SPRINGER BRIEFS IN STATISTICS
JSS RESEARCH SERIES IN STATISTICS

Gareth William Peters
Tomoko Matsui *Editors*

Modern Methodology and Applications in Spatial-Temporal Modeling

Japan Statistical Society
JSS
since 1931 日本統計学会

Springer

SpringerBriefs in Statistics

JSS Research Series in Statistics

The current research of statistics in Japan has expanded in several directions in line with recent trends in academic activities in the area of statistics and statistical sciences over the globe. The core of these research activities in statistics in Japan has been the Japan Statistical Society (JSS). This society, the oldest and largest academic organization for statistics in Japan, was founded in 1931 by a handful of pioneer statisticians and economists and now has a history of about 80 years. Many distinguished scholars have been members, including the influential statistician Hirotugu Akaike, who was a past president of JSS, and the notable mathematician Kiyosi Itô, who was an earlier member of the Institute of Statistical Mathematics (ISM), which has been a closely related organization since the establishment of ISM. The society has two academic journals: the Journal of the Japan Statistical Society (English Series) and the Journal of the Japan Statistical Society (Japanese Series). The membership of JSS consists of researchers, teachers, and professional statisticians in many different fields including mathematics, statistics, engineering, medical sciences, government statistics, economics, business, psychology, education, and many other natural, biological, and social sciences.

The JSS Series of Statistics aims to publish recent results of current research activities in the areas of statistics and statistical sciences in Japan that otherwise would not be available in English; they are complementary to the two JSS academic journals, both English and Japanese. Because the scope of a research paper in academic journals inevitably has become narrowly focused and condensed in recent years, this series is intended to fill the gap between academic research activities and the form of a single academic paper.

The series will be of great interest to a wide audience of researchers, teachers, professional statisticians, and graduate students in many countries who are interested in statistics and statistical sciences, in statistical theory, and in various areas of statistical applications.

More information about this series at http://www.springer.com/series/13497

Gareth William Peters · Tomoko Matsui
Editors

Modern Methodology and Applications in Spatial-Temporal Modeling

Springer

Editors

Gareth William Peters
Department of Statistical Science
University College London
London
UK

Tomoko Matsui
The Institute of Statistical Mathematics
Tachikawa, Tokyo
Japan

ISSN 2191-544X ISSN 2191-5458 (electronic)
SpringerBriefs in Statistics
ISSN 2364-0057 ISSN 2364-0065 (electronic)
JSS Research Series in Statistics
ISBN 978-4-431-55338-0 ISBN 978-4-431-55339-7 (eBook)
DOI 10.1007/978-4-431-55339-7

Library of Congress Control Number: 2015953790

Springer Tokyo Heidelberg New York Dordrecht London

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made.

Printed on acid-free paper

Springer Japan KK is part of Springer Science+Business Media (www.springer.com)

Preface

The idea to create this book arose as a response to the discussions and presentations that took place in the first and second annual international workshops on spatial and temporal modeling (STM2013 and STM2014) and the first workshop on complex systems modeling and estimation challenges in big data (CSM2014), all of which were held in the Institute of Statistical Mathematics (ISM), Tokyo, Japan. These workshops were cohosted by Prof. Tomoko Matsui (ISM) and Dr. Gareth W. Peters (UCL). It was apparent after these workshops were completed that the wide range of participants from various backgrounds including probability, statistics, applied mathematics, physics, engineering, and signal processing as well as speech and audio processing had been recently developing a variety of new theory, models, and methods for dealing with spatial and temporal problems that would be beneficial to document for a wider scientific audience.

Therefore, this book is intended to bring together a range of new innovations in the area of spatial and temporal modeling in the form of self-contained tutorial chapters on recent areas of research innovations. Since it is based around contributions from a selection of world experts in spatial and temporal modeling who participated in the workshop, it reflects a cross section of specialist information on a range of important topics in spatial and temporal modeling and application. It is the aim of such a text to provide a means to motivate further research, discussion, and cross fertilization of research ideas and directions among the different research fields representative of the authors who contributed.

Whilst this book covers more of the practical and methodological aspects of spatial-temporal modeling, its companion book, also in the Springer Briefs series, titled *Theoretical Aspects of Spatial-Temporal Modeling*, complements this book for theoreticians as it covers a range of new innovations in theoretical aspects of modeling. The chapters in this book cover the topics summarized in the figure.

This book aims to provide a modern introductory tutorial on specialized methodological and applied aspects of spatial and temporal modeling. The areas covered involve a range of topics which reflect the diversity of this domain of research across a number of quantitative disciplines. For instance, the first chapter covers nonparametric Bayesian inference via a recently developed framework known as kernel mean embedding that has had a significant influence in machine learning disciplines. The second chapter covers nonparametric statistical methods for spatial field reconstruction and exceedance probability estimation based on Gaussian process-based models in the context of wireless sensor network data. The third chapter covers signal processing methods applied to acoustic mood analysis based on music signal analysis. The final chapter covers models that are applicable to time series modeling in the domain of speech and language processing. This includes aspects of factor analysis, independent component analysis in an unsupervised learning setting. Then it moves to cover more advanced topics on generalized latent variable topic models based on hierarchical Dirichlet processes which have been developed recently in nonparametric Bayesian literature.

Applications and the Bayesian approaches

Applications
Wireless sensor network
Speech processing
Language processing
Music processing

Bayesian approaches	
Parametric	**Nonparametric**
S-BLUE Spatial-covariance regression	Kernel mean embedding
	Gaussian process
	Hierarchical Dirichlet process

We first note that each chapter of this book is intended to be a self-contained research-level tutorial on modern approaches to the practical and methodological study of some aspect of spatial and temporal statistical modeling. However, to guide the reader in considering the sections of this book, we note the following relationships between chapters. The first and second chapters cover recent advances in machine learning-based methodologies for nonparametric estimation procedures. The first chapter addresses the recent topic of kernel mean embedding methods, which are now becoming popular approaches to performing high-dimensional state-space modeling problems as well as addressing problems with intractable likelihood in filtering applications. These recent nonparametric inference methods with positive definite kernels have been developed to utilize the kernel mean expression of distributions. In this approach, the distribution of a variable is represented by the kernel mean, which is the mean element of the random feature vector defined by the kernel function, and the relation among variables is expressed by covariance operators. This general methodology is starting to have important applications in many spatial and temporal modeling settings.

The second and third chapters also consider nonparametric models, focussing on the class of Gaussian process models, the second chapter looking at spatial models, and the third chapter looking at state-space models. In the second chapter new methods to model spatial data via combinations of Gaussian process models with observations of mixed type, discrete, and continuous. It develops a framework for spatial field reconstruction and establishes efficient spatial best linear unbiased estimators for this spatial field estimation given observations. In addition, an estimation framework based on a covariance regression model is established to perform parameter estimation and introduce covariates into the spatial covariance function structure. In the third chapter state-space models with Gaussian process state or observation equations are considered in the application of speech and music emotion recognition.

The final chapter also studies speech and language processing, this time focusing on topic models for structural learning and temporal modeling from unlabeled sequential patterns. The nonparametric models developed in this chapter are based on the family of hierarchical Dirichlet processes and are considered in a Bayesian formulation. The chapter also discusses, in addition to construction of such models, the variational Bayes- and MCMC-based estimation procedures for such models.

Tokyo, Japan Gareth William Peters
August 2015 Tomoko Matsui

Acknowledgments

We are very grateful to Research Organization of Information and Systems (ROIS), The Institute of Statistical Mathematics (ISM), UK Royal Society International Exchange Grant, and Ministry of Education, Culture, Sports, Science and Technology (MEXT) undertake project “Cooperation with Math Program” to support the first and second annual international workshops on spatial and temporal modeling (STM2013 and STM2014) and the first workshop on complex systems modeling and estimation challenges in big data (CSM2014). The idea to create this book arose as a response to the discussions and presentations that took place in the workshops.

We would like to express our sincere thanks to all the following presenters in the workshops.

