data, probabilities are estimated as relative frequencies of each class in a node. For probabilistic data, the probability of class $l \in \{1,...,s\}$ in node k is estimated by:

$$P([Z=l]|\beta_k \wedge \mu_k) := \frac{\sum_{\omega\in\mathit{\Omega}}(\beta_k \wedge [Z=l] \wedge \mu_k)(\omega)}{\sum_{\omega\in\mathit{\Omega}}(\beta_k \wedge \mu_k)(\omega)}$$

The value of the information content measure is the negative of the value of a weighted uncertainty for the Z variable in decisional nodes. The extended information content measure is based on internal uncertainty in strata in successor nodes and measures the lost of uncertainty of Z in each stratum node when splitting a node of the tree. The decisional node criterion is based on a threshold for these internal uncertainties.

Let X be the set of exploratory nodes in an algorithm iteration. The algorithm main steps are shown in Figure 1 and very briefly described here:

Step 0: Initialisation and evaluation of IC at first algorithm iteration, $IC\{T,\mathit{\Omega}\}$, that is, the negative of the Z uncertainty in the whole population. The only exploratory node contains the whole population (and all strata).

Step 1: Check admissibility condition. For each $r \in X$ (if any), build $B_r \subseteq B$ the set of admissible splitting statements to be explored from node r

and considering *maximum depth level* permitted (and the information given by NA rules when they occur).

Step 2: Obtain the best split. For each $r \in X$, maximise in $b \in B_r$, the EIC measure, $EIC\{T, r, b, \Omega\}$ of T expanded from node r by splits b and b^c with respect to Ω. Maximise in $r \in X$ these measures and select the best node r' and split b. This node r' is removed from X (given that we explore it now). Make the split.

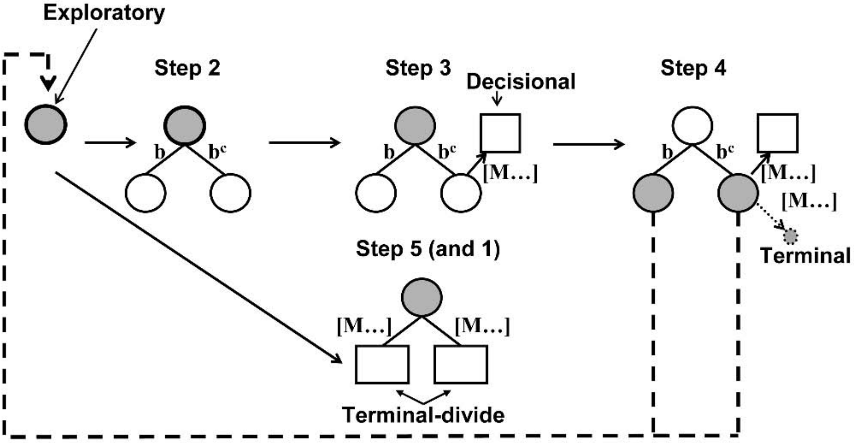


Fig. 1. Algorithm main steps.

Step 3: Decisional node criterion. For the new children nodes, i.e., the new exploratory nodes, check the set of strata for which the decisional node condition is satisfied (e.g. minimum probability for a class of Z in a stratum to be split in a decisional node). These strata are split from an exploratory node to form one or several decisional nodes.

Step 4: Strata terminal node condition. For the new exploratory nodes (when at least one strata belongs to it), check the set of strata for which the stopping propagation condition from the node is satisfied (e.g. low weight). Remove these strata from an exploratory node to form a terminal node.

Step 5: Check minimum improvement of IC. If the improvement of the value of IC for the new tree obtained in *Steps 2-4*, is relatively small to the previous IC value, then: (1) algorithm *Steps 2-4* are undone and; (2) the explored *parent* node r' is split into two terminal nodes, splitting the subset of strata by their quality of prediction of Z. This latter action is also taken when the maximum depth level is attained or an exploratory node has no admissible splits (in *Step 1*). These nodes are called terminal-divide in Figure 1.

Update node descriptions. In this step, update of node descriptions and IC measure are obtained. When *Steps 2-4* have not been undone, the descriptions of new exploratory/decisional/terminal nodes in Y_j add to its parent node descriptions in Y_j, the description of split b, b^c. The description in M of exploratory/decisional/terminal nodes obtained in *Steps 2-5* identify the strata they contain. Initially, at *Step 2*, exploratory nodes contain the same strata as parent r' node. Obtain for all these nodes the description in Z. Compute $IC\{T, \Omega\}$, go to *Step 1.*

3 Symbolic object description of strata

The advantages of the method presented here are the analysis of symbolic data, the *generalisation of a stratum by symbolic objets* that represent prediction rules for the class variable by the predictors giving a conjoint interpretation of strata in the *context* of all strata and not isolatedly, and *classification of strata by common prediction rules.* Also that inclusion of strata information in all steps of the algorithm gives in only one tree common prediction rules for strata and favors good predictors for some strata.

With the method, it is possible to identify strata with antagonistic rules, that is, the same predictor values can predict a different class, depending on the stratum; characterize some strata before others, that is, that go out of the recursive process before; identify strata that predict the same class with a common rule with the exception of the values of one predictor; and, definitively, classify strata with common prediction rules and identify strata with different prediction rules. The formalisation of the method allows for the extension to other symbolic data. The measures presented here are extended to other symbolic data in Bravo (2004a); these data are interval, fuzzy and possibilistic data (Diday (1995), Bravo(2004a)).

As an output of the algorithm, a symbolic object description of each stratum S_i is obtained. This description is composed by the rules that can be applied for the stratum and gives the relative importance they have in this stratum. A stratum is described by different 'segments' of objects described by the values of the predictors and the value for the prediction of the class Z in these segments, as well as by certain weights these segments have in the stratum. Each stratum is described by an *organised* set of *weighted assertions* (Bravo and García-Santesmases (2000a, 2000b), Bravo (2004a)). Stratum S_i $(i = 1, ..., m)$, can be described by:

$$S_i : \{w_k^i(\beta_k \wedge \alpha_k) \mid k = 1, \ldots, K\} \tag{4}$$

where: $w_k^i := P(\beta_k \wedge \mu_k|[M = i]) \in [0, 1]$ is the *relative contribution* of the decisional node k (with $\mu_k = [M \in S^k]$) to the stratum S_i. For monoevaluated data, the value of w_k^i is given by:

$$w_k^i = \begin{cases} \frac{Card(Ext_{S_i}(\beta_k))}{Card(S_i)} & i \in S^k \\ 0 & \text{otherwise} \end{cases} \quad \text{for } k = 1, ..., K \tag{5}$$

For probabilistic data, the value of w_k^i is given by:

$$w_k^i = \begin{cases} \frac{\sum_{\omega \in S_i} \beta_k(\omega)}{Card(S_i)} & i \in S^k \\ 0 & \text{otherwise} \end{cases} \quad \text{for } k = 1, ..., K \tag{6}$$

where $\sum_{\omega \in S_i} \beta_k(\omega)$ is the weight of stratum S_i in node k.

In both cases, β_k and μ_k are the assertion and event of decisional node k in Y_j and Z, respectively. These contributions verify:

$$\sum_{k \in \{1,...,K\}} w_k^i = 1 \text{ for all } S_i \in E$$

Relative contributions help strata interpretation. w_k^i measures the relative importance of node k to stratum S_i.

The *absolute contribution* of a stratum to a node measures the importance of a stratum in a node and characterises nodes by strata. Let t_k be a tree node, absolute contributions of strata to node t_k (with $\mu_k = [M \in S^k]$) are $wa_i^k := P([M = i] | \beta_k \wedge \mu_k)$ for $i \in \{1, ..., m\}$. For monoevaluated data, the value of wa_i^k is given by:

$$wa_i^k = \begin{cases} \frac{Card(Ext_{S_i}(\beta_k))}{Card(Ext_{\Omega}(\beta_k \wedge \mu_k))} & i \in S^k \\ 0 & \text{otherwise} \end{cases} \quad \text{for } i \in \{1, ..., m\} \tag{7}$$

while for probabilistic data, the value of wa_i^k is given by:

$$wa_i^k = \begin{cases} \frac{\sum_{\omega \in S_i} \beta_k(\omega)}{\sum_{\omega \in \Omega} (\beta_k \wedge \mu_k)(\omega)} & i \in S^k \\ 0 & \text{otherwise} \end{cases} \quad \text{for } i \in \{1, ..., m\} \tag{8}$$

The value wa_i^k measures the relative importance of stratum S_i in node k. In both cases, these contributions verify:

$$\sum_{i \in \{1,...,m\}} wa_i^k = 1 \text{ for all } k = 1, ..., K$$

High wa_i^k values identify strata that characterize a decisional node k.

Looking at relative contributions we may detect strata that share rules, that is, those with non zero relative contributions in the same node. The importance of these rules in each stratum is given by these relative contributions. From another point of view, a node describes several strata with the same prediction rule. The importance of strata in the nodes or rules associated are given by *absolute contributions*.

4 Example

The method, implemented in the SODAS software (Bravo (2000, 2004a, 2004b)), has been applied to probabilistic data obtained from *T25IT Italy: Monthly earnings by local unit size, NACE (economic activity) and ISCO (profession)* data about statistics on the structure and distribution of earnings (SES) in 1995. Consolidated original data refers to 5.000.000 employees. More details on original data may be found in Bravo & García-Santesmases (2000), Bravo (2004a).

The set Ω consists in 720 data units described by modal probabilistic data, considering input data weights of original data. The groups described come from the combination of the categories of 22 economic sectors, 7 professions and 7 company sizes. The set of strata, $E \subset \mathcal{P}(\Omega)$, contains subsets of data units that belong to the same $NACE$ sector ($m = 5$). The $NACE$ sectors considered are: mining and quarrying; manufacturing; electricity, gas and water supply; construction; and services. The class variable is the binary (yes/no) variable $manual$ ($s = 2$). Predictors are sex and binary (yes/no) variables for thresholds in original variable quartiles: $h50$ for mean weekly hours; $sal75$ for mean gross monthly earnings; $b25$ for mean monthly value of periodic bonuses, $salh50$ for mean gross hourly earnings and cvm median for monthly earnings coefficient of variation. An example of the description of one data unit $\omega \in \Omega$ is:

$$(sex(\omega) = (f(0.64), m(0.36)), sal75(\omega) = yes, b25(\omega) = (yes(0.04), no(0.96))$$
$$salh50(\omega) = (yes(0.64), no(0.36)), nace(\omega) = services, manual(\omega) = no)$$

that represents a set of individuals of the $services$ sector, $non - manual$, with mean gross salary below the third quartile and probability distributions for sex $(f(0.64), m(0.36))$, for $b25$, $(yes(0.04), no(0.96))$ and for $salh50$, $(yes(0.64),$ $no(0.36))$.

Figure 2 shows the decisional tree built in 3 levels. The initial information content measure is -0.677260 and the final value is -0.432755. Round nodes are exploratory nodes and square nodes are decisional nodes. At level 3, five terminal nodes are obtained because the maximum level condition is attained (see steps 5 and 1 of the algorithm). Light grey nodes (on the left side of an exploratory node) represent prediction rules for $manual$ employees and dark grey nodes (on the right side) for $non - manual$ employees. Nodes show weights and estimated probabilities for $non - manual$ employees. Decisional nodes show strata as well. The decisional nodes in this tree are:

$11d1$: $[sal75 \sim no] \wedge [manual \sim (no0.96, yes0.04)] \wedge [nace \in \{manufact, services,$ $construc, mining, electric\}]$

$30d0$: $[sal75 \sim yes] \wedge [salh50 \sim yes] \wedge [b25 \sim yes] \wedge [manual \sim (no0.18, yes0.82)] \wedge$ $[nace \in \{manufact, services, construc, mining\}]$

$31d1$: $[sal75 \sim yes] \wedge [salh50 \sim yes] \wedge [b25 \sim no] \wedge [manual = no] \wedge [nace = construc]$

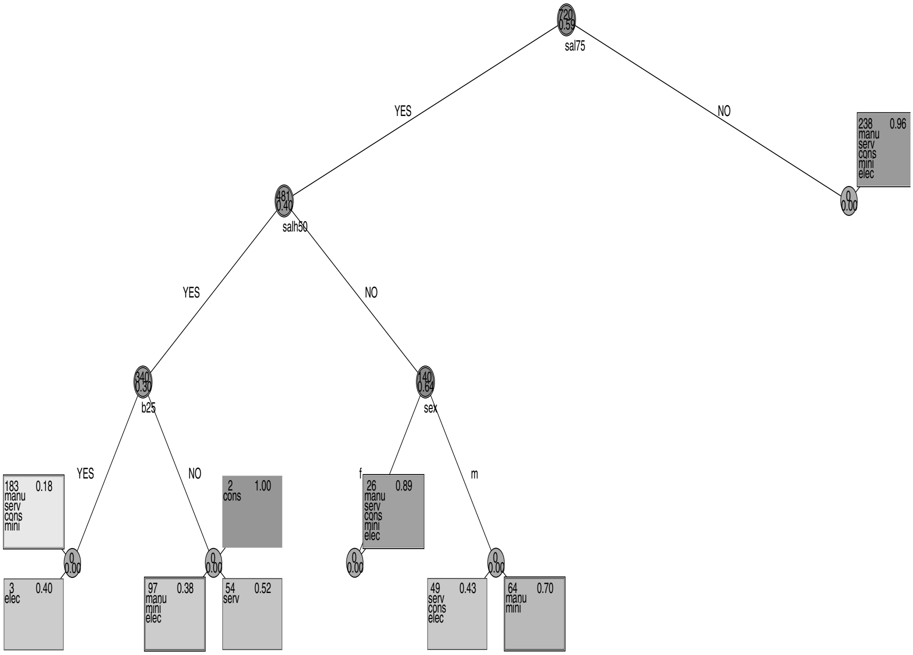


Fig. 2. Decision Tree on SES data, 3 levels

$$32d1 : [sal75 \sim yes] \wedge [salh50 \sim no] \wedge [sex\sim f] \wedge [manual \sim (no0.89, yes0.11)] \wedge$$
$$[nace \in \{manufact, services, construc, mining, electric\}]$$

An illustration of a node obtained by the maximum tree level condition is:

$$33td1 : [sal75 \sim yes] \wedge [salh50 \sim no] \wedge [sex\sim m] \wedge [manual \sim (no0.7, yes0.3)] \wedge$$
$$[nace \in \{manufact, mining\}]$$

Node identification is $nm\mathbf{d}x$ or $nm\mathbf{td}x$ with d for decisional nodes obtained in step 3 of the algorithm by the decisional node condition, td for decisional nodes obtained in step 5 or 1 of the algorithm, n the node tree level and $x = 0$ when the higher probability class is $manual$ and $x = 1$ when the higher probability class is $non-manual$. Then, two nodes with the same nm and different values of x define antogonistic rules.

Table 1 gives relative and absolute contributions for these decisional nodes. Columns under Cr and Ca are relative and absolute contributions. The sum of the elements in a Cr column is not 1, because in the table only the decisional nodes obtained by the decisional node condition are considered. All NACE sectors share the $11d1$ rule, with a relative importance around 30% or 35%. Rule 30d0 is shared by $mining$, $manufacturing$, $construction$ and *services* sectors, having a relative importance into the first three of 30%, while in the *services* sector it decreases to 11%. NACE sectors with the exception of *construction* share rule $32d1$ with a relative importance of 4%.

	mining		*manuf*		*electr*		*constr*		*servic*	
	Cr	Ca	Cr	Ca	Cr	Ca	Cr	Ca	Cr	Ca
$11d1$	0.34	0.05	0.32	0.57	0.34	0.05	0.3	0.05	0.37	0.28
$30d0$	0.37	0.05	0.32	0.77			0.28	0.05	0.11	0.13
$31d1$							0.08	1		
$32d1$	0.04	0.05	0.04	0.6	0.04	0.05	×	0.04	0.04	0.26

Symbol × specifies a value lower than 0.01

Table 1. Table of relative and absolute contributions for probabilistic SES data.

$Mining$ and $manufacturing$ sectors share 3 rules $11d1$, $30d0$ and $32d1$ with similar relative importance in both strata ($Cr = 0.34$, 0.37 and 0.04 for *mining* and $Cr = 0.32$, 0.32 and 0.04 for $manufacturing$). Thus, about 70% of *manual* explanation in both sectors is for the same prediction rules. The *services* sector also shares these rules, the first and third with similar relative importance while the second decreases to 11%. The $manufacturing$ sector characterises more than the others rules $11d1$, $30d0$ and $32d1$ ($Ca = 0.57, 0.77$ and 0.6), while *services* do it less ($Ca = 0.28$, 0.13 and 0.26) and *mining* and *construction* do it much less (Ca bellow 0.06 in all cases). The *construction* sector characterises by itself the rule 31d1 ($Ca = 1$), which has a relative importance in this sector of 8% ($Cr = 0.08$).

As an example, the *mining* sector is described by:

$$\begin{gathered} mining : \{0.34([sal75 \sim no] \wedge [manual \sim (no(0.96), yes(0.04)]), \\ 0.37([sal75 \sim yes] \wedge [salh50 \sim yes] \wedge [b25 \sim yes] \wedge \\ [manual \sim (no(0.18), yes(0.82))]), \\ 0.04([sal75 \sim yes] \wedge [salh50 \sim no] \wedge [sex \sim f] \wedge \\ [manual \sim (no(0.89), yes(0.11))]), \\ 0.25 Other\} \end{gathered}$$

that is, by three rules with respective relative importance in the sector of 0.34, 0.37 and 0.04. In Figure 2, nodes with a double surrounding line show nodes where the *mining* sector is present.

5 Conclusion

A classification of strata by common prediction rules is obtained with the method presented here, which provides a conjoint interpretation of strata in the *context* of all strata. *Relative* and *absolute contributions* have been defined to aid strata interpretation after the application of the method. We have showed how these contributions identify common prediction rules in different strata, together with their relative importance. Also rules can be characterized by some strata. Together with the generalisation of strata by symbolic objects, these contributions identify the rules applicable to a stratum with their relative importance. These measures contribute to the interpretation of the results of the method.

References

BOCK, H.-H., DIDAY, E., Eds. (2000): *Analysis of Symbolic Data. Exploratory Methods for Extracting Statistical Information from Complex Data.* Springer, Heidelberg.

BRAVO, M.C. (2000): Strata decision tree symbolic data analysis software. In: H.A.L. Kiers, J.P. Rasson, P.J.F Groenen, M. Shader (Eds.): *Data Analysis, Classification and Related Methods.* Springer, Heidelberg, 409-415.

BRAVO LLATAS, M.C. (2004a): *Análisis de Segmentación en el Análisis de Datos Simbólicos.* Ed. Universidad Complutense de Madrid. Servicio de Publicaciones. ISBN: 8466917918.

BRAVO, M.C. (2004b): SDT User Manual. Strata decision tree. In: *User manual for SODAS 2 Software, ASSO (Analysis System of Symbolic Official Data) Project (IST-2000-25161)*, 12 pp.

BRAVO, M.C., GARCÍA-SANTESMASES, J.M. (1997): Segmentation trees for stratified data. In: J. Jansen, C.N. Lauro (Eds): *Applied Stochastic Models and Data Analysis: The Ins/Outs of Solving Real Problems.* Curto, Naples, 37-42.

BRAVO, M.C., GARCÍA-SANTESMASES, J.M. (2000a): Segmentation trees for stratified data. In: H.-H. Bock and E. Diday (Eds.): *Analysis of Symbolic Data. Exploratory Methods for Extracting Statistical Information from Complex Data.* Springer, Heidelberg, 266-293.

BRAVO, M.C., GARCÍA-SANTESMASES, J.M. (2000b): Symbolic object description of strata by segmentation trees. In: *Computational Statisticss 15, 13-24.*

BREIMAN, L., FRIEDMAN, J. H., OLSHEN,R.A., STONE, C.J. (1984): *Classification and Regression Trees.* Wadsworth, Belmond, Ca.

CIAMPI, A., DIDAY, E., LEBBE, J., PÉRINEL, E. and VIGNES, R. (1996): Recursive partition with probabilistically imprecise data. In: E. Diday et al. (Eds.): *Ordinal and Symbolic Data Analysis.* Springer Verlag. 201–212.

DIDAY, E. (1995): Probabilistic, possibilist and belief objects, *Annals of Operations Research, 55, 227-276.*

QUINLAN, J.R. (1990): Probabilistic decision trees. In:Y. Kodratoff, R. Michalski (Eds.). *Machine Learning, an Artificial Intelligence Approach, III.* Kaufmann, 140-152.

Classification and Generalized Principal Component Analysis

Henri Caussinus[1] and Anne Ruiz-Gazen[2]

[1] Laboratoire de Statistique et Probabilités, U.M.R. - C.N.R.S. C5583, Université Paul Sabatier, 118, route de Narbonne, 31062 Toulouse cedex 4, France, *caussinus@cict.fr*

[2] LSP, Université Paul Sabatier and GREMAQ, Université Toulouse 1, 21, allée de Brienne, 31000 Toulouse, France, *ruiz@cict.fr*

Abstract. In previous papers, we propose a generalized principal component analysis (GPCA) aimed to display salient features of a multidimensional data set, in particular the existence of clusters. In the light of an example, this article evidences how GPCA and clustering methods are complementary. The projections provided by GPCA and the sequence of eigenvalues give useful indications on the number and the type of clusters to be expected; submitting GPCA principal components to a clustering algorithm instead of the raw data can improve the classification. The use of a convenient robustification of GPCA is also evoked.

1 Introduction

Visualizing and classifying are complementary purposes of exploratory data analysis. If the data consist of an objects × variables real matrix, principal component analysis (PCA) and related techniques (e.g. correspondence analysis) are the most popular tools of visualization and are often used to complement partition-type clustering techniques (e.g. k-means or other methods belonging to the class of "nuées dynamiques" according to Diday's terminology). There are two main aspects of this complementary use.

- Low dimensional displays of the data set allow the user to verify whether or not the groups obtained by a clustering algorithm make sense and/or which kind of groups they are: do the data present a "natural" partition into groups, or do they arise from a more or less artificial dividing of a fairly homogeneous data set? In other words, does the data cloud look like cumulus, stratus or cirrus? Moreover, by representing the variables on the same display, biplots (Gabriel (1971)) highlight the variables or their combinations that are the most responsible for the visualized structure.
- Since principal components are supposed to contain the most relevant information, they can be submitted to the clustering algorithm instead of the whole set of variables. This can be expected to eliminate uninteresting noise and thus facilitate the retrieval of clusters. The efficiency of such an approach has been investigated by many authors. To cite only one recent paper, see e.g. Chae and Warde (2006).

However, while clustering is related to the search of some data structure, PCA displays the data according to the criterion of maximal dispersion, which is not necessarily the best way to visualize clusters or any salient feature of the data. On the contrary projection pursuit techniques aim to display the data by maximizing a criterion of heterogeneity, which is more closely related to the search of a partition. In previous papers (Caussinus and Ruiz-Gazen (1993, 1995), Caussinus et al. (2003b)) we show that suitable generalizations of PCA work as projection pursuit techniques able to display interesting structures of the data. Among the various projection pursuit techniques, the present paper further investigates our approach as a complement of clustering methods through a real life example.

Combinations of cluster analysis and PCA have been considered for a long time. Chapters 8 and 9 of Diday et al. (1979) are devoted to such developments. Bock (1987) draws attention to the fact that PCA is not designed for the purpose of classification and proposes alternatives he calls "projection pursuit clustering"; he notes that one of his proposals is close to Diday's "Analyse typologique discriminante" (Diday et al. (1979), chapter 9). Stute and Zhu (1995) propose a "dimension-reducing k-means clustering". Our mathematical tools and our practical approach are a little different from those of these authors but our aim is very close to theirs.

2 Generalized PCA

Let X be a $n \times p$ matrix (objects × variables). The transpose of the i^{th} row is denoted by X_i, the empirical mean of the X_i's by $\bar{X}$ and the empirical matrix of variances and covariances by V, which is assumed non singular. For any column p-vector x we set $\|x\|_{V^{-1}} = x'V^{-1}x$ where x' is the transpose of x. Let us set

$$T(\beta) = \frac{\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} w_{ij}(\beta)\,(X_i - X_j)\,(X_i - X_j)^T}{\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} w_{ij}(\beta)}$$

with $w_{ij}(\beta) = \exp\left(-\frac{\beta}{2}\, \|X_i - X_j\|^2_{V^{-1}}\right)$, β being a tuning parameter (in practice close to 2: see the above mentioned papers).

Generalized PCA (GPCA) consists in projecting the X_i's onto the subspace spanned by the m eigenvectors of $VT^{-1}(\beta)$ associated with the m largest eigenvalues. An important property of these projections is to be invariant under any affine transformation of the data rows. In particular, raw or standardized data provide the same display.

Another useful property concerns the sequence of eigenvalues. Within a fairly general probabilistic model, it can be shown that the dimensions associated with theoretical eigenvalues lower than $\beta + .5$ merely contain noise, which gives a valuable information about the number of principal components to take into consideration. In practice, due to sampling variability and possible inadequacy of the model, this cutting off value must be cautiously used. Testing procedures have been developed and turn out to be interesting in some cases (Caussinus et al. (2003b)), but they will not be used in the exploratory data analysis framework of the present paper.

The biplots obtained by means of GPCA can be interpreted as usual ones (Gabriel (2002)).

In practice, the presence of outliers may slant GPCA towards their detection rather than the detection of clusters (an outlier is a small cluster). To get round this problem, we exchange V for the robust estimate of the variance proposed by Ruiz-Gazen (1996). This estimate S depends on a tuning parameter α in such a way that $S(0) = V$ and $S(\alpha)$ becomes more robust when α increases; here, α will be set to .2 except when more robustification will be required.

The next section emphasizes the interaction between plotting and clustering from an empirical point of view, while our previous papers were mainly devoted to the production of displays and their theoretical properties. At this place, it is worth mentioning Art et al. (1982) who consider the empirical properties of a “local variance” similar to T for cluster analysis.

3 Getting and visualizing clusters

The interaction between GPCA and clustering will now be analyzed through an example. The olive oil data set has been analyzed by several authors: Forina et al. (1983) seem at the origin of the statistical study of these data; Glover and Hopke (1992) and Cook et al. (2004) use them to illustrate projection pursuit approaches. The data can be found on the web: http://www2.chemie.uni-erlangen.de/publications/ANN-book/datasets/

The data consist of the percentage composition of $p = 8$ fatty acids found in the lipid fraction of $n = 572$ Italian olive oils. The 572 samples come from known regions subdivided into areas as shown in Table 1.

Region	Area	Size
A Southern Italia	1 Northern Apulia	25
A Southern Italia	2 Calabria	56
A Southern Italia	3 Southern Apulia	206
A Southern Italia	4 Sicily	36
B Sardinia	5 Inland Sardinia	65
B Sardinia	6 Coastal Sardinia	33
C Northern Italia	7 Eastern Liguria	50
C Northern Italia	8 Western Liguria	50
C Northern Italia	9 Umbria	51

Table 1. Regions and areas of olive oil samples

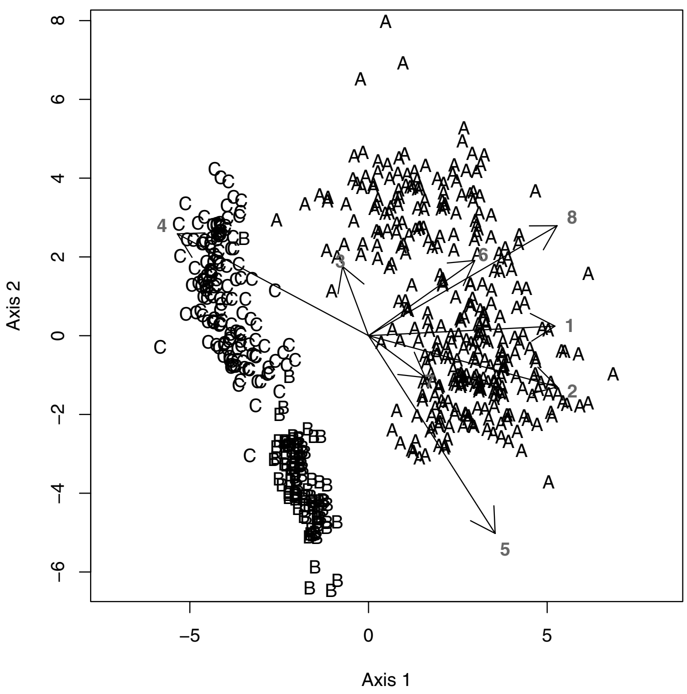


Fig. 1. GPCA biplot of olive oil data with region labels.

Since the oils come from known areas, these data have been previously processed by means of various supervised clustering techniques. For example, Cook and al. (2004) describe how to combine classifiers (support vector machines) with visual (tour) methods. The challenge in the present paper will be to process the data by a visualization/classification method without taking into account the provenience of the oils, except to evaluate the results of the analyses.

Step 1. To get a first insight into the data, we project both objects (letters corresponding to the region) and variables (arrows) onto the first principal plane of the GPCA described in the previous section (Figure 1). It seems clear that there are three well characterized clusters which correspond to the regions up to very few cases (ordinary or standardized PCA are far from giving so a comprehensive display). Moreover, the biplot shows that variable 8 is the most responsible for the separation of one region (South A) from the two others (Sardinia B and North C). In fact, a look at the data shows that variable 8 is zero for B and C and strictly positive for A. Variables 5 and - to a lower extent - 2, 3 and 4, are the most relevant ones to distinguish between

B and C. On the other hand, the sequence of eigenvalues (10.99, 7.23, 4.99, 4.02, 3.31, 2.89, 2.65, 2.52) indicates that higher principal components still contain information about the structure of the data set at least up to the fifth or sixth one. Since the inspection of further principal planes does not clearly display such a structure, we shall study each of the three groups in a second step of the analysis. Now, resting on the suggestions of Figure 1, the data are subdivided into three classes by means of a clustering algorithm. In all the paper we use k-means (Hartigan and Wong (1979)) with 100 random starts. Although this is not suggested by Figure 1, a subdivision between nine classes is also performed to be compared with the known prior subdivision into nine areas. Table 2 shows the Rand coefficients (Rand (1971)) comparing the known classification A, B, C (resp. 1 to 9) and the three (resp. nine) classes found by applying the algorithm successively to the original data, the six first principal components of ordinary and standardized PCA and the six first principal components of GPCA.

	Raw data	PCA	st. PCA	GPCA
3 groups vs. regions	0.761	0.761	0.738	0.960
9 groups vs. areas	0.896	0.812	0.806	0.905

Table 2. Rand coefficients for four clustering approaches

Retrieval of the groups is fairly good with the original data, worse with PCA components, better with the first 6 principal components of GPCA. To save space, we do not discuss at length what happens with different numbers of principal components: the results are very similar for 5 to 7 components, with a slight improvement for recovering the 3 regions (resp. the 9 areas) when the dimension is smaller (resp. larger).

Table 3 cross-classifies the true regions A, B and C versus the clusters (A^*, B^* and C^*) obtained by the algorithm with the first six principal components of GPCA.

	A	B	C	Total
A^*	319	0	0	319
B^*	0	97	20	117
C^*	4	1	131	136
Total	323	98	151	572

Table 3. Olive oil data: regions vs. retrieved clusters

Step 2. We analyse now the three groups obtained at the first step by k-means from the six first principal components of GPCA.

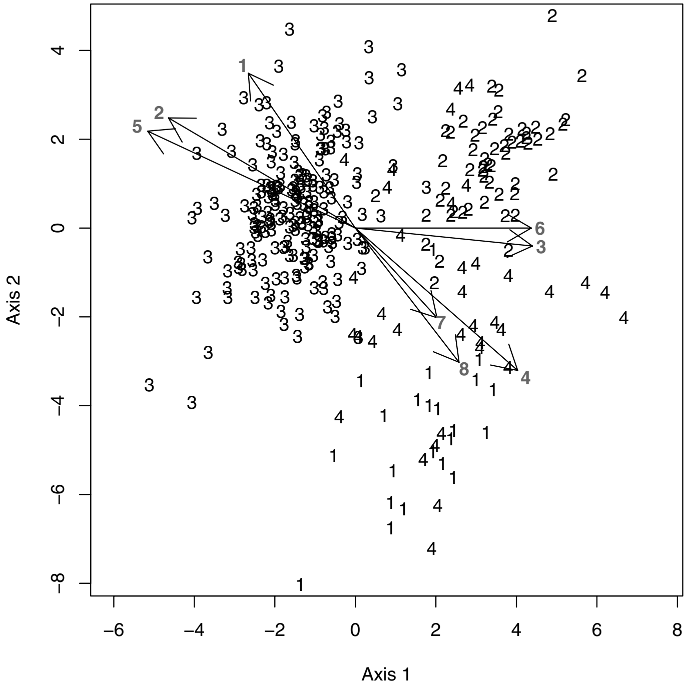


Fig. 2. GPCA biplot of olive oil data, class A*, with area labels.