- Prof. Nourddine Azzaoui (Université Blaise Pascal)
- Prof. Jen-Tzung Chien (National Chiao Tung University)
- Prof. Arnaud Doucet (Oxford University)
- Prof. Norikazu Ikoma (KIT)
- Prof. Kenji Fukumizu (ISM)
- Prof. Konstatin Markov (Aizu University)
- Prof. Daichi Mochihashi (ISM)
- Prof. Pierre Del Moral (UNSW)
- Prof. Tor Andre Myrvoll (SINTEF)
- Dr. Ido Nevat (Institute for Infocomm Research, A-Star)
- Prof. Yosihiko Ogata (ERI, University of Tokyo and ISM, ROIS)
- Dr. Takashi Owada (Technion)
- Prof. Daniel P. Palomar (HKUST)
- Prof. François Septier (Telecom Lille 1)
- Prof. Taiji Suzuki (Tokyo Institute of Technology)
- Prof. Kazuya Takeda (Nagoya University)
- Prof. Mario Wüthrich (ETH Zurich)

Contents

1 Nonparametric Bayesian Inference with Kernel Mean Embedding 1
Kenji Fukumizu

2 How to Utilize Sensor Network Data to Efficiently Perform Model Calibration and Spatial Field Reconstruction 25
Gareth W. Peters, Ido Nevat and Tomoko Matsui

3 Speech and Music Emotion Recognition Using Gaussian Processes 63
Konstantin Markov and Tomoko Matsui

4 Topic Modeling for Speech and Language Processing 87
Jen-Tzung Chien

Editors and Contributors

About the Editors

Dr. Gareth William Peters Department of Statistical Science, University College London, UK e-mail: gareth.peters@ucl.ac.uk

He is an assistant professor in the Department of Statistical Science, Principle Investigator in Computational Statistics and Machine Learning, and Academic Member of the UK Ph.D. Center of Financial Computing at University College London. He is also an adjunct scientist in the Commonwealth Scientific and Industrial Research Organization, Australia; Associate Member Oxford-Man Institute at the Oxford University; and associate member in the Systemic Risk Center at the London School of Economics. Dr. Peters is also a visiting professor at the Institute of Statistical Mathematics, Tokyo, Japan, where he has visited since 2009. Dr. Peters obtained a B.Sc. (hons 1st) in Mathematics and Physics and a B.Eng. (hons 1st) from the University of Melbourne in 2003, a M.Sc. (research) from the University of Cambridge in 2006 and a Ph.D. in Statistics (by publication) from the University of New South Wales in 2009.

Dr. Peters research interests range over several areas of mathematical statistics, time series and state-space modeling, heavy tailed stochastic processes and Levy processes, dependence structure modeling and a range of applications in insurance, risk management, econometrics, finance, signal processing, ecology, and medical statistics. Dr. Peters has lectured in the Department of Mathematics and Statistics in the University of New South Wales, Sydney, Australia (2008–2012) where he still holds an associate lecturer position. He has worked as a lead quantitative analyst in the Commonwealth Bank of Australia for 3 years and has more than 5 years industry experience in financial trading in the hedge fund sector and asset management.

Prof. Tomoko Matsui The Institute of Statistical Mathematics, Tokyo, Japan email: tmatsui@ism.ac.jp

She received the Ph.D. degree from the Computer Science Department, Tokyo Institute of Technology, Tokyo, Japan, in 1997. From 1988 to 2002, she was with

NTT, where she worked on speaker and speech recognition. From 1998 to 2002, she was with the Spoken Language Translation Research Laboratory, ATR, Kyoto, Japan, as a senior researcher and worked on speech recognition. From January to June 2001, she was an invited researcher in the Acoustic and Speech Research Department, Bell Laboratories, Murray Hill, NJ, working on finding effective confidence measures for verifying speech recognition results. She is currently a professor in the Institute of Statistical Mathematics, Tokyo, working on statistical modeling for speech and speaker recognition applications. Prof. Matsui received the paper award of the Institute of Electronics, Information, and Communication Engineers of Japan (IEICE) in 1993.

Contributors

Prof. Jen-Tzung Chien Department of Electrical and Computer Engineering, National Chiao Tung University, Taiwan

He received his Ph.D. degree in electrical engineering from National Tsing Hua University, Hsinchu, Taiwan, ROC, in 1997. During 1997–2012, he was with the National Cheng Kung University, Tainan, Taiwan. Since 2012, he has been with the Department of Electrical and Computer Engineering, National Chiao Tung University, Hsinchu. He held the visiting researcher position with the IBM T.J. Watson Research Center, Yorktown Heights, NY in 2010. His research interests include machine learning, speech recognition, blind source separation, information retrieval, and face recognition.

Dr. Chien served as the associate editor of the IEEE Signal Processing Letters in 2008–2011, the guest editor of the IEEE Transactions on Audio, Speech, and Language Processing in 2012, and the tutorial speaker of the Interspeech in 2013 and the ICASSP in 2012 and 2015. He coauthored the textbook "Bayesian Speech and Language Processing," Cambridge University Press, 2015. He received the Distinguished Research Award from the Ministry of Science and Technology, Taiwan, and the Best Paper Award of 2011 IEEE Automatic Speech Recognition and Understanding Workshop. He currently serves as an elected member of the IEEE Machine Learning for Signal Processing Technical Committee.

Prof. Kenji Fukumizu The Institute of Statistical Mathematics, Tokyo, Japan

He is a professor at The Institute of Statistical Mathematics, where he serves as director of the Research Center for Statistical Machine Learning. His research interests include machine learning and mathematical statistics. Before he joined the current institute in 2000, he worked as a researcher in the Research and Development Center, Ricoh Co., Ltd. from 1989 to 1998. He received Ph.D. of Science from Kyoto University in 1996. He then worked as a research scientist at the Brain Science Institute of the Institute of Physical and Chemical Research (RIKEN) from 1999 to 2000. He was a visiting scholar at the Department of Statistics, UC Berkeley in 2002, and worked in the Max Planck Institute for Biological Cybernetics as a Humboldt Fellow in 2006. Since 2010, he has been a visiting professor at the Department of Computational Intelligence and Systems Science, Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology. He serves as Chief Editor of Annals of the Institute of Statistical Mathematics, and as an associate editor of Journal of Machine Learning Research and Foundations and Trends in Machine Learning.

Prof. Konstantin Markov The University of Aizu, Fukushima, Japan

He was born in Sofia, Bulgaria. After graduating with honors from the St. Petersburg State Polytechnic University, he worked for several years as a research engineer at the Communication Industry Research Institute, Sofia, Bulgaria. He received his M.Sc. and Ph.D. degrees in electrical engineering from Toyohashi University of Technology, Japan, in 1996 and 1999, respectively. In 1998, he received the Best Student Paper Award from the IEICE Society. In 1999, he joined the research development department of ATR, Japan, and in 2000 became an invited researcher at the ATR Spoken Language Translation (SLT) Research Laboratories. Later, he became a senior research scientist at the Acoustics and Speech Processing Department of ATR SLT. In 2009 he joined the Human Interface Laboratory of the Information Systems Division, University of Aizu, Japan. He is a member of IEEE and ISCA. His research interests include audio signal processing, Bayesian statistical modeling, machine learning and pattern recognition.

Dr. Ido Nevat Institute for Infocomm Research, A*STAR, Singapore

He received the B.Sc. degree in electrical engineering from the Technion-Israel Institute of Technology, Haifa, Israel in 1998 and the Ph.D. degree in Electrical Engineering from the University of NSW, Sydney, Australia in 2010. From 2010 to 2013 he was a research fellow at the Wireless and Networking Technologies Laboratory at CSIRO, Australia. Currently, he is a scientist at the Institute for Infocomm Research (I2R), Singapore. His main areas of interests include statistical signal processing and Bayesian.

Chapter 1
Nonparametric Bayesian Inference with Kernel Mean Embedding

Kenji Fukumizu

Abstract Kernel methods have been successfully used in many machine learning problems with favorable performance in extracting nonlinear structure of high-dimensional data. Recently, nonparametric inference methods with positive definite kernels have been developed, employing the kernel mean expression of distributions. In this approach, the distribution of a variable is represented by the kernel mean, which is the mean element of the random feature vector defined by the kernel function, and relation among variables is expressed by covariance operators. This article gives an introduction to this new approach called *kernel Bayesian inference*, in which the Bayes' rule is realized with the computation of kernel means and covariance expressions to estimate the kernel mean of posterior [11]. This approach provides a novel nonparametric way of Bayesian inference, expressing a distribution with weighted sample, and computing posterior with simple matrix calculation. As an example of problems for which this kernel Bayesian inference is applied effectively, nonparametric state-space model is discussed, in which it is assumed that the state transition and observation model are neither known nor estimable with a simple parametric model. This article gives detailed explanations on intuitions, derivations, and implementation issues of kernel Bayesian inference.

1.1 Introduction

Recent data analysis often involves voluminous high-dimensional data, which may include continuous and complex-structured variables. Classical toolboxes of statistical data analysis may not be sufficient to derive useful information or make reliable predictions in such problems, since the methods often assume low-dimensional simple structure for data such as Gaussian distributions in Euclidean space. It is highly desirable to develop more flexible approaches to tackle those modern data analysis.

K. Fukumizu (✉)
The Institute of Statistical Mathematics, Tokyo, Japan
e-mail: fukumizu@ism.ac.jp

G.W. Peters and T. Matsui (eds.), *Modern Methodology and Applications in Spatial-Temporal Modeling*, JSS Research Series in Statistics,
DOI 10.1007/978-4-431-55339-7_1

Kernel methods have been developed as useful tools for generalizing linear statistical approaches to nonlinear settings. The main idea of kernel methods is to embed original data to a high-dimensional feature space, called a reproducing kernel Hilbert space (RKHS), and apply some linear methods of data analysis for the embedded feature vectors. With this approach, nonlinear features of data can be efficiently handled by virtue of the special way of computing the inner product, which is often called kernel trick. Since the proposal of support vector machines, a number of methods, such as kernel principle component analysis and kernel ridge regression, have been proposed along this discipline and successfully applied in many fields.

The aim of this article is to review recent development of kernel methods for nonparametric statistical inference. In the methods, the mean of the feature vector in the RKHS is considered as a summary for the distribution of feature vectors. We call it *kernel mean*. Although it might be thought that taking the mean loses information of the underlying distribution of data, if a kernel is chosen appropriately, the kernel mean maintains all the information of the distribution. This is possible by the fact that the kernel mean is a function with infinite degree of freedom in an infinite-dimensional RKHS. With this kernel mean approach, probability distributions are expressed by the corresponding kernel means, and linear operations with Gram matrices yield various algorithms for statistical inference, which includes homogeneity test [13–15, 26], independence test [16, 17], conditional independence test [9], and Bayes' theorem [11]. See [29] for a gentle introduction to these researches.

This article focuses on nonparametric kernel methods for Bayesian inference. In Bayesian inference, the sum rule, product rule, and Bayes' rule are important building blocks of inference procedures. The general kernel implementation of these three rules is first presented to realize a nonparametric method for Bayesian inference. As a basis, the conditional kernel mean is introduced and a new theoretical result on the convergence rate of its estimator is shown. A particularly important building block is the kernel implementation of Bayes' rule, called *Kernel Bayes' Rule* [11]. The KBR has special properties in comparison with other methods for Bayesian computation: (a) unlike other popular methods of computing posterior distributions such as Markov Chain Monte Carlo and sequential Monte Carlo, the KBR computes the kernel mean of posterior simply with linear operations of Gram matrices with no need of numerical integration or advanced approximate inference, (b) the ingredients for the Bayesian inference, prior and conditional probability (likelihood), are provided in the form of samples nonparametrically. Thus, this KBR approach is a purely nonparametric Bayesian inference.

A particularly useful application of the kernel Bayes' rule is nonparametric state-space model, for which sequential application of Bayes' rule realizes filtering, prediction, and smoothing. This paper particularly focuses on filtering with nonparametric state-space models, in which it is assumed that the state transition $p(x_{t+1}|x_t)$ and the observation model $p(y_t|x_t)$ are unknown but paired data for the state and observation variables are available for training. The detailed derivation of the kernel filtering algorithm based on the kernel Bayes' rule is presented.

The purpose of this article is to explain the kernel Bayesian inference with details together with some new results. In particular, as building blocks, kernel sum rule, kernel product rule, and kernel Bayes' rule are explained in detail including intuitions and derivations. A new theoretical result on the convergence rate of the conditional kernel mean estimator is presented using the decay rate of eigenvalues of the covariance operator. Additionally, as a typical application, details on the KBR filter are discussed including efficient low-rank approximation.

1.2 Representing Distributions with Kernel Mean Embedding

1.2.1 Preliminary: General Kernel Methods

We first give a brief review of positive definite kernels and kernel methods. A standard reference for readers unfamiliar with kernel methods is [28].

Given a set Ω, a ($\mathbb{R}$-valued) *positive definite kernel* k on Ω is a symmetric kernel $k : \Omega \times \Omega \to \mathbb{R}$ that satisfies positive semidefiniteness, i.e., $\sum_{i,j=1}^{n} c_i c_j k(x_i, x_j) \geq 0$ for arbitrary number of points $x_1, \ldots, x_n$ in Ω and real numbers $c_1, \ldots, c_n$. The matrix $(k(x_i, x_j))_{i,j=1}^{n}$ is called a *Gram matrix*. It is known [1] that a positive definite kernel on Ω uniquely defines a Hilbert space $\mathscr{H}$ consisting of functions on Ω such that the following three conditions hold: (i) $k(\cdot, x) \in \mathscr{H}$ for any $x \in \Omega$, (ii) $\mathrm{Span}\{k(\cdot, x) \mid x \in \Omega\}$ is dense in $\mathscr{H}$, and (iii) $\langle f, k(\cdot, x)\rangle = f(x)$ for any $x \in \Omega$ and $f \in \mathscr{H}$ (the reproducing property), where $\langle \cdot, \cdot \rangle$ is the inner product of $\mathscr{H}$. The Hilbert space $\mathscr{H}$ is called the *reproducing kernel Hilbert space* (RKHS) associated with k.

In kernel methods, Ω is a space where data exist, and a positive definite kernel k is prepared for Ω. The corresponding RKHS $\mathscr{H}$ is used as a feature space, and a nonlinear mapping (feature mapping) from data space Ω to the feature space $\mathscr{H}$ is defined by

$$\Phi : \Omega \to \mathscr{H}, \quad x \to k(\cdot, x),$$

where $k(\cdot, x) \in \mathscr{H}$ should be interpreted as a function of the first argument with x fixed. A data is thus mapped to a function, and this functional representation of data extracts various nonlinear features of data. From computational side, the reproducing property provides an efficient way of extracting nonlinear features in data analysis, without expanding the original variables with basis functions, which causes an intractably large number of components for high-dimensional original variables.

The traditional way of kernel methods considers the mapping of data $X_1, \ldots, X_n$ in the original space Ω to feature vectors $\Phi(X_i), \ldots, \Phi(X_n)$ in the RKHS, and apply some linear method of data analysis, such as principal component analysis, to those feature vectors. By the reproducing property, the inner product of two feature vectors is reduced to evaluation of the kernel, that is

$$\langle \Phi(x), \Phi(y) \rangle = k(x, y).$$

This fact is sometimes referred to as *kernel trick*, providing one of the essential elements in kernel methods. More generally, given two linear combinations of the feature vectors, say $f = \sum_{i=1}^{n} \alpha_i \Phi(X_i)$ and $g = \sum_{j=1}^{n} \beta_j \Phi(X_j)$, then the inner product between f and g is given by

$$\langle f, g \rangle = \alpha^T G_X \beta,$$

where $G_{X,ij} = k(X_i, X_j)$ is the Gram matrix. Given that computation of an analysis method for Euclidean data relies on the inner product among data points, the method can be extended to a nonlinear version with the above inner products among feature vectors. The computational cost thus does not depend on the dimensionality of data, once the Gram matrices are computed. This is computational advantage of kernel methods for handling high-dimensional data.

Computation with Gram matrices is obviously expensive if the sample size is large. It is known, however, that low-rank approximation of a Gram matrix reduces the size of the involved matrix drastically, while maintaining the approximation accuracy reasonably. As typical methods for low-rank approximation, the incomplete Cholesky decomposition [6] and Nyström approximation [38] approximate a Gram matrix G of size n to the form $G \approx RR^T$ with $n \times r$ matrix R in computational time proportional to n. Once the low-rank approximation is done, inversion $(G + \lambda I_n)^{-1}$ can be approximated by $I_n - R(R^T R + \lambda I_r)^{-1} R^T$ (Woodbury's formula), in which the inverse is taken for a matrix of size r. Here I_m denotes the $m \times m$ identity matrix. The merit of this approximation will be discussed in Sect. 1.4.2.

1.2.2 Kernel Mean Representation of Probability Distributions

In the recent development of kernel methods for nonparametric inference, the mean of the random feature vector $\Phi(X) = k(\cdot, X)$ is considered to represent a probability distribution on the random variable X.