Let us first consider A*. The first principal plane of GPCA (Figure 2) suggests 3 clusters (subclusters of A*) and the sequence of eigenvalues (5.84, 3.40, 3.16, 3.06, 2.90, 2.54, 2.47, 2.32) suggests the relevance of five dimensions. In order to understand what we can expect from the clustering algorithm on these 5 dimensions, the resulting classes from k-means (labeled 1*, 2*, 3*) are compared to the true areas (1, 2, 3, 4) in Table 4. While objects from areas 1, 2 and 3 are put together with few exceptions, those from area 4 are dispersed between the other classes, mainly between two of them, as can be expected from the display (clustering into four classes does not recover area 4 but rather split the 2* class in two). A similar feature is pointed out by Cook et al. (2004) in the context of supervised classification: we refer to these authors for a possible explanation (importation of olives to Sicily from other areas); the point is that the absence of a "Sicilian cluster" corresponds to a real fact. The clustering algorithm has also been worked out on raw data with similar but somewhat less satisfactory results; since three classes have to be compared to four "expert" ones, the quality of the results is measured

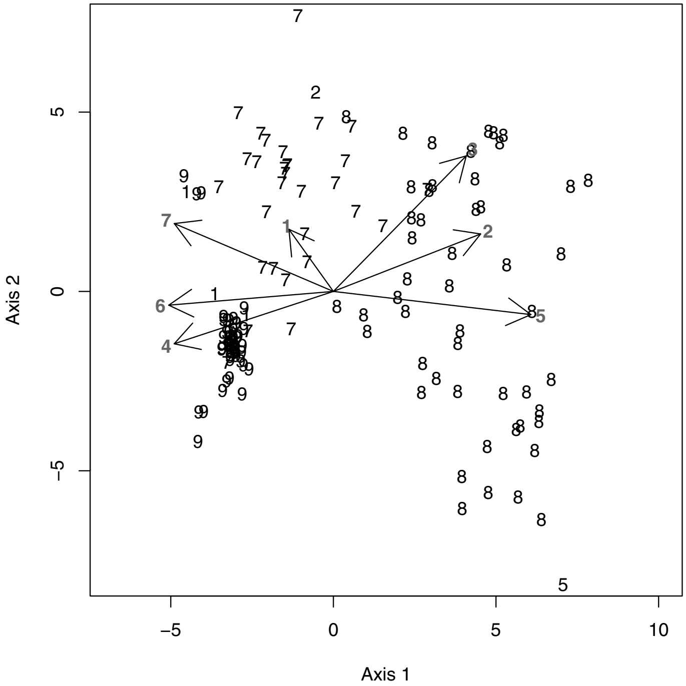


Fig. 3. GPCA biplot of olive oil data, class C*, with area labels.

by an asymmetric Rand coefficient (Chavent et al. (2001)) whose values are .97 for our proposal (Table 4) and .95 when clustering on raw data.

	1	2	3	4
1^*	21	0	0	16
2^*	0	1	201	5
3^*	1	54	5	15

Table 4. Olive oil data, class A*: areas 1, 2, 3, 4 vs. clusters 1^*, 2^*, 3^*

Let us now consider C*. Figure 3 shows the projections of the objects on the first principal plane of GPCA together with the 7 variables (the 8th variable takes the value 0 for all objects and thus has been dropped). The display visualizes a compact cluster and a more scattered one which could be thought of as one or two clusters. The clustering algorithm has thus been performed to look for two or three classes. The results from the raw data or

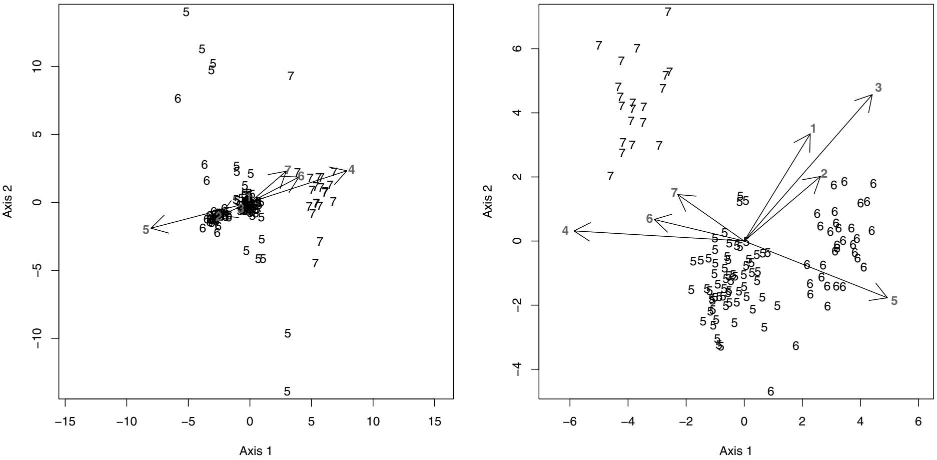


Fig. 4. Olive oil data, class B*, GPCA biplots with area labels, $\alpha = .2$ (left) and $\alpha = .5$ (right).

the first principal components of GPCA are very similar: the three classes are very close to the three "true" areas. The subdivision into two classes heavily depends on the clustering algorithm that, roughly speaking, can put together 7 and 8 or 7 and 9.

Let us finally consider B*. Figure 4 (left) is the first two-dimensional projection provided by GPCA with the same tuning parameters as in previous analyses ($\alpha = .2$, $\beta = 2$). The main feature of the display is now the presence of overdispersed values. On the one hand, this fact is interesting by itself but will not be discussed in detail here (note only that (i) Figure 4 (right) in Cook et al. (2004) suggests a similar though less striking feature, (ii) almost all the "outliers" belong to area 5). On the other hand, the display does not give a good insight into the main set of objects concentrated around the centre of the graphic. This drawback can be overcome by a more robust analysis, that is by increasing α (see section 2). Figure 4 (right) gives the projection obtained with $\alpha = .5$ (and still $\beta = 2$). Three clusters appear. Up to only two "errors", the clustering algorithm from the principal components of GPCA finds the two Sardinian areas and separates the 20 areas which "should not be in this subgroup". In a sense, this attenuates the major "misclassification" of step 1 (see Table 2). But is it a misclassification? It may also happen that these oils of region C are somewhat different from the others in that region and then the analysis would reveal a substantial feature. In this case, a supervised classification fails to discover this aspect of the data (for such a discussion, see Diday et al. (1979), p. 259-260); this can explain that previous analyzes (Glover and Hopke (1992); Cook et al. (2004)) did not find it. However, this can also be an artefact: further analysis is necessary, e.g. the analysis of $B^* + C^*$.

To summarize the results, the method we advocate seems efficient to get relevant clusters. In fact, the provenience of the oils is recovered to a large extent with two notable exceptions. One of them, concerning Sicily, turns out to point to a true problem: the cluster that is not found is likely not to exist; the other addresses a question to the chemist by creating a class whose specificity should be interesting to investigate.

4 Further comments and conclusion

Let us first draw the main lessons from the analysis of the example.
(i) Visual inspection of the data by means of GPCA is useful to get an idea of the number of possible homogeneous classes and their major characteristics.
(ii) Clustering from the first principal components of GPCA rather than the raw data improves efficiency in many circumstances; the sequence of eigenvalues provides a good guideline for choosing the suitable number of components; incidentally, using the results of GPCA gets round the problem of possible linear transformations of the data, in particular standardization, since GPCA is invariant under any affine transformation.
(iii) Robustification of GPCA gets rid of discording observations that are likely to spoil the displays as well as the clustering (in fact, non robust GPCA is useful to detect outliers, but another generalization of PCA is simpler and more efficient for that: see Caussinus et al. (2003a)).

With its two steps, the analysis is basically hierarchical. It could then be claimed that hierarchical clustering would be more appropriate. Nevertheless, (i) from a formal point of view, when starting the analysis the structure of the data is not known, (ii) from the practical point of view, the hierarchical classifications which we did perform do not bring much more insight into the data set. However, it would be interesting for further research to compare the results of interacting GPCA with various clustering methods.

This paper deals with numerical data. Caussinus and Ruiz-Gazen (2006) consider the projection pursuit approach for categorical data and its connection with the search for a latent class structure. As a further step, it would certainly be worth considering a similar approach for symbolic data.

References

ART, D., GNANADESIKAN, R. and KETTENRING, J.R. (1982): Data-based metrics for cluster analysis. *Utilitas Mathematica, 21A, 75-99.*

BOCK, H.H. (1987): On the interface between cluster analysis, principal component analysis, and multidimensional scaling. In: H. Bozdogan and A.K. Gupta (Eds.): *Multivariate Statistical Modeling and Data Analysis.* D. Reidel Publishing Company, 17-34.

CAUSSINUS, H., FEKRI, M., HAKAM, S. and RUIZ-GAZEN, A. (2003a): A monitoring display of multivariate outliers. *Computational Statistics and Data Analysis 44, 237-252.*

CAUSSINUS, H., HAKAM, S. and RUIZ-GAZEN, A. (2003b): Projections révélatrices contrôlées: groupements et structures diverses. *Revue de Statistique Appliquée 51(1), 37-58.*

CAUSSINUS, H. and RUIZ-GAZEN, A. (1993): Projection pursuit and generalized principal component analyses. In: S. Morgenthaler, E. Ronchetti, W.A. Stahel (Eds.): *New directions in statistical data analysis and robustness.* Birkhauser Verlag, Basel Boston Berlin, 35-46.

CAUSSINUS, H. and RUIZ-GAZEN, A. (1995): Metrics for finding typical structures by means of principal component analysis. In: Y. Escoufier and C. Hayashi (Eds.): *Data Science and its Applications.* Academic Press, Tokyo, 177-192.

CAUSSINUS, H. and RUIZ-GAZEN, A. (2006): Projection pursuit approach for categorical data. In: M. Greenacre and J. Blasius (Eds.): *Multiple Correspondence Analysis and Related Methods.* Chapman and Hall/CRC, London, 405-418.

CHAE, S. S. and WARDE, W.D. (2006): Effect of using principal coordinates and principal components on retrieval of clusters. *Computational Statistics and Data Analysis 50, 1407-1417.*

CHAVENT, M., LACOMBLEZ, C. and PATOUILLE, B. (2001): Critère de Rand asymétrique. *Huitièmes rencontres de la Société Francophone de Classification*, Pointe à Pitre, 82-88.

COOK, D., CARAGEA, D. and HONAVAR, H. (2004): Visualization in classification problems. In: J. Antoch (Ed.): *Proceedings in Computational Statistics (COMPSTAT 2004)*, Springer, Berlin, 799-806.

DIDAY, E. et collaborateurs (1979): *Optimisation en classification automatique.* INRIA, Roquencourt.

FORINA, M., ARMANINO, C. LANTERI, S. and TISCORNIA, E. (1983): Classification of olive oils from their fatty acid composition. In: H. Martens and H. Russwurm Jr. (Eds.): *Food Research and Data Analysis.* Applied Science Publishers, London, 189-214.

GABRIEL, K.R. (1971): The biplot: graphical display of matrices with application to principal component analysis. *Biometrika 58 453-467.*

GABRIEL, K.R. (2002): Le biplot : outil d'exploration des données multidimensionnelles. *Journal de la Société Française de Statistique 143 (3-4), 5-55.*

GLOVER, D.M. and HOPKE, P.K. (1992): Exploration of multivariate chemical data by projection pursuit. *Chemometrics and Intelligent Laboratory Systems 16, 45-59.*

HARTIGAN, J. and WONG, M.A. (1979): A k-means clustering algorithm. *Applied Statistics, 28, 100-108.*

RAND, W.M. (1971): Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association 66 (336), 846-850.*

RUIZ-GAZEN, A. (1996): A very simple robust estimator of a dispersion matrix. *Computational Statistics and Data Analysis 21, 149-162.*

STUTE, W. and ZHU, L.X. (1995): Asymptotics of k-means clustering based on projection pursuit. *Sankhya 57, series A (3), 462-471.*

Locally Linear Regression and the Calibration Problem for Micro-Array Analysis

Antonio Ciampi[1], Benjamin Rich[1], Alina Dyachenko[1], Isadora Antoniano Villalobos[2], Carl Murie[3], and Robert Nadon[3,4]

[1] Epidemiology & Biostatistics, McGill University, Montreal, Québec, Canada
antonio.ciampi@mcgill.ca
[2] IIMAS-UNAM, Ciudad de México, México
[3] McGill University and Genome Quebec Innovation Centre, Montreal, Québec, Canada
[4] Department of Human Genetics, McGill University, Montreal, Québec, Canada

Abstract. We review the concept of locally linear regression and its relationship to Diday's *Nuées Dynamiques* and to tree-structured linear regression. We describe the calibration problem in microarray analysis and propose a Bayesian approach based on tree-structured linear regression. Using the proposed approach, we analyze a subset of a large data set from an Affymetrix microarray calibration experiment. In this example, a tree-structured regression model outperforms a multiple regression model. We calculated 95% Credible Intervals for a sample of the data, obtaining reasonably good results. Future research will consider and compare several other approaches to locally linear regression.

1 Introduction

In flexible modeling, the relationship between a variable y and a vector of other variables $x = (x_1, x_2, \ldots, x_p)$, is studied by a variety of tools, among which linear regression plays a central role. One of the great merits of linear regression is that it can be easily extended to provide even more flexible tools. An important example of possible extension is what we will call in this paper *locally linear regression.* Suppose we have a data set, $(Y^{(i)}, X^{(i)})$, $i = 1, 2, \ldots, N$, where Y and X represent, respectively, $N \times 1$ and $N \times p$ vectors of measurements of y, and x. If a linear model doesn't fit the data sufficiently well, it is natural to think of partitioning the data set into subsets, in the hope that a set of linear models, one for each set of the partition, describes the data better than a unique, global linear model. In the simplest case, if we study the relationship between y and a single variable x, we might consider partitioning the x axis so that on each portion of it a linear model seems reasonable. This simple idea is currently used in many software packages as the key step in the construction of smoothers, powerful tools which draw smooth lines (instead of straight lines) through scatter plots.

The partitioning can be done in many different ways, but the original idea, in its most general form, is due to Diday and collaborators (Charles

(1977); Diday (1979)) as an application of the general approach to clustering known as '*Nuées Dynamiques*', sometimes translated into English as 'dynamic clustering'. Independently, Späth (1979) proposed a similar approach, but outside the framework of '*nuées dynamiques*'. In both works, the data set is partitioned so that the global fit of the distinct regressions is optimal. This justifies the expression *cluster wise regression* which is generally used in the English literature to refer to this approach. The original French term was *regression typologique.*

Both Diday's and Späth works now appear as ground breakers and it is hardly surprising that they have remained relatively obscure for several years. It was the need for flexible tools in data mining that revamped interest in cluster wise regression. There is a good deal of recent literature that aims to reformulate the optimization problem of cluster wise regression and/or to propose new, more efficient algorithms for its solution. See for example Caporossi and Hansen (2005), and Mirkin (2005). Interestingly, Mirkin uses the term regression wise clustering instead of cluster wise regression.

The approach to locally linear regression proposed here is based on tree-growing. It starts from a more general formulation of the cluster wise regression problem, which, however, can easily be accommodated within the *nuées dynamiques* framework. We consider one dependent variable y and two vectors of independent variables x, and $z = (z_1, \ldots, z_m)$. While the relationship between y and x is considered of primary importance and assumed linear, the vector z is seen as background information that can affect y and the relationship between y and x. Notice that it is possible that some or all of the components of z are also components of x. In general, we will assume that there is an important association between x, z, and y. Suppose we have data in the form: $(Y^{(i)}, X^{(i)}, Z^{(i)})$, $i = 1, 2, \ldots, N$. Now, we can seek a partition of z space such that, if we fit a regression model of y on x on each subset, then the global fit of these local models is optimal. It is possible to develop optimization algorithms of the K-means type, hence a particular case of the *Nuées Dynamiques.* However we propose here a suboptimal, heuristic algorithm for constructing a tree with linear regressions of y on x at its leaves.

This work was motivated by an important application: the calibration problem in the analysis of microarray data. In section 2 we describe this problem. In section 3 we outline our tree-based approach to linear regression (also known as tree-structured regression). Section 4 is devoted to an example of calibration using our approach. In section 5, a discussion concludes the paper

2 Calibration in microarray analysis

The classical calibration problem concerns two measuring instruments for the same quantity. These instruments produce measure y and x, and it is assumed that x is considerably more expensive to obtain than y. A calibration experiments is the simultaneous measurement of y and x, performed in order to extract the information that permits to predict x from y. In fact, in future experiments, only y will be measured and x is to be inferred from y. There are well-known solution to the problem of predicting x from y, based on both classical (Draper and Smith, 1966) and Bayesian statistics (Hunter and Lamboy (1981)).

In the last ten years microarray technology has become an essential tool in genomic analysis. There are various types of microarray, but we will focus here only the Affymetrix array, one of the most popular. An Affymetrix microarray is a 2-dimensional physical array of 'spots' on a small square surface. Multiple identical copies of a 25 base pair oligonucleotide, called probes, are fixed to each spot; each spot is assigned a unique probe sequence from the genetic alphabet ACGT. A microarray is used to simultaneously perform tens of thousands of hybridization experiments, one at each spot. A sample containing fluorescently labeled mRNA is added to the array. The sample contains many different mRNA's that may reflect the state of gene expression within cells of a particular tissue. Hybridazation occurs when a probe, which is chosen to be reverse complimentary to a sequence that only occurs within one gene, binds to its matching mRNA in the sample.

In the typical microarray experiment, x is the log-concentration of messenger RNA (mRNA) and y is a simple transformation of the intensity of fluorescent luminance of a spot on the array. This luminance is proportional to the amount of hybridization occurring at the spot. It will depend not only on the concentration (x) of mRNA at the spot, but also on specific characteristics of the probe sequence. These characteristics are summarized by a vector of variables, which we will call z.

Microarray analysis generates a new type of calibration problem. Though we are still interested in inferring x from y, it is clear that y will depend on z as well as x, and this will be reflected in the inverse equation. While the first idea would be to assume linearity of y in z as well as in x, this, depending on the nature of z, may be an oversimplification. We propose a locally linear model with the partition defined by z. Thus, conditional on z, we will still assume linearity, but the effect of z on y may be highly non-linear. Clearly, locally linear regression may be a useful approach to consider.

The first step of our approach consists of building a reasonably good locally linear predictive model for y given x and z, assuming linear the relationship between y and x. We present here a RECPAM tree-structured regression of y on x, with the tree structure defined by z.

The second step consists in inverting the linear relationship between y and x locally on each of the tree leaves. Since in future experiments x will be unknown, while y and z will be known, a Bayesian approach will consider x as a parameter and develop inference on it accordingly (Hunter and Lamboy (1981)). We will need to impose a reasonable prior on x; this done, it is relatively straightforward to obtain an expression for the posterior distribution of x given y and z. In what follows we choose a non informative (improper) prior on x, and instead of calculating the theoretical expression, which would be of limited usefulness in our case, we will calculate posterior Credible Intervals (Bayesian equivalent of Confidence Intervals) by simulating the posterior distribution.

3 Tree-structured regression in RECPAM

The term RECursive Partition and Amalgamation (RECPAM) refers to a family of tree-growing algorithms. Given a data set $(Y^{(i)}, X^{(i)}, Z^{(i)})$, $i = 1, 2, \ldots, N$, as in the introduction, RECPAM may be used to construct a tree with splits defined by the z-variables and having at its leaves a linear regression equation of y on x. Therefore, when used this way, RECPAM is a tool to construct locally linear models.

The RECPAM construction consists of three steps. In the first step, a large tree is constructed recursively. The algorithm starts with the whole data set, which is represented as the root node of a tree (a dark circle in the diagram). All binary splits of the data set based on a single component of z are considered. If z_i, is a categorical predictor, then a split on z_i is generated by a binary question of the form: is '$z_i \in A$', where A is a subset of the levels of z_i. If z_i is continuous, then the split-generating binary question has the form: 'is [$z_i \leq a$]?'. For each split, the algorithm computes the Likelihood Ratio Statistic (LRS) comparing two separate regressions, one for each branch of the split, to the single regression model fitted on the whole data set. The split with the largest LRS is selected, and the data set is split accordingly into two subsets. In the tree diagram, the question defining the best split is represented under the root, from which two branches issue and point to two nodes (clear circles), one for each subset. This same search is carried out recursively on the nodes issuing from the root and on nodes issuing from other nodes (parents), until node size falls below a user-defined threshold. Then the nodes that cannot be split further are termed *terminal nodes* or *leaves* and are represented by square boxes.

The second step is called *pruning* and consists of building a sequence of sub-trees of the large tree, from which one is chosen, on the basis of the AIC or similar criteria, as our 'honest' tree.

In the third step, *amalgamation*, one compares leaves from different parents using the appropriate LRS, and then merges them starting from the pair with the smallest LRS. This also leads to a sequence of amalgamations, going from the large tree to the root. Again, AIC and/or BIC are used to select where to stop, i.e. which element of the amalgamation sequence.

The result of the RECPAM construction is an induction diagram with distinct linear regressions at its leaves. In the case of our calibration problems, such a model is useful to represent a linear relationship between y and x with

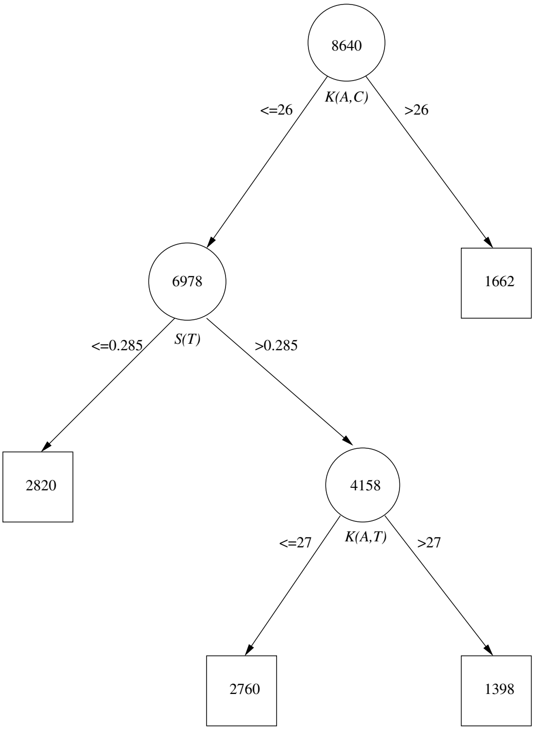


Fig. 1. RECPAM local regression tree.

coefficients varying according to the 'environmental conditions' described by the z variables.

4 An example: Affimetrix data

The data we analyze here come from a series of microarray experiments conducted by Affymetrix. The purpose was to identify probes that could best be used to predict concentration from intensity. Thus, the probes constitute the background within which the relationship between intensity y and log-concentration x is studied. Our goal is to assess, through a locally linear model, the effect of the background on this relationship, and construct a good model for calibration.

Utilizing a Latin square design, 85 genes clustered in 16 groups were assessed at each of 16 predetermined concentrations, ranging from 0 to 512 picomoles. The 16 experiments were replicated 4 times each, resulting in 64 intensity readings per probe.

Our data set contains data from a unique gene examined through 360 possibly overlapping probes. Its columns represent, log-concentrations (y), intensity (x), and 58 continuous features extracted from the probe sequences

Probe sequence	Intensity	True log-concentration	Posterior 95% C.I.
ACCCCTCGTGACCGTCCTTCCCTTG	13.445403	5.0000000	(2.104, 5.416)
CCCGTCTGGGACGCTCGTCTTTCTG	9.236014	1.0000000	(-1.246, 2.059)
CCAGCCGTAGGTCCCTGCGGAGGAG	8.518850	0.5849625	(-0.883, 2.433)
TCTTTCTGACGGGTGTCGCGGGGAA	8.066089	0.0000000	(-0.140, 3.176)
CCAGAACGAGAGCCCGACGGAGGTC	7.748193	1.5849625	(0.0383, 3.350)
AATTTACTCTCGAACCAGAACGAGA	9.062856	4.0000000	(1.896, 5.528)
AGGGAAGGTTCGTGCCAGTGTTACG	8.906891	1.5849625	(0.507, 4.116)
TTACGTCTTCCACTACTACTCTTGT	8.954196	0.0000000	(-2.789, 0.841)
ACGGAGGTCTACTGCGGGCACTGGT	8.174926	0.5849625	(-0.972, 2.627)
TCGAACCAGAACGAGAGCCCGACGG	12.721313	7.0000000	(6.642, 10.264)
AGGTTCGTGCCAGTGTTACGTCTTC	10.270295	2.584963	(1.504, 4.320)
TCTCGAACCAGAACGAGAGCCCGAC	8.880502	1.584963	(0.996, 3.825)
TCAGGAACGAACCAGCCGTAGGTCC	10.237210	3.584963	(2.323, 5.162)
AGGTTCGTGCCAGTGTTACGTCTTC	13.206251	7.000000	(6.267, 9.083)
AGGTCCCTGCGGAGGAGCGACACGG	7.581201	0.000000	(-1.706, 1.137)
CTACGAGTCCTACCCCTCGTGACCG	11.728048	2.5849625	(0.653, 3.777)
GATATTCCGACCAACTTTACAAGTG	13.147523	9.0000000	(6.153, 9.275)
CGGGCACTGGTGCCGTGTCTCCTCC	9.221587	0.0000000	(-1.088, 2.028)
CGGTATCGGTTTCATCTACTACTTC	8.684749	0.5849625	(-2.036, 1.093)
CGTAGGTGTGTCGTTTGGCCTGGGT	11.187971	2.5849625	(1.866, 5.001)

Table 1. Posterior 95% credible intervals of log-concentration for randomly selected values of log-intensity and feature vector z.

(z). The description of the features will appear elsewhere. It suffices here to say that each feature captures certain statistical properties of the distributions of the ACGT letters, e.g. Skewness (S) in the probes or of pairs of such distributions, e.g. Kolmogorov distance between the distribution of two letters (K). After having excluded concentrations less than 1 pm, each probe constituted 48 rows in our data matrix. We then randomly selected a subset of 8640 rows from a total of 17,280.

Our first step was to build a stepwise linear regression model to predict y from x and the features z. Besides log-concentration, 29 features were selected using the Bayesian model selection approach (Raftery (1995)). We obtained a multiple adjusted R^2 of 88.2%.

To improve the prediction, we used RECPAM to build a local regression tree, with a linear regression equation of y on x at its leaves. Figure 1 gives the tree structure. Finally, we constructed at each leaf of the tree a stepwise linear regression model of y on x and z as above. Table 1 gives the posterior credible intervals of log-concentration for randomly selected values of log-intensity and feature vector z (we selected at random 5 probes from each leaf).

5 Discussion

We have reviewed the concept of locally linear regression, which is close to Diday's *regression typologique*, emphasizing the role of tree-growing in the construction of such models from data. We have also analyzed a complex data set, constructing a tree-structured linear predictor with good properties. Finally, we have applied the Bayesian calibration approach to obtain a reasonable inverse prediction equation at each leaf of the tree.

Calibration in microarray experiments is a fundamental problem. While we obtain impressive results for a small portion of a large experiment, we are far from having solved the calibration problem, since we have not addressed the generalizability of our model to the totality of genes. Further work is necessary to adapt our approach to the full complexity of the Affymetrix data. This will require extending the tree-growing algorithm to add random effects describing genes.

Another important direction of research is to develop a *Nuées Dynamiques* approach to locally linear regression for our problem and compare it with the one presented in this paper.

References

CAPOROSSI, G. and HANSEN, P. (2005): Variable Neighborhood search for least squares clusterwise regression. *Les Cahiers du GERAD, G-2005-61*, Montreal.

CHARLES, C. (1977): Régression typologique et reconnaissance des formes, Thèse de doctorat 3ème cycle, Université Paris IX.

DIDAY, E. et al. (1979): *Optimization en Classification Automatique.* INRIA, Le Chesnay.

DRAPER, N.R. and SMITH, H. (1966): *Applied Regression Analysis.* Wiley, New York.

HUNTER, W.G. and LAMBOY, W.F. (1981): A Bayesian analysis of the linear calibration experiment. *Technometrics 23: 323-328.*

MIRKIN, B. (2005): *Clustering for Data Mining.* Chapman&Hall/CRC, London.

RAFTERY, A.E. (1995): Bayesian model selection in social research. In: P.V. Marsden (Ed.): *Sociological Methodology 1995.* Blackwells, Cambridge, Mass., 111–196.

Sanskrit Manuscript Comparison for Critical Edition and Classification*

Marc Csernel[1] and Patrice Bertrand[2]

[1] INRIA, Projet AXIS, Domaine de Voluceau, Rocquencourt BP 105 78153 Le Chesnay Cedex, France, *Marc.Csernel@inria.fr*

[2] GET - ENST Bretagne, 2 rue de la Châtaigneraie, 35576 Cesson-Sévigné Cedex, France, *Patrice.Bertrand@enst-bretagne.fr*

Abstract. A critical edition takes into account all the different known versions of the same text in order to show the differences related to any two distinct versions. The construction of a critical edition is a long and, sometimes, tedious work. In order to make it easier, softwares helping the philologist are nowadays available for the European languages. Because of its complex graphical characteristics, which involve computationally expensive solutions to problems occurring in text comparisons, such softwares do not yet exist for Sanskrit language.

This paper describes the Sanskrit characteristics that make text comparisons different, presents computationally feasible solutions for the elaboration of the computer assisted critical edition of Sanskrit texts, and provides, as a byproduct, a distance between two versions of the edited text.

1 Introduction

When a text is known through a great number of manuscripts that include non trivial differences, the critical edition looks often rather daunting for readers unfamiliar with the subject. The edition is then formed mainly by footnotes enlightening the differences between manuscripts, while the main text (the text of the edition) is rather short, sometimes a few lines in a page. Note that in either case, the main text is established by the editor through his own knowledge. More explicitly, the main text can be either a particular manuscript, or a "mean" text built according to some specific criteria chosen by the editor. Building a critical edition by comparing the texts two by two, especially if they are manuscripts, is a task which is certainly long and, sometimes, tedious. This is why, for a long time, computer programs have been helping philologists in their work, but most of them are dedicated to texts written in Latin (sometimes Greek) scripts. For example, the Institute for New Testament Textual Research (2006) provides an interactive critical edition of the gospels which have induced a considerable amount of studies. In this paper we focus on critical edition of manuscripts written in Sanskrit.

* This paper was supported by the ACI CNRS "histoire des savoirs" and the Asia IT & C contract 2004/091-775.

Sanskrit is an ancient Indo-European language mainly used as a liturgical language which enjoys nowadays, in the Indian subcontinent, a position similar to that of Latin during the 19th century in Europe. The texts we will have to deal with, are ancient, scientific, either mathematical or grammatical. Our approach will be presented and illustrated on paragraphs and sentences that are extracted from a collection of manuscripts of the "Benares gloss", or *kāśikāvṛtti* in Sanskrit (Kāśi is the name of Benares). The Benares gloss, which was written around the 7th century A.D., and is the most widespread, the most famous, and one of the most pedagogical comments of the notorious Pāṇini[1] grammar.

Pāṇini grammar is known as the first **generative** grammar and was written around the fifth century B.C. as a set of aphorisms. These aphorisms cannot been understood without the explanation provided by a comment such as the *kāśikāvṛtti.*

In what follows we will first describe the characteristics of Sanskrit that matter for text comparison algorithms as well as for the classification of the whole set of manuscripts. Notice that since some manuscripts have been damaged by mildew, insects, rodents ... they are not all complete. In particular, they do not include all chapters, generally around fifty different texts are available for comparison at the same time. We will also present briefly the textual features we use to identify and to quantify the differences between manuscripts of the same Sanskrit text. We will show that such a comparison requires to use a lemmatized[2] Sanskrit text as the main text. The revealed differences, which as a whole form the critical edition, provide all the information required to build distances between the manuscripts, and consequently, to build phylogenetic trees assessing filiations between these manuscripts. Finally, we will thus discuss the definition of a method of computation of faithful distances between any two Sanskrit texts, provided one of them is lemmatized.