More formally, let $(\mathscr{X}, \mathscr{B}_{\mathscr{X}})$ be a measurable space, X be a random variable taking values in $\mathscr{X}$ with probability distribution P on $\mathscr{X}$, and k be a measurable positive definite kernel on $\mathscr{X}$ such that $E[\sqrt{k(X, X)}] < \infty$. The associated RKHS is denoted by $\mathscr{H}$. The *kernel mean* m_X (also written by m_P) of X in $\mathscr{H}$ is defined by the mean $E[k(\cdot, X)]$ of the $\mathscr{H}$-valued random variable $\Phi(X)$.[1] Here, the mean

[1] As the kernel mean depends on k, it should be written by m_X^k rigorously. We will, however, generally write m_X for simplicity, where there is no ambiguity.

is interpreted as Bochner integral, which exists by the assumption $E[\|k(\cdot, X)\|] = E[\sqrt{k(X, X)}] < \infty$.

By the reproducing property, the kernel mean satisfies the relation

$$\langle f, m_X \rangle = E[f(X)] \tag{1.1}$$

for any $f \in \mathscr{H}$. Plugging $f = k(\cdot, u)$ into this relation yields

$$m_X(u) = E[k(u, X)] = \int k(u, \tilde{x}) dP(\tilde{x}), \tag{1.2}$$

which is an explicit integral form of the kernel mean.

To represent probabilities, an important notion is the characteristic property. A positive definite kernel k is called *bounded* if $\sup_{x \in \mathscr{X}} k(x, x) < \infty$. A bounded measurable positive definite kernel k on a measurable space $(\Omega, \mathscr{B})$ is called *characteristic* if the mapping from a probability Q on $(\Omega, \mathscr{B})$ to the kernel mean $m_Q \in \mathscr{H}$ is injective [7, 8, 32]. This is equivalent to assuming that $E_{X \sim P}[k(\cdot, X)] = E_{X' \sim Q}[k(\cdot, X')]$ implies $P = Q$ by definition: probabilities are uniquely determined by their kernel means on the associated RKHS. A popular example of a characteristic kernel defined on Euclidean space is the Gaussian RBF kernel $k(x, y) = \exp(-\|x - y\|^2 / (2\sigma^2))$. A characteristic kernel provides a RKHS that contains a rich class of functions so that the moments $E[f(X)]$ for all $f \in \mathscr{H}$ can identify the underlying distribution. Various conditions for a kernel to be characteristic can be found in [12, 31, 32].

By the unique representation property of characteristic kernels, statistical inference problems on probability distribution can be converted to the inference problems on the kernel means, which are easier to handle by the special properties of RKHS. This is the principle of the nonparametric inference with kernel means. Various inference methods have been proposed under this discipline. If we consider a two-sample problem, which aims at determining whether or not given two samples come from the same distribution, it can be cast as the problem of comparing the corresponding two kernel means in a RKHS [13]. The problem of independence test can be solved by comparing the kernel means of joint distributions and the product of the marginals [15].

When the relation of two random variables is discussed, covariance is useful in addition to means. In the kernel mean framework, covariance of the two feature vectors on the RKHS's is considered, and it is called covariance operator. More precisely, let $(\mathscr{X}, \mathscr{B}_{\mathscr{X}})$ and $(\mathscr{Y}, \mathscr{B}_{\mathscr{Y}})$ be measurable spaces, (X, Y) be a random variable on $\mathscr{X} \times \mathscr{Y}$ with distribution P, and $k_{\mathscr{X}}$ and $k_{\mathscr{Y}}$ be measurable positive definite kernels with respective RKHS $\mathscr{H}_{\mathscr{X}}$ and $\mathscr{H}_{\mathscr{Y}}$ such that $E[k_{\mathscr{X}}(X, X)] < \infty$ and $E[k_{\mathscr{Y}}(Y, Y)] < \infty$.[2] The (uncentered) *covariance operator* $C_{YX} : \mathscr{H}_{\mathscr{X}} \to \mathscr{H}_{\mathscr{Y}}$

[2] These conditions guarantee existence of the covariance operator. Note also $E[k(X, X)] < \infty$ is stronger than the condition for kernel mean, $E[\sqrt{k(X, X)}] < \infty$; this is obvious from Cauchy–Schwarz inequality.

is defined by

$$C_{YX} = E[k_{\mathscr{Y}}(\cdot, Y)\langle k_{\mathscr{X}}(X, \cdot), *\rangle],$$

or equivalently, for $f \in \mathscr{H}_{\mathscr{X}}$,

$$(C_{YX}f)(y) = E[k_{\mathscr{Y}}(y, Y)f(X)] = \int k_{\mathscr{Y}}(y, \tilde{y})f(\tilde{x})dP(\tilde{x}, \tilde{y}). \tag{1.3}$$

From the reproducing property, the covariance operator is a linear operator that satisfies

$$\langle g, C_{YX}f\rangle_{\mathscr{H}_{\mathscr{Y}}} = E[f(X)g(Y)]$$

for all $f \in \mathscr{H}_{\mathscr{X}}$, $g \in \mathscr{H}_{\mathscr{Y}}$. We also define C_{XX} by the operator on $\mathscr{H}_{\mathscr{X}}$ that satisfies $\langle f_2, C_{XX}f_1\rangle = E[f_2(X)f_1(X)]$ for any $f_1, f_2 \in \mathscr{H}_{\mathscr{X}}$.

The covariance operator is a natural extension of an ordinary covariance matrix: given two random vectors Z and W on Euclidean spaces, the covariance matrix can be regarded as a linear mapping $a \mapsto E[WZ^T]a$. Replacing Z and W with $k_{\mathscr{X}}(\cdot, X)$ and $k_{\mathscr{Y}}(\cdot, Y)$, respectively, yields the covariance operator $E[k_{\mathscr{Y}}(\cdot, Y)\langle k_{\mathscr{X}}(\cdot, X), *\rangle]$. Readers who are unfamiliar with the notion of operators can simply think of linear mappings on infinite-dimensional vector spaces to grasp the general ideas in this article.

Note also that by identifying the dual element $\langle k_{\mathscr{X}}(\cdot, X), *\rangle$ with $k_{\mathscr{X}}(\cdot, X)$, the covariance operator C_{YX} can be identified with the kernel mean $m_{YX} = E[k_{\mathscr{Y}}(\cdot, Y)\, k_{\mathscr{X}}(\cdot, X)]$ in the direct product $\mathscr{H}_{\mathscr{Y}} \otimes \mathscr{H}_{\mathscr{X}}$, which is given by the product kernel $k_{\mathscr{Y}}k_{\mathscr{X}}$ on $\mathscr{Y} \times \mathscr{X}$ [1]. This fact will be used in deriving kernel Bayes' rule.

Given i.i.d. sample $(X_1, Y_1), \ldots, (X_n, Y_n)$ with law P, the empirical estimators of the kernel mean and covariance operator are given straightforwardly by the empirical mean and covariance as

$$\widehat{m}_X = \frac{1}{n}\sum_{i=1}^{n} k_{\mathscr{X}}(\cdot, X_i), \qquad \widehat{C}_{YX}^{(n)} = \frac{1}{n}\sum_{i=1}^{n} k_{\mathscr{Y}}(\cdot, Y_i) \otimes k_{\mathscr{X}}(\cdot, X_i),$$

where $\widehat{C}_{YX}^{(n)}$ is written in the tensor form. These estimators are known to be $\sqrt{n}$-consistent in appropriate norms, and $\sqrt{n}(\widehat{m}_X - m_X)$ converges to a Gaussian process on $\mathscr{H}_{\mathscr{X}}$ [3].

1.3 Bayesian Inference with Kernel Means

There are three basic operations used in general Bayesian inference: sum rule, product rule, and Bayes' rule, which are summarized in Table 1.1. Correspondingly, in the framework of Bayesian inference with kernel means, these operations are realized

Table 1.1 Operations for Bayesian inference.

Density form		Kernel version $\widehat{m}_{\Pi} = \sum_j \gamma_j k_{\mathscr{X}}(\cdot, U_j)$, $(X_i, Y_i) \sim P$
Sum rule	$q_{\mathscr{Y}}(y) = \int p(y\|x)\pi(x)dx$	$\widehat{m}_{Q_{\mathscr{Y}}} = \sum_i w_i k_{\mathscr{Y}}(\cdot, Y_i)$, $w = (G_X + n\varepsilon_n I_n)^{-1} G_{XU}\gamma$
Product rule	$q(x, y) = p(y\|x)\pi(x)$	$\widehat{m}_Q = \sum_i w_i k_{\mathscr{X}}(\cdot, X_i) \otimes k_{\mathscr{Y}}(\cdot, Y_i)$, $w = (G_X + n\varepsilon_n I_n)^{-1} G_{XU}\gamma$
Bayes' rule	$q(x\|y_{\text{obs}}) = \frac{p(y_{\text{obs}}\|x)\pi(x)}{\int p(y_{\text{obs}}\|x)\pi(x)dx}$	$\widehat{m}_{Q_{x\|y_{\text{obs}}}} = \sum_i w_i k_{\mathscr{X}}(\cdot, X_i)$, $\Lambda = \text{Diag}\{(G_X + n\varepsilon_n I_n)^{-1} G_{XU}\gamma\}$, $w = \Lambda G_Y((\Lambda G_Y)^2 + \delta_n I_n)^{-1} \Lambda \mathbf{k}_Y(y_{\text{obs}})$

In the kernel version, $G_X = (k(X_i, X_j))$, $G_Y = (k(Y_i, Y_j))$, and $G_{XU} = (k(X_i, U_j))$

in terms of kernel means. This section first provides an intuitive explanation for the population version of the kernel realization, which may not be rigorous in handling operator inversion, and then shows rigorous empirical expressions, which can be proved to be consistent.

In the framework, each distribution is represented by the corresponding kernel mean or its empirical estimate. An empirical estimator of the kernel mean of a probability P is, in general, given by a weighted sum of feature vectors

$$\widehat{m}_P = \sum_{i=1}^{n} w_i k(\cdot, X_i),$$

where $(X_i)_{i=1}^n$ is some sample, which may not be generated by P.

1.3.1 Conditional Kernel Mean

For Bayesian inference with kernels, a basis is how to express or estimate the conditional kernel mean. It is not straightforward, however, to have an empirical expression of kernel mean of the conditional probability of Y given X. If we had many samples of Y for each value of x, we could just use the samples or their feature vectors to represent the kernel mean of Y given x. It is unlikely, however, that we have such *conditional samples*, if the variable X is continuous and random. We then need an alternative way of expressing the kernel mean of a conditional probability. We assume that there is a probability P with density $p(x, y)$ that gives a conditional density $p(y|x)$, and we have data (X_i, Y_i) generated by P.

The theoretical basis of the conditional kernel mean is the following theorem.

Theorem 1.3.1 *([7]) If, for $g \in \mathscr{H}_{\mathscr{Y}}$, $E[g(Y)|X = x]$ is included in $\mathscr{H}_{\mathscr{X}}$ as a function of x, then*

$$C_{XX}E[g(Y)|X = \cdot] = C_{XY}g.$$

The proof is easy from the fact $\langle C_{XX}E[g(Y)|X = \cdot], f\rangle = E[g(Y)f(X)] = \langle C_{XY}g, f\rangle$ for any $f \in \mathscr{H}_{\mathscr{X}}$. From this theorem, if C_{XX} is invertible, we have

$$E[g(Y)|X = \cdot] = C_{XX}^{-1}C_{XY}g.$$

Taking the inner product with $k_{\mathscr{X}}(\cdot, x)$ derives

$$\langle E[g(Y)|X = \cdot], k_{\mathscr{X}}(\cdot, x)\rangle_{\mathscr{H}_{\mathscr{X}}} = \langle C_{XX}^{-1}C_{XY}g, k_{\mathscr{X}}(\cdot, x)\rangle_{\mathscr{H}_{\mathscr{X}}},$$

which implies

$$\langle g, E[k_{\mathscr{Y}}(\cdot, Y)|X = x]\rangle_{\mathscr{H}_{\mathscr{Y}}} = \langle g, C_{YX}C_{XX}^{-1}k_{\mathscr{X}}(\cdot, x)\rangle_{\mathscr{H}_{\mathscr{Y}}}.$$

If $E[g(Y)|X = \cdot] \in \mathscr{H}_{\mathscr{X}}$ holds for any $g \in \mathscr{H}_{\mathscr{Y}}$, it follows that

$$E[k_{\mathscr{Y}}(\cdot, Y)|X = x] = C_{YX}C_{XX}^{-1}k_{\mathscr{X}}(\cdot, x). \tag{1.4}$$

Since the left-hand side of Eq. (1.4) is exactly the kernel mean of conditional probability of Y given $X = x$, this equation provides an expression of its kernel mean in terms of the covariance operator of the joint distribution (X, Y). Note, however, that the above reasoning involves a strong assumption: C_{XX} is invertible. In fact, this does not hold if the dimensionality of $\mathscr{H}_{\mathscr{X}}$ is infinite and C_{XX} has arbitrarily small or zero eigenvalues. This occurs in typical cases with a bounded kernel of infinite-dimensional RKHS, since the trace of the infinite-dimensional linear map C_{XX} is finite [10].

Nonetheless, from the expression Eq. (1.4), we can introduce an empirical estimator of the kernel mean of $p(y|x)$, namely, given i.i.d. sample $(X_1, Y_1), \ldots, (X_n, Y_n)$ following the joint distribution P, an estimator is defined by

$$\widehat{m}_{Y|X=x} := \widehat{C}_{YX}^{(n)}\big(\widehat{C}_{XX}^{(n)} + \varepsilon_n I\big)^{-1}k_{\mathscr{X}}(\cdot, x), \tag{1.5}$$

where I is the identity operator and ε_n is a regularization constant so that the operator can be inverted. This estimator is rigorously defined and proved to be consistent to $E[k_{\mathscr{Y}}(\cdot, Y)|X = x]$ under the sufficient condition in the following Theorem 1.3.2.

To describe the following convergence result, decay rate of eigenvalues is introduced. The eigenvalues of a positive compact operator C are said to *decay at rate b* if there is a constant $\beta > 0$ such that $\lambda_\ell \leq \beta\ell^{-b}$ for all ℓ, where (λ_ℓ) is the positive eigenvalues of C in descending order. (See [4]). The following theorem shows the convergence rate of the conditional kernel mean estimator.

Theorem 1.3.2 *Assume that* $E[k(X,\tilde{X})|Y=\cdot,\tilde{Y}=*]\in\mathscr{R}(C_{YY}\otimes C_{YY})$, *where* $(\tilde{X},\tilde{Y})$ *is an independent copy of* (X,Y), *and that the eigenvalues of* C_{YY} *decay at rate* b $(1<b<+\infty)$. *Then, with* $\varepsilon_n=n^{-b/(4b+1)}$,

$$\left\|\widehat{C}^{(n)}_{XY}(\widehat{C}^{(n)}_{YY}+\varepsilon_n I)^{-1}k_{\mathscr{Y}}(\cdot,y_0)-E[k_{\mathscr{X}}(\cdot,X)|Y=y_0]\right\|_{\mathscr{H}_{\mathscr{X}}}=O_p(n^{-b/(4b+1)})$$

as $n\to\infty$.

See the appendix for the proof. The decay rates of eigenvalues of a covariance operator are known in some typical cases; see [36, 37]. Note that the assumption of the decay rate of the covariance operator is related to the entropy number, and standard in discussing the behavior of kernel regression [33]. The assumption $E[k(X,\tilde{X})|Y=\cdot,\tilde{Y}=*]\in\mathscr{R}(C_{YY}\otimes C_{YY})$ requires the smoothness of the conditional expectation when the kernel is smooth such as Gaussian kernel; the range space consists of smoother functions by the smoothing effect of the integral in Eq. (1.3). To the best of our knowledge, the convergence rate of the conditional kernel mean in the above form has not been presented in existing literatures.[3]

1.3.2 Kernel Sum Rule and Kernel Product Rule

For the sum and product rules, this subsection gives intuitive explanation rather than rigorous convergence results. See [11] for the results.

For the kernel mean implementation of the sum rule, let Π be a probability on $\mathscr{X}$ with density $\pi(x)$. As in the previous subsection, we assume that there is a joint distribution P on $\mathscr{X}\times\mathscr{Y}$ with density $p(x,y)$ of which the conditional p.d.f. is equal to the given $p(y|x)$. Suppose that the sum rule gives $Q_{\mathscr{Y}}$ with density $q_{\mathscr{Y}}(y)$, i.e.,

$$q_{\mathscr{Y}}(y)=\int p(y|x)\pi(x)dx.$$

The kernel mean of $Q_{\mathscr{Y}}$ is then given by

$$m_{Q_y}=\int\int k_{\mathscr{Y}}(\cdot,y)p(y|x)\pi(x)dxdy.$$

[3] Some previous literatures derived a convergence rate at unrealistic assumptions. For example, Theorem 6 in [30] assumes $k(\cdot,y_0)\in\mathscr{R}(C_{YY})$ to achieve the rate $n^{-1/4}$, but in typical cases there is no function $f\in\mathscr{H}_{\mathscr{Y}}$ that satisfies $\int k(y,z)f(z)dP_Y(z)=k(y,y_0)$. Theorem 1.3.2 shows that if the eigenvalues decay sufficiently fast the rate approaches $n^{-1/4}$. As a relevant result, Theorem 11 in [11] shows a convergence rate of the kernel sum rule. While the conditional kernel mean is a special case of kernel sum rule with prior given by Dirac's delta function at x, the faster rate ($n^{-1/3}$ at best) is not achievable by Theorem 1.3.2, since the former assumes that π/p_X is a function in the RKHS and smooth enough.