2 How to compare Sanskrit manuscripts

2.1 The Sanskrit and its graphical characteristics

Sanskrit is written mostly using a script called Devanāgari that has a 48 letter alphabet, which can also be considered as a syllabary, because this alphabet reveals the pronunciation in the writing.

Due to a long English presence in India, a tradition of writing Sanskrit with the Latin alphabet (a transliteration) has been established for a long

[1] The polysemy of the word Pāṇini, in Europe, is a surprise for Indian scholars.

[2] The lematization is, roughly speaking, a morpho-linguistic process which makes each word appears under its base form, generally followed by a suffix indicating its inflected form. For example *walking*, consists of the base form *walk* followed by the suffix *ing* which indicates the continuous form. After a lematization each word will, at least, appear as separated from the others.

time. Sanskrit transliteration was originally carried out to be used with traditional printing. It was adapted for computers by Frans Velthuis (1991), more specifically for a TEX transliteration. According to the Velthuis transliteration scheme, each Sanskrit letter is written using one, two or three Latin letters.

A long time ago Sanskrit was written with the Brāhmī script, but nowadays Devanāgari is the most common script. Other scripts may be used, such as Bengali or Telugu in south India. In Europe, an equivalent (but fictive) situation would be using either Latin, Cyrillic, or Greek alphabets to write Latin.

In ancient manuscripts, Sanskrit is written without blanks, and from our point of view, this is an important graphical specificity, because it increases greatly the complexity of text comparison algorithms. One may remark that Sanskrit is not the only language where blanks are missing in the text, Roman epigraphies and European Middle Age manuscripts are also good examples.

2.2 The different comparison methods

Comparing manuscripts can be achieved in two ways:

- When building a critical edition, the notion of word is central, and an absolute precision is required. For example, the critical edition must indicate that the word `gurave` is replaced by the word `ga.ne"saaya` in manuscripts 3 and 19, and that the word `"srii` is omitted in manuscripts 5, 8, 12, 19.
- When establishing some filiation relations between the manuscripts (or a classification between them), the notion of word can be either ignored, or taken into account. The only required information, is the one needed to build a distance between texts. Texts can be considered either as letter sequences, or as a sequence of words.

Considering each text as a letter sequence, Le Pouliquen (2007) proposed an approach that determines the so called *"Stemma codicum"* (in other words, filiation trees) of a set of Sanskrit manuscripts. The first step consists in the construction of a distance according to the Gale and Church (1993) algorithm. This algorithm was first developed to provide sentence alignments in a multilingual corpus, for example a text in German and its English translation. It uses a statistical method based on the sentence length. Gale and Church showed that the correlation between two sentence lengths follows a normal distribution. Once the distance is computed, a phylogenic tree is built using the N-J algorithm (Saitou and Nei (1987)).

On the other hand, each critical edition deals with the notion of word. Since electronic Sanskrit lexicons such as the one built by Huet (2006) do not cope with grammatical texts, we must be able to identify each Sanskrit word within a character string, without the help of either a lexicon or blanks to separate the words.

The solution comes from the lemmatization of one of the two texts: the text of the edition. The lemmatized text which is prepared *by hand* by the editor is called the *padapāṭha*, this name coming from a special kind of recitation where the words are well separated.

From this lemmatized text, we will build the text of the edition, it is called *saṃhitapāṭha* according to the name given to an oral mode of recitation where words are pronounced fluently one after the other. The transformation of the *padapāṭha* into the *saṃhitapāṭha* is not straightforward because the Sanskrit writing reflects the pronunciation. A text with separators (such as blanks) between words, can look rather different (the letter string can change greatly) from a text where no separators occur.

The typed text corresponding to each manuscript is called *mātṛkāpāṭha.* Each *mātṛkāpāṭha* contains the text of the manuscript and some annotation commands. These commands allow some elements visible on the manuscript, but which are not part of the text, such as ink color, margin notes to be taken into account. They provide a kind of meta-information.

3 Comparing the *padapāṭha* with a manuscript

In this section we aim to compare the *padapāṭha* and a *mātṛkāpāṭha*, this comparison cannot be simple.

- As previously seen, the *padapāṭha*, must be transformed into a virtual *saṃhitapāṭha* before being compared with the *mātṛkāpāṭha.*
- The comparison must be based on common words, so words must be identified in the *mātṛkāpāṭha* but with no lexicon available.
- Since only one text is lemmatized, if the texts differ in a simple way, it would be easy to stress the words where differences occur. But as soon as the differences include a lot of characters, the algorithmic complexity may grow quickly, and if a new part of text is inserted in a manuscript, no lemmatization will be possible and the software will find it difficult to "take a decision".

To cope with these difficulties, we propose the following two step procedure:

- First step: A twofold lexical preprocessing. On the one hand the *padapāṭha* is transformed into a virtual *saṃhitapāṭha.* The transformation consists in removing all the separations between the words and then in applying some morpho-phonology rules called *sandhi. Sandhi* are perfectly defined by the Sanskrit grammar. Not every language has *sandhis*, but French does. A good example in French could be as if "Les enfants" (= the children) was written "Lezenfants". In English, an illustration is provided when comparing the decomposed form "syn + pathy" with its usual form "sympathy". This virtual *saṃhitapāṭha* will form the text of the edition, and will be compared to the *mātṛkāpāṭha.* On the other hand, the lexical treatment of a *mātṛkāpāṭha* consists mainly in keeping the collation commands out of the texts to be compared.

- Second step: Alignment of a *mātṛkāpāṭha* and the virtual *saṃhitapāṭha*. The Longest Common Subsequence algorithm is applied to these two texts. The aim is to identify, as precisely as possible, the words in the *mātṛkāpāṭha*, using the *padapāṭha* as a pattern. Once the words of the *mātṛkāpāṭha* have being determined, we can see those that have been added, replaced or suppressed.

The comparison is done paragraph by paragraph, the different paragraphs being perfectly determined in each manuscript by the scholar who collated it. In a first stage, the comparison is performed on the basis of a Longest Common Subsequence. Each of the obtained alignments, together with the lemmatized *padapāṭha*, suggests an identification of the words of the *mātṛkāpāṭha*. However, due to the Sanskrit specificities, the answer is not straightforward, and a consistent amount of the original part of this work concerns this identification process. Surprisingly the different rules used for this determination are not based on any Sanskrit knowledge, but on common sense. The result of the application of these rules has been validated by Sanskrit philologists.

We remark that the kind of results expected for the construction of a critical edition (which words have been added, suppressed, replaced in the manuscript) is similar to the formulation of an edit distance, but in terms of *words*. The results we obtain from the construction of the critical edition can be transformed into a distance between the manuscripts.

3.1 The Longest Common Subsequence algorithm

The Longest Common Subsequence (LCS) algorithm is a well-known algorithm[3] used in string sequence comparison. The goal of this algorithm is to provide a longest common substring between two character strings. More precisely, given a sequence $X = \langle x_1, x_2, ..., x_m \rangle$, another sequence $Z = \langle z_1, z_2, ..., z_n \rangle$ is a subsequence of X if there is a strictly growing set of indices $\langle i_1, i_2, ...i_k \rangle$ such that $z_j = x_{i_j}$ for each $j \in [1 : k]$. For example, if $X = \langle A, B, C, D, A, B, C \rangle$ then $Z = \langle B, D, B, C \rangle$ is a subsequence of X.

A common subsequence to sequences X and Y is a subsequence of both X and Y. Generally there is more than one LCS. Once the computation of a LCS is achieved, one can compute an alignment of the two sequences. Most of the time one considers any of the alignments as equivalent. It will not be the case here, because the comparison should be based on words, not only on characters.

In the following, we describe the LCS algorithm giving an example, then we explain why the result cannot be a solution in our case. We will see how we can improve this result by using the *padapāṭha*. Finally we will see that the LCS algorithm cannot overcome all the difficulties, because it works in terms of characters whereas a critical edition is built in terms of words.

[3] the Unix *diff* command is based on this algorithm.

Computing the LCS is equivalent to the computation of an Edit distance between two character strings. An Edit distance between sequences X and Y is the minimum number of operations such as suppression, addition and replacement (in term of characters) needed to change the sequence X into Y. An Edit distance that is computed without the replacement operation, is sometimes called *LCS distance* by some authors. This function is a kind of dual of the length of a LCS between X and Y(see, for more details, chapter 7 of Crochemore *et al.* (2001)). The length of a LCS between X and Y will be denoted *lcs(X,Y)* or simply *lcs* if there is no ambiguity. Edit distance and *lcs* can be computed efficiently by the dynamic programming algorithm.

Example 1. Let us compute the *lcs* between two (simple) Sanskrit texts: X = `yamaan`, Y = `yamin`. Note that according to the Velthuis transliteration **aa** is a single letter: **long a**.

		y	a	m	i	m
	0	0	0	0	0	0
y	0	1	1	1	1	1
a	0	1	2	2	2	2
m	0	1	2	3	3	3
aa	0	1	2	3	3	3
m	0	1	2	3	3	4

Table 1. Computation of a LCS matrix T.

The value of the *lcs*, here 4, is displayed in the bottom right corner of the matrix T. The matrix is initialised to zero, and each score is computed by:

$$T[i,j] = \begin{cases} T[i-1,j-1]+1 & \text{if } X[i] = Y[j], \\ max\{T[i-1,j],T[i,j-1]\} & \text{otherwise.} \end{cases}$$

The score $T[i,j]$ gives the value of *lcs* between subsequences $X[i]$ and $Y[j]$, these subsequences being defined as the first i letters of X and j letters of Y respectively.

Each score $T[i,j]$ can be computed using some adjacent scores as shown in the previous formula. The complexity of the matrix computation is obviously in $O(|X||Y|)$. In this example, the LCS matrix generates exactly the two following symmetrical alignments.

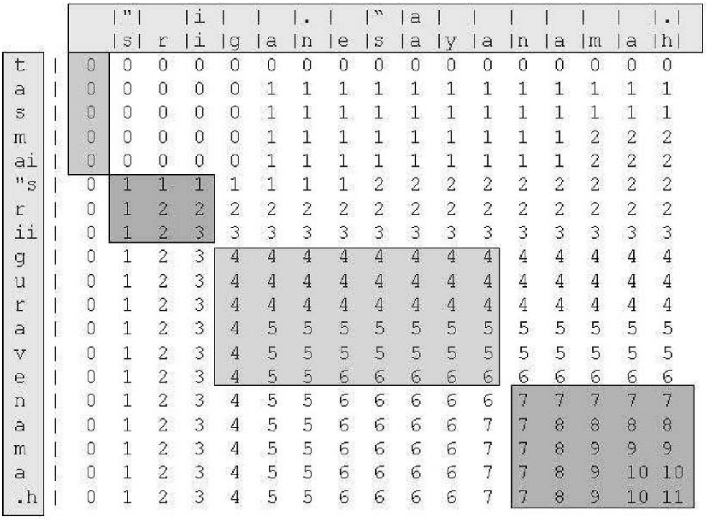
y	a	m	i	_	m
y	a	m	_	aa	m

y	a	m	_	i	m
y	a	m	aa	_	m

The alignment can be read in the following way: when letters are present up and down, they belongs to the LCS When a letter l is present with an opposite '-', then l can be considered either added in the line where it appears, or suppressed from the line where the opposite '-' is present.

Example 2. The comparison between two short sentences as shown in Table 2 describes the way we proceed and what kind of result can be expected.

The sentences to compare in this example are `tasmai "srii_gurave namas` and `"sriiga.ne"saaya nama.h`. Note that the first sentence belongs to the *padapāṭha*, and that the character '_' (underscore) is a lemmatization sign.



		"		i			.		"	a							.
		s	r	i	g	a	n	e	s	a	y	a	n	a	m	a	h
t	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
a	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
s	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
m	0	0	0	0	0	1	1	1	1	1	1	1	1	1	2	2	2
ai	0	0	0	0	0	1	1	1	1	1	1	1	1	1	2	2	2
"s	0	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2
r	0	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
ii	0	1	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3
g	0	1	2	3	4	4	4	4	4	4	4	4	4	4	4	4	4
u	0	1	2	3	4	4	4	4	4	4	4	4	4	4	4	4	4
r	0	1	2	3	4	4	4	4	4	4	4	4	4	4	4	4	4
a	0	1	2	3	4	5	5	5	5	5	5	5	5	5	5	5	5
v	0	1	2	3	4	5	5	5	5	5	5	5	5	5	5	5	5
e	0	1	2	3	4	5	5	6	6	6	6	6	6	6	6	6	6
n	0	1	2	3	4	5	5	6	6	6	6	6	7	7	7	7	7
a	0	1	2	3	4	5	5	6	6	6	6	7	7	8	8	8	8
m	0	1	2	3	4	5	5	6	6	6	6	7	7	8	9	9	9
a	0	1	2	3	4	5	5	6	6	6	6	7	7	8	9	10	10
.h	0	1	2	3	4	5	5	6	6	6	6	7	7	8	9	10	11

Table 2. A second example.

The rectangle in Table 2 contains all the possible alignments, one of them being the alignment in Table 3. We can see that the word *tasmai* is missing in the *mātṛkāpāṭha*, that the word *srii* is present in both sentences, that *gurave* is replaced by *ga.ne”saaya*, and that the word *nama.h* is present in both sentences but under two different aspects: *nama.h* and *namas*. The rule that states the equivalence between character ”.h” and character ”s” is one of the *sandhis*. The following alignment is one of the possible results, the separation between words of the *padapāṭha* being represented by double vertical lines.

t	a	s	m	ai	”s	r	ii	g	u	r	a	_	v	e	_	_	_	_	n	a	m	a	s
_	_	_	_	_	”s	r	ii	g	_	_	a	.n	_	e	”s	aa	y	a	n	a	m	a	.h

Table 3. The correponding alignment.

3.2 Sailing trough the LCS matrix

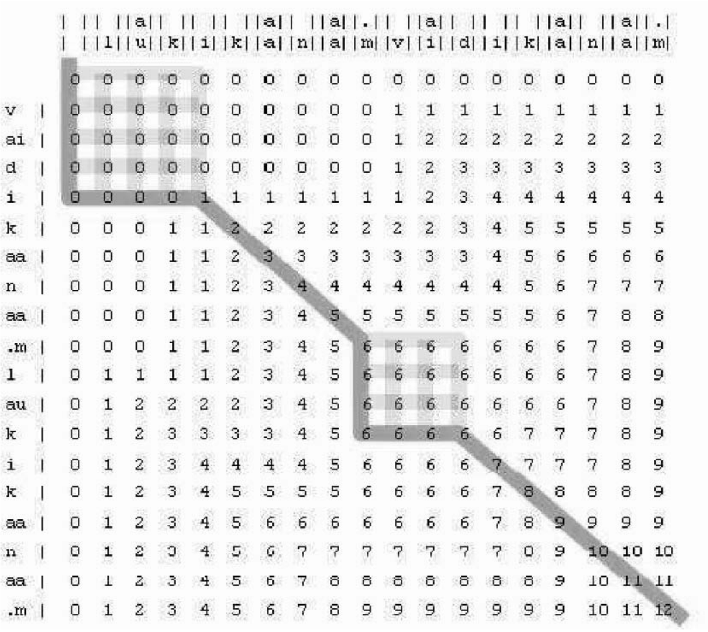
As already mentioned in the section 3.1, our approach uses the LCS matrix, but only to find which words are different.

Note first that the matrix provides alignments in the following way.

1) if $T[i,j] < T[i+1,j+1]$ then move (down and right) from $T[i,j]$ to $T[i+1,j+1]$ and in this case, the score which is increased by 1, indicates that a (common) letter is added to the alignment;

2) otherwise, move either vertically down one row or horizontally right one column, without increasing the current score. In this case, each vertical move corresponds to the addition of a $\binom{\text{X}}{\text{-}}$ to the alignment, and each horizontal one to the addition of a $\binom{\text{-}}{\text{X}}$.

Table 4 presents the alignments provided by the LCS algorithm. The dark grey line depicts the chosen alignment, and the clear grey lines represent other alignments also provided by LCS algorithm. The sequence X belonging to the *padapāṭha*, the alignment is selected in order to maximize consecutive letters belonging to X. This choice reduces the risk that two parts of the same word in the *padapāṭha* be identified with two different subsequences of the *mātṛkāpāṭha*.



		l	au	k	i	k	aa	n	aa	.m	v	ai	d	i	k	aa	n	aa	.m
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
v	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
ai	0	0	0	0	0	0	0	0	0	0	1	2	2	2	2	2	2	2	2
d	0	0	0	0	0	0	0	0	0	0	1	2	3	3	3	3	3	3	3
i	0	0	0	0	1	1	1	1	1	1	1	2	3	4	4	4	4	4	4
k	0	0	0	1	1	2	2	2	2	2	2	2	3	4	5	5	5	5	5
aa	0	0	0	1	1	2	3	3	3	3	3	3	3	4	5	6	6	6	6
n	0	0	0	1	1	2	3	4	4	4	4	4	4	4	5	6	7	7	7
aa	0	0	0	1	1	2	3	4	5	5	5	5	5	5	5	6	7	8	8
.m	0	0	0	1	1	2	3	4	5	6	6	6	6	6	6	6	7	8	9
l	0	1	1	1	1	2	3	4	5	6	6	6	6	6	6	6	7	8	9
au	0	1	2	2	2	2	3	4	5	6	6	6	6	6	6	6	7	8	9
k	0	1	2	3	3	3	3	4	5	6	6	6	6	6	7	7	7	8	9
i	0	1	2	3	4	4	4	4	5	6	6	6	6	7	7	7	7	8	9
k	0	1	2	3	4	5	5	5	5	6	6	6	6	7	8	8	8	8	9
aa	0	1	2	3	4	5	6	6	6	6	6	6	6	7	8	9	9	9	9
n	0	1	2	3	4	5	6	7	7	7	7	7	7	7	0	9	10	10	10
aa	0	1	2	3	4	5	6	7	8	8	8	8	8	8	8	9	10	11	11
.m	0	1	2	3	4	5	6	7	8	9	9	9	9	9	9	9	10	11	12

Table 4. The different alignments within the matrix.

The chosen alignment corresponding to the dark grey line is:

v	ai	d	i	_	_	_	_	k	aa	n	aa	.m	l	au	k	_	_	_	i	k	aa	n	aa	.m
_	_	_	_	l	au	k	i	k	aa	n	aa	.m	_	_	_	v	ai	d	i	k	aa	n	aa	.m

We may remark that when the possible alignments form a square the number of possible alignments grows very quickly. If N is the size of the square, the number of different alignments generated by each square is $\binom{2\text{N}}{\text{N}}$. To provide a good idea of the possible number of paths, if we have in a matrix which contains two ten by ten squares we got approximately $39 * 10^9$ different possible alignments. This number expresses how complicated the comparison of Sanskrit texts is, and excludes any method that requires all the solutions generated by LCS algorithm to be examined.

3.3 Local improvement of the initial LCS alignment

The identification of words in the *mātṛkāpāṭha*, as implicitly defined from the previous alignments, is not completely satisfactory. Indeed the maximisation of *lcs* cannot satisfy our purpose, because the value of *lcs* is related only to the notion of character, whereas our aim is to compare the texts word by word.

Once the alignment is obtained, the words of the *mātṛkāpāṭha* are not really identified. To improve this alignment we propose a procedure which consists of a local change of the alignment to satisfy the following two rules:

(1) Two words cannot be considered as similar if they do not share at least 50% of their characters (very short words must be considered apart).
(2) Considering that words can be suppressed, added, or replaced in the *mātṛkāpāṭha*, the desired alignment has to minimize the number of these operations.

Notice that the second rules matches exactly the definition of the edit distance, but in terms of words instead of characters as is usually the case. The results provided by these two rules were approved by the philologist in charge of the Sanskrit critical edition. To illustrate our approach let us compare the following two texts:

padapāṭha : upadis̈yate mahaa .n
mātṛkāpāṭha : upadi.syata.n

The LCS algorithm provides an alignment with a *lcs* of 10 that does not fulfil rule (1).

¨ u p a d i s _ y a t e	a m a h a	. n
. u p a d i _ s y a t _	_ a _ _	. n

This involves the following conclusions:

- The word *upadis̈yate* is substituted by *upadi.syat*
- The word *mahaa* is substituted by *a*

Next alignment it is not optimal for the *lcs* criterion, because its *lcs* is only 9, but is preferable because rule (1) is satisfied:

¨ u p a d i s _ y a t e _	a m a h a	. n
. u p a d i _ s y a t _ a	_ _ _ _	. n

- the word *upadiš̈yate* is substituted by *upadi.syata*
- the word *mahaa* is missed

It appears that the improvement of the initial alignment consists of asserting that the word *maha* is missing instead of stating that the word *maha* is replaced by *a*.

4 Conclusion

Recent advances in information technology have been so great that computer science has almost becomes essential for the studies of ancient manuscripts: back-up and electronic transmission, interactive critical edition, computer built phylogenetic trees, etc. In this paper we have proposed a method for comparing two versions of the same Sanskrit text. The alignment provided by the LCS algorithm between the two texts, considered as a sequence of characters, is not always sufficient, but provides a good initialisation for further processing that considers each of the two texts as sequences of words.

The critical edition provided such improved alignments has been submitted to philologists and has been approved in its essential part. Nevertheless a more intense use of the software should enable to improve and justify the setting of our empirical approach.

However, the absence of a Sanskrit lexicon constitutes a limit to our approach: in case of addition of long sentences within a manuscript, it is impossible to detect words that are added, we can only consider the addition in terms of sequence of characters.

References

CROCHEMORE, M., HANCART, C. and LECROQ, T. (2001): *Algorithmique du texte.* Vuibert, Paris.

GALE, W.A. and Church, K.W. (1993): A program for aligning sentences in bilingual corpora. *Computational Linguistics 19(3), 75–102.*

HUET, G. (2006): *Héritage du Sanskrit : Dictionnaire Français-Sanskrit.* http://sanskrit.inria.fr/Dico.pdf.

LE POULIQUEN, M. (2007): Filiation de manuscrits Sanskrit et arbres phylogénétiques. *Submitted to Mathématiques & Sciences Humaines.*

INSTITUTE FOR NEW TESTAMENT TEXTUAL RESEARCH (2006): *Digital Nestle-Aland.* Münster University.
http://nestlealand.uni-muenster.de/index.html.

SAITOU, N. and NEI, M. (1987): The neighbour-joining method: a new method for reconstructing phylogenetic trees. *Molecular Biology Evolution 4, 406–425.*

VELTHUIS, F. (1991): *Devanagari for TEX, Version 1.2, User Manual*, University of Groningen.

Divided Switzerland

Yadolah Dodge[1], Gérard Geiser[2], and Valentin Rousson[3]

[1] Statistical Institute, University of Neuchâtel
Pierre-à-Mazel 7, 2000 Neuchâtel, Switzerland, *yadolah.dodge@unine.ch*
[2] Neuchâtel Cantonal Statistical Office
Le Château, 2001 Neuchâtel, Switzerland, *gerard.geiser@ne.ch*
[3] Department of Biostatistics, University of Zürich
Hirschengraben 84, 8001 Zürich, Switzerland, *rousson@ifspm.uzh.ch*

Abstract. On the 6th of December, 1992, the Swiss population voted against the "Adhesion of Switzerland to the European Economic Area". Swiss German cantons, except Basel-Stadt and Basel-Land, voted against, and all French speaking cantons voted in favour of adhesion. Shocked by this outcome, the media, the politicians, and the population itself took this date as the beginning of the divided Switzerland. The purpose of this article is to show that what happened on that day was not a new phenomenon but was in line with more than a century of votations.

1 Introduction

Switzerland is a small but diverse country. Although it has an area of only about 41'000 km^2 and a little more than 7 million inhabitants, four very different official languages are spoken and Catholics and Protestants are present with similar proportions. Its geographical situation is particular too. Switzerland lies at the heart of Europe between countries with influential cultures such as France, Germany, Austria and Italy. Consequently, it is a country containing many different mentalities and it is divided in 26 cantons (or half-cantons).

Swiss direct democracy gives the people (and its government) various ways of expressing their opinions. The most frequent procedure used is the compulsory referendum which concerns mainly constitutional amendments. Another one is the popular initiative. For the acceptance of a referendum or a popular initiative, the majority of the people and the majority of the cantons must stand behind. Thus, all the cantons have equal weight (importance) in the final decision, regardless of their population sizes. For example, a canton like Zürich which in 1990 had more than one million inhabitants has the same weight as Glarus with less than 40'000. According to the constitution, this ensures representation of the minorities (small cantons).

A major problem with such a federal system of voting is cultural differences such as language, mentality and traditions. In particular there are only 6 French cantons whereas there are 16 Swiss-German ones (13 cantons and 6 half-cantons). If these two groups of cantons do not agree about an object, there is no doubt about the final result. The only Italian speaking canton is still more minoritary. The famous vote of the 6th of December, 1992, about the “Adhesion of Switzerland to the European Economic Area” is a typical example of such a situation. A real linguistic cleavage was observed on this occasion, and the media, the politicians and the population believed this to be a new phenomenon.

The primary aim of this study is to show how official statistics can be useful in the analysis of some cultural phenomena in a given country. The official statistics are rarely used for decision making purposes. They are usually summarized in tables, percentages or graphical displays such as pies and bar charts to be presented to the general public. In fact official statistics are one of the best sources of information to understand the political, social, economical or cultural behaviors of a nation when combined with some simple statistical techniques. More importantly, an attempt has been made to “picture the mass of data” in a constructive way. The secondary aim of the article is to show that the linguistic cleavage has always existed in Switzerland. To achieve this, we took into consideration all the voting results from 1866 to 1998 and used a simple multivariate statistical method, namely principal components analysis.

This article is organized as follows. The data are presented in Section 2. In Section 3, the main aspects of principal component analysis are recalled. The analyses of the data for the three periods considered are given in Section 4. In Section 5, an attempt is made to compare the voting results with other variables that describe Switzerland. Finally, some conclusions are drawn in Section 6.

2 The data

The results of all federal votes from 1866 to 1998 are the basis of our study (Bundesblatt, 1866, 1872, 1874–1876, 1878–1880, 1882, 1884-1885, 1887, 1889-1898, 1900, 1902–1903, 1905–1908, 1912–1915, 1918–1935, 1937–1939, 1941–1942, 1944–1998). These include the compulsory referendums, the popular initiatives, as well as the optional referendums (although the latter do not need the double majority to be accepted). The total number of topics voted on is 405. For each votation and each canton, the percentage of “yes” has been recorded. The blank and the nonvalid bulletins have been excluded since they were in negligible quantities. A few examples of the data are given in Table 1. We have divided the votes in three periods, and this for several reasons. First, we wanted to observe if the voting behavior has changed with time. Second, we had to consider the creation of the canton Jura in 1978, and third, we wanted to see if the results of the votes since the 6th of December, 1992, were different. The three periods are:

1. From 1866 to 1978 (256 votations),
2. From 1979 (date of the entry of the canton of Jura into the Confederation) to December 6, 1992 (96 votations),

	14.01.1866	14.01.1866	...	25.10.1908	...	14.03.1948	...	06.12.1992	...
	vote #1	vote #2		vote #67		vote #143		vote #352	
Cantons	% of yes	% of yes		% of yes		% of yes		% of yes	
Zürich (ZH)	92.71	92.94		93.63		33.66		48.48	
Bern (BE)	38.59	37.19		83.21		40.55		47.59	
Luzern (LU)	21.76	19.20		92.79		32.74		39.31	
Uri (UR)	10.76	16.22		58.33		25.18		25.13	
Schwyz (SZ)	23.06	25.20		59.62		31.79		26.69	
Obwalden (OW)	73.74	71.37		79.97		37.83		28.20	
Nidwalden (NW)	20.64	15.68		77.08		37.42		33.86	
Glarus (GL)	78.66	65.48		89.23		31.88		31.95	
Zug (ZG)	12.27	14.09	...	85.97	...	33.80	...	43.83	...
Fribourg (FR)	21.27	46.50		83.76		54.41		64.89	
Solothurn (SO)	71.58	71.46		90.72		31.21		42.59	
Basel-Stadt (BS)	53.02	53.97		97.66		18.36		55.43	
Basel-Land (BL)	58.45	58.82		85.45		30.40		53.18	
Shaffhausen (SH)	48.20	47.00		92.42		47.58		38.51	
Appenzell-AR (AR)	41.51	40.38		82.92		18.12		36.73	
Appenzell-IR (AI)	4.79	1.59		47.78		29.80		29.05	
St.Gallen (SG)	26.32	20.22		75.43		32.36		38.44	
Graubünden (GR)	11.05	8.12	...	72.48	...	43.80	...	32.44	...
Aargau (AG)	57.39	56.15		78.76		33.34		39.94	
Thurgau (TG)	77.27	77.73		81.70		43.14		35.96	
Ticino (TI)	66.87	78.45		73.61		43.99		38.46	
Vaud (VD)	14.19	10.90		90.09		40.93		78.31	
Valais (VS)	14.91	13.30		79.87		42.30		55.84	
Neuchâtel (NE)	83.44	80.76		89.92		23.06		79.96	
Genève (GE)	75.71	69.05		98.65		51.03		78.14	
Jura (JU)							...	77.15	...

Table 1. A subset of the Swiss votation data used in our study (source: Bundesblatt, 1866–1998).

3. From 1993 to June 7, 1998 (53 votations).

The 26 cantons are the units of the present analysis. One might think that a large canton is too heterogeneous to serve as an interesting unit. However, according to Joye (1987) if one wishes to observe extensive cultural divisions like the linguistic, a canton is a good unit because it is most of the time a geographical area well recognized by its inhabitants. Other analyses based on smaller units such as the communities within cantons are certainly possible and will be the subject of further investigation.

3 Principal Components Analysis (PCA)

This well-known methodology was originally proposed by K. Pearson in 1901 as a means of fitting planes by orthogonal least squares, and was developed by Hotelling in 1933 for the particular purpose of analyzing correlation structures.

A sample of p measurements $\mathbf{X}_1, \cdots, \mathbf{X}_p$ taken on n individuals can be represented by a matrix $\mathbf{X}$ of n rows and p columns (an x_{ij} element of this matrix being the j^{th} measurement on the i^{th} individual) or by a cloud of n points in a p-dimensional space, which is hard to visualize if p is greater than 2 or 3. It is therefore difficult to summarize such a sample using elementary descriptive statistics techniques and to get a global idea of what the data contain. Principal components analysis allows to deal with this difficulty by representing a p-dimensional cloud of points in a well chosen subspace of dimension smaller than p, for example in a 2-dimensional subspace. The idea is to project the n individuals in a subspace in which the distances between the (projected) individuals are the largest possible. The optimal 2-dimensional subspace hence obtained is called the principal plane of the sample in question, and the axes that generate it are the first two principal components. The procedure for a principal components analysis is as follows:

1. Standardize the p variables $\mathbf{X}_j$, i.e. replace the initial data matrix $\mathbf{X}$ by the matrix $\mathbf{Y}$ with elements y_{ij} such that $y_{ij} = (x_{ij} - \overline{x}_j)/s_j$, where $\overline{x}_j$ and s_j are the estimated mean and standard deviation of the variable $\mathbf{X}_j$ (for $j = 1, \cdots, p$).
2. Compute $\mathbf{C} = \mathbf{Y}'\mathbf{Y}/(n-1)$. This is the estimated correlation matrix of the variables $\mathbf{X}_1, \cdots, \mathbf{X}_p$.
3. Find the eigenvalues $\lambda_1, \cdots, \lambda_p$ and the associated eigenvectors $e_1, \cdots, e_p$ of $\mathbf{C}$. Order them so that λ_1 is the largest eigenvalue and λ_p the smallest one (they are all positive). Retain the first two eigenvectors e_1 and e_2.
4. Calculate the variables $\mathbf{Z}_1$ and $\mathbf{Z}_2$ as follows:

$$\mathbf{Z}_1 = e_{11}\mathbf{Y}_{1+}e_{21}\mathbf{Y}_2 + \cdots + e_{p1}\mathbf{Y}_p$$

$$\mathbf{Z}_2 = e_{12}\mathbf{Y}_{1+}e_{22}\mathbf{Y}_2 + \cdots + e_{p2}\mathbf{Y}_p$$

   where e_{ij} is the i^{th} coordinate of the j^{th} eigenvector. These linear combinations of the p initial variables are the first two principal components that we are searching for.

Thus, one has reduced the number of dimensions from p $(\mathbf{X}_1, \mathbf{X}_2, \cdots, \mathbf{X}_p)$ to 2 ($\mathbf{Z}_1$ and $\mathbf{Z}_2$), and one can now visualize the n individuals in the principal plane generated by $\mathbf{Z}_1$ and $\mathbf{Z}_2$. If one desires to add a third dimension, one can consider the third principal component, i.e. the linear combination of the original variables defined by the third eigenvector e_3, and similarly for further dimensions. Recall also that the eigenvalue λ_i associated with the principal component $\mathbf{Z}_i$ is the variance of the n individuals projected on $\mathbf{Z}_i$, while the ratio λ_i/p is the percentage of total variance of the n individuals represented (preserved) by the principal component $\mathbf{Z}_i$. For more details, see for example chapters in the books of Diday et al. (1982), Manly (1986) or Jolliffe (2002).

4 Data analysis

We performed a principal components analysis for each period. The Swiss cantons are the n individuals (or units), and the voting results (the percentages of “yes”) are the p variables. For the first period we have 25 individuals in a 256-dimensional space, for the second period we have 26 individuals (with the new canton of Jura) in a 95-dimensional space, and for the third period we have 26 individuals in a 53-dimensional space. Actually, as there are more variables than individuals, the real dimension of the cloud of points is $(n-1)$ (that is 24 or 25 according to the period) in the same way that 2 points in a 3-dimensional space lie on a line (a one-dimensional subspace).

As explained in Section 3, PCA gives an approximate display in two dimensions of a cloud of points situated in a 24- or in a 25-dimensional space. Figures given in next section represent the Swiss cantons in principal planes. The horizontal axis represents the first principal component, and the vertical axis the second one.

These principal planes give us an idea of voting similarities among the different cantons. If two cantons are close to each other in such a plane, it means that they voted similarly, and if they are distant from each other, they voted differently (at least if the principal plane gives a good approximation of the real situation, that is if λ_1 and λ_2 are high). Note also that cantons situated near the origin of the graph were generally close to the Swiss mean (their opinions were often in line with the majority).

4.1 Analysis of the period 1866-1978

Figure 1 represents the Swiss cantons in the principal plane for the period 1866-1978 (the abbreviations for the cantons are given in Table 1). The first axis accounted for 36% of the total variance and the second axis for 17%. This graph provides us with a summary picture of more than a century of votations. Observe for example that the cantons of Basel-Stadt (at the extreme top right of the graph) and Appenzell-IR (at the opposite side) voted very differently from each other, while cantons like Zug or Graubünden were the closest to the Swiss mean.

On the left side of Figure 1 we found the small and rural cantons like Appenzell-IR, Obwalden, Nidwalden, Uri and Schwyz, while at the opposite side, we found cantons with big cities like Basel-Stadt, Zürich and Genève. Not surprisingly, cantons with small population densities voted in a different way than those with higher population densities. Another factor correlated with the first axis was religion. Cantons on the right side of the graph were rather protestant, cantons on the left side were rather catholic and cantons in the middle of the graph were often semi-protestant and semi-catholic. The role played by religion in the voting results may for example explain the surprising distance found between Appenzell-IR and Appenzell-AR. However, this remark did not hold true for the non Swiss-German cantons. Genève

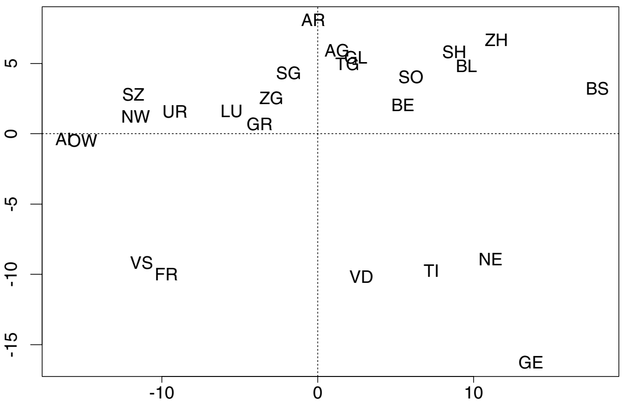


Fig. 1. First two principal components of 112 years of federal votation from 1866 to 1978.

and Ticino, with catholic majorities were close to Neuchâtel and Vaud, with protestant majorities. For a canton like Genève, this was not too much surprising given its history.

Interpretation of the second axis was more straightforward. All Swiss-German cantons were clearly situated in the top part of the graph whereas the French cantons and Ticino were in the bottom part. From all French cantons, the bilingual Valais and Fribourg were also the closest to the Swiss-German ones. From this analysis one can conclude that the difference in voting results between the German and the French cantons is a more deeply rooted phenomenon than the December 6, 1992 voting result.

4.2 Analysis of the period 1979-1992

Figure 2 represents the Swiss cantons in the principal plane for the period 1979-1992. The first axis accounted for 37% of the total variance and the second one for 29%.

Just entered into the Helvetic Confederation, the canton of Jura adopted a very special position, lying at the very right bottom part of the graph, still more extreme than Genève. All French cantons had actually quite special positions, each one being somewhat isolated in the plane. This was also the case of Ticino. Fribourg was a bit closer to the other French cantons (especially Vaud) than in Figure 1. The case of the Swiss-German cantons was quite different. With the exception of the two Basels, Zürich and Appenzell-IR, they were remarkably concentrated together.

Note that religion seems to have lost some of its influence. For example, the protestant cantons Bern and Schaffhausen were found in the neighbor-

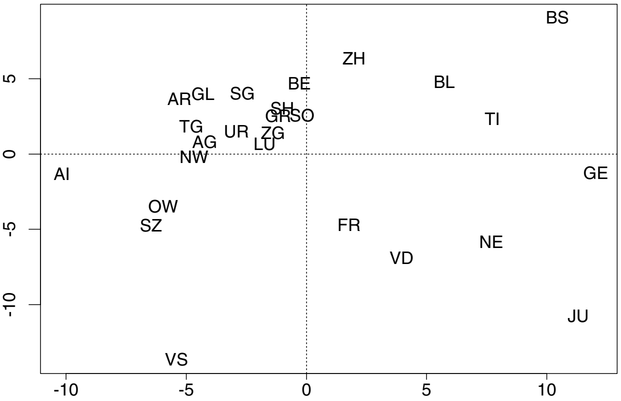


Fig. 2. First two principal components of votation results from 1979 to 1992.

hood of the catholic cantons Luzern and Uri. This was actually not really surprising since religion is nowadays less important in citizens' lives than in the past.

4.3 Analysis of the period 1992-1998

Figure 3 represents the Swiss cantons in the principal plane for the period 1993-1998. The first axis accounted for 39% of the total variance and the second one for 27%.

The linguistic separation between cantons was again pronounced, even more than for the previous periods, since the distinction was made here on the first axis, not on the second one. The French cantons stood on the right side of the graph whereas the Swiss-German cantons stood on the left side. Ticino had an intermediary position between the two groups. The homogeneity among French cantons was here comparable to the homogeneity among Swiss-German cantons. Valais was closer to French cantons than to Swiss-German ones, even if still a bit extreme. Among Swiss-Germans, Basel-Stadt and Basel-Land were the closest to the French cantons.

The linguistic factor seemed to play an important role among the Swiss-Germans cantons themselves! Swiss-German cantons where the percentage of German speaking people was particularly high (like Uri with 93.2%) were generally found more on the left side of the graph than Swiss German cantons where this percentage was smaller (like Basel-Stadt with 78.6%). Similarly, among the French cantons, the bilingual Fribourg and Valais remained the closest to the Swiss-German cantons. The correlation coefficient between the percentage of German speaking people and the coordinates on the first axis

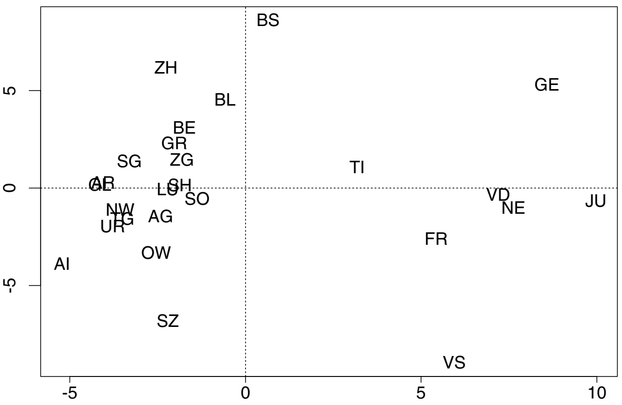


Fig. 3. First two principal components of votation results from 1993 to 1998.

was −0.95! Thus the spoken language was very much related to Swiss citizens' opinions.

5 Other description of Switzerland

In this section, we investigated how the Swiss cantons differ from each other according to other characteristics than votations. We performed a PCA using the 20 variables of general interest listed in Table 2 describing Switzerland in 1990. These data were published by OFS (1990, 1991-1994). The principal plane obtained is plotted in Figure 4. The first axis accounted for 32% of the total variance and the second one for 22%. Interestingly enough, the position of the Swiss cantons were very similar like in the principal plane of Figure 1 (if we ignore the canton of Jura not present in Figure 1). As in Figure 1, French cantons were found in the bottom part of the graph, with Genève at the right extremity, and with Fribourg and Valais nearly close to the Swiss-German cantons. The latter were covering the entire top part of the graph with Basel-Stadt and Appenzell-IR at both extremities and with big cities more on the right. The correlations between the canton's coordinates on Figure 1 and canton's coordinates on Figure 4 (if we ignore Jura) were of 0.87 for the first axis and of 0.85 for the second one! Thus, the picture of Switzerland was quite similar by considering more than one century of votations or by considering variables of general interest describing the Swiss cantons.

1. % of total population	11. % of 20 to 64 years old people
2. Population density (per km^2)	12. % of unemployment
3. % of German speaking people	13. % of married people
4. % of French speaking people	14. % of women
5. % of Protestant	15. % of women in cantonal parliament
6. % of Catholics	16. Infantile mortality
7. % of foreigners	17. % of road accidents
8. % of Swiss from another canton	18. Inhabitant income
9. % of students in gymnasium	19. Fiscal charge
10. % of students in university	20. % of pure agriculture exploitation

Table 2. Twenty variables describing Switzerland.

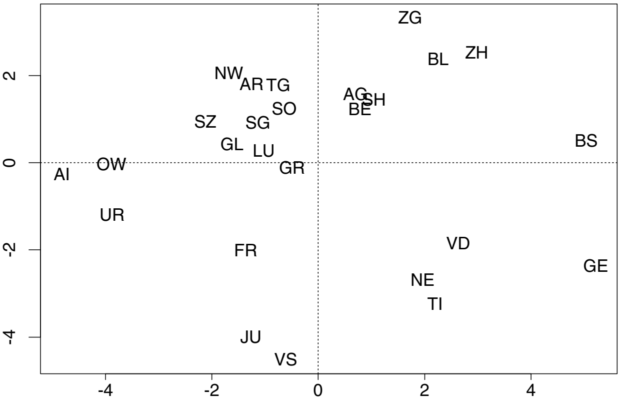


Fig. 4. First two principal components of 20 variables characterizing Switzerland.

6 Conclusion

An attempt has been made to answer the important question following the federal vote of the 6th of December, 1992, whether this date has to be interpreted as the beginning of a divided Switzerland. Using official statistics (the results of federal votations from 1866 to 1998) and a simple statistical technique (principal components analysis), we came to the conclusion that this division is not a new phenomenon. The fact that voting results have always been related to linguistic factors appears clearly in this analysis, even if other cleavages are also important. One should admit that Switzerland has faced such differences without too much difficulties during more than a century.

References

BUNDESBLATT (1866, 1872, 1874-1876, 1878-1880, 1882, 1884-1885, 1887, 1889-1898, 1900, 1902-1903, 1905-1908, 1912-1915, 1918-1935, 1937-1939, 1941-1942, 1944-1998). Swiss Federal Chancellery, Berne.

DIDAY, E., LEMAIRE, J., POUGET, J. and TESTU, F. (1982): *Eléments d'analyse de données*. Dunod, Paris.

JOLLIFFE, I.T. (2002). *Principal Component Analysis*. Springer, New York.

JOYE, D. (1987): Développement méthodologique et analyse du vote. In *Schweizerisches Jahrbuch für politische Wissenschaft, Abstimmungen und Wahlen*, 17–32. Berne.

MANLY, B.F.J. (1986): *Multivariate Statistical Methods, a Primer*. Chapman & Hall, London.

OFS (1991-1994): *Annuaires Statistiques de la Suisse*. Swiss Federal Office of Statistics, Neuchâtel.

OFS (1990): *Recensement de la Population 1990, un Profil de la Suisse*. Swiss Federal Office of Statistics, Neuchâtel.

Prediction with Confidence

Alexander Gammerman

Computer Learning Research Centre, Royal Holloway, University of London
Egham, Surrey TW20 0EX, England, *alex@cs.rhul.ac.uk*

Abstract. The paper outlines an efficient way to complement predictions, produced by new and traditional machine-learning methods, with measures of their accuracy and reliability. These measures are not only valid and informative, but they also take full account of the special features of the object to be predicted. They are based on computable approximations of Kolmogorov's algorithmic notion of randomness. In using these measures it becomes possible to control the number of erroneous predictions by selecting a suitable confidence level. Further development of these ideas can lead to establishing useful links with the Diday's Symbolic Data Analysis.

1 Background

Symbolic Data Analysis (SDA) originally suggested by Edwin Diday has a profound implication on the type of analysis that can be done. Among the areas affected by the SDA is pattern recognition and machine learning. Machine learning has made significant progress and now have a wide range of algorithms that often works very well in practice: decision trees, neural networks, nearest neighbours algorithms, and naive Bayes methods have been used for decades. There are several new algorithms that have been developed recently, including support vector machines and boosting.

From a theoretical point of view, machine learning's most significant contributions to learning are comprised by *statistical learning theory*. This theory, which began with the discovery of VC dimension by Vapnik and Chervonenkis in the late 1960s has produced both deep mathematical results and learning algorithms that work very well in practice.

Given a training set of examples, statistical learning theory produces what we call a *prediction rule* – a function mapping the objects into the labels. Formally, the value taken by a prediction rule on a new object is a *simple prediction* – a guess that is not accompanied by any statement concerning how accurate it is likely to be. The theory does guarantee, however, that as the training set becomes bigger and bigger these predictions will become more and more accurate with greater and greater probability: *probably approximately correct.*

What is less clear is how probably and how approximately? This question has not been answered as well as we might like. This is because the theoretical results that might be thought to answer it, the bounds that demonstrate arbitrarily good accuracy with sufficiently large sizes of the training set, are usually too loose to tell us anything interesting for training sets that we actually have.

This happens in spite of the empirical fact that the predictions often perform very well in practice. Consider, for example, the problem of recognizing hand-written digits. Here we are interested in giving an upper bound on the probability that our learning algorithm fails to choose the right digit; we might like this probability to be less than 0.05, for example, so that we can be 95% confident that the prediction is correct. Unfortunately, typical upper bounds on the probability of error provided by the theory, even for relatively clean data sets such as the USPS data set are greater than 1. We outline below how this problem can be solved and the advantages of using confidence predictors.

2 Confidence predictors

Confidence estimation is a well-studied area of both parametric and non-parametric statistics; however, usually only low-dimensional problems are considered. In this paper we review the approach that has been developed at the Computer Learning Research Centre, Royal Holloway, University of London - see www.clrc.rhul.ac.uk/research/universaltransductionoverview.htm.

It is based on recently developed approximations to the universal measures of confidence given by the algorithmic theory of randomness. The connection between testing for randomness and prediction is, of course, well understood and have been discussed at length by philosophers and statisticians.

In the recently published book by Vovk et al.(2005) it has been shown how some popular prediction algorithms can be transformed into randomness tests and, therefore, be used for producing so-called hedged predictions.

The problem of hedged prediction is intimately connected with the problem of testing randomness. Different versions of the “universal” notion of randomness were defined by Kolmogorov, Martin-Löf and Levin based on the existence of universal Turing machines. Adapted to our current setting, Martin-Löf’s definition is as follows. Let $\mathbf{Z}$ be the set of all possible examples; as each example consists of an object and a label, $\mathbf{Z} = \mathbf{X} \times \mathbf{Y}$, where $\mathbf{X}$ is the set of all possible objects and $\mathbf{Y}$, $|\mathbf{Y}| > 1$, is the set of all possible labels. We will use $\mathbf{Z}^*$ as the notation for all finite sequences of examples. A function $t : \mathbf{Z}^* \to [0,1]$ is a *randomness test* if

1. for all $\epsilon \in (0,1)$, all $n \in \{1,2,\dots\}$ and all probability distributions P on $\mathbf{Z}$,
$$P^n \{z \in \mathbf{Z}^n, t(z) \leq \epsilon\} \leq \epsilon; \tag{1}$$
2. t is upper semicomputable.

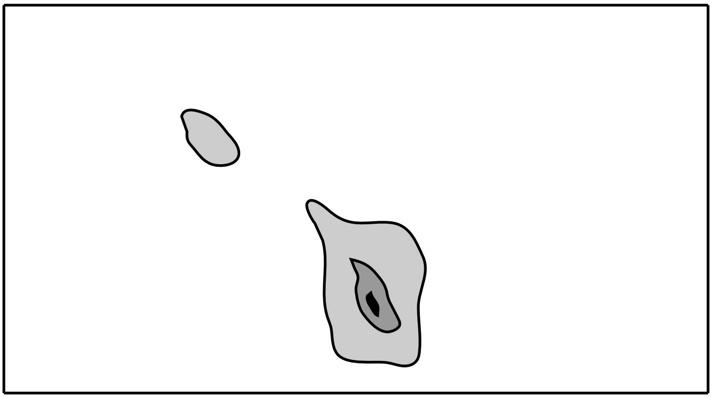


Fig. 1. An example of a nested family of prediction sets (casual prediction in black, confident prediction in dark grey, and highly confident prediction in light grey.

The first condition means that the randomness test is required to be valid: if, for example, we observe $t(z) \leq 1\%$ for our data set z, then either the data set was not generated independently from the same probability distribution or a rare (of probability at most 1%, under any P) event has occurred. The second condition means that we should be able to compute the test, in a weak sense (we cannot require computability in the usual sense, since the universal test can only be upper semicomputable: it can work forever to discover *all* patterns in the data that makes it non-random). Martin-Löf (developing Kolmogorov's earlier ideas) proved that there exists a smallest, to within a constant factor, randomness test.

This new approach allows computing prediction and estimate confidence of the prediction for high-dimensional data. This measure of confidence is given as a number useful for solution of practical problems, and not some asymptotic statement. The only assumption made is the iid assumption (the examples are generated from the same probability distribution independently of each other).

The main idea can be illustrated in case of regression as follows: let's choose a range of "confidence levels" $1 - \epsilon$, and for each of them specify a *prediction set*, the set of labels deemed possible at the confidence level $1 - \epsilon$.

A *confidence predictor* is a function that maps each training set, each new object, and each confidence level $1 - \epsilon$ (formally, we allow ϵ to take any value in $(0, 1)$) to the corresponding prediction set Γ^ϵ. For the confidence predictor to be *valid* the probability that the true label will fall outside the prediction set Γ^ϵ should not exceed ϵ, for each ϵ.

We might, for example, choose the confidence levels 99%, 95% and 80%, and refer to the 99% prediction set $\Gamma^{99\%}$ as the highly confident prediction, to the 95% prediction set $\Gamma^{95\%}$ as the confident prediction, and to the 80%

prediction set $\Gamma^{80\%}$ as the casual prediction. Figure 1 shows how such a family of prediction sets might look in the case of a rectangular.

The casual prediction pinpoints the target quite well, but we know that this kind of prediction can be wrong with probability 20%. The confident prediction is much bigger. If we want to be highly confident (make a mistake only with probability 1%), we must accept an even lower accuracy; there is even a completely different location that we cannot rule out at this level of confidence.

In principle, a confidence predictor outputs prediction sets for all confidence levels, and these sets are nested, as in the Figure 1. This approach is a transductive one since we do not use any inductive rule to label new examples, but move directly from old examples to the prediction about the new object. These prediction sets also form some sort of "symbolic objects" and can be interpreted as a symbolic computing approach. This approach is being developed further, and the results will be reported.

References

VOVK, V., GAMMERMAN, A., SHAFER, G. (2005): *Algorithmic Learning Theory in a Random World.* Springer, Berlin.

Which Bootstrap for Principal Axes Methods?

Ludovic Lebart

CNRS GET-Télécom Paris
46 rue Barrault, 75013, Paris, France, *lebart@enst.fr*

Abstract. This paper deals with validation techniques in the context of exploratory techniques involving singular values decomposition, namely: Principal Components Analysis, Simple and Multiple Correspondence Analysis. We briefly show that, according to the purpose of the analysis, at least five types of resampling techniques could be carried out to assess the quality of the obtained visualisations: a) Partial bootstrap, that considers the replications as supplementary data, without diagonalization of the replicated moment-product matrices. b) Total bootstrap type 1, that performs a new diagonalization for each replicate, with corrections limited to possible changes of signs of the axes. c) Total bootstrap type 2, which adds to the preceding one a correction for the possible exchanges of axes. d) Total bootstrap type 3, that implies Procrustean transformations of all the replicates striving to take into account both rotations and exchanges of axes. e) Specific bootstrap, implying a resampling at a different level (case of a hierarchy of statistical units). An example is presented for each type of resampling.

1 Introduction

Our aim is to assess the results of principal axes methods (PAM), i.e.: multivariate descriptive techniques involving singular values decomposition (SVD) such as principal components analysis (PCA), simple and multiple correspondence analyses (CA and MCA). These methods provide useful data visualisations but their outputs (parameter estimates, graphical displays) are difficult to assess. Computer intensive techniques allow us to go far beyond the criterion of interpretability of the results that was frequently used during the first phases of the upsurge of data analytic methods thirty years ago (see, e.g., Diday and Lebart (1976)). To compute the precision of estimates, the classical analytical approach is both unrealistic and analytically complex. The bootstrap (see: Efron and Tibshirani (1993)), on the contrary, makes almost no assumption about the underlying distributions, and gives the possibility to master every statistical computation for each sample replication and therefore to deal with parameters computed through the most complex algorithms.

2 Basic principles of the bootstrap, a reminder

The *nonparametric bootstrap* consists in drawing with replacement K samples of size n out of n statistical units. Then, parameter estimates such as means, variances, eigenvectors are computed on the K new obtained samples. A current value of K is 200, but it can vary from 10 to several thousands according to the type of application. Empirical evidence suggests that 30 is an acceptable value for K in the context of PAMs. We have at this stage K samples (the replicates) drawn from a new theoretical population defined by the empirical distribution of the original data set, and, as a consequence, K estimates of the parameters of interest. Briefly and under rather general assumptions, it has been proved that we can estimate the variance (and other statistical parameters) of these parameters directly from the set of their K values. In the PCA case, variants of bootstrap do exist for active variables and supplementary variables, both continuous and nominal. Numerous papers have contributed to select the relevant number of axes, and have proposed confidence intervals for points in the subspace spanned by the principal axes. The s^{th} eigenvector of a replicated correlation matrix is not necessarily homologous of the s^{th} eigenvector of the original matrix, because of possible rotations, permutations or changes of sign of the axes. In addition, the expectations of the eigenvalues of the replicated matrices are not the original eigenvalues (see, e.g., Alvarez et al.(2004), Lebart (2006)). Several procedures have been proposed to overcome these difficulties (Chateau and Lebart (1996)): partial replications using supplementary elements (partial bootstrap), use of a three-way analysis to process simultaneously the whole set of replicates and filtering techniques involving reordering of axes and Procrustean rotations (Markus (1994), Milan and Whittaker (1995), Gower and Dijksterhuis (2004)).

3 The illustrative example

An open-ended question has been included in a multinational survey conducted in seven countries around 1990 (Hayashi et al. (1992)). The respondents were asked: "What is the single most important thing in life for you?". The illustrative example is limited to the British sample. The counts for the first phase of numeric coding are as follows: Out of 1043 responses, there are 13669 occurrences (tokens), with 1413 distinct words (types). When the words appearing at least 16 times are selected, there remain 10357 occurrences, with 135 distinct words. The same questionnaire contained also the socio-demographics of the respondents. In this example we focus on a partitioning of the sample into 9 categories, obtained by cross-tabulating age (3 categories) with educational level (3 categories). Figures 1 to 5 will contain an excerpt (four words) of the principal plane produced by a CA of the contingency table cross-tabulating the previous 135 words with the 9 categories. The entry (i, j) of such table is the number of occurrences n_{ij} of word i in the responses of individuals belonging to category j (see: Lebart et al. (1998)).

4 Partial bootstrap

The *partialbootstrap* makes use of projections of replicated elements onto the original principal subspace provided by the eigen-decomposition of the covariance matrix of the original data matrix. It has several advantages. From a descriptive standpoint, this initial subspace is better than any subspace implying the replicates. In fact, unlike the eigenvalues, this subspace is the expectation of all the replicated subspaces having undergone perturbations. The plane spanned by the first two axes, for instance, provides an optimal two-dimensional view on the data set. To apply the partial bootstrap to PCA, one may project the K replicates of variables in the common reference subspace, and compute confidence regions (ellipses or convex hulls) for the locations of these replicates. Then, for each variable-point and each pair of principal axes, a confidence ellipse is derived from a PCA of the two-dimensional cloud of the K replicates. The lengths of the two principal diameters of these ellipses are normatively fixed to four standard deviations. The corresponding ellipses contain then approximately 90% of the replicates. Confidence ellipses may also be replaced by convex hulls. Both ways of visualizing the uncertainty around each variable-point are complementary: ellipses take into account the density of the cloud of replicated points whereas convex hulls pinpoint peripheral points and possible outliers. Gifi (1980), Greenacre (1984) first addressed a similar problem for CA and MCA.

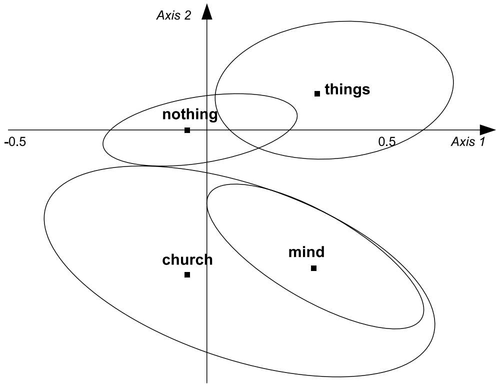


Fig. 1. Partial bootstrap: Confidence ellipses for the location of 4 words in the principal plane of a CA [contingency table crossing 135 words and 9 categories of respondents].

Figure 1 shows confidence ellipses for the location of four words.The words corresponding to markedly overlapping ellipses could not be deemed to be significantly distinct with regard to their distributions among the nine categories. Thus, the words *church* and *mind*, despite their distinct locations, correspond to the same profile of respondents (profile described by the nine categories). Such profile is significantly distinct from those of the words *nothing* and *things*.

5 The total bootstrap and its three options

The *totalbootstrap* consists in performing a specific PAM for each replicate. Evidently, the absence of a common reference subspace may induce a pessimistic view of the variances of the coordinates of the replicates on the principal axes. The most obvious change concerns the directions of the axes, which are in fact unpredictable. We can also observe exchange of axes from one replicate to another, and also rotations of these axes (see: Milan et Whittaker (1995)). We have then to perform a series of transformations to identify the homologous axes during the successive diagonalizations of the K replicated covariance matrices $\mathbf{C}_k$ ($\mathbf{C}_k$ corresponding to the k-th replicate). Three types of transformations lead to three distinct tests for the stability of the observed structure:

- 1. Total bootstrap type 1 (very conservative) : simple change (when necessary) of signs of the axes found to be homologous (merely to remedy possible reflections of the axes). A simple scalar product between homologous original and replicated axes allows for this elementary transformation.
- 2. Total bootstrap type 2 (rather conservative) : correction for possible exchanges of axes. Replicated axes are sequentially assigned to the original axes with which the correlation (in fact its absolute value) is maximum. Then, change of the signs of axes, if needed, as previously.
- 3. Total bootstrap type 3 (could be lenient if the procrustean rotation is performed in a space spanned by many axes) : a procrustean rotation (see: Gower and Dijksterhuis (2004)) aims at superimposing as much as possible the original and replicated axes.

Total bootstrap type 1 ignores the possible exchanges and rotations of axes. It allows for the validation of stable and robust structures. Each replicate is supposed to produce the original axes with the same ranks (order of the eigenvalues). Total bootstrap type 2 is ideally devoted to the validation of axes considered as latent variables, without paying attention to the order of the eigenvalues. Total bootstrap type 3 allows for the validation of a whole subspace. If, for instance, the subspace spanned by the first four replicated axes can coincide with the original four-dimensional subspace, one could find a rotation that can put into coincidence the homologous axes. The situation

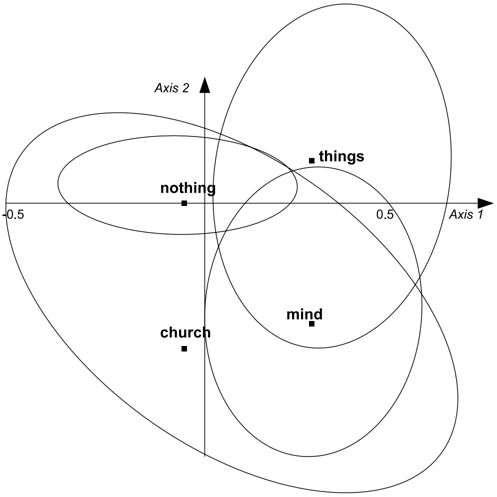


Fig. 2. Total bootstrap type 1: Confidence ellipses for the same word-points in the same original principal plane.

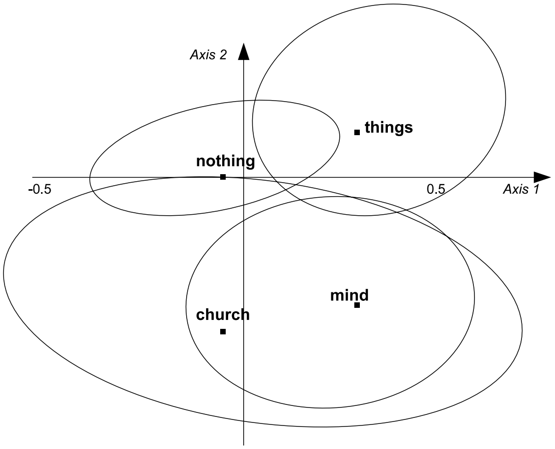


Fig. 3. Total bootstrap type 2: Confidence ellipses for the same word-points in the same original principal plane after correction of the possible exchanges of axes.

is then somewhat similar to that of partial bootstrap. Figure 2 shows the case of total bootstrap of type 1: evidently, the ellipses are much larger. Figure 3 introduces the corrections implied by possible exchange of axes. The pattern observed in figure 1 reappears, albeit less clearly. This improvement means that some axes exchanges were responsible for the perturbations of figure 2.

Some stable dimensions may exist, but their order of appearance (order of the corresponding eigenvalues) can vary from one replicate to another. Figure 4 is similar to figure 1 as far as the size of the ellipses is concerned. In fact, the procrustean transformations depends on the number of axes taken into considerations. They have been performed here in a 5-dimensional space, and the original space can be retrieved without difficulty, leading to a procedure similar to the partial bootstrap. The lack of space does not allow for displaying all the other dimensions.

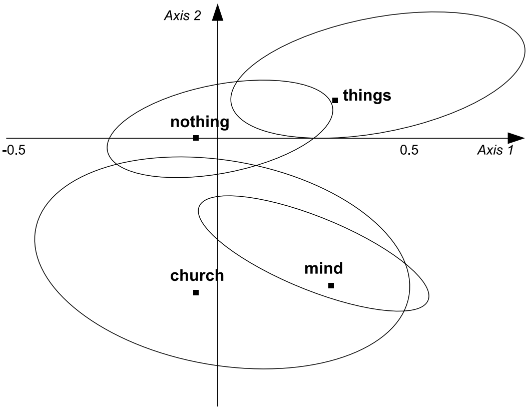


Fig. 4. Total bootstrap type 3: Confidence ellipses for the same words in the same original principal plane, with correction of the possible exchanges of axes and of possible rotations (procrustean transformations).