From Eq. (1.4), we already know the (non-rigorous) expression

$$\int k_{\mathscr{Y}}(\cdot, y)p(y|x)dy = C_{YX}C_{XX}^{-1}k_{\mathscr{X}}(\cdot, x).$$

Plugging this into the previous equation, we have (the population version of) *kernel sum rule*:

$$m_{Q_{\mathscr{Y}}} = \int C_{YX}C_{XX}^{-1}k_{\mathscr{X}}(\cdot, x)\pi(x)dx = C_{YX}C_{XX}^{-1}m_{\Pi}. \tag{1.6}$$

There is another way to derive Eq. (1.6) in terms of density functions. Suppose that the density ratio π/p_X is included in $\mathscr{H}_{\mathscr{X}}$. From Eqs. (1.2) and (1.3), we see

$$m_{\Pi} = \int k_{\mathscr{X}}(\cdot, x)\pi(x)dx = \int k_{\mathscr{X}}(\cdot, x)\frac{\pi(x)}{p_X(x)}dP_X(x) = C_{XX}\left(\frac{\pi}{p_X}\right),$$

from which we obtain

$$C_{XX}^{-1}m_{\Pi} = \frac{\pi}{p_X}.$$

It follows from Eq. (1.3) that

$$\begin{aligned} C_{YX}C_{XX}^{-1}m_{\Pi} &= C_{YX}\Big(\frac{\pi}{p_X}\Big) = \int k_{\mathscr{Y}}(\cdot, y)\frac{\pi(x)}{p_X(x)}dP(x, y) \\ &= \int\int k_{\mathscr{Y}}(\cdot, y)p(y|x)\pi(x)dxdy = m_{Q_{\mathscr{Y}}}, \end{aligned}$$

which agrees with Eq. (1.6).

Given a consistent estimator $\widehat{m}_{\Pi}$ of m_{Π} and i.i.d. sample $(X_1, Y_1), \ldots, (X_n, Y_n)$ from P, the empirical version of the kernel sum rule is defined based on Eq. (1.6);

$$\widehat{m}_{Q_{\mathscr{Y}}} = \widehat{C}_{YX}^{(n)}\big(\widehat{C}_{XX}^{(n)} + \varepsilon_n I\big)^{-1}\widehat{m}_{\Pi}. \tag{1.7}$$

In a Gram matrix expression, given

$$\widehat{m}_{\Pi} = \sum_{j=1}^{\ell}\gamma_j k_{\mathscr{X}}(\cdot, U_j),$$

we have

$$\widehat{m}_{Q_{\mathscr{Y}}} = \sum_{i=1}^{n} w_i k_{\mathscr{Y}}(\cdot, Y_i), \quad w = (G_X + n\varepsilon_n I_n)^{-1}G_{XU}\gamma,$$

where $G_X = (k_{\mathscr{X}}(X_i, X_j))_{ij}$ and $G_{XU} = (k_{\mathscr{X}}(X_i, U_j))_{ij}$. The convergence of this estimator to the true $m_{Q_{\mathscr{Y}}}$ and its convergence rate are shown in Theorems 8 and

11 of [11]. For the convergence, it is assumed that the sample size ℓ for the prior increases as $n \to \infty$.

The kernel version of product rule can be derived as a special case of the kernel sum rule. Consider the conditional density $\tilde{p}(y, \tilde{x}|x) = p(y|x)\delta_x(\tilde{x})$ on the product space $\mathscr{Y} \times \mathscr{X}$, where δ_x is Dirac's delta function with mass concentrated at x. Let Q be a probability distribution on $\mathscr{Y} \times \mathscr{X}$ with density $p(y|x)\pi(x)$, i.e., the density given by the product rule. The population version of kernel sum rule applied to $\tilde{p}(y, \tilde{x}|x)$ and $\pi(x)$ with the product kernel then yields

$$m_Q = \int\int\int k_{\mathscr{Y}}(\cdot, y) \otimes k_{\mathscr{X}}(\cdot, \tilde{x})\tilde{p}(y, \tilde{x}|x)\pi(x)d\tilde{x}dydx = C_{(YX)X}C_{XX}^{-1}m_{\Pi},$$

where $C_{(YX)X} : \mathscr{H}_{\mathscr{X}} \to \mathscr{H}_{\mathscr{Y}} \otimes \mathscr{H}_{\mathscr{X}}$ is the covariance operator for the random variable $(X, (X, Y))$. Based on the (non-rigorous) population expression, we define the empirical kernel product rule by

$$\widehat{m}_Q := \widehat{C}^{(n)}_{(YX)X}\big(\widehat{C}^{(n)}_{XX} + \varepsilon_n I\big)^{-1}\widehat{m}_{\Pi}, \tag{1.8}$$

or in Gram matrix expression

$$\widehat{m}_Q = \sum_{i=1}^{n} w_i k_{\mathscr{Y}}(\cdot, Y_i) \otimes k_{\mathscr{X}}(\cdot, X_i), \quad w = (G_X + n\varepsilon_n I_n)^{-1}G_{XU}\gamma, \tag{1.9}$$

Note that the weight vectors of Eqs. (1.7) and (1.9) are exactly the same, while the feature vectors or the spaces of interest are different.

1.3.3 Kernel Bayes' Rule

As demonstrated in this subsection, by combining the kernel product rule and conditional kernel mean, we can easily derive the kernel Bayes' rule. As in the previous subsection, let Π be the prior and P be a probability on $\mathscr{X} \times \mathscr{Y}$ with conditional density $p(y|x)$. The distribution of the variable (X, Y) is P. The posterior distribution given y_{obs} is denoted by $Q_{x|y_{\text{obs}}}$.

From the expression of Bayes' rule

$$q(x|y_{\text{obs}}) = \frac{p(y|x)\pi(x)}{\int p(y|x)\pi(x)dx},$$

we see that the posterior is simply the conditional distribution of x given y_{obs} with the joint distribution Q given by the product rule. Once we have covariance operators for Q, Theorem 1.3.1 tells how to derive the conditional kernel mean, that is the kernel mean of posterior. The remaining task is thus to construct the covariance operators for Q.

Let (Z, W) denote a random variable taking values on $\mathscr{X} \times \mathscr{Y}$ with distribution Q. Then, from Eq. (1.8),

$$\widehat{m}_{(WZ)} = \widehat{C}^{(n)}_{(YX)X}\big(\widehat{C}^{(n)}_{XX} + \varepsilon_n I\big)^{-1}\widehat{m}_{\Pi} \quad \text{and} \quad \widehat{m}_{(WW)} = \widehat{C}^{(n)}_{(YY)X}\big(\widehat{C}^{(n)}_{XX} + \varepsilon_n I\big)^{-1}\widehat{m}_{\Pi},$$

where the second relation can be obtained in a similar way to the first one. Recall that the covariance operators C_{WZ} and C_{WW} are identified with the kernel means m_{WZ} and m_{WW}, respectively, on the product spaces, as discussed in Sect. 1.2.2. We can therefore obtain the estimator of $\widehat{C}^{(n)}_{WZ}$ and $\widehat{C}^{(n)}_{WW}$ from the above empirical version of kernel product rule. Namely, when the kernel product rule provides the empirical expressions

$$\widehat{m}_{(WZ)} = \sum_{i=1}^{n} \widehat{\mu}_i k(\cdot, Y_i) \otimes k(\cdot, X_i) \quad \text{and} \quad \widehat{m}_{(WW)} = \sum_{i=1}^{n} \widehat{\mu}_i k(\cdot, X_i) \otimes k(\cdot, X_i)$$

with

$$\widehat{\mu} = (G_X + n\varepsilon_n I_n)^{-1} G_{XU}\gamma, \tag{1.10}$$

the empirical estimators of covariance operators for Q are given by

$$\widehat{C}^{(n)}_{WZ} = \sum_{i=1}^{n} \widehat{\mu}_i k_{\mathscr{Y}}(\cdot, Y_i)\langle k_{\mathscr{X}}(\cdot, X_i), *\rangle, \quad \widehat{C}^{(n)}_{WW} = \sum_{i=1}^{n} \widehat{\mu}_i k_{\mathscr{Y}}(\cdot, Y_i)\langle k_{\mathscr{X}}(\cdot, Y_i), *\rangle.$$

Note that the coefficients to the feature vectors are the same for $\widehat{C}^{(n)}_{WZ}$ and $\widehat{C}^{(n)}_{WW}$.