6 Specific bootstrap

When dealing with textual data, resampling techniques can help to solve the problem of plurality of statistical units (see, in the case of responses to open questions: Tuzzi and Tweedie (2000)). In fact, two (or more) levels of statistical units coexist in textual data analysis. On the one hand, the individuals (with their usual meaning in statistics) could be respondents (case of sample surveys). On the other hand, within the produced corpus of textual responses, the individuals could be the occurrences of words. Replications can be obtained by drawing with replacement either respondents or words. Owing to the discrepancies of responses sizes, the location of a word could be significant when the statistical unit is the word, and not relevant if the statistical unit is the respondent. Figure 5 shows again the same set of four words after a partial specific bootstrap consisting of drawings with replacement the 1043 respondents and projecting the replicates as supplementary points. If we compare the ellipses with those of figure 1, we observe for example that the location of the word *things* is now less precise: this is due to the fact that some respondents use several times that word. Consequently, a drawing of respondents induces a larger perturbation of the data. The specific bootstrap is however the right procedure for inferring the results to the universe of respondents.

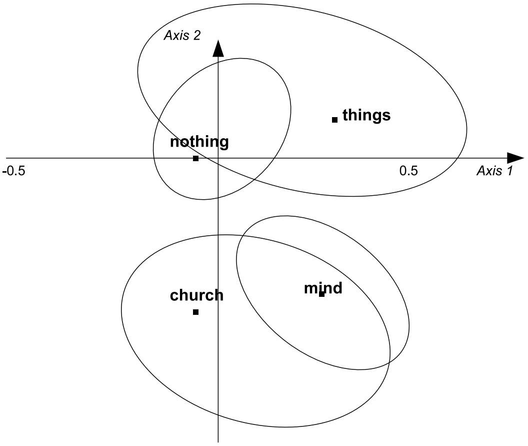


Fig. 5. Specific two-level partial bootstrap: Bootstrapping the observations (i.e.: respondents) instead of the words. This figure should be compared only with Figure 1 (both of them use partial bootstrap).

7 Conclusion

The bootstrap stipulates that the observed sample can serve as an approximation of the population. It takes into account the multivariate nature of the observations and involves simultaneously all the axes. Bootstrapping can also be used to process weighted data (circumstances occurring in most sample surveys) and to draw confidence intervals around the location of supplementary variables in PAM. In the case of multilevel samples (for example: sample of respondents, and samples of words within the responses), the replications can involve separately the different levels, and allows for studying the different components of the observed variance. From a practitioner's standpoint, PAM are particularly profitable when they consider the principal space spanned by the first dimensions as a predictive map which purports to receive all the remaining information contained in the data file (set of supplementary questions).That approach, closely related to regression, is widely used in practice. In all these cases, assessment procedures are difficult to carry out in a classical statistical framework. Bootstrap techniques are the versatile tools able confer to the obtained visualizations a scientific status.

Software note: The used software (DTM: *Data and Text Mining*) as well as the data set serving as an illustration can be freely downloaded from: *http : //www.lebart.org*.

References

ALVAREZ, R., BECUE, M. and VALENCIA, O. (2004): Etude de la stabilité des valeurs propres de l'AFC d'un tableau lexical au moyen de procédures de rééchantillonnage. In: G. Purnelle, C. Fairon, A. Dister (Eds.): *Le poids des mots*. PUL, Louvain, 42-51.

CHATEAU, F. and LEBART, L. (1996): Assessing sample variability in visualization techniques related to principal component analysis: bootstrap and alternative simulation methods. In: A. Prats (Ed.): *COMPSTAT 1996*. Physica Verlag, Heidelberg, 205-210.

DIDAY, E. and LEBART, L. (1977): L'analyse des données. *La Recherche, 74, 15-25.*

EFRON, B. and TIBSHIRANI, R.J. (1993): *An Introduction to the Bootstrap*. Chapman and Hall, New York.

GIFI, A. (1990): *Non Linear Multivariate Analysis*. J. Wiley, Chichester [updated from: A. Gifi (1980) (same title), Dept of Data theory, University of Leiden].

GOWER, J.C. and DIJKSTERHUIS, G.B. (2004): *Procrustes Problems*. Oxford Univ. Press, Oxford.

GREENACRE, M. (1984): *Theory and Applications of Correspondence Analysis*. Academic Press, London.

HAYASHI, C., SUZUKI, T. and SASAKI, M. (1992): *Data Analysis for Social Comparative research: International Perspective*. North-Holland, Amsterdam.

LEBART, L., PIRON, M. and MORINEAU, A. (2006): *Statistique exploratoire multidimensionnelle*. Dunod, Paris.

LEBART, L., SALEM, A. and BERRY, L. (1998): *Exploring Textual Data*. Kluwer, Dordrecht.

LEBART, L. (2006): Validation techniques in multiple correspondence analysis. In: M. Greenacre and J. Blasius (Eds.): *Multiple Correspondence Analysis and Related Methods*. Chapman an Hall, Boca Raton, 179-196.

MARKUS, M.Th. (1994): Bootstrap confidence regions for homogeneity analysis; the influence of rotation on coverage percentages. In: R. Dutter and W. Grossmann (Eds.): *COMPSTAT 1994*. Physica Verlag, Heidelberg, 337-342.

MILAN, L. and WHITTAKER, J. (1995): Application of the parametric bootstrap to models that incorporate a singular value decomposition. *Applied Statistics 44 (1), 31-49.*

TUZZI, A. and TWEEDIE, F.J. (2000): The best of both worlds: Comparing Mocar and Mcdisp. In: M. Rajman and J.-C. Chappelier (Eds.): *JADT2000 (Cinquièmes Journées Internationales sur l'Analyse des Données Textuelles)*. EPFL, Lausanne, 271-276.

PCR and PLS for Clusterwise Regression on Functional Data

Cristian Preda[1] and Gilbert Saporta[2]

[1] Faculté de Médecine, Université de Lille 2
CERIM - Département de Statistique
1, Place de Verdun, 59045 Lille Cedex, France, *cpreda@univ-lille2.fr*
[2] Chaire de Statistique Appliquée, CEDRIC, CNAM
292, Rue Saint Martin, 75141 Paris Cedex 03, France, *saporta@cnam.fr*

Abstract. Clusterwise regression is applied to functional data, using PCR and PLS as regularization methods for the functional linear regression model. We compare these two approaches on simulated data as well as on stock-exchange data.

1 Introduction

Clusterwise linear regression method provides classification of data such that each cluster is generated by some linear regression model. More precisely, if $\{Y, X_1, \ldots, X_q\}$, $q \geq 1$, are real-valued random variables, the homogeneity of subjects within a cluster is given not only by similarities of the observed values of these variables but mainly by the proximity of subjects with respect to some linear model. One can consider that data is generated by a mixture of several regression models (DeSarbo and Cron (1988)), Hennig (1999),(2000)), that is, there exists a latent categorical random variable $\mathcal{G}$, $\mathcal{G} \in \{1, \ldots, K\}$, $K \geq 2$, defining the clusters such that for $\forall k \in 1, \ldots K$, $\mathbb{P}(\mathcal{G} = k) \neq 0$ and

$$\mathbb{E}(Y \big| X_1 = x_1, \ldots, X_q = x_q) = \beta_0^k + \beta_1^k x_1 + \ldots + \beta_q^k x_q,$$

where $\{\beta_i^k\}_{i=0,\ldots,q}$ are the regression coefficients for the cluster defined by $\{\mathcal{G} = k\}$.

The estimation aspects in clusterwise linear regression was addressed firstly by the pioneering works of Bock (1969) and Diday (1976) who propose a piecewise linear regression algorithm as a special case of k-means clustering with a criterion based on the minimization of the squared residuals instead of the classical within-class dispersion. The problem of multicollinearity and overfit under the least squares criterion is the subject of works of Charles (1977) which establish properties and conditions for convergence of the alternating algorithm proposed by Diday(1976) and introduce the ridge regression as a regularization method for the clusterwise procedure. One can also mention the works of Spaeth(1979) which propose an estimation procedure of clusterwise regression models by an exchange algorithm.

These clusterwise algorithms are largely used nowadays but few significant modifications have been done since then (Plaia (2004)). Recent contributions in this area are due mainly to the development of techniques for estimating the linear models within clusters subject to different inconsistency issues : multicollinearity of predictors, number of observations within a cluster smaller then the number of predictors, etc.

In this paper we are interested in clusterwise linear regression when the set of explanatory variables (predictors) are of functional type, i.e., data are functions or curves of some continuous parameter t (usually time or wavelength). A well accepted model for this kind of data is to consider them as paths of a stochastic process $X = \{X_t\}_{t \in T}$ taking values in a Hilbert space H of functions on some set T. For example, a second order stochastic process $X = \{X_t\}_{t \in [0,1]}$, L_2–continuous with sample paths in $L_2([0,1])$ can be used as model for index stock-exchange evolution during a time period or for the knee flexion angle measure over a complete gait cycle.

There is a rich and recent literature devoted to functional data, the last contributions being reported by the monographs of Ferraty and Vieu (2006), Ramsay and Silverman (1997, 2002). As an alternative to the work of Abraham et al. (2002) on unsupervised classification of functional data, Preda and Saporta (2005b) proposed the PLS approach for clusterwise regression on functional data.

We propose a comparative study of the partial least squares (PLS) and the regression on principal components (PCR) approaches for estimating coefficient regression functions within clusters in the context of clusterwise linear regression with predictors of functional type. The paper is divided into three parts. After a brief introduction to PCR and PLS regularization methods for functional data, we describe the clusterwise linear model using the estimations given by PCR and PLS. In the last section we present a simulation study as well as an application on stock exchange data.

2 PCR and PLS for functional data

Let us consider the functional data as sample paths of a stochastic process $\mathbf{X} = \{X_t\}_{t \in [0,T]}$ with continuous time, and $\mathbf{Y} = (Y_1, Y_2, \ldots, Y_p)$, $p \geq 1$, a random vector defined on the same probability space as $\mathbf{X}$, $(\Omega, \mathcal{A}, P)$. We assume that $\{X_t\}_{t \in [0,T]}$ and $\mathbf{Y}$ are of second order, $\{X_t\}_{t \in [0,T]}$ is L_2-continuous and for any $\omega \in \Omega$, $t \mapsto X_t(\omega)$ is an element of $L_2([0,T])$. Without loss of generality we assume also that $E(X_t) = 0$, $\forall t \in [0,T]$ and $E(Y_i) = 0$, $\forall i = 1, \ldots, p$.

The functional linear regression model assumes that

$$\mathbf{Y} = \int_0^T \beta(t) X(t) dt + \varepsilon, \tag{1}$$

where β is a $\mathbb{R}^p$-valued function on $[0,T]$ and ε is the random error term.

It is well known that the approximation of $\mathbf{Y}$ obtained by the classical linear regression on $\mathbf{X} = \{X_t\}_{t\in[0,T]}$, i.e., $\hat{\mathbf{Y}} = \int_0^T \beta(t)X_t dt$, is such that β is in general a distribution rather than a function of $L_2([0,T])$ (Saporta (1981)). This difficulty appears also in practice because one has generally more predictors than the number of observations, the least squares criterion providing inconsistent estimators (infinite number of solutions). Regression on principal components (PCR) of $\mathbf{X}$ (Deville (1978)) and PLS approach (Preda and Saporta (2005a)) give satisfactory solutions to this problem.

2.1 Linear regression on principal components (PCR)

The principal components of the stochastic process $\mathbf{X} = \{X_t\}_{t\in[0,T]}$ are linear combinations of X_t, $t \in [0,T]$, given by the eigenfunctions of the covariance operator of $\mathbf{X}$:

$$\xi_i = \int_0^T f_i(t)X_t dt,$$

where $\{f_i\}_{i\geq 1}$ are solution of the eigenvalue equation

$$\int_0^T C(t,s)f_i(s)ds = \lambda_i f_i(t),$$

and $C(t,s) = \mathrm{cov}(X_t, X_s)$, $\forall t, s \in [0,T]$.

Observe that the principal components $\{\xi_i\}_{i\geq 1}$ are eigenvectors of the Escoufier operator, $\mathbf{W}^X$, defined by

$$\mathbf{W}^X Z = \int_0^T E(X_t Z)X_t dt,$$

for any real-random variable Z in $L_2(\Omega)$ (Escoufier (1970)).

As in the classical setting, the process $\{X_t\}_{t\in[0,T]}$ and the set of its principal components, $\{\xi_k\}_{k\geq 1}$, span the same linear space. Thus, the regression of $\mathbf{Y}$ on $\mathbf{X}$ is equivalent to the regression on $\{\xi_k\}_{k\geq 1}$ and we have $\hat{\mathbf{Y}} = \sum_{k\geq 1} \frac{E(\mathbf{Y}\xi_k)}{\lambda_k}\xi_k$.

In practice one has to choose an approximation of order q, $q \geq 1$:

$$\hat{\mathbf{Y}}_{PCR(q)} = \sum_{k=1}^{q} \frac{E(\mathbf{Y}\xi_k)}{\lambda_k}\xi_k = \int_0^T \hat{\beta}_{PCR(q)}(t)X_t dt, \tag{2}$$

where

$$\hat{\beta}_{PCR(q)} = \sum_{k=1}^{q} \frac{E(\mathbf{Y}\xi_k)}{\lambda_k} f_k(t)$$

is the estimator of the coefficient regression function β obtained with the first q principal components.

Using the first q principal components raises a problem since they are computed independently of the response. Principal components with a great power of explanation yield generally stable models but could be uncorrelated with the response, whereas the principal components highly correlated with the response could be less explanatory for $\mathbf{X}$. Moreover, for functional data, the number of principal components could be infinite. Thus, the choice of principal components is a trade-off between stability of the linear model and its predictive power (see also Escabias et al. (2004)). A solution to this problem is the PLS approach.

2.2 Partial least squares regression (PLS)

The PLS approach offers a good alternative to the PCR method by replacing the least squares criterion with that of maximal covariance between $\{X_t\}_{t\in[0,T]}$ and $\mathbf{Y}$ (Preda and Saporta (2005a)).

One obtains a set of PLS components $\{t_i\}_{i\geq 1}$ using an iterative procedure. At each step, the PLS component being defined as the linear combination of X_t variables that attains maximum covariance with the response or between residuals :

Let $X_{0,t} = X_t$, $\forall t \in [0,T]$ and $\mathbf{Y}_0 = \mathbf{Y}$. At step q, $q \geq 1$, of the PLS regression of $\mathbf{Y}$ on $\{X_t\}_{t\in[0,T]}$, we define the q^{th} PLS component, t_q, by the eigenvector associated to the largest eigenvalue of the operator $\mathbf{W}^X_{q-1}\mathbf{W}^Y_{q-1}$, where $\mathbf{W}^X_{q-1}$, respectively $\mathbf{W}^Y_{q-1}$, are the Escoufier's operators associated to $\{X_{q-1,t}\}_{t\in[0,T]}$, respectively to $\mathbf{Y}_{q-1}$. The PLS step is completed by the ordinary linear regression of $X_{q-1,t}$ and $\mathbf{Y}_{q-1}$ on t_q. Let $X_{q,t}$, $t \in [0,T]$ and $\mathbf{Y}_q$ be the random variables which represent the error of these regressions : $X_{q,t} = X_{q-1,t} - p_q(t)t_q$ and $\mathbf{Y}_q = \mathbf{Y}_{q-1} - \mathbf{c}_q t_q$.

Then, for each $q \geq 1$, $\{t_q\}_{q\geq 1}$ forms an orthogonal system in $L_2(X)$ and the following decomposition formulas hold :

$$\begin{aligned}
&\mathbf{Y} = \mathbf{c}_1 t_1 + \mathbf{c}_2 t_2 + \ldots + \mathbf{c}_q t_q + \mathbf{Y}_q,\\
&X_t = p_1(t)t_1 + p_2(t)t_2 + \ldots + p_q(t)t_q + X_{q,t}, \quad t \in [0,T].
\end{aligned}$$

The PLS approximation of $\mathbf{Y}$ by $\{X_t\}_{t\in[0,T]}$ at step q, $q \geq 1$, is given by :

$$\hat{\mathbf{Y}}_{PLS(q)} = \mathbf{c}_1 t_1 + \ldots + \mathbf{c}_q t_q = \int_0^T \hat{\beta}_{PLS(q)}(t) X_t dt. \tag{3}$$

Notice that de Jong (1993) and Phatak (2001) show that for a fixed q, the PLS regression fits closer than PCR, in that sense

$$R^2(\mathbf{Y}, \hat{\mathbf{Y}}_{PCR(q)}) \leq R^2(\mathbf{Y}, \hat{\mathbf{Y}}_{PLS(q)}),$$

where R is the multiple correlation coefficient.

The number of PLS components used for regression is generally determined by cross-validation.

3 Clusterwise regression model and functional data

Let us suppose that the response Y is one dimensional ($p = 1$). The clusterwise linear model assumes that there exists a random variable $\mathcal{G}$, $\mathcal{G} \in \{1, 2, \ldots, K\}$, $K \geq 2$, such that for each cluster defined by $\{\mathcal{G} = i\}$ one has

$$\begin{array}{l} \mathbb{E}(Y \big| \mathbf{X} = x, \mathcal{G} = i) = \alpha^i + \int_0^T \beta^i(t)x(t)dt, \\ V(Y \big| \mathbf{X} = x, \mathcal{G} = i) = \sigma_i^2 > 0, \quad x \in L_2([0,T]), \forall i = 1, \ldots, K. \end{array} \tag{4}$$

i.e.,

$$Y_{|\mathcal{G}=i} = \alpha^i + \int_0^T \beta^i(t)X(t)dt + \varepsilon_i, \quad \forall i = 1, \ldots, K.$$

Let us assume that K is known and the homoscedasticity hypothesis holds, i.e. the variance of the random error term ε_i within each cluster are equals, $\sigma_i^2 = \sigma^2$, $\forall i = 1, \ldots K$.

In such a model, the parameters that have to be estimated are the regression coefficient functions for each cluster $\{(\alpha^i, \beta^i)\}_{i=1,\ldots,K}$ and σ^2. Charles (1997) and Bock (1969) use the following criterion for estimating the linear models within clusters, $\{\alpha^i, \beta^i\}_{i=1}^K$:

$$\min_{\{\alpha^i, \beta^i\}_{i=1}^K, \mathcal{L}(\mathcal{G})} \left\{ \mathrm{V}(Y - \hat{Y}^L) \right\}, \tag{5}$$

where $\hat{Y}^L = \sum_{i=1}^{K} \hat{Y}^i \mathbf{1}_{\mathcal{G}=i}$ and $\hat{Y}^i = \alpha^i + \langle \hat{\beta}^i, \mathbf{X} \rangle$ is the approximation of Y given by the linear regression of Y on $\mathbf{X}$ within the cluster i, $i = 1, \ldots, K$.

If n data points $\{x_i, y_i\}_{i=1}^n$ have been collected, the cluster linear regression algorithm finds simultaneously an optimal partition of the n points, $\hat{\mathcal{G}}$ (as estimation of the distribution of $\mathcal{G}$, $\mathcal{L}(\mathcal{G})$), and the regression models for each cluster (element of partition) $(\hat{\alpha}, \hat{\beta}) = \{\hat{\alpha}^i, \hat{\beta}^i\}_{i=1}^K$, which minimize the criterion :

$$\mathcal{V}(K, \hat{\mathcal{G}}, \hat{\alpha}, \hat{\beta}) = \sum_{i=1}^{K} \sum_{\hat{\mathcal{G}}(j)=i} \left(y_j - (\hat{\alpha}^i + \langle \hat{\beta}^i, x_j \rangle) \right)^2. \tag{6}$$

In order to minimize (6), the clusterwise linear regression algorithms iterates the following two steps :

i) For given $\hat{\mathcal{G}}$, $\mathcal{V}(K, \hat{\mathcal{G}}, \hat{\alpha}, \hat{\beta})$ is minimized by the LS-estimator $(\hat{\alpha}^i, \hat{\beta}^i)$ from the points (x_j, y_j) with $\hat{\mathcal{G}}(j) = i$.

ii) For given $\{\hat{\alpha}^i, \hat{\beta}^i\}_{i=1}^K$, $\mathcal{V}(K, \hat{\mathcal{G}}, \hat{\alpha}, \hat{\beta})$ is minimized according to

$$\hat{\mathcal{G}}(j) = \arg \min_{i \in \{1, \ldots, K\}} \left(y_j - (\hat{\alpha}^i + \langle \hat{\beta}^i, x_j \rangle) \right)^2. \tag{7}$$

That is, $\mathcal{V}(K, \hat{\mathcal{G}}, \hat{\alpha}, \hat{\beta})$ is monotonely decreasing if the steps *i*) and *ii*) are carried out alternately :

$$\underbrace{\hat{\mathcal{G}}_0 \Rightarrow (\hat{\alpha}_0, \hat{\beta}_0)}_{\mathcal{V}_0} \underset{\geq}{\Rightarrow} \underbrace{\hat{\mathcal{G}}_1 \Rightarrow (\hat{\alpha}_1, \hat{\beta}_1)}_{\mathcal{V}_1} \underset{\geq}{\Rightarrow} \ldots \underset{\geq}{\Rightarrow} \underbrace{\hat{\mathcal{G}}_l \Rightarrow (\hat{\alpha}_l, \hat{\beta}_l)}_{\mathcal{V}_l} \underset{\geq}{\Rightarrow} \ldots \quad (8)$$

where the index of each term denotes the iteration step, $\hat{\mathcal{G}}_0$ being an initial partition of the n data points.

When the predictor is of functional type, the classical linear regression is not adequate to provide estimators for the linear models within clusters, $\{\alpha^i, \beta^i\}_{i=1}^K$. We propose to adapt the PLS and PCR regression approaches for the clusterwise algorithm in order to overcome this problem. The convergence of the clusterwise algorithm using these regularization methods is discussed in Preda and Saporta (2005b).

Let us denote by $\{\hat{\alpha}^i_{PLS}, \hat{\beta}^i_{PLS}\}_{i=1}^K$, respectively by $\{\hat{\alpha}^i_{PCR}, \hat{\beta}^i_{PCR}\}_{i=1}^K$ the estimators for the coefficient regression functions within clusters.

As a quality measure of the fit in clusterwise regression one can use the square of the correlation coefficient between the response (Y) and the predictor $(\mathbf{X})$ within each cluster. If a clusterwise linear model underlies data, it is interesting to compare each cluster regression quality with that obtained by the linear model without clusters. For comparison of several techniques for estimating the clusterwise model (for example, PLS and PCR) the criterion given in (7) is a natural choice.

4 Numerical results

In this section we compare the clusterwise PLS and PCR approaches in the context of functional data both on simulated and real data.

Firstly we consider simulated data with two clusters each having its own linear structure with respect to a one dimensional response Y and a set of curves $\{X_t, t \in [0, T]\}$ drawn from the one-dimensional Brownian motion. The aim is to check the capability of the clusterwise regression to identify these two clusters. The second application concerns stock exchange data and the aim is to "predict" the last five minutes of the evolution of a particular share, considered on a certain interval of time.

We quote by CW-PLS(K) and CW-PCR(K) the clusterwise PLS, respectively PCR, regressions with K clusters, by PCR and PLS, the global linear regression models obtained with the principal components, respectively on the first PLS components. The number of components considered for regression (PLS and PCR) is determined by cross-validation (leave-one-out) using a significance level of 95%.

4.1 Simulation study

Let us consider that the stochastic process underlying the functional data is the standard Brownian motion on the interval $[0,1]$, $\mathbf{X} = \{X_t\}_{t\in[0,1]}$, $\mathbb{E}(X_t) = 0$, $\mathbb{E}(X_t X_s) = \inf\{t,s\}$, $\forall t, s \in [0,1]$. The response variable Y is defined with respect to a group variable, $\mathcal{G}$, with two modalities in the following way :

$$\text{Class 1 :} \qquad Y = \int_0^1 tX_t dt + \varepsilon_1$$

$$\text{Class 2 :} \qquad Y = \int_0^1 (1-t)X_t dt + \varepsilon_2$$

where ε_1 and ε_2 are Gaussian noises such that $\sigma_1^2 = \sigma_2^2 = \sigma^2$. We consider two situations $\sigma^2 = 0.01$ and $\sigma^2 = 0.02$ which correspond to the following ratios $\sigma^2/\mathbb{V}(Y)$:

	$\sigma^2 = 0.01$	$\sigma^2 = 0.02$
Class 1	0.069	0.137
Class 2	0.167	0.285

Table 1. Noise to response ratio, $\sigma^2/\mathbb{V}(Y)$.

Our simulation is based on the following conditions:

- the trajectories of X are discretized in 101 equidistant points.
- values of Y as well as the principal and PLS components are computed using integration by trapezoidal interpolation.
- the training sample sizes are identical for both groups, $n = 500$.
- 100 simulations.

Table 2 presents the performance values of PLS and PCR models in terms of response variance explained by the predictor, i.e. the multiple correlation coefficient, R^2. For clusterwise models we present also the error classification rate (ECR). Both measures are averaged over 100 samples.

Figure 1 plots $\hat{\beta}^i_{PLS}$, $i = 1, 2.$, the two regression coefficient functions obtained with the PLS approach attaining the best model with respect to the criterion given by (7).

The results obtained on this example show that PLS fits slightly better than PCR especially when the noise to response ratio is increasing. This is mainly due to the fact that the PLS takes into account, for computing PLS components, the correlation between the response and predictor, whereas that is not the case for PCR. Notice that these results are in agreement with those obtained by Barker and Rayens (2003) and Preda et al. (2007) on the capability of PLS models for classification purpose.

Model	$\sigma^2 = 0.01$			$\sigma^2 = 0.02$		
R^2-PCR	0.718			0.597		
R^2-PLS	0.724			0.612		
	cluster 1	cluster 2	ECR	cluster 1	cluster 2	ECR
CW-PCR(2)	0.882	0.794	0.112	0.752	0.625	0.322
CW-PLS(2)	0.908	0.812	0.103	0.826	0.674	0.260

Table 2. Model quality : R^2 and error classification rate (ECR) averaged over 100 simulations.

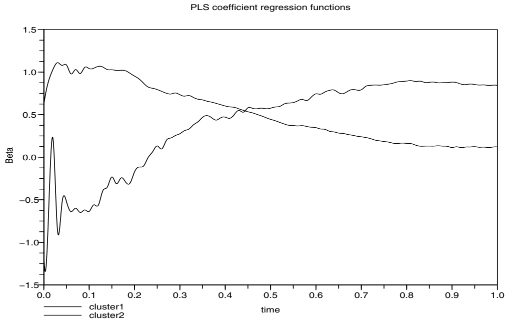


Fig. 1. Cluster-specific regression coefficient functions for PLS approach ($\sigma^2 = 0.01$).

4.2 Application on stock exchange data

We have 84 shares quoted at the Paris stock exchange, for which we know the whole behavior of the growth index during one hour (between 10^{00} and 11^{00}). Notice that a share is likely to change every second. We also know the evolution of the growth index of a new share (indexed 85) between 10^{00} and 10^{55}.

Linear models for this data set were fitted with PLS and PCR regression techniques in order to predict the way in which the new share will behave between 10^{55} and 11^{00} (Preda and Saporta (2005a). We have shown (Preda and Saporta (2005b)) that this prediction is improved when the clusterwise approach is considered.

Since the curves are completely known, we use the time average approximation developed in Preda (2000) by taking an equidistant discretization of the interval $[0, 3600]$ (time expressed in seconds) in 60 subintervals. The forecasts obtained will then match the average level of the growth index of share 85 considered on each interval $[60 \cdot (i-1), 60 \cdot i)$, $i = 56, \ldots, 60$.

The results of the best models are presented in the Table 3.

	$\hat{m}_{56}(85)$	$\hat{m}_{57}(85)$	$\hat{m}_{58}(85)$	$\hat{m}_{59}(85)$	$\hat{m}_{60}(85)$	SSE
Observed	**0.700**	**0.678**	**0.659**	**0.516**	**-0.233**	**-**
PLS	0.312	0.355	0.377	0.456	0.534	0.911
PCR	0.613	0.638	0.669	0.825	0.963	1.511
CW-PLS(3)	0.643	0.667	0.675	0.482	0.235	0.215
CW-PLS(4)	0.653	0.723	0.554	0.652	-0.324	0.044

Table 3. Forecasts for share 85.

Using the sum of squared errors (SSE) as measure of fit, let us observe that the clusterwise models give better results than the global ones. The clusterwise models predict better the crash of the share 85 for the last 5 minutes, whereas the global models do not. For the PLS model with 4 clusters, the size of each cluster is given by the distribution $(\frac{17}{84}, \frac{32}{84}, \frac{10}{84}, \frac{25}{84})$. Following the K-NN procedure proposed by Charles (1977), the share 85 belongs to the first cluster.

5 Conclusion

PLS and PCR approaches are regularization techniques for linear regression used with success when the least squares criterion produces inconsistent estimators, in particular, when multicollinearity and sample size problems occur. This is the case for functional data (multicollinearity) and the clusterwise algorithm (cluster size less than the number of predictors). We show by a simulation study and an application on stock-exchange data the efficiency of these two methods and point out the accuracy of PLS with respect to PCR.

References

ABRAHAM, C., CORNILLON, P., MATZNER-LÖBER, E. and MOLINARI, N. (2003): Unsupervised curve clustering using B-splines. *Scand. J. Statist., 30, 581–595.*

BARKER, M. and RAYENS, W. (2003): Partial least squares for discrimination. *Journal of Chemometrics, 17, 166–173.*

BOCK, H.-H. (1989): *The equivalence of two extremal problems and its application to the iterative classification of multivariate data.* Lecture note, Mathematisches Forschungsinstitut Oberwolfach.

CHARLES, C. (1977): *Régression Typologique et Reconnaissance des Formes.* Thèse de doctorat, Université Paris IX.

DE JONG, S. (1993): PLS fits closer than PCR. *Journal of Chemometrics, 7, 551-557.*

DESARBO, W.S. and CRON, W.L. (1988): A maximum likelihood methodology for clusterwise linear regression. *Journal of Classification, 5, 249-282.*

DEVILLE, J.C. (1978): Analyse et prévision des séries chronologiques multiples non stationnaires. *Statistique et Analyse des Données, 3, 19-29.*

DIDAY, E. (1976): *Classification et sélection de paramètres sous contraintes.* Rapport de Recherche IRIA-LABORIA, no. 188.

ESCABIAS, M., AGUILERA, A.M., and VALDERRAMA, M.J. (2004): Principal component estimation of functional logistic regression : discussion of two different approaches. *Journal of Nonparametric Statistics, 3–4, 365–385.*

ESCOUFIER, Y. (1970) *Echantillonage dans une population de variables aléatoires réelles.* Publications de l'Institut de Statistique de l'Université de Paris, 19, Fasc. 4, 1-47.

FERRATY, F. and VIEU, P. (2006): *Nonparametric Functional Data Analysis. Theory and Practice.* Springer Series in Statistics.

HENNIG, C. (1999): Models and methods for clusterwise linear regression. In: *Classification in the Information Age*, Springer, Berlin, 179-187.

HENNIG, C. (2000): Identifiability of models for Clusterwise linear regression. *Journal of Classification, 17, 273-296.*

PLAIA, A. (2004): Constrained clusterwise linear regression. In: M. Vichi, P. Monari, S. Mignani, A. Montanari (Eds): *New Developments in Classification and Data Analysis*, Springer, 78–86.

PHATAK, A. and DE HOOG, F. (2001): PLSR, Lanczos, and conjugate gradients. *CSIRO Mathematical & Information Sciences*, Report No. CMIS 01/122, Canberra.

PREDA, C. and SAPORTA, G. (2005a): PLS regression on a stochastic process. *Computational Statistics and Data Analysis, 48, 149–158.*

PREDA, C. and SAPORTA, G. (2005b): Clusterwise PLS regression on a stochastic process. *Computational Statistics and Data Analysis, 49, 99–108.*

PREDA, C., SAPORTA, G. and LÉVÉDER, C. (2007): PLS classification of functional data. *Computational Statistics, In Press, doi : 10.1007/s00180-007-0041-4.*

RAMSAY, J.O. and SILVERMAN, B.W. (1997): *Functional Data Analysis.* Springer Series in Statistics, Springer-Verlag, New York.

RAMSAY, J.O. and SILVERMAN, B.W. (2002): *Applied Functional Data Analysis. Methods and Case Studies.* Springer, Berlin-Heidelberg.