In applying Eq. (1.5), there is another technical point. The estimated covariance operator $\widehat{C}^{(n)}_{WW}$ may not be positive definite, since the coefficients $\widehat{\mu}_i$ are not necessarily positive as the solution of the matrix operation Eq. (1.10). We use a more involved regularization to make the operator inversion possible, and introduce

$$\widehat{m}_{Q_x|y_{\text{obs}}} := \widehat{C}_{ZW}\big(\widehat{C}^2_{WW} + \delta_n I\big)^{-1}\widehat{C}_{WW} k_{\mathscr{Y}}(\cdot, y_{\text{obs}}). \tag{1.11}$$

This gives an estimator of the posterior kernel mean, and is called *Kernel Bayes' Rule* (KBR).

Theorem 1.3.3 (Kernel Bayes' Rule [11]) *For any $y_{\text{obs}} \in \mathscr{Y}$, the estimator $\widehat{m}_{Q_x|y_{\text{obs}}}$ of the posterior kernel mean is given by*

$$\widehat{m}_{Q_x|y_{\text{obs}}} = \sum_{i=1}^{n} w_i k(\cdot, X_i), \qquad w = \Lambda G_Y((\Lambda G_Y)^2 + \delta_n I_n)^{-1}\Lambda \mathbf{k}_Y(y_{\text{obs}}), \tag{1.12}$$

where $\Lambda = \operatorname{diag}(\widehat{\mu})$ is a diagonal matrix with elements $\widehat{\mu}_i$ in Eq. (1.10), and $\mathbf{k}_Y(y_{\text{obs}}) = (k_{\mathscr{Y}}(y_{\text{obs}}, Y_1), \ldots, k_{\mathscr{Y}}(y_{\text{obs}}, Y_n))^T \in \mathbb{R}^n$.

It is known that under some conditions the estimator $\widehat{m}_{Q_x|y_{\text{obs}}}$ converges to the true kernel mean of the posterior in probability, and an upper bound of its convergence rate is also known (Theorem 4, [11]).

The expression Eq. (1.12) takes the form of a weighted sum of feature vectors $k(\cdot, X_i)$, and is regarded as the kernel mean of the signed measure $\sum_{i=1}^{n} w_i \delta_{X_i}$. The KBR thus provides a weighted sample expression $(w_i, X_i)_{i=1}^{n}$ of the posterior. Note again that the weights may include negative values, which is different from ordinary weighted sample expression used popularly in importance sampling and particle filters. Figure 1.1 illustrates the procedure of KBR.

The above estimator provides the kernel mean of the posterior, and not the posterior itself. We need to develop methods for decoding necessary information of posterior from the kernel mean expression. Two methods are discussed below: estimation of expectation with respect to posterior and point estimation with the posterior.

If our aim is to estimate the expectation of a function $f \in \mathscr{H}_{\mathscr{X}}$ with respect to the posterior, the reproducing property of Eq. (1.1) gives an estimator

$$\langle f, \widehat{m}_{Q_x|y_{\text{obs}}} \rangle = \sum_{i=1}^{n} w_i f(X_i). \tag{1.13}$$

In fact, it is known that, under some conditions, the estimator Eq. (1.13) for any $f \in \mathscr{H}_{\mathscr{X}}$ converges to the expectation of f w.r.t. the true posterior, and its convergence rate is also known (Theorems 6 and 7, [11]). A recent work has shown that the consistency of $\sum_{i=1}^{n} w_i f(X_i)$ to $\int f(x) q_{x|y_{\text{obs}}}(x)dx$ is true for a wider class of functions than $\mathscr{H}_{\mathscr{Y}}$ [19]. This fact confirms similarity of (w_i, X_i) in KBR to the standard weighted sample expression.

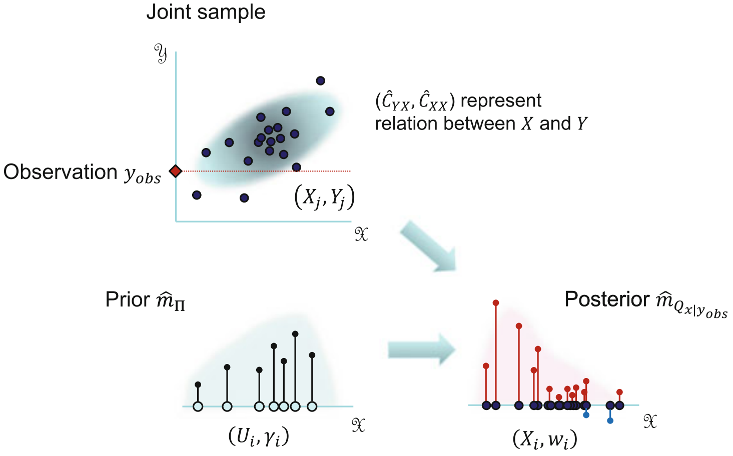
Joint sample
$\mathcal{Y}$
$(\hat{C}_{YX}, \hat{C}_{XX})$ represent relation between X and Y
Observation y_{obs}
(X_j, Y_j)
$\mathcal{X}$
Prior $\widehat{m}_{\Pi}$
Posterior $\widehat{m}_{Q_{x|y_{obs}}}$
(U_i, γ_i)
$\mathcal{X}$
(X_i, w_i)
$\mathcal{X}$

Fig. 1.1 Kernel Bayes' rule

If our aim is to obtain a point estimate based on the posterior, such as MAP, we can use a point $x \in \mathscr{X}$ such that the feature vector is the closest to the kernel mean of posterior [11, 30], i.e.,

$$\widehat{x} = \arg\min_{x \in \mathscr{X}} \left\| k_{\mathscr{X}}(\cdot, x) - \widehat{m}_{Q_x|y_{\text{obs}}} \right\|^2.$$

In the case of Gaussian kernel $k(x, x') = \exp(-\frac{1}{2\sigma^2}\|x - x'\|^2)$, from $\|k(\cdot, x)\| = 1$, the above minimization is equivalent to

$$\widehat{x} = \arg\max_{x \in \mathscr{X}} \sum_{i=1}^{n} w_i \exp\Big(-\frac{1}{2\sigma^2}\|x - X_i\|^2\Big),$$

which is similar to the MAP estimation, though $\sum_i w_i k(x, X_i)$ may not be a density function.

The above optimization problem can be solved in the same manner as the pre-image problem [24]. Taking the derivative of the squared norm in the right-hand side, we obtain the consistence equation

$$\widehat{x} = \frac{\sum_{i=1}^{n} w_i \exp(-\frac{1}{2\sigma^2}\|\widehat{x} - X_i\|^2)}{\sum_{i=1}^{n} \exp(-\frac{1}{2\sigma^2}\|\widehat{x} - X_i\|^2)},$$

which yields an iterative method for solving the point estimate:

$$\widehat{x}^{(t+1)} = \frac{\sum_{i=1}^{n} w_i \exp(-\frac{1}{2\sigma^2}\|\widehat{x}^{(t)} - X_i\|^2)}{\sum_{i=1}^{n} \exp(-\frac{1}{2\sigma^2}\|\widehat{x}^{(t)} - X_i\|^2)}.$$

Note that the objective function of pre-image problem is not necessarily convex and there may be local optima. The initial point of the above iteration must be chosen carefully. One possible method for initialization is to use the posterior mean. In the filtering problem discussed in Sect. 1.4, the estimate in the previous time step can serve as an initial point. Other pre-image methods [21] can be also applied to the above point estimation problem.

1.4 Kernel Bayesian Inference for State-Space Models

We discuss applications of KBR to the sequential Bayesian inference with state-space models. A time-invariant state-space model is defined by

$$p(X, Y) = \pi(X_1) \prod_{t=1}^{T+1} p(Y_t|X_t) \prod_{t=1}^{T} q(X_{t+1}|X_t),$$

where Y_t is an observation and X_t is a hidden state variable. The index t indicates time. The conditional probability $q(x_{t+1}|x_t)$ and $p(y_t|x_t)$ are called the state transition and observation model, respectively. With this model of time series, given $Y_1, \ldots, Y_t$, we wish to estimate the posteriors $p(X_s|Y_1, \ldots, Y_t)$. Filtering, prediction, and smoothing refer to as the case $s = t$, $s > t$, and $s < t$, respectively. This article discusses only the filtering problem for simplicity, while the other cases can be solved similarly.