SAPORTA, G. (1981): *Méthodes exploratoires d'analyse de données temporelles.* Cahiers du B.U.R.O., No. 37-38, Université Pierre et Marie Curie, Paris.

SPAETH, H. (1979): Clusterwise linear regression. *Computing 22, 367-373.*

A New Method for Ranking n Statistical Units

Alfredo Rizzi

Dipartimento di Statistica, Probabilità e Statistiche Applicate
Università di Roma "La Sapienza", Rome, Italy, *alfredo.rizzi@uniroma1.it*

Abstract. In many research problems it is useful to summarize some indices or indicators to express a synthetic, indirect measure of a concept which is revealed by p variables observed in each statistical unit. This is because the p variables are considered to be indirect measures of a complex (perhaps indefinable) concept. Within this context and for ranking the n statistical units the author suggests the index:

$$R_i = (\operatorname{sgn} c_{i1})(\sum_r c_{ir}^2)^{1/2}$$

where the c_{ir} $(i = 1, 2, \ldots, n;\ r = 1, 2, \ldots, p)$ represent the values of the p principal components connected with the *i-th* statistical unit. This index is applied for ranking the 20 Italian Regions for *quality of life* for the years 2000-2002. The results are compared with those that are furnished by the *single source method.*

1 Introduction

When p quantitative characters are observed in a finite population $\mathbf{P}$ of n statistical units, the information on the structure of the population is contained in the matrix: $\mathbf{X}_{n,p} \equiv x_{i,j}$, $(i = 1, 2, \ldots n;\ j = 1, 2, \ldots, p)$ where $\mathrm{x}_{i,j}$ represents the realization of variable j relative to the statistical unit i. Row i, therefore, is related to unit i and column j is related to character j. The data matrix $\mathbf{X}_{n,p}$ can be represented by n points in a vector space of p dimensions or as p points on a vector space of n dimensions.

In applications of interest to statistics, n is nearly always greater than p and no variable is proportional to or a linear combination of other variables. Moreover, the rank of the matrix can be assumed to be equal to p.

Each of the p variables observed is defined by:

1. A set Θ called the space of the observations;
2. A algebraic structure s on the Θ ;
3. An mapping V of Ω into Θ, where Ω is the finite set of the statistical units.

Different types of variables can be distinguished by the cardinality of the set Θ and by the algebraic structure s; for example:

1. Quantitative ordinals, such as judgements, quality;
2. Quantitative measurements or intensities, such as income in dollars or imported goods in quintals;

3. Quantitative ordinals, such as location of home, socio-professional category;
4. Absolute frequencies such as the number of inhabitants in a area at a particular date.

We can also obtain other tables of great statistical interest from matrix $\mathbf{X}_{n,p}$. For example: contingency or frequency tables; standardized matrices in their different forms; matrices of the deviations from the mean expressed in terms of standard deviation; matrices of variance and covariance, matrices of correlation, matrices of distances and similarity.

When a researcher collects an $n \times p$ data array $\mathbf{X}_{n,p}$ he generally has two goals (Escoufier(2006)):

1. Comparison of the variables.
2. Comparison of the observations.
   Generally to compute a distance between the observations in $\mathbb{R}^p$, we need a $p \times p$ symmetric positive definite matrix $\mathbf{Q}_{p,p} = \mathbf{L'}_{p,p}\mathbf{L}_{p,p}$, where $\mathbf{L}_{p,p}$ is a $p \times p$ matrix of rank p which can be viewed as a linear transformation of $\mathbf{X}_{n,p}$ such that $\mathbf{Y}_{n,p} = \mathbf{X}_{n,p}\mathbf{L}_{p,p}$ will replace $\mathbf{X}_{n,p}$.

This paper considers the ranking of the observations which is a particular case of comparing the observations.

2 Synthesis of the indicators

The following distinctions between the diverse methodologies for the design of elementary indicators can be proposed:

- the ordinal approach;
- the cardinal approach (through the arithmetic mean of the values transformed into index numbers, through the arithmetic mean of the values in proportion with the field of variations, through the arithmetic mean of the values transformed into standard deviations, through the sum of the values transformed into percentages);
- the transformation of elementary indicators into comparative indicators;
- the synthesis of the elementary indicators through the taxonomic method of Wraclaw which considers the distance of the units from an ideal unit and the regrouping of homogenous territorial units through taxonomic graphics;
- the synthesis of the elementary indicators through the method of principal components (single source factor);
- factorial analysis.

The general problem of ranking n statistical units on which the modalities of p characters have been observed was dealt with by V. Barnett, (1976), in an important article which appeared in the Journal of the Royal Statistical Association in 1976. Following this article there were interesting discussions by R.L. Plackett, K.V. Mardia, R.M. Loynes, G.M. Paddle, T. Lewis, G.A. Barnard, A.M. Walker, F. Downton, P.J. Green, M. Kendall, A. Rizzi, M. Robinson, Allen Scheult and D.H. Young. The concept of sub-ranking was introduced and marginal ranking, reduced ranking, partial ranking and conditioned ranking were defined. This approach was presented in both probabilistic as well as descriptive terms. The research concluded with the affirmation that there was no reasonable basis for a complete ranking of a set of multivariate modalities.

To rank statistical units through a synthetic index one could proceed, for example, in the following way.

We add the values of every statistical unit after having standardized them with the field of variation which leaves us with a value lying between 0 and 100. Calculating this index for every statistical unit gives us the ability to rank them in function of their value. This will vary between zero and $100p$ where p is the number of variables, taking the value $100p$ in the case of a single unit which absorbs the total of each phenomenon. Dividing this index by p yields a value of between 0 and 100. This procedure presents the inconvenience of summing modalities which are highly correlated. In the case of their application to Italian regions presented below, the economic variables are very highly correlated. Therefore, summing the modalities by row leads to duplication of the information characterizing the statistical units.

This redundancy in information is not, in itself, a negative element for statistical analysis. In some situations, for example in communication between human beings, it is this redundancy of information itself which leads to better understanding of diverse phenomena. In our case, however, the choice of variables is nearly always conditioned by the availability of statistical data. The correlation between the variables can derive, as is nearly always the case in operative reality, from this choice conditioned by the availability of data.

For the statistical value of the unit i, the synthetic index used in the applications can be written as:

$$I_i = (1/p)\sum_{j=1}^{p} x_{i,j}100 \quad (i = 1, 2, \ldots n)$$

One has:

$$0 \leq I_i \leq 100, \quad 0 \leq x_{i,j} \leq 1 \quad \forall i, j$$

This can be formally considered and written in the following way:

$$I_i = \sum_{j=1} x_{i,j} q_j 100 \quad (i = 1, 2, \ldots n)$$

where q_j (with $\sum q_j = 1$; $0 \leq q_j$) is a generic weight to be attributed to variable X. The choice of weights can be made in a subjective manner, there are no rules for their assignment in an objective manner. It is known that in many researches the weights assigned are all equal.

Another method to obtain a synthetic index which allows the ranking of statistical units is to substitute each category assumed by a variable with an integer number equal to the position occupied in the ascending (or descending) rank of variable X_j (j=1,2,..., p). Each statistical unit is therefore assigned p integer between 1 and n, where n is the number of statistical units.

Therefore, the synthetic index is:

$$R_i = \sum_j g_{i,j} \quad (i = 1, 2, \ldots, n)$$

where $g_{i,j}$ is the positions assigned to *ith* unit in the ranking in question.

This index will vary between p and np (in the case in which the statistical units are found in the *nth* position in each of the rankings), i. e.:

$$p \leq R_i \leq np$$

To have an index varying between 0 and 1 we can propose the following index:

$$R_i = (R_i - p)/(np - p)$$

This index could require some type of standardization; it does not take into account the value of the diverse modalities, only their relative positions.

In applications one uses particular short-cuts to adapt the general procedure to specific situations. For example, the possible points are divided into classes either a priori or after the fact according to particular quantiles of the distributions or standard secondary intervals (for example, multiple prefixes of the mean quadratic deviation).

Some situations have furnished interesting results both in graphical analysis and in the procedure of clustering statistical units based on the observed characters and which can serve, in the judgement of the researcher, as an indirect measure of complex concepts.

In social statistics these procedures are preceded by careful analysis of the nature of the data, for example, whether the analysis in question are made in descriptive or normative terms, whether the data are expressed in the same units of measurement and/or ranking in size, whether the parameters are scores of a subjective type, etc.

3 A new method for ranking n statistical units

It is known that an infinite number of linear transformations of data matrix $\mathbf{X}_{n,p}$ exist which allow the elimination of correlations between the columns of the matrix.

From the geometric point of view this means relocating the origins of the system of reference at the mean point and rotating the axes such that the matrix of correlation is reduced to the unitary matrix (Pompilj (1952), Casella and Berger (2002)).

Now, with matrix $\mathbf{X}_{n,p}$ we can consider the matrix of the percentages and then the matrix S of the deviation from the mean.

Matrix:

$$\mathbf{Y}_{n,p} = \mathbf{SQ\Lambda}^{-1/2}$$

has the unit matrix as correlation matrix; in this expression $\mathbf{Q}$ is the matrix of the eigenvectors associated with the correlation matrix of $\mathbf{X}_{n,p}$, $\mathbf{\Lambda}^{-1/2}$ has on its principal diagonal the inverse of the square roots of the eigenvalues of $\mathbf{X}'_{n,p}\mathbf{X}_{n,p}$.

The principal components are also equal to the number p of the variables if, as is supposed here, matrix $\mathbf{X}_{n,p}$ has rank equal to p. These are well known to be orthogonal. This means that the correlation coefficients calculated in the components are always null.

The indices proposed here measure the distance of each statistical unit from its origin. Each statistical unit is represented as a point in space of p dimensions. The distance is calculated with respect to the orthogonal reference system of the principal components. The sign is that of the first principal component.

$$R_i = (sgn\, c_{i1})(\sum_r c_{ir}^2)^{1/2}$$

where $c_{ir}(i = 1, 2, \ldots, n, r = 1, 2, \ldots, p)$ are the component scores associated to the ith unit.

The proposed indices take into account of all essential information contained in the matrix of the principal components in that considers all the components and not just those concerning data on the percentage of variation. Therefore the problem of choosing the number of components to be retained is overcome.

The sign of the first principal component is retained because it is, by definition, that one which will account for the majority of the variability with respect to the other components.

In general if there are p components, the n statistical units are represented as n points in space singled out from the p non-correlated components. The space is divided into two parts with p axes. The positive sign is assumed for the points found in the second part of the space in which the first component is positive.

4 Application

The goal is to measure the quality of life in Italian regions through the construction of a multidimensional ranking based on a set of objective and descriptive social indicators and a methodology for the construction of their syntheses. In this application, therefore, the study of the quality of life is conceived as an analysis of the standard of living, that is, through directly observable dimensions (objective) which can be analyzed in relation to the regions in all their complexity.

The goal of these analyses is to rank the Italian Regions with regard to many empirical social indicators of the quality of life through synthetic indices of these very indicators.

Eighteen distinct indicators were considered relative to these seven areas:

a) Social-Demographic: indices of old-age, life expectancy at birth for males and for females

b) Health: infant mortality, availability of hospital beds and their use

c) Jobs and Employment: non-members of the work force, members of the work force, unemployed

d) Social Safety: assaults, traffic accidents

e) Stress and Social Hardship: suicide

f) Economic Well-Being: per capita income, food consumption, private cars

g) Culture and Free Time: recreation consumption, TV subscription, university graduates.

The source of all data is the Italian National Statistical Institute (ISTAT). The period of reference is the three years of 2000, 2001 and 2002. The analysis is based on the mean values within these three years.

In Table 1 we find the matrix $\mathbf{Z}_{n,p}$ of the standardized deviation from the mean of the three year period 2000-2002. These standardized deviations are obtained, for every value, as ratios of the deviation from the mean and the standard deviation.

Table 2 display the correlation matrix. The correlation coefficients vary between -0.8 and 0.8. That means that many variables are strongly correlated.

In Table 3 we find the matrices of the 18 eigenvalues obtained with standard software. The first is the principal component reproducing a consistent variance quota equal to 49.7% and together with the two successive ones we obtain a percentage equal to 73.5% of the total.

With reference to the generic statistical unit a_i, to calculate the index proposed in paragraph 3 the quantity $\sum_r c_{ir}^2$ is taken directly from the matrix of principal components. This is given by the product of the matrix of standardized deviation multiplied by the matrix of the normalized eigenvalues.

In Table 4 are reported the values of the indices in ascending order. The first positions on the classification, which represent a higher quality of life, are occupied by the central, northern regions, in particular and in this order: Toscana, Valle D'Aosta, Liguria, Lazio, Umbria. At the bottom of the list we find the southern regions, Calabria, Basilicata, Sicilia and Sardegna.

In our application we have made reference to all of the principal components, 18 variables in all. We are not limited to either the first or to the first few principal components. In this way no information is lost, not even from the less important principal components. Naturally one could ignore the components where the calculation yields results which are truly negligible. For example, in our case the sum of the contributions of the eleventh principal component is less than one percent. Considering only the indices based on the first eleven principal components yields substantially the same results as does the calculation based on all the components.

The analysis was repeated but limited to only the first principal component. This is known as the single source factor. In our case the first principal components accounts for about 50% of the variance and therefore is highly significant.

The ranking derived in this manner is: Emilia Romagna, Trentino-Alto Adige,Veneto, Friuli-Venezia Giulia, Toscana, Valle D'Aosta, Marche, Umbria, Liguria, Piemonte, Lombardia, Lazio, Abruzzo, Molise, Sardegna, Puglia, Basilicata, Calabria, Sicilia, Campania .

The two classifications presented do not show differences. The Spearman's rank coefficient is equal to 0.67. This low value indicates that there is no great concordance between the two classifications. It is therefore undeniable that the method that uses the sign of the first principal component offers more complete information than that of the single source factor.

The southern regions hold the last positions in both methods. This occurs because in such regions most of the information is contained in the first principal component. For the northern regions the information is contained in both the first and the second principal components. Therefore, for these regions information is partially lost with the single source factor method.

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r
Pie	0,71	-0,67	-0,42	-0,64	-0,14	0,43	-0,51	0,75	-0,62	0,25	0,10	0,37	0,22	0,72	-0,44	1,83	0,06	-0,23
VdA	0,15	-0,67	-0,42	0,57	-0,14	0,86	-0,19	1,51	-0,80	-1,57	0,91	3,70	1,60	1,23	-1,67	-1,67	0,02	-
Lom	-0,15	-0,83	0,17	-0,64	0,19	0,40	0,48	1,00	-0,81	0,29	-1,51	0,04	-0,52	1,27	-0,59	0,71	0,39	0,33
TAA	-0,86	0,31	1,48	-1,02	1,34	-1,09	0,09	1,27	-1,00	-1,63	0,27	-0,48	1,03	1,39	-1,90	-0,29	0,20	-0,89
Ven	-0,19	-0,02	0,90	-1,20	1,01	0,19	0,14	0,97	-0,88	-1,70	0,10	-0,05	0,50	0,68	-0,92	0,96	0,50	-0,01
FVG	1,07	-0,83	-0,12	-1,39	1,01	-0,56	-0,08	0,53	-0,80	-0,38	0,27	0,07	1,85	0,62	-0,47	0,08	0,74	0,66
Lig	2,24	-0,50	-0,42	0,01	1,67	0,65	-1,36	-0,21	-0,41	0,51	-1,83	-0,60	0,46	0,48	-0,17	0,21	0,78	-0,01
Emi	1,19	0,31	0,46	-0,55	0,02	2,23	-1,60	1,37	-0,85	-1,40	-0,22	0,27	0,63	1,17	-0,88	1,71	0,76	1,77
Tos	1,10	0,80	0,46	-1,20	0,19	-0,07	-1,06	0,50	-0,64	0,62	-0,86	0,25	-0,11	0,53	-0,36	0,83	0,91	0,88
Umb	0,94	1,28	0,75	-1,02	-0,14	-0,27	-1,04	0,04	-0,57	-0,05	-0,06	0,54	1,16	-0,06	-0,29	0,21	0,55	0,88
Mar	0,57	1,77	1,48	-0,36	0,19	-0,73	-1,02	0,46	-0,72	-0,26	-0,86	0,16	-0,19	0,10	-0,17	1,33	0,79	0,88
Laz	-0,45	-0,34	-0,56	0,20	-0,14	0,73	0,54	0,06	-0,02	-0,08	-0,86	0,71	-0,80	0,63	-0,25	-0,29	-0,31	1,43
Abr	-0,01	0,80	0,75	0,20	0,02	-0,31	-0,42	-0,63	-0,46	1,18	0,10	-0,16	-0,43	-0,53	0,20	-0,29	0,48	0,22
Mol	0,08	0,80	0,75	0,57	0,68	1,44	-0,85	-0,63	0,46	1,40	0,10	-0,73	0,06	-0,84	0,69	-1,17	0,20	-1,22
Cam	-1,61	-2,62	-2,61	0,76	-2,61	0,65	2,02	-1,39	1,67	0,44	0,27	-0,48	-1,62	-1,30	1,47	-0,54	-2,60	0,10
Pug	-1,20	0,63	-0,27	1,41	-0,14	0,47	1,36	-1,37	0,71	-1,04	1,56	-0,91	-1,58	-1,28	1,21	-0,42	0,56	-0,56
Bas	-0,66	0,31	-0,12	1,13	0,02	-1,45	0,31	-1,24	0,81	0,63	2,37	-0,93	-0,39	-1,14	1,17	0,08	-0,01	-2,22
Cal	-1,03	0,96	-0,56	1,41	-1,79	-1,36	1,06	-1,14	2,07	0,48	0,27	-0,81	-1,41	-1,48	1,40	-0,79	-1,96	-1,45
Sic	-1,15	-0,50	-1,88	2,06	-1,29	-1,70	0,76	-1,56	1,60	1,38	-1,03	-0,39	-1,00	-1,29	1,47	-1,54	-1,94	-0,45
Sar	-0,75	-0,99	0,17	-0,27	0,02	-0,49	1,38	-0,28	1,25	0,92	0,91	-0,59	0,54	-0,91	0,50	-0,92	-0,10	-0,12

Table 1. Standardized values. a) indices of old-age, b) life expectancy at birth for males c) life expectancy at birth for females d) infant mortality e) availability of hospital beds f) occupancy of hospital beds g) non-members of the work force h) members of the work force i) unemployed j) assaults k) traffic accidents l) suicides m) per capita income n) food consumption o) number of cars p) recreation consumption q) TV subscription r) number university graduates.

a	1,00																	
b	0,24	1,00																
c	0,38	0,71	1,00															
d	-0,59	-0,07	-0,60	1,00														
e	0,60	0,28	0,71	-0,59	1,00													
f	0,36	-0,19	0,01	-0,20	0,15	1,00												
g	-0,88	-0,52	-0,58	0,49	-0,59	-0,32	1,00											
h	0,51	0,04	0,54	-0,77	0,54	0,37	-0,52	1,00										
i	-0,67	-0,23	-0,65	0,79	-0,72	-0,34	0,70	-0,86	1,00									
j	-0,07	-0,05	-0,31	0,37	-0,30	-0,27	0,10	-0,64	0,53	1,00								
k	-0,45	0,02	-0,01	0,27	-0,15	-0,17	0,38	-0,25	0,29	-0,19	1,00							
l	0,26	-0,10	0,02	-0,19	0,03	0,31	-0,26	0,60	-0,46	-0,42	-0,05	1,00						
m	0,60	0,04	0,48	-0,67	0,65	0,13	-0,55	0,72	-0,68	-0,41	0,01	0,49	1,00					
n	0,56	-0,04	0,44	-0,73	0,59	0,38	-0,54	0,95	-0,89	-0,59	-0,39	0,56	0,65	1,00				
o	-0,49	-0,07	-0,57	0,74	-0,63	-0,30	0,52	-0,96	0,89	0,67	0,22	-0,60	-0,76	0,93	1,00			
p	0,53	0,20	0,39	-0,63	0,30	0,24	-0,51	0,48	-0,58	-0,27	-0,26	-0,08	0,14	0,49	-0,36	1,00		
q	0,69	0,47	0,79	-0,65	0,85	0,28	-0,69	0,58	0,81	-0,33	-0,07	0,15	0,58	0,57	-0,60	0,52	1,00	
r	0,49	-0,04	0,12	-0,54	0,09	0,48	-0,38	0,53	-0,52	-0,27	-0,55	0,83	0,27	0,55	-0,47	0,48	0,33	1,00

Table 2. Correlation matrix (average 2000-2002) a) indices of old-age, b) life expectancy at birth for males c) life expectancy at birth for females d) infant mortality e) availability of hospital beds f) occupancy of hospital beds g) non-members of the work force h) members of the work force i) unemployed j) assaults k) traffic accidents l) suicides m) per capita income n) food consumption o) number of cars p) recreation consumption q) TV subscription r) number university graduates.

	ABS	%VAR	%CUM
1	8.95	49.69	49.69
2	2.40	13.31	63.01
3	1.90	10.53	73.54
4	1.08	5.99	79.52
5	0.97	5.40	84.92
6	0.85	4.73	89.66
7	0.58	3.22	92.88
8	0.45	2.49	95.37
9	0.31	1.75	97.11
10	0.27	1.48	98.59
11	0.08	0.46	99.06
12	0.07	0.40	99.45
13	0.04	0.22	99.67
14	0.02	0.12	99.80
15	0.02	0.10	99.90
16	0.01	0.06	99.96
17	0.01	0.03	99.98
18	0.00	0.02	100.00

Table 3. Matrix of Eigenvalues, percentage (simple and cumulative).

REGION	R_i
Toscana	4,248
Liguria	4,248
Valle d'Aosta	4,248
Lazio	4,244
Umbria	4,218
Friuli Venezia Giulia	4,201
Lombardia	4,179
Emilia Romagna	4,162
Piemonte	4,121
Trentino-Alto Adige	4,083
Marche	3,660
Veneto	3,593
Abruzzo	-4,026
Campania	-4,168
Molise	-4,170
Puglia	-4,186
Calabria	-4,203
Basilicata	-4,203
Sicilia	-4,217
Sardegna	-4,248

Table 4. Ranking of Italian regions using index R.

References

BARNETT, V. (1976): The ordering of multivariate data. *Journal of Royal the Statistical Society Series A 139 (3), 318-358.*

CASELLA, G. and BERGER, R.L. (2002): *Statistical Inference (second edition).* Duxbury Press.

CHATFIELD, C. and COLLINS, A.J. (1980): *Introduction to multivariate analysis.* Chapman and Hall, London.

ESCOUFIER, Y. (2006): Operator related to a data matrix: a survey. In: A. Rizzi and M. Vichi (Eds): *Compstat 2006.* Physica Verlag, Heidelberg, 285-297.

POMPILJ, G. (1952): *Teoria dei campioni.* Veschi, Roma.

RIZZI, A. (1988): Un metodo di graduazione di più unità statistiche. *Statistica applicata 21 (1), 49-64.*

About Relational Correlations

Yves Schektman

1bis rue des potiers, 31000 Toulouse, France

Abstract. Using particular euclidean geometries called relational, one can go deeper into the usual concepts as well as the Data Analysis methods and even generalizes or proposes new ones. Inner products in these particular euclidean spaces are built using correlations between principal components of observed sets of variables. A summary of the main topics on an essay in process is proposed.

1 Introduction

More than three years ago, E. Diday friendly advised me to leave my very comfortable and self imposed retirement many years ago. So, an essay is in process. Structured main ideas may be found here. Interval Data are naturally treated in, because (i) this gives me the opportunity to acknowledge Edwin who suggested me, in particular, to think about them (Cazes et al. (1997)) and (ii) what one might name Relational Geometry (RG), briefly described here, needed a lightly deepening, in order to take into account these kind of data.

Observed and experimental data are collected on Statistic Units (SU) through sets of variables. In the data array, both SU and variables define each other. Moreover, in statistical interpretation, variables and SU mutually emphasize each other. In probability models, SU are at the service of variables and SU subsets are at the service of controlled experiment factors. In the sixties, thanks to the emergence of computing tools and following the new practices, geometrical models rendered to SU its specific part in Data Analysis.

Accepting that the role of SU as well as that of variables depends on the kind of analyses, one had to go further. Since spaces associated to sets of variables are put into a unique variable-space F, the same had to be done for the corresponding sets of SU, i.e more precisely for sets of weighted points or Massive Shape (MS). Let us recall the "double duality schema" or the "mixed scatter-plots in Correspondence", where matched MS were in different SU-spaces. Besides, that embedding was usefull in order to extend my results on constrained principal components.

To make the SU-space E useful, a semantic solution was chosen, i.e to enrich it with an Inner Product (IP), called Relational (RIP), briefly defined in \$3. Interval Data may be studied in RG, because usual concepts (inertia, ...) and well known results on finite MS are still valid with a well behaved weight-function $\{ \mathrm{f(t)} \, ; (\mathrm{t} \in \mathrm{D}) / \mathrm{f(t)} \in [0,1] \, , \int_D \mathrm{f(t)} \, \mathrm{dt} = 1\}$, when D is not finite. In the properties described, weight-functions are naturally supposed well behaved.

For "paper" saving reasons, mathematical singularities, mainly due to qualitative variables, are generally avoided. The same symbol is used for a vector and its image in a greater dimension space, similarly for a bilinear mapping and its associated matrix. The bounds on the domains of variation are not given when they are not useful, symbolic notations are used like $\mathrm{P}_y[\,\mathrm{N}_x]$ instead of $\{\,\mathrm{P}_y(x)\,;x \in \mathrm{N}_x\}$, as well as integrals on matrices and shorthand notations. Finally, D means Definition, P Property, C Consequence, N Note, iof "if and only if", $\forall$ j "for all j", "$\subset$" means also "is located in".

2 Mahalanobis Inner Products (MIP) - Sterilized variables and equalized Inertia Structure (IS)

In a RG, the role of MIP is of great importance. As far as usual Data Analysis concepts and methods are concerned, MIP are a semantic reference which guides us to formulate new view points.

To the set of variables $\{\,x^j\,;(\mathrm{j}{=}1,\mathrm{p})\,\}$ it corresponds:

(i) a vector function $[\,x(t)=\{x^j(t)\,,(\mathrm{j}{=}1,\mathrm{p})\,\}\,;(\mathrm{t}\in\mathrm{D})\,]$ from D to $\mathrm{E}_x=\mathrm{R}^p$ and it is denoted by M_x the IP in E_x or its matrix.

(ii) a MS $\mathrm{N}_x=\{\,[\,x_t={}^t[x_t^1..\,x_t^j..\,x_t^p]\,,\mathrm{f(t)}\,]\,;(\mathrm{t}\in\mathrm{D})\,\}\subset\mathrm{E}_x$, where $x_t^j=x^j(t)$.

Without lost of generality $\{\,\int_D \mathrm{f(t)}\; x_t^j\;\mathrm{dt}=0\;\;\forall\mathrm{j}\,\}$ is supposed, that it is written as $\int\mathrm{f(t)}\,x_t\,\mathrm{dt}=0$.

D1: The usual matrix for the MIP is V_x^{-1}, where $[\,\mathrm{V}_x]_r^s=\int\mathrm{f(t)}\;\;[x_t\,{}^tx_t]_r^s\;\;\mathrm{dt}$.

P1-D2: For the IS $[\mathrm{N}_x\,,\mathrm{M}_x]$, $\mathrm{M}_x=\;\mathrm{V}_x^{-1}$ iof any orthonormal basis in $[\,\mathrm{E}_x\,,\mathrm{M}_x]$ is a Normalized Principal Vectors (NPV) basis, associated to the Principal Moment (PM) of value 1 and order p. It is said that N_x is equalized by V_x^{-1}.

D3-P2: For a given IS $[\,\mathrm{N}_x\,,\mathrm{M}_x]$, denoting by $\{\,\mathrm{e}_j^{M_x}\}$ the dual basis of the canonical basis $\{\mathrm{e}_j\}$ in $[\,\mathrm{E}_x\,,\mathrm{M}_x]$, by P_{e_j} the orthogonal projection operator onto the Δe_j axis , by I the Inertia, by PI the Inertia Product, and as

$$\text{- VE}[\,x^j/\;\mathrm{M}_x]=\int\mathrm{f(t)}\,[\,\mathrm{M}_x(e_j^{M_x}\;/\parallel e_j^{M_x}\parallel\,,\,x_t)\,]^2\,\mathrm{dt}\;\;(\,=\mathrm{I}\{\,\mathrm{P}_{e_j^{M_x}}[\mathrm{N}_x]\,/\,\mathrm{M}_x\}\,)$$
$$=\mathrm{Var}[x^j]\qquad\qquad\text{if }\{\mathrm{e}_j\}\text{ is an orthonormal basis}$$

$$\text{- PI}e_re_s^{M_x}[\,\mathrm{N}_x/\,\mathrm{M}_x]=\int\mathrm{f(t)}\;\mathrm{M}_x(\,e_r/\parallel e_r\parallel\,,\,x_t)\;\;\mathrm{M}_x(\,e_s^{M_x}\;/\parallel e_s^{M_x}\parallel\,,\;x_t)\;\mathrm{dt}$$

$$\text{- AE}[\,x^r;x^s/\,\mathrm{M}_x]=\mathrm{PI}e_re_s^{M_x}[\mathrm{N}_x/\,\mathrm{M}_x]\,/\,[\,\mathrm{I}\{\,\mathrm{P}e_r[\,\mathrm{N}_x]\,/\,\mathrm{M}_x\}\;\;\mathrm{VE}[\,x^s/\mathrm{M}_x]\,]^{1/2}$$
$$=\rho[\,x^r,\,x^s]\qquad\text{if }\{\mathrm{e}_j\}\text{ is an orthonormal basis ,}$$

where Var and ρ denote usual variance and correlation coefficients

then $\{\mathrm{VE}[\,x^j/\,\mathrm{MIP}]=1\;\forall\mathrm{j}\,\}$ and $\{\,\mathrm{AE}[\,x^r;x^s/\,\mathrm{MIP}]=0\;\forall(r,s)\,;\,r\neq s\,\}$.

It is said: $\mathrm{M}_x=\mathrm{MIP}$ eliminates the Variability (VE) and the Association Effects (AE) of the set of variables $\{x^j\}$ on the MS N_x .

N1: One can show that if $\mathrm{AE}[x^r;x^s\;/\,\mathrm{M}_x]=0\;\;(\,\forall\,\mathrm{s}\,/\,\mathrm{s}{\neq}\mathrm{r})$ then generally $\Delta\mathrm{e}_r$ is a principal axis of $[\,\mathrm{N}_x\,,\mathrm{M}_x]$ and $\{\,\mathrm{M}_x\;=\;k^2\;\mathrm{MIP}\,/\,(k\in\mathrm{R})\}$.

So, to eliminate AE is "quite nearly equivalent" to MIP.

D4-P3: Being A any regular linear operator in E_x , it is written $[\mathrm{N}_x , \mathrm{M}_{xA} = {}^t\mathrm{A}\,\mathrm{M}_x\,\mathrm{A}] \sim [\mathrm{N}_{xA} = \mathrm{A}\ \mathrm{N}_x = \{ x_t^A = \mathrm{A}(x_t) ; (x_t \in \mathrm{N}_x)\} , \mathrm{M}_x]$ because these two IS , called similars, may be considered as two expressions of the same IS in two different bases.

C1: $[\mathrm{N}_x , \mathrm{MIP}] \sim [\mathrm{N}_{x_n} = \{ x_t^n = (\mathrm{V}_x\,\mathrm{M}_x)^{-1/2}(x_t) ; (x_t \in \mathrm{N}_x)\} , \mathrm{M}_x]$. So N_{x_n} is equalized by any M_x . $\{ x_n^j = [\,[(\mathrm{V}_x\ \mathrm{M}_x)^{-1/2}\,]_j\ x_t ; (x_t \in \mathrm{N}_x)\,] ; (\mathrm{j}{=}1,\mathrm{p})\}$ is so that $\mathrm{V}_{x_n} = \mathrm{M}_x^{-1}$. Thus it is said that variables $\{x_n^j\}$ are sterilized.

N2: If $\{x^j\}$ are items of a qualitative variable x then V_x is not regular. In this case, the Khi-square IP (χ_x^2) has the same properties as V_x^{-1} , with some minor differences. Both are denoted by MIP. The neutral IP , used in $5 , whose matrix is the unity matrix, is called Canonical IP (CIP).

3 Relational Inner Products (RIP)

Let **(i)** $\{ (\lambda_j \neq 0\ ,\ c_j \in \mathrm{E}_x) ; (\mathrm{j}{=}1,\mathrm{p}) \}$ and $\{ (\mu_k \neq 0\ , \mathrm{d}_k \in \mathrm{E}_y) ; (\mathrm{k}{=}1,\mathrm{q}) \}$ be the PM and the NPV of $[\,\mathrm{N}_x , \mathrm{M}_x\,]$ and $[\,\mathrm{N}_y , \mathrm{M}_y\,]$ respectively,
(ii) $\{ \mathrm{C}^j = \{ \mathrm{M}_x(c_j , x_t) ; (\mathrm{t} \in \mathrm{D}) \} ; (\mathrm{j}{=}1,\mathrm{p}) \}$ and $\{ \mathrm{D}^k = \{ \mathrm{M}_y(d_k , y_t) ; (\mathrm{t} \in \mathrm{D}) \} ; (\mathrm{k}{=}1,\mathrm{q}) \}$ be the corresponding Principal Components (PC).

D5: In $[\,\mathrm{E} = \mathrm{E}_x \oplus \mathrm{E}_y , \mathrm{M}[\mathrm{M}_x , \mathrm{M}_y]\,]$, M is the RIP for $(\,[\mathrm{N}_x , \mathrm{M}_x] , [\mathrm{N}_y , \mathrm{M}_y]\,)$, denoted by R_{xy} , iof $\cos_M(\,c_j , d_k) = \rho[\,C^j , D^k]$ $[\forall\,(\mathrm{j}{=}1,\mathrm{p}) , \forall\,(\mathrm{k}{=}1,\mathrm{q})\,]$. (1)

P4: $\mathrm{ER}_{xy}[\mathrm{M}_x , \mathrm{M}_y]$ being Extra-diagonal matrix of $\mathrm{R}_{xy}[\mathrm{M}_x , \mathrm{M}_y]$, it comes : (1) $\Leftrightarrow \{\mathrm{ER}_{xy}[\mathrm{M}_x , \mathrm{M}_y] = \mathrm{M}_x\ (\mathrm{V}_x\ \mathrm{M}_x)^{-1/2}\ \mathrm{V}_{xy}\ \mathrm{M}_y\ (\mathrm{V}_y\ \mathrm{M}_y)^{-1/2}\}$. (2)

Syntactical Coherence: For bases changing in E_x and/or in E_y , the different expressions for the matrix of $\mathrm{R}_{xy}[\mathrm{M}_x , \mathrm{M}_y]$ are those of the usual change of bases. Besides, if some PM are multiple then (2) remains the same.

Semantic Coherence: Fortunately the following equalities,

$$\mathrm{Q} = \mathrm{R}_{(x\cup y)z}[\mathrm{R}_{xy}[\mathrm{M}_x , \mathrm{M}_y]\ , \mathrm{M}_z\,] = \mathrm{R}_{x(y\cup z)}[\mathrm{M}_x\ , \mathrm{R}_{yz}[\,\mathrm{M}_y\ , \mathrm{M}_z\,]\,]$$
$$= \mathrm{R}_{xyz}[\,\mathrm{M}_x\ , \mathrm{M}_y\ , \mathrm{M}_z\,]$$

hold and are valid for any number of sets of variables.

N3: Having $\mathrm{E} = \mathrm{E}_x \oplus \mathrm{E}_y \oplus .. \oplus \mathrm{E}_z$, in a well behaved mathematical context , if U_s is the local isometry from $[\,\mathrm{E}_s , \mathrm{M}_s]$ into $[\,\mathrm{F} , \mathrm{N}\,]$, where N is the covariance IP, so that the images of the NPV are the normalized PC , then $\{\mathrm{M} = \mathrm{R}_{xy..z}[\mathrm{M}_x , \mathrm{M}_y ,.., \mathrm{M}_z\,]\,\}$ iof $\mathrm{U} = \sum \mathrm{U}_s\,\mathrm{Pr}_s$, where Pr_s is the canonical projection from E onto E_s , is a local isometry from $[\,\mathrm{E} , \mathrm{M}\,]$ into $[\,\mathrm{F} , \mathrm{N}\,]$. Besides, (i) if $\{\,\mathrm{M}_s = \mathrm{V}_s\}$ then the variable vectors of the set of variables s are the images of the canonical basis vectors in E_s , and (ii) if $\{\mathrm{M}_s = \mathrm{MIP}\ (\forall\, s)\}$ then the canonical variables are the images of the canonical vectors.

C2-"Relational Meccano": One may update any set of variables without distorting the restriction of RIP to the remaining ones.

P5-RIPcharacterizations: Let Hρ be $\sum [\,\rho^c_h(\mathrm{M})\,]^2 = \sum [\,\rho^c_h\,]^2$, where $\{\,\rho^c_h\}$ are the usual canonical correlations and $\{\rho^c_h(\mathrm{M})\}$ those of $[\,\mathrm{E}_x\,,\mathrm{E}_y\,,\mathrm{M}\,]$ defined in $ 4., P_y be the orthogonal projection operator onto E_y and y^n_t being defined in the same manner as x^n_t ,

$\{\,\mathrm{M} = \mathrm{R}_{xy}[\,\mathrm{M}_x\,,\mathrm{M}_y]\,\}$ is true iof :

(i) $(\,\mathrm{H}\rho\,)$ holds and $\int \mathrm{f(t)} \parallel y^n_t - x^n_t \parallel^2_M$ dt is minimum.

(ii) $\int \mathrm{f(t)} \parallel y^n_t - \mathrm{P}_y(x^n_t) \parallel^2_M$ dt is minimum (= q - ($\sum [\rho^c_h]^2$).

(iii) $(\mathrm{V}_y\ \mathrm{M}_y\,)^{1/2}\,\mathrm{Pr}_y\mathrm{P}_y[\,\mathrm{N}_{x_n}] = \mathrm{V}_{yx}\,\mathrm{V}^{-1}_x\,[\mathrm{N}_x]$, i.e iof the q coordinates of $(\mathrm{V}_y\ \mathrm{M}_y)^{1/2}\,\mathrm{Pr}_y\,\mathrm{P}_y(x^n_t)$ are the adjusted values of $\{\,y^k_t\,;(\mathrm{k=1,q})\}$ obtained by the q multiple linear regressions $[\,y^k/x^1..x^p\,]\,(\mathrm{k=1,q})$.

Particular case: Using $\mathrm{Pr}_y\,\mathrm{P}_y[\,\mathrm{N}_x]$ in (iii) and $(x_t\,,y_t)$ instead of $(x^n_t\,,y^n_t)$ in (i and ii), if $\{\,\mathrm{M}_x = \mathrm{MIP}$ and $\mathrm{M}_y = \mathrm{MIP}\,\}$ then P5 holds.

P6-Strong property of RIP: If $\{\,\mathrm{M} = \mathrm{R}_{xy}[\,\mathrm{M}_x\,,\mathrm{M}_y\,]\,\}$ then $\int \mathrm{f(t)} \parallel y^n_t - \mathrm{b}\ \mathrm{P}_y(x^n_t) \parallel^2$ dt is minimum for and only for $\mathrm{b} = 1$.

As $\mathrm{N}_{x_n} = \sum \mathrm{P}_{c^n_j}[\,\mathrm{N}_{x_n}]$, where $\{c^n_j\}$ is any orthonormal basis in E_x , let set $\mathrm{N}_{x_{cn}} = \{\,\sum a_j\,\mathrm{P}_{c^n_j}[\,\mathrm{N}_{x_n}]\,/\,(a_j \geq 0\ \ \forall \mathrm{j}\,)\}$.

P7:(i) If $(\,c^n_j = c_j$ and $a_j = \lambda^{1/2}_j\ \ \forall \mathrm{j}\,)$ then $\mathrm{N}_{x_{cn}} = \mathrm{N}_x$.

(ii) $\{\,(a_j)^2,\,c^n_j\}$ are the non ordered PM and NPV of $[\,\mathrm{N}_{x_{cn}},\,\mathrm{M}_x]$.

(iii) Given M_x and $[\,\mathrm{N}_y\,,\mathrm{M}_y]$ then $\mathrm{R}_{xy}[\,\mathrm{M}_x\,,\mathrm{M}_y]$ is an invariant for and only for all the MS as $\mathrm{N}_{x_{cn}}$.

D6: All the IS as $[\,\mathrm{N}_{x_{cn}},\,\mathrm{M}_x]$ are called Relationally Compatible (RC) and all the IS as $[\,\mathrm{N}_{x_{cn+}} = [\,\mathrm{N}_{x_{cn}}\,/\,(\,a_j \geq\ a_{j+1}\ \forall \mathrm{j}\,)]\,,\,\mathrm{M}_x\,]$, i.e having $\{c^n_j\}$ as common ordered NPV basis, are Relationally Hyper Compatible (RHC).

4 [Relational] Canonical Vectors (CV)

D7-P8: $(\,\alpha_h \in \mathrm{E}_x\ ,\,\beta_h \in \mathrm{E}_y)$ are the CV of $[\,\mathrm{E}_x\,,\mathrm{E}_y\,,\mathrm{M}[\mathrm{M}_x\,,\mathrm{M}_y]\,]$

iof $\mathrm{P}_y(\alpha_h) = \rho^c_h(\mathrm{M})\ \beta_h$ and $\mathrm{P}_x(\beta_h) = \rho^c_h(\mathrm{M})\ \alpha_h$, where $\rho^c_h(\mathrm{M}) \geq 0$,

or iof $\mathrm{M}^{-1}_x\ \ \mathrm{M}_{xy}\ \ \mathrm{M}^{-1}_y\ \ \mathrm{M}_{yx}\,\alpha_h = [\ \rho^c_h(\mathrm{M})\ ]^2\ \alpha_h$

and $\mathrm{M}^{-1}_y\ \ \mathrm{M}_{yx}\ \ \mathrm{M}^{-1}_x\ \ \mathrm{M}_{xy}\ \beta_h = [\ \rho^c_h(\mathrm{M})\,]^2\ \beta_h$, where $\mathrm{M}_{yx} = \mathrm{Pr}_y\,\mathrm{M}\,\mathrm{In}_x$.

If $\{\,\mathrm{M} = \mathrm{R}_{xy}\}$ then the CV, called relational CV, are the solutions

of $(\,V_x\ \mathrm{M}_x)^{-1/2}\ V_{xy}\ V^{-1}_y\ V_{yx}\ \mathrm{M}_x\ (\,V_x\ \mathrm{M}_x)^{-1/2}\ \ \alpha_h = [\ \rho^c_h\ ]^2\ \ \alpha_h$

and of $(\,V_y\ \mathrm{M}_y)^{-1/2}\ V_{yx}\ V^{-1}_x\ V_{xy}\ \mathrm{M}_y\ \ (\,V_y\ \mathrm{M}_y)^{-1/2}\ \ \beta_h = [\ \rho^c_h\ ]^2\ \ \beta_h$.

If $\{\,\mathrm{M}_x = \mathrm{MIP}$ and $\mathrm{M}_y = \mathrm{MIP}\}$ then the relational CV are identical to usual CV $(\,\alpha^u_h\ ,\ \beta^u_h)$ and $(\,\alpha_h = (\mathrm{V}_x\ \mathrm{M}_x)^{1/2}\,\alpha^u_h\ ,\ \beta_h\ = (\mathrm{V}_y\ \mathrm{M}_y)^{1/2}\,\beta^u_h\,)$.

If the shapes of N_x or N_y , or the expressions of M_x or M_y , is modified, in any fashion, then the relative location of E_x and E_y generally changes. This is not the case for all the IS $[\mathrm{N}_{x_{cn}}, \mathrm{M}_x]$. Now P9 goes deeper and specifies P3.

P9: For any given $\{c_j^n\}$, if $\mathrm{N}_{x'_{cn}} = \{ \sum a'_j \, \mathrm{P}_{c_j^n}[\mathrm{N}_{x_n}] \,/\, a'_j \geq 0 \ \ \forall j \ \}$ then $[\mathrm{N}_{x'_{cn}}, \mathrm{M}_{x'_{cn}}] \sim [\mathrm{N}_{x_{cn}}, \mathrm{M}_x]$ for and only for $\mathrm{M}_{x'_{cn}} = \mathrm{M}_x \sum (\, a_j \,/ a'_j \,)^2 \, \mathrm{P}_{c_j^n}$.

5 Relational correlations

Within a RG, one cannot generalize, for two multidimensional MS, the usual angular point of view of correlation between two variables x and y , i.e $\rho[x,y] = cos_R(\, e_x \,, e_y)$. On the contrary, one can generalize the usual variability point of view, i.e $\rho^2[x,y] = (\, Var[x] - Var[x/y]\,) \,/\, Var[x]$, in terms of inertia as : ($\mathrm{I}[\mathrm{N}_x]$ - $\mathrm{I}\{\, \mathrm{P}_y^{\perp}[\mathrm{N}_x]\,\}$) / $\mathrm{I}[\mathrm{N}_x]$ $=$ $\mathrm{I}\{\, \mathrm{P}_y[\mathrm{N}_x]\,\}$ / $\mathrm{I}[\mathrm{N}_x]$.

P10: If $\{\, \mathrm{M} = \mathrm{R}_{xy}[\, \mathrm{M}_x \ , \ \mathrm{M}_y]\,\}$, then

(i) $\mathrm{I}\{\, \mathrm{P}_y[\, \mathrm{N}_{x_n}] \ \} = \sum (\rho_h^c)^2$.

(ii) $\mathrm{I}\{\, \mathrm{P}_y[\, \mathrm{N}_{x_{cn}}]\,\} = \sum (\rho_h^c)^2 \sum \{\, a_j \ cos(c_j \,, \alpha_h = [\sum (a_j)^2 \, \mathrm{P}_{c_j^n}]^{1/2} \ \alpha_h^u\,)\}^2$ (3)

(iii) $\mathrm{I}\{\, \mathrm{P}_y[\, \mathrm{N}_x \ ] \ \} = \mathrm{trace}[\, V_{xy} \ V_y^{-1} \, V_{yx} \ \mathrm{M}_x]$

$= \mathrm{trace}[\, V_{xy} \ \chi_y^2 \ V_{yx} \ \mathrm{M}_x]$ if y is a qualitative variable.

P11: $\sum (\rho_h^c)^2 = \mathrm{I}\{\, \mathrm{P}_y[\mathrm{N}_{x_n}] \,/\, \mathrm{M}_x\} = \mathrm{I}\{\, \mathrm{P}_y[\, \mathrm{N}_x] \,/\, \mathrm{M}_x = \mathrm{MIP}\,\}$
$= \mathrm{I}\{\mathrm{P}_x[\mathrm{N}_{y_n} = (V_y \, M_y)^{-1/2} \, \mathrm{N}_y] \,/\, \mathrm{M}_y\} = \mathrm{I}\{\mathrm{P}_x[\mathrm{N}_y] \,/\, \mathrm{M}_y = \mathrm{MIP}\}$.

C3: According to the nature of the variables and the values of p and q ,

(i) P11 synthesizes the usual symmetrical association indices,

(ii) $\mathrm{I}\{\, \mathrm{P}_y[\, \mathrm{N}_x] \,/\, \mathrm{M}_x = \mathrm{CIP}\,\}$ synthesizes [cf. P10-iii] the usual dependence indices $(y \rightarrow x)$ [Stewart-Love, Goodman-Kruskal].

If $\{c_j^n \neq c_j\}$ then generally it is impossible to have $\mathrm{N}_{x_{cn}} = \mathrm{N}_x$. So, if one wants to define non symmetrical correlations, including the usual ones, then he has to suppose $\{c_j^n = c_j\}$ and to go away reasonably from $[\,\mathrm{N}_{x_n}$, CIP] towards $[\,\mathrm{N}_x$, CIP] using $[\,\mathrm{N}_{x_{cn}}$, CIP] : starting from statistics for testing independence in probability or no effect of a factor, one goes towards classical measures of dependence, giving progressively life to AE and VE of $\{x^j\}$ via $\{a_j\}$. This process is reasonable because all the $[\,\mathrm{N}_{x_{cn}}$, CIP] are RC, so ($\forall$ $\{a_j\}$) orthogonal projections are calculated with the same RIP [cf. P7-(iii)].

D8: Given $\{a_j\}$, $\mathrm{RAC}[\,\mathrm{N}_y \rightarrow \mathrm{N}_x] = \mathrm{I}\{\, \mathrm{P}_y[\, \mathrm{N}_{x_{cn}}] \,/\, \mathrm{M}_x = \mathrm{CIP}\}$ is the Relational Association Coefficient (with respect to CIP and $\{a_j\}$).
$\mathrm{RAC}[\,\mathrm{N}_y \rightarrow \mathrm{N}_x] \,/\, \mathrm{I}[\,\mathrm{N}_{x_{cn}} \,/\, \mathrm{M}_x = \mathrm{CIP}\,]$ is the corresponding Relational Correlation Coefficient (RCC).

N4: **(i)** One may use, extended CIP , i.e M_x so that $[\mathrm{M}_x]_r^s = \delta_{rs} \,/\, Var[x^r]$ which eliminates VE of $\{x^j\}$ and also go further than $\mathrm{N}_{x_{cn}} = \mathrm{Nx}$.

(ii) In order to simplify calculi one may use $\mathrm{M}_y = \mathrm{MIP}$ [cf.(2) and P10-(iii)].

(iii) In a RG, (3) is one of the expressions of the algebraic structure of the RAC and an illustration of the AE and the VE. Besides the maximum of (3) is obtained for and only for $\{c_h = \alpha_h \, ; \, (\,\forall\, h \,/\, \rho_h^c \neq 0\,)\}$.

(iv) To choose the "good" $\mathrm{N}_{x_{cn}}$, i.e the "good" $\{a_j\}$, one may, (a) want to have RHC, so one must use $\mathrm{N}_{x_{cn+}}$, in order to smooth the MS transformation with also the help of AE, VE and RAC gradient variations too, (b) be helped by endogenous or exogenous criteria, etc...

(v) One can deduce from P12 below, that $\mathrm{RAC}[\mathrm{N}_y \rightarrow \mathrm{N}_x]$ is the part of $\mathrm{I}[\,\mathrm{N}_{x_{cn}} \,/\, \mathrm{CIP}\,]$ (= VP) "viewed" from N_y, i.e linearly "explained by E_y",

$\mathrm{OP} = \sum \, [\; \mathrm{I}\{\, \mathrm{P}_{\alpha_h}[\mathrm{N}_{x_{cn}}]\,\} \; ; \, (\,\forall\, h \,/\, \rho_h^c \neq 0\,)\,]$ is the "observable" part ,

$\mathrm{HP} = \sum \, [\; \mathrm{I}\{\, \mathrm{P}_{\alpha_h}[\mathrm{N}_{x_{cn}}]\,\} \; ; \, (\,\forall\, h \,/\, \rho_h^c = 0\,)\,]$ is the "hidden" part

and finally NPP = OP - VP is the "no perceived" part, i.e the part of OP not linearly explained by E_y. Of course, VP, HP, OP and NPP change with $\{a_j\}$ and may give useful complementary informations.

(vi) If p=1 then all the $\mathrm{RCC}[\mathrm{N}_y \rightarrow \mathrm{N}_x]$ are equal, besides if q=1 they are equal to all the $\mathrm{RCC}[\mathrm{N}_x \rightarrow \mathrm{N}_y]$. Note that for two variables x and y the HP and the AE do not exist. In this case, one possibility to have non symmetrical RCC would be to code x and y into qualitative variables in order to create HP and AE again and then to get "richer association" too.

RAC decomposition into "Russian dolls": Briefly, one may do

$$\mathrm{N}_{x_{cn}}^1 = \mathrm{P}_y[\mathrm{N}_{x_{cn}}] \quad , \quad \mathrm{I}[\mathrm{N}_{x_{cn}}] = \mathrm{I}[\mathrm{N}_{x_{cn}}^1] \;+ \mathrm{I}\{\mathrm{P}_y^{\perp}[\mathrm{N}_{x_{cn}}]\,\} \;,$$
$$..$$
$$\mathrm{N}_{x_{cn}}^{\tau+1} = \mathrm{P}_y[\mathrm{N}_{x_{cn}}^{\tau}\,] \quad , \quad \mathrm{I}[\mathrm{N}_{x_{cn}}^{\tau}\,] = \mathrm{I}[\mathrm{N}_{x_{cn}}^{\tau+1}\,] + \mathrm{I}\{\mathrm{P}_y^{\perp}[\mathrm{N}_{x_{cn}}^{\tau}\,]\} \;, \text{ etc...}$$
$$\text{untill } \; \mathrm{RAC}[\,\mathrm{N}_y \rightarrow \mathrm{N}_{x_{cn}}^{\tau+\cdot\cdot}\,] = \mathrm{I}\{\,\mathrm{P}_y[\mathrm{N}_{x_{cn}}^{\tau+\cdot\cdot}\,]\,\} = 0 \;.$$

6 Scattering decompositions - basic relational models

From the IS $[\,\mathrm{P}_y[\mathrm{N}_{x_{cn}} / \,\mathrm{M}_x = \mathrm{CIP}\,]\,, \mathrm{M}_y = \mathrm{MIP}\,]$, one gets the PM of RAC and the principal scatterplots to describe it, with respect to SU. One may also analyse the other parts of $\mathrm{N}_{x_{cn}}$ [cf. N4-v]. Note that $\mathrm{M}_{x_{cn}}$ exists [cf. P9] so that:

$$[\,\mathrm{P}_y[\mathrm{N}_{x_{cn}} \,/\, \mathrm{CIP}\,]\;,\; \mathrm{MIP}\,] \sim [\,\mathrm{P}_y[\mathrm{N}_x \,/\, \mathrm{M}_{x_{cn}}]\;,\; \mathrm{MIP}] \qquad (4).$$

In a RG, P12 illustrates the relational effect on the shape of $\mathrm{N}_{x_{cn}}$ when one orthogonally projects it onto E_y.

P12: $\mathrm{P}_{\beta_h}(x_t^{cn}) = \rho_h^c \; \mathrm{M}_x(\alpha_h \,, x_t^{cn}) \; \beta_h \;, \mathrm{I}\{\mathrm{P}_{\beta_h}[\mathrm{N}_{x_{cn}}]\} = (\rho_h^c)^2 \; \mathrm{I}\{\mathrm{P}_{\alpha_h}[\mathrm{N}_{x_{cn}}]\}$

$\mathrm{P}_y(x_t^{cn}) = \sum \rho_h^c \; \mathrm{M}_x(\alpha_h \,, x_t^{cn}) \; \beta_h \;, \mathrm{I}\{\,\mathrm{P}_y[\mathrm{N}_{x_{cn}}]\,\} = \sum \; (\rho_h^c)^2 \; \mathrm{I}\{\,\mathrm{P}_{\alpha_h}[\mathrm{N}_{x_{cn}}]\}.$

Particular cases: $[\,\mathrm{P}_y[\mathrm{N}_{x_n}\,]\;,\; \mathrm{MIP}\,] = [\,\mathrm{P}_y[\mathrm{N}_x \,/\, \mathrm{MIP}]\;,\; \mathrm{MIP}\,]$ has PM and NPV which are the $\{\,(\rho_h^c)^2\}$ and the usual CV belonging to E_y. So, according to the nature of variables, here are relational formal definitions of the usual Discriminant or Correspondence Analyses in a RG :

$[\,\mathrm{P}_y[\mathrm{N}_{x_n}\,]\;,\; \mathrm{MIP}\,]$ is (i) one of the two IS of Correspondence Analysis (ii) a new IS for defining Discriminant Analysis if y is the qualitative variable [cf.

P5-iii] or the set of gravity centres if x is the qualitative one. Note that these methods describe only the relational effect on equalized MS.

Two useful subspaces are introduced below. To have a more general property, easier for reading, one uses N_x instead of $\mathrm{N}_{x_{cn}}$ [cf.(4)] and any M_x.

D9: Adjusted and residual subspaces, denoted by E_{x^a} and E_{x^r} are associated to $\mathrm{N}_{x^a} = \{ V_{xy} V_y^{-1} [\mathrm{N}_y]$ or $V_{xy} \chi^2_y [\mathrm{N}_y]$ if y is a qualitative variable$\}$ and to $\mathrm{N}_{x^r} = \mathrm{N}_x - \mathrm{N}_{x^a}$, which is written: $\{ [x_t - x^a_t , \mathrm{f(t)}] ; (\mathrm{t} \in \mathrm{D}) \}$.

P13: In $\mathrm{E} = \mathrm{E}_x \oplus \mathrm{E}_y \oplus \mathrm{E}_{x^a} \oplus \mathrm{E}_{x^r}$, it comes :

(i) Denoting $\theta_k = \int \mathrm{f(t)}\ y^k_t\ \mathrm{dt}$, if y is a qualitative variable then N_{x^a} is the MS of gravity centres: $\mathrm{G}_{x/y} = \{ [g_{x/y^k} = \int [\mathrm{f(t)} / \theta_k]\, y^k_t\ x_t\ \mathrm{dt}\ , \theta_k] ; (\mathrm{k=1,q}) \}$.

(ii) $\mathrm{V}_{x^a} = \mathrm{V}_{xy} \mathrm{V}_y^{-1} \mathrm{V}_{yx}$ (or $\mathrm{V}_{xy}\, \chi^2_y\, \mathrm{V}_{yx}$), $\mathrm{V}_{x^r} = \mathrm{V}_x - \mathrm{V}_{x^a}$, $\mathrm{V}_{x^a x^r} = 0 = \mathrm{V}_{yx^r}$.

(iii) Furthermore, if $\{ \mathrm{M}_{xyx^ax^r} = \mathrm{R}_{xyx^ax^r}[\mathrm{M}_x , \mathrm{M}_y , \mathrm{M}_x , \mathrm{M}_x]\}$ then

(a) $\mathrm{E}_{x^a} \perp \mathrm{E}_{x^r}$, $\mathrm{E}_y \perp \mathrm{E}_{x^r}$ [cf. (ii) then (2)].

(b) $(\forall x^a \in \mathrm{E}_{x^a})\ \| x^a - \mathrm{P}_y(x^a) \| = 0$, $(\forall x \in \mathrm{E}_x)\ \|\mathrm{P}_y(x) - \mathrm{P}_{x^a}(x) \| = 0$ $\|\mathrm{P}^{\perp}_y(x) - \mathrm{P}_{x^r}(x) \| = 0$ and $\| x - [\mathrm{P}_{x^a}(x) + \mathrm{P}_{x^r}(x)] \| = 0$.

So, with respect to the RIP chosen, one may use $\mathrm{P}_{x^a}[\mathrm{N}_x]$ and $\mathrm{P}_{x^r}[\mathrm{N}_x]$ instead of $\mathrm{P}_y[\mathrm{N}_x]$ and $\mathrm{P}^{\perp}_y[\mathrm{N}_x]$.

(c) $[\mathrm{P}_{x^a}[\mathrm{N}_x], \mathrm{M}_x]$ and $[\mathrm{N}_{x^a}, \mathrm{M}_x]$ [resp. $[\mathrm{P}_{x^r}[\mathrm{N}_x], \mathrm{M}_x]$ and $[\mathrm{N}_{x^r}, \mathrm{M}_x]$] have the same PM and the same NPV .

N5: **(i)** V_{x^r} is the covariance matrix of $\{x/y\}$, so $\mathrm{V}^{-1}_{x^r}$ eliminates the AE and the VE of $\{x^j\}$ not linearly explained by $\{y^k\}$. It is meaningful to note that the anti-RIP ($\mathrm{ER}^-_{xy} = -\,\mathrm{ER}_{xy}$) is so that $\mathrm{R}^-_{xy}[V^{-1}_{x^r}, V^{-1}_{y^r}]$ is equal to $\mathrm{V}^{-1}_{x \cup y}$, i.e it eliminates, (a) the remaining part of AE and VE of $\{x^j\}$ [resp.$\{y^k\}$] linearly explained by $\{y^k\}$ [resp.$\{x^j\}$], (b) the relational effect due to $\{x^j\}$ and $\{y^k\}$. This is a new manifestation of the geometrical semantic coherence of RIP.

(ii) The same IP is chosen in E_{x^a}, E_{x^r} and E_x in order to "look" N_{x^a} and N_{x^r} in the same way that N_x is.

(iii) $\mathrm{R}_{xyx^ax^r}$ is obviously a non regular RIP.

(iv) One may imagine the projections one can do onto principal planes of $[\mathrm{N}_{x^a}, \mathrm{M}_x]$: for example, in relational Correspondence Analysis one has all the following mixed simultaneous scatter-plots in E_{x^a} :

$$\{ \mathrm{N}_{x^a} = \mathrm{G}_{x/y} \} \ \cup\ \{ \mathrm{P}_{x^a}[\mathrm{N}_x] = \mathrm{G}_{y/x} \} \ \cup\ \mathrm{P}_{x^a}\{ \mathbf{G}_{x/y} = \mathrm{P}_x[\mathrm{N}_y] \}$$

where $\mathbf{G}_{x/y}$ is the "clone" of $\mathrm{G}_{x/y}$ in E_x .

Some total variation and RAC decompositions

(i) $\mathrm{I}[\mathrm{N}_x] = \{ \mathrm{RAC}[\mathrm{N}_y \rightarrow \mathrm{N}_x] = \mathrm{I}\{ \mathrm{P}_{x^a}[\mathrm{N}_x]\} = \mathrm{I}[\mathrm{N}_{x^a}]\} + \{ \mathrm{I}\{ \mathrm{P}_{x^r}[\mathrm{N}_x]\} = \mathrm{I}[\mathrm{N}_{x^r}]\}$

(ii) With three variables x, y and z, it comes:

$$\mathrm{I}[\mathrm{N}_x] = \mathrm{RAC}[\mathrm{N}_{y\cup z} \rightarrow \mathrm{N}_x] + \{ \mathrm{I}[\mathrm{N}_{x^r}] = \mathrm{I}\{ \mathrm{P}^{\perp}_{y\cup z}[\mathrm{N}_x]\} \}$$

with $\mathrm{RAC}[\mathrm{N}_{y\cup z} \rightarrow \mathrm{N}_x] = \mathrm{RAC}[\mathrm{N}_y \rightarrow \mathrm{N}_x] + \mathrm{RAC}[\mathrm{N}_z \rightarrow \mathrm{N}_x]$
+ linear relational effect (y,z).

(iii) Deepening RAC decompositions when y and z are controlled experiment factors: here are ways to extend MANOVA (Lawley criteria),

a) $\mathrm{RAC}[\mathrm{N}_y \!\to\! \mathrm{N}_x] = \mathrm{I}[\mathrm{N}_{x^a}] = \mathrm{I}\{\mathrm{P}_{x^a}[\mathrm{N}_x]\} = \mathrm{I}\{\mathrm{P}_{x^a}[\mathbf{G}_{x/y}]\} + \sum \theta_k\, \mathrm{I}\{\mathrm{P}_{x^a}[\mathrm{N}_{x_k}]\}$
where $\mathrm{N}_{x_k} = \{\, [\, x_t - g_{x/y^k} \ , \mathrm{f(t)} / \theta_k] \,;\, [\, (x_t \in \mathrm{N}_x) \,/\, y_t^k{=}1 \,]\} \subset \mathrm{E}_x$.

b) $\mathrm{I}[\,\mathrm{N}_x] = \mathrm{RAC}[\,\mathrm{N}_{y\cap z} \!\to\! \mathrm{N}_x] + \mathrm{I}[\,\mathrm{N}_{x^r}]$
with $\mathrm{RAC}[\,\mathrm{N}_{y\cap z} \!\to\! \mathrm{N}_x] = \mathrm{RAC}[\,\mathrm{N}_y \!\to\, \mathrm{N}_x] + \mathrm{RAC}[\,\mathrm{N}_z \!\to\! \mathrm{N}_x]$
$+ \{\,$interaction effect (y×z)$\,\}$.

N6: In (a), one may calculate (i) the contributions of items $\{y^k\}$ to the RAC with N_{x^a} points and (ii) the SU absolute, average and differential contributions with $\mathrm{P}_{x^a}[\,\mathrm{N}_x]$, $\mathrm{P}_{x^a}[\,\mathrm{G}_{x/y}]$ and $\mathrm{P}_{x^a}[\mathrm{N}_{x_k}]$ points respectively.
In (b), if one has some troubles with non regular RIP, he may design the shape of N_x as he likes via $\mathrm{M}_{x_{cn}}$, or by dilations or contractions of N_x along its principal axes, then build its clone in F, having $\{\mathit{\Delta}C^j\}$ as principal axes, and finally does what he likes in F, using well known geometrical results.

7 Massive shapes designed under relational effect and the influence of a given massive shape

Relational forecasts: To forecast linearly $[\,\mathrm{N}_y \,, \mathrm{M}_y]$ with $[\,\mathrm{N}_x \,, \mathrm{M}_x]$, one should have in mind the formula given in P5-(iii). For avoiding the trivial q usual regressions, in the same manner as for the RAC, one has to go away from MIP according to the following process, in order ”to give live” to the AE of $\{y^k\}$. Having no condition to impose to N_x and to M_x let us take

$\mathrm{N}_{x_{\alpha n}} = \sum \{\, a_h\ \ \mathrm{P}_{\alpha_h}[\,\mathrm{N}_x] \,;\, (h \,/\, \rho_h^c \neq 0)\}$, i.e the observable part of N_x from E_y, for example and choose MIP in E_x to simplify calculi.

(e1) Go away from N_{y_n} towards N_y (or further) using $\mathrm{N}_{y_{dn}} = \sum\, b_k\, \mathrm{P}_{d_k}[\mathrm{N}_{y_n}]$ and choose a ”good” $\mathrm{N}_{y_{dn}}$, with respect to a ”reasonable” $\mathrm{RAC}[\,\mathrm{N}_x \!\to\! \mathrm{N}_y]$.

(e2) As $y_t = \sum\, [\, \mu_k^{1/2} / \, b_k \,]\ \ \mathrm{P}_{d_k}(y_t^{dn})$, find the ”best” $\mathrm{N}_{x_{\alpha n}}$ by minimizing $\int \mathrm{f(t)} \parallel y_t - y_t^a \parallel^2 \mathrm{dt}$, to propose the best $y_t^a = \sum\, [\, \mu_k^{1/2} / \, b_k \,]\, \mathrm{P}_{d_k}(x_t^{\alpha n})$. (5)

(e3) Having forced, (a) the relational effect to $\mathrm{N}_{x_{\alpha n}}$ by projection on E_y, (b) the influence of N_y to $\mathrm{P}_y[\,\mathrm{N}_{x_{\alpha n}}]$ by applying (5) , one finally gets one among the expressions of y_t^a , the one where appear $\{\alpha_h\}$ and $\{\beta_h\}$:
$y_t^a = \sum \sum\, a_h\ (\mu_k^{1/2} / \, b_k)\ \ \rho_h^c\ \ cos(d_k \,,\, \beta_h)\ M_x(\alpha_h \,,\, x_t^n)\ d_k$, where
$a_h = (\rho_h^c)^{-1} \{\sum\, cos(d_r \,,\, \beta_h)\, cos(d_r \,,\, \alpha_h)\, \mu_r / b_r\} \,/\, \{\sum\, [\, cos(d_s \,,\, \beta_h) / b_s]^2\, \mu_s\}$

N7: In this process N_y is partially equalized when one applies the relational effect to a non equalized $\mathrm{N}_{x_{\alpha n}}$. So, recovering a part of AE , more or less important, in accordance with reasonable choices made, one hopes that y_t^a will be a multidimensional adjusted value of y_t which will have some interests in some specific contexts.

Relational proximities: One wants to determine the nearest $\mathrm{N}_{x_{cn}}$ of a given $\mathrm{N}_{y_{dn}}$. Here, one decides to respect at best the shape of N_x . One may want to have $[\,\mathrm{N}_{x_{cn}}, \mathrm{M}_x]$ and $[\,\mathrm{N}_x, \mathrm{M}_x]$ RHC, or if one is less demanding one may accept a compromise between $\int \mathrm{f(t)} \parallel y_t^{dn} - x_t^{cn} \parallel^2$ dt and a criteria like

$$\int \mathrm{f(t)} \mid \mathrm{I}\{\mathrm{P}_{\alpha_h}[\mathrm{N}_{x_{cn}}]\} - \mathrm{I}\{\mathrm{P}_{\alpha_h}[\mathrm{N}_x]\} \mid \mathrm{dt} \;\text{ or }\; \int \mathrm{f(t)} \parallel \sum \mathrm{P}_{\alpha_h}(x_t^{cn}) - x_t \parallel^2 \mathrm{dt}\ .$$

8 A relational model for interval data

p intervals $\{\,[\,m_i^j\,, M_i^j]\,;\,(\mathrm{j}=1,\mathrm{p})\}$ of p quantitative variables $\{x^j\}$ are observed on n SU $\{\,\mathrm{I}_i\ ;\ (\mathrm{i}=1,\mathrm{n})\,\}$. Besides, the SU are regrouped into $(q \leq n)$ clusters by the q items $\{\,y^k\ ;\ (\mathrm{k}=1,\mathrm{q})\,\}$ of a qualitative variable y . In accordance with what one can learn from Chouakria (1998), three relational models may be defined. Here one gives main elements for only one:

- $\mathrm{D}_i = \prod\,[\,m_i^j, M_i^j] \subset \mathrm{R}^p$, $\mathrm{P}_i = \mathrm{D}_i\ \forall\ (\mathrm{i}=1,\mathrm{n})$, $\mathrm{D} = \cup\ \mathrm{D}_i$, $\mathrm{P} = \cup\ \mathrm{P}_i$
- f(t) is so that $\mathrm{f(t)} = \theta_i\ \mathrm{f}_i(\mathrm{t})$ if $(\mathrm{t}{\in}\mathrm{D}_i)$, with $\int_{D_i} \mathrm{f}_i(\mathrm{t})\,\mathrm{dt} = 1$ and
  $\mathrm{f}_i(\mathrm{t}) = \mathrm{h}(\mathrm{t}\,,\varphi_i)$, where φ_i is a set of parameters specifying I_i
- $\mathrm{N}_x\ = \{\,[\,x_t = x(t) = t \in \mathrm{P}\,,\ \mathrm{f(t)}\,]\,;\,(t \in \mathrm{D})\ \} \subset \mathrm{E}_x = \mathrm{R}^p$
- $\mathrm{N}_{g_{x/I}} = \{\,[\,\mathrm{g}_{x/I_i} = \int_{D_i} \mathrm{f}_i(\mathrm{t})\,x_t\ \mathrm{dt}\ ,\ \theta_i = \int_{D_i} \mathrm{f(t)}\ \mathrm{dt}\ ]\ ;\ (\mathrm{i}=1,\mathrm{n})\,\} \subset \mathrm{P}$
- $\mathrm{N}_{x/I_i} = \{\,[\,x_t - \mathrm{g}_{x/I_i}\ ,\ \mathrm{f}_i(\mathrm{t})\,]\ ;\ (\mathrm{t} \in \mathrm{D}_i)\,\}$
- $\mathrm{N}_y\ = \{\,[\,y_t = y(t)\ ,\ \mathrm{f(t)}\,]\,;\,(\mathrm{t} \in \mathrm{D})\,\} \subset\ \mathrm{E}_y = \mathrm{R}^q$ so that
  $\forall\ [\,(\mathrm{t}{\in}\mathrm{D}_i)\,,(\mathrm{i}=1,\mathrm{n})\,]\ [\,y_t^k = \psi_i \in \{0{,}1\}\ \ \forall\ (\mathrm{k}=1,\mathrm{q})\,]$ and $\sum_k y_t^k{=}1$
- $\mathrm{g}_{y^k}\ = \int_D \mathrm{f(t)}\ y_t^k\ \mathrm{dt} = \sum_{I_k} \theta_i = \theta_k^y$, with $\mathrm{I}_k = \{\,\mathrm{i}\,/\,\forall\,(\mathrm{t} \in \mathrm{D}_i) \Rightarrow y_t^k = 1\}$
- $\mathrm{g}_{x/y^k} = \int_D\ [\,\mathrm{f(t)}\ /\ \theta_k^y\,]\ y_t^k\ x_t\ \mathrm{dt}\ =\ \sum_{I_k}\,[\,\theta_i\,/\,\theta_k^y\,]\ \mathrm{g}_{x/I_i}$
- $\mathrm{N}_{x/y^k} = \{\,[\,x_t\ \text{-}\ \mathrm{g}_{x/y^k}\ ,\ (\,\mathrm{f(t)}\ /\ \theta_k^y\,)\ ]\ ;\ (\,\mathrm{t} \in [\cup_{I_k} \mathrm{D}_i\,]\ )\ \}$
- $\mathrm{N}_{g_{x/y}} = \{\,[\ \mathrm{g}_{x/y^k}\ ,\ \theta_k^y\ ]\ ;\ (\mathrm{k}=1,\mathrm{q})\ \}$
- $V_x\ = \int_D\ \mathrm{f(t)}\ x_t\ {}^t x_t\ \mathrm{dt}$
- $[V_{xy}]^k = \int_D\ \mathrm{f(t)}\ x_t\ (y_t^k - \theta_k^y)\ \mathrm{dt}\ =\ \theta_k^y\ \ \mathrm{g}_{x/y^k} = \sum_{I_k}\ \theta_i\ \ \mathrm{g}_{x/I_i}$
- $[V_y]_r^s\ = \int_D\ \mathrm{f(t)}\ (y_t^r - \theta_r^y)\ (y_t^s - \theta_s^y)\ \mathrm{dt} = \theta_r^y\ (1\ \text{-}\ \theta_r^y)$ if (r = s)
  $= \text{-}\ \theta_r^y\ \theta_s^y$ if $(\mathrm{r} \neq \mathrm{s})$.

So, **(i)** one may now define the RIP of this relational model and use all the results described above. It is proposed to represent $\{I_i\}$ by the concentration ellipsoids with respect to the corresponding principal subspaces.

(ii) $\mathrm{I}[\,\mathrm{N}_x] = \mathrm{I}[\,\mathrm{N}_{g_{x/I}}] + \sum\ \theta_i\ \mathrm{I}[\,\mathrm{N}_{x/I_i}] = \mathrm{I}[\,\mathrm{N}_{g_{x/y}}] + \sum\ \theta_k^y\ \mathrm{I}[\,\mathrm{N}_{x/y^k}]$
with $\mathrm{I}[\,\mathrm{N}_{g_{x/I}}] = \mathrm{I}[\,\mathrm{N}_{g_{x/y}}] + \sum\ \theta_k^y\ \mathrm{I}[\,\mathrm{N}_{(g_{x/I})/y^k}]$.

(iii) One may enjoy using other relational models not described till now : for example, one may propose clusters on $\{I_i\}$, by maximizing $\mathrm{RAC}[\,\mathrm{N}_x{\rightarrow}\mathrm{N}_z]$, where z is an unknown qualitative variable, then with the same relational model compare y and z , (a) by analysing their RAC, (b) by determining a central subspace E_w between E_y and E_z which minimizes

$$|\mathrm{RAC}[\,\mathrm{N}_y \rightarrow \mathrm{N}_w] - \mathrm{RAC}[\,\mathrm{N}_z \rightarrow \mathrm{N}_w]\,|\,,$$

(c) by proposing changing clusters $\mathrm{w}(\tau)$ between y and z which maximizes at each step τ :

$$(1 - \tau)\ \mathrm{RAC}[\,\mathrm{N}_y \rightarrow \mathrm{N}_w] + \tau\ \mathrm{RAC}[\,\mathrm{N}_z \rightarrow \mathrm{N}_w]\ , \text{ for example.}$$

(iv) Finally, one may also extract principal clusters $\mathrm{z}(\tau)$ of $[\,\mathrm{N}_x\,, \mathrm{M}_x]$, where $\mathrm{z}(\tau)$ is a qualitative variable, by maximizing at each step τ :

$$\mathrm{RAC}[\,\mathrm{N}_{z(1)} \rightarrow \mathrm{N}_x]\,, .., \mathrm{RAC}\{\,\mathrm{N}_{z(\tau+1)} \rightarrow \mathrm{P}^{\perp}_{z(\tau)} \,\mathrm{o}\, \mathrm{P}^{\perp}_{z(\tau-1)} \,\mathrm{o}\, .. \,\mathrm{o}\, \mathrm{P}^{\perp}_{z(1)}[\,\mathrm{N}_x]\,\}\,.$$

N8: As geometrical structure of RAC are those of their corresponding MS, Correspondence Analysis, MANOVA got on changing time data or on sets of statistic units virtually matched for example, one may analyse their RAC.

9 Conclusion

Some youthful ideas are organized and described here. I do that out of duty. But now, what shall we do about, (i) a statistic unit-space which expands as far as you want (principal clusters, Russian dolls, ...), (ii) deeper relational scattering decompositions, (iii) relational association coefficients between relational association coefficients [see N8], ... In what frame of mind are we, as we have to choose between an infinite numbers of choices (massive shapes designed, relational association coefficients, ...), having to justify each decision! Wanting to restore the statistic units specific part in Data Analysis, as a boomerang one gets results which give a bad headache.

Finally, one may say : (i) multiway tables methods can be defined in a relational geometry, for Multiple Correspondence Analysis or STATIS, this job has been done (Schektman (1989)), so understanding better their relational structures perhaps one could extend them more easily, and (ii) relational geometry are well fitted (a) to analyse data changing in time, mainly to forecast as I indicated a long time ago in my thesis, (b) supposing that absolute or differential contributions of statistic units to relational association coefficients are values of elementary attractive forces between matched points of massive shapes, to propose, among others, new approaches to study stability and protection of relational models, as it was done in theses I supervised.

References

CAZES, P., CHOUAKRIA, A., DIDAY, E. and SCHEKTMAN, Y. (1997): Extension de l'Analyse en Composantes Principales à des données de type intervalle. *Revue de Statistique Appliquée XLV (3), 5-24.*

CHOUAKRIA, A. (1998): *Extension des Méthodes d'Analyse Factorielle à des Données de Type Intervalle.* Thèse, Université Paris Dauphine - Paris IX.

SCHEKTMAN, Y. (1989): Inner products and association indices for analysing some multiway tables. In: R.Coppi and S.Balasko (Eds): *Multiway Data Analysis.* North-Holland, 203-212.

Dynamic Features Extraction in Soybean Futures Market of China

Huiwen Wang and Jie Meng

School of Economics and Management, Beihang University
Beijing 100083, China, *wanghw@vip.sina.com*, *mengjie517@gmail.com*

Abstract. By applying Symbolic Data Analysis (SDA), this paper investigates the dynamic features of soybean futures market of Dalian Commodity Exchange (DCE) of China during 2002 to 2004. First, interval data is created by classifying mass futures contracts by different years and different maturity dates; and then DIV clustering method is applied on these interval data which produces further simplified three-way interval symbolic data and greatly reduces the sample size. Based on that, factor analysis of interval data is adopted to extract dynamic principal characteristics of soybean futures, which reduces the dimension of the variable space. The results of the empirical research, which are rightly coincident with the realities, verify the application value of SDA in analyzing mass, dynamic and complex data.

1 Introduction

In the analysis of large scale data set, high dimension of both sample and variable spaces leads to complex computation and it is also difficult to obtain the integral structure of the data set. To solve the problem, E.Diday (1988) proposed a brand-new way of data analysis - Symbolic Data Analysis (SDA), which is a kind of multivariate statistic analysis technique oriented to large scale database retrieval and capable of multilevel analysis. Extended from traditional data, cells of data table in SDA could be not only quantitative or qualitative but also a concept, multivalue, interval or distribution. Because of those advantages of SDA, it is especially effective in knowledge exploring to huge amounts of data.

In traditional multivariate statistic analysis, principal component analysis provides an efficient way for dimension reduction. In the field of SDA, Cazes (1997) presented Principal Component Analysis (PCA) on interval data, and Lauro, Verde and Palumbo proposed factor discriminant analysis method on symbolic data. Moreover, PCA has also been developed onto three-way interval data and Wang, Hu (2003) successfully applied it in features extraction in stock market of China. There are three advantages of this method: 1) SDA realizes dimension reduction in sample space; 2) PCA performs dimension reduction in variable space; 3) analysis on three-way data set explores dynamic features of the complex system.

Based on that, this paper applies global PCA on three-way interval data to the dynamic features extraction in soybean futures market of China. In section 2, modeling method of global PCA on three-way interval data is introduced. After that, section 3 adopts the method to analyze principal factors of the soybean futures market of China. Finally, section 4 gives a conclusion of the paper.

2 Global PCA on three-way interval data

The main idea of global PCA on interval data is: first, transform the three-way interval data into a numerical matrix; and then apply the classical PCA on the transformed numerical data table; finally, construct the interval principal components from the numerical principal components. And the procedures of its algorithm are summarized as follows:

(1) Three-way interval data and its transformation

Denote Z as a three-way interval data which is composed of T periods of plane interval data table Z^t $(t = 1, \cdots, T)$, that is

$$Z = \begin{bmatrix} Z^1 \\ \vdots \\ Z^T \end{bmatrix} \tag{1}$$

where $Z^t = \begin{pmatrix} x_1^t \\ \vdots \\ x_n^t \end{pmatrix} = \begin{pmatrix} [\underline{x}_{11}^t, \overline{x}_{11}^t] & \cdots & [\underline{x}_{1p}^t, \overline{x}_{1p}^t] \\ \vdots & \ddots & \vdots \\ [\underline{x}_{n1}^t, \overline{x}_{n1}^t] & \cdots & [\underline{x}_{np}^t, \overline{x}_{np}^t] \end{pmatrix}$, $(t = 1, \cdots, T)$, and the observation x_i^t is an interval object with p dimension.

x_i^t, a hyperrectangle in the p-dimension space, can be described by a matrix with 2^p rows and p columns where each row contains the coordinates of one vertex of the hyperrectangle in R^p, which can be denoted as

$$V_i^t = \begin{pmatrix} \underline{x}_{i1}^t & \underline{x}_{i2}^t & \cdots & \underline{x}_{ip}^t \\ \overline{x}_{i1}^t & \underline{x}_{i2}^t & \cdots & \underline{x}_{ip}^t \\ & \cdots & \cdots & \\ \overline{x}_{i1}^t & \overline{x}_{i2}^t & \cdots & \overline{x}_{ip}^t \end{pmatrix}_{2^p \times p} \tag{2}$$

Compile all the transformed numerical matrix $V_i^t (i = 1, \cdots, n; t = 1, \cdots, T)$ as

$$V = \begin{bmatrix} V^1 \\ \vdots \\ V^T \end{bmatrix} \tag{3}$$

where $V^t = \begin{pmatrix} V_1^t \\ \vdots \\ V_n^t \end{pmatrix} = \begin{pmatrix} \begin{pmatrix} \underline{x}_{11}^t & \cdots & \underline{x}_{1p}^t \\ \vdots & \ddots & \vdots \\ \overline{x}_{11}^t & \cdots & \overline{x}_{1p}^t \end{pmatrix}_{2^p \times p} \\ \vdots \\ \begin{pmatrix} \underline{x}_{n1}^t & \cdots & \underline{x}_{np}^t \\ \vdots & \ddots & \vdots \\ \overline{x}_{n1}^t & \cdots & \overline{x}_{np}^t \end{pmatrix}_{2^p \times p} \end{pmatrix}_{(n \cdot 2^p) \times p}$, $(t = 1, \cdots, T)$ is the transformed numerical matrix at t.

(2) PCA on the transformed numerical matrix V

Apply the classical PCA on the transformed numerical matrix V and obtain the first m numerical principal components $\widetilde{F}_1, \cdots, \widetilde{F}_m$ which can be denoted as

$$\widetilde{F}_j = \begin{bmatrix} \widetilde{F}_j^1 \\ \vdots \\ \widetilde{F}_j^T \end{bmatrix}, j = 1, 2, \cdots, m \tag{4}$$

where $\widetilde{F}_j^t = \begin{pmatrix} \widetilde{f}_{1j}^t \\ \vdots \\ \widetilde{f}_{nj}^t \end{pmatrix}, \widetilde{f}_{ij}^t = \begin{pmatrix} f_{ij}^{(t,1)} \\ \vdots \\ f_{ij}^{(t,2^p)} \end{pmatrix}, (i = 1, \cdots, n; t = 1, \cdots, T; j = 1, 2, \cdots, m)$.

(3) Construct the interval principal components of Z

Let $\underline{f}_{ij}^t = \min\{f_{ij}^{(t,1)}, \cdots, f_{ij}^{(t,2^p)}\}$, $\overline{f}_{ij}^t = \max\{f_{ij}^{(t,1)}, \cdots, f_{ij}^{(t,2^p)}\}$; then $f_{ij}^t = \left[\underline{f}_{ij}^t, \overline{f}_{ij}^t\right]$ is the interval value of the i interval object on the j principal component at t. And the interval principal components of Z, denoted as $F_1, F_2, , F_m$, are conducted by

$$F_j = \begin{bmatrix} F_j^1 \\ \vdots \\ F_j^T \end{bmatrix}, j = 1, 2, \cdots, m \tag{5}$$

where $F_j^t = \begin{pmatrix} f_{1j}^t \\ \vdots \\ f_{nj}^t \end{pmatrix} = \begin{pmatrix} \left[\underline{f}_{1j}^t, \overline{f}_{1j}^t\right] \\ \vdots \\ \left[\underline{f}_{nj}^t, \overline{f}_{nj}^t\right] \end{pmatrix}, (t = 1, \cdots, T; j = 1, 2, \cdots, m).$

Finally, according to the loading plot and the rectangle projections of the interval objects on the factorial plane, we can find the dynamic features of the original complex system with much integrated and simplified results.

3 Factorial analysis on soybean futures market

In this section, global PCA on three-way interval data is applied to the soybean futures of DCE to analyze the dynamic marketing features of different contracts in different time. We select eight exchange indexes "open price, maximum price, minimum price, closing price, balance price, trading volume, turnover, open interest" of the contract daily records from 2002 to 2004.

3.1 Classification of the futures contracts

In the futures market, futures contracts have particular characters: there are more than one contracts of each kind of futures at the same time; besides, every contract has a valid trading period of time. Therefore, from the static point of view, contracts with different maturity dates need to be considered in the meantime; and from the dynamic point of view, a single contract can't form a continuous time series, that's the main problems in the research of futures market.

Proposed solution to the above difficulties in this paper is: tag every contract with its maturity date at every point of time; therefore, there are 19 classes each year from 2002 to 2004; finally, construct interval data table by selecting the minimum and maximum values of every index in each class. Obviously, it is a kind of dynamic classification that samples in each class change as time goes on and every contract passes in and out of all classes during its whole period of validity. Resultingly, 1) thousands of daily contract records are transformed to 19 interval objects of each year which greatly reduces complexity of the research; 2) 19 classes cover the whole trading period without missing information; 3) each class of contract can form a continuous time series since there are new contract timely passing in and out; 4) class features can be easily explored and compared which seems much integrated and efficient.

To further summarize and simplify the data set, DIV method is applied to cluster the 19 interval objects. Variables of trading volume and open interest are selected as the clustering criterion and the results are listed in table 1.

Table 1. DIV clustering results

Class	*ClassI*	*ClassII*	*ClassIII*	*ClassIV*	*ClassV*
Contracts *time to maturity dates*	$1 \sim 3$ *months*	$4 \sim 6$ *months*	$7 \sim 9$ *months*	$10 \sim 12$ *months*	$13 \sim 19$ *months*

To clearly illustrate different features of different classes, the mid values of the 19 interval objects in trading volume and open interest are selected and joint in fig.1, which shows a rightly discriminated result of the DIV clustering.

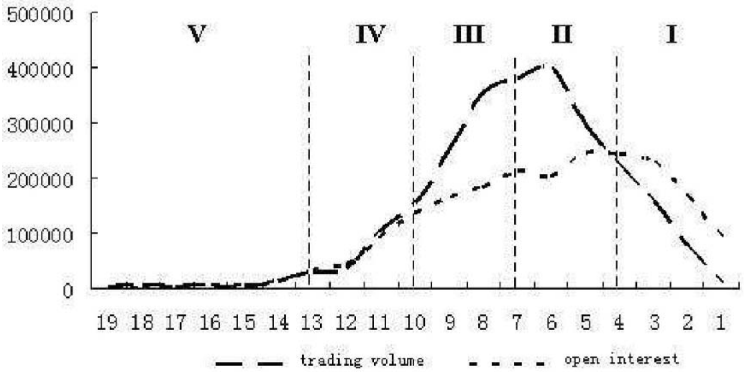


Fig. 1. Change tendencies of the contracts in trading volume and open interest.

3.2 Dynamic factor analysis of the five classes of contracts

Apply global PCA on three-way interval data of the five classes of the contracts from 2002 to 2004. The cumulate contribution proportion of the first two principal components is 77%, and the loading plot is shown in fig.2 which exposes the relationships of the first two principal components and the original variables.

It is clear in fig.2 that: 1) the price variables “open price, maximum price, minimum price, closing price, balance price”, which are highly correlated

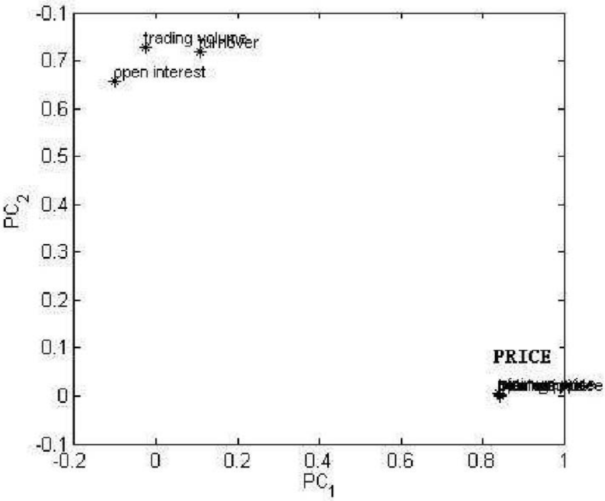


Fig. 2. Loading plot.

with each other, reflect the most notable feature in the soybean futures market of DCE. Actually, the balance prices of the five classes of soybean futures contracts have been greatly increased during 2002 to 2004 (seen from fig.3); 2) the three trading variables “trading volume, turnover, open interest” also have high correlations and show the second feature in the soybean futures market of DCE.

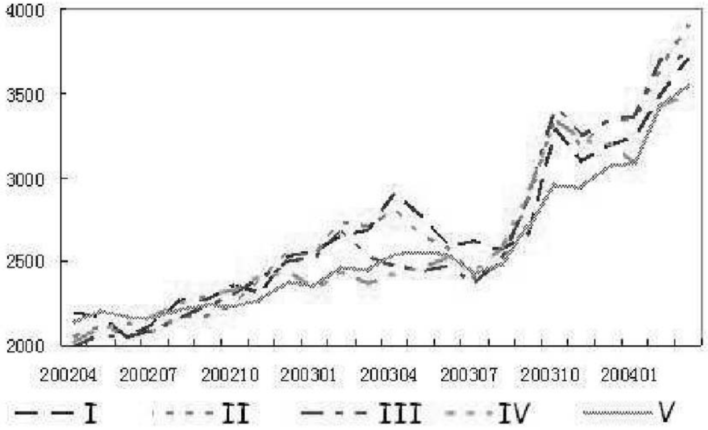


Fig. 3. Price trendlines of the five classes of contracts.

Furthermore, project the five classes of contracts from 2002 to 2004 on the principal plane in fig.4 which illustrates the features and change tendencies of the soybean futures market in price (component 1) and trading (component 2) aspects.

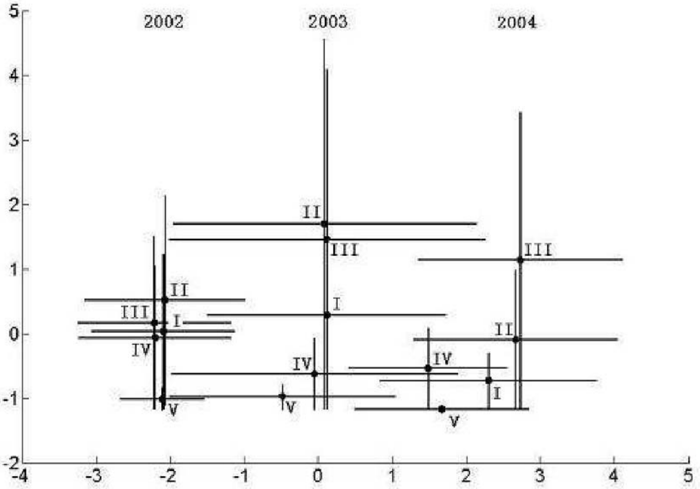


Fig. 4. Interval principal components of the five classes of contracts.

From the direction of the first component in fig.4, we can find the following features of the futures market in price: 1) contracts are fairly divided by time, which implies a yearly rise of the price in the futures market; 2) contract intervals in 2003 range larger, which implies higher fluctuations of contract price in 2003; 3) the disparities of the average price of different class of contracts is enlarged in 2004, where class I, II, III are higher than class IV, V.

Besides, we can see from the direction of the second component in fig.4 that: 1) contract trading of 2003 is more active than that of 2002 and 2004; 2) from 2002 to 2003, trading activity of class II, III is higher than class IV, V and the trading disparities between different classes tend to expand, while in 2004, the trading of class III and IV increased a little, which implies the trading time of the speculators in the soybean futures market generally advanced.

3.3 Radar-chart of the five classes of contracts in 2002 $\sim$ 2004

To compare with the above results of factor analysis, radar-graphs of the five classes of contracts in 2002 $\sim$ 2004 are listed below. In fig.5, each row gives comparisons of the five classes of contracts in the variables of “balance price, trading volume and open interest” at each year; while each column reflects the dynamic change tendencies of the five classes of contracts through the three years at each index.

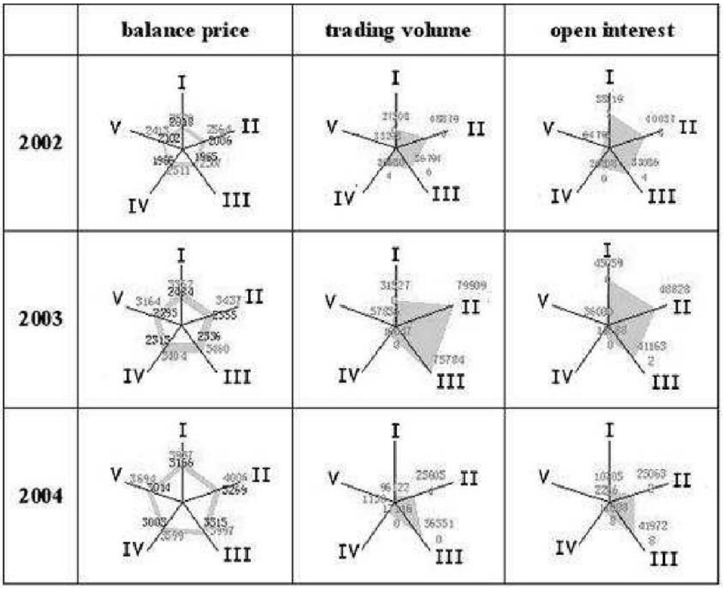


Fig. 5. Comparisons of the five classes of contracts in 2002 ∼ 2004.

It is similar with the above results of PCA: in the same year, there is no great difference between the five classes of contracts in price, but large disparity in their trading volume and open interest where class II, III are more than class IV, V; besides, the contract price has been increasing from 2002 to 2004, while the transaction in 2003 is more active.

4 Conclusion

This paper applies SDA technique to overcome difficulties of traditional modeling methods on large scale data set. Following the idea of “data package”, it realizes the reduction in both sample and variable spaces but without destruction to the original internal logic relationship of the dataset, which efficiently solves the contradiction of difficult analysis on the mass data to its easy collection.

In this paper, SDA is applied to simplification and dynamic factor extraction in the soybean futures market of DCE. By classifying and clustering mass futures contracts by different years and different maturity dates, three-way interval symbolic data is constructed, which greatly reduces scale of the data set. Based on that, global PCA on interval data is adopted to extract dynamic principal characteristics of soybean futures. The results of the case study are proved same with the actual status, which verifies the validity and rationality of the modeling method in integrating and extracting information of the multidimensional and dynamic complex system.

Acknowledgements This research is supported by the National Natural Science Foundation of China (NSFC).

References

BOCK, H.-H. and DIDAY, E. (2000): *Analysis of Symbolic Data.* Springer, Berlin, Heidelberg.

WANG, H. (1999): *Partial Least-Squares Regression and its Application.* National Defense Publishing, Beijing.

HU, Y. and WANG, H. (2004): A new data mining method based on huge data and its application. *Journal of Beijing University of Aeronautics and Astronautics(Social Sciences Edition) 17, 40-44.*

Index