1.4.1 KBR Filter

It is well known that, under the assumption of state-space models, application of Bayes' rule derives a sequential algorithm of filtering, which consists of two steps: prediction and correction steps.

Prediction step: Given an estimate of $p(x_t|y_1, \ldots, y_t)$, the conditional probability $p(x_{t+1}|y_1, \ldots, y_t)$ is estimated. This is done by the sum rule,

$$p(x_{t+1}|y_1, \ldots, y_t) = \int q(x_{t+1}|x_t)p(x_t|y_1, \ldots, y_t)dx_t. \tag{1.14}$$

Correction step: Given a new observation y_{t+1}, Bayes' rule derives the estimate of $p(x_{t+1}|y_1, \ldots, y_{t+1})$ with the prior $p(x_{t+1}|y_1, \ldots, y_t)$ and likelihood $p(y_t|x_t)$,

$$p(x_{t+1}|y_1, \ldots, y_{t+1}) = \frac{p(y_{t+1}|x_{t+1})p(x_{t+1}|y_1, \ldots, y_t)}{\int p(y_{t+1}|x_{t+1})p(x_{t+1}|y_1, \ldots, y_t)dx_{t+1}}. \tag{1.15}$$

If the state transition and observation model are given by linear mapping plus Gaussian noise, Kalman filter is the well-known filtering procedure. If they are written by known nonlinear dynamics, nonlinear extensions of Kalman filter, such as the extended Kalman filter (EKF) and unscented Kalman filter (UKF, [35]), are popular choices. In more general setting, given the state transition and observation model are known upto constant, the particle filter or sequential Monte Carlo [5] gives a weighted sample expression of the sequential update. These methods, however, require the precise knowledge on the functional form of the state transition and observation model, and not applicable unless they are known.

The KBR can be effectively applied to inference with the nonparametric setting of state-space models. In the nonparametric state-space models, it is not assumed that the conditional probabilities $p(Y_t|X_t)$ and $q(X_{t+1}|X_t)$ are known explicitly, nor estimated them with simple parametric models. Rather, it is assumed that training data $(X_1, Y_1), \ldots, (X_{T+1}, Y_{T+1})$ are given for both the observable and state variables in the *training phase*. In the *testing phase*, the state x_t is inferred based on a different sequence of observations $\tilde{y}_1, \ldots, \tilde{y}_t$ without knowing the corresponding state variables.

In the training phase, given the training sample, the observation model $p(y_t|x_t)$ and the state transition $q(x_{t+1}|x_t)$ are represented using the empirical covariances operators[4]: $\widehat{C}_{YX} = \frac{1}{T}\sum_{i=1}^{T} k_{\mathscr{Y}}(\cdot, Y_i) \otimes k_{\mathscr{X}}(\cdot, X_i)$, $\widehat{C}_{YY} = \frac{1}{T}\sum_{i=1}^{T} k_{\mathscr{Y}}(\cdot, Y_i) \otimes k_{\mathscr{Y}}$
$(\cdot, Y_i)$, $\widehat{C}_{XX} = \frac{1}{T}\sum_{i=1}^{T} k_{\mathscr{X}}(\cdot, X_i) \otimes k_{\mathscr{X}}(\cdot, X_i)$, and $\widehat{C}_{X_{+1}X} = \frac{1}{T}\sum_{i=1}^{T} k_{\mathscr{X}}(\cdot, X_{i+1}) \otimes k_{\mathscr{X}}(\cdot, X_i)$. In practice, we compute

$$G_X = (k_X(X_i, X_j))_{i,j=1}^T, \quad G_Y = (k_Y(Y_i, Y_j))_{i,j=1}^T, \quad \text{and} \quad G_{XX_{+1}} = (k_X(X_i, X_{j+1}))_{i,j=1}^T,$$

where $G_{XX_{+1}}$ is the "transfer" matrix.

In the testing phase, given new observations $\tilde{y}_1, \ldots, \tilde{y}_t$, the prediction and correction steps are kernelized. Suppose we already have an estimate of the kernel mean of $p(x_t|\tilde{y}_1, \ldots, \tilde{y}_t)$ in the form

$$\widehat{m}_{x_t|\tilde{y}_1,\ldots,\tilde{y}_t} = \sum_{s=1}^{T} \alpha_s^{(t)} k_{\mathscr{X}}(\cdot, X_s),$$

where $\alpha_i^{(t)} = \alpha_i^{(t)}(\tilde{y}_1, \ldots, \tilde{y}_t)$ are the coefficients at time t. For the prediction step (1.14), we can simply apply the kernel sum rule (1.7) to estimate the kernel mean of $p(x_{t+1}|\tilde{y}_1, \ldots, \tilde{y}_t)$:

$$\widehat{m}_{x_{t+1}|\tilde{y}_1,\ldots,\tilde{y}_t} = \widehat{C}^{(n)}_{X_{+1}X}\big(\widehat{C}^{(n)}_{XX} + \varepsilon_T I\big)^{-1}\widehat{m}_{x_t|\tilde{y}_1,\ldots,\tilde{y}_t} =: \sum_{j=1}^{T} \beta_j^{(t+1)} k(\cdot, X_{j+1}),$$
$$\text{where} \quad \beta^{(t+1)} = (G_X + T\varepsilon_T I_T)^{-1} G_X \alpha^{(t)}, \tag{1.16}$$

In the correction step (1.15), the kernel Bayes' rule first computes $\widehat{m}_{(y_{t+1}x_{t+1})|\tilde{y}_1,\ldots,\tilde{y}_t} = \widehat{C}^{(n)}_{(YX)X}(\widehat{C}^{(n)}_{XX} + \varepsilon_T I)^{-1}\widehat{m}_{x_{t+1}|\tilde{y}_1,\ldots,\tilde{y}_t}$ and $\widehat{m}_{(y_{t+1}y_{t+1})|\tilde{y}_1,\ldots,\tilde{y}_t} = \widehat{C}^{(n)}_{(YY)X}(\widehat{C}^{(n)}_{XX} + \varepsilon_T I)^{-1}$ $\widehat{m}_{x_{t+1}|\tilde{y}_1,\ldots,\tilde{y}_t}$, of which the coefficients are given by

$$\mu^{(t+1)} = \big(G_X + T\varepsilon_T I_T\big)^{-1} G_{XX_{+1}} \beta^{(t+1)}, \tag{1.17}$$

and next takes the conditioning, which yields

$$\alpha^{(t+1)} = \Lambda^{(t+1)} G_Y \big((\Lambda^{(t+1)} G_Y)^2 + \delta_T I_T\big)^{-1} \Lambda^{(t+1)} \mathbf{k}_Y(\tilde{y}_{t+1}), \tag{1.18}$$

where $\Lambda^{(t+1)} = \mathrm{diag}(\mu_1^{(t+1)}, \ldots, \mu_T^{(t+1)})$. Equations (1.16)–(1.18) describe the sequential update rule of the KBR filtering. The initial estimate $\widehat{m}^{(1)}_{x_1|\tilde{y}_1} = \sum_{i=1}^{T} \alpha_i^{(1)}$ $k(\cdot, X_i)$ can be computed by applying the KBR. We can also use the estimate of the

[4] Although the samples are not i.i.d., we assume an appropriate mixing condition and thus the empirical covariances converge to the covariances with respect to the stationary distribution as $T \to \infty$.

Table 1.2 Algorithm of the KBR filter

Input: Training data $(X_1, Y_1), \ldots, (X_T, Y_T)$, regularization constants ε_T, δ_T, kernels k_X, k_Y.

Training phase:

- Compute $G_X = (k_X(X_i, X_j))_{i,j=1}^T$, $G_Y = (k_Y(Y_i, Y_j))_{i,j=1}^T$, $G_{XX_{+1}} = (k_X(X_i, X_{j+1}))_{i,j=1}^T$.

Testing phase:

- Compute the initial estimate $\alpha^{(1)}$ given $\tilde{y}_1$.
- For $t = 1, 2, \ldots$, given $\tilde{y}_{t+1}$, do the following
  1. $\beta^{(t+1)} = (G_X + T\varepsilon_T I_T)^{-1} G_X \alpha^{(t)}$.
  2. $\mu^{(t+1)} = \big(G_X + T\varepsilon_T I_T\big)^{-1} G_{XX_{+1}} \beta^{(t+1)}$, $\Lambda^{(t+1)} = \mathrm{Diag}(\mu^{(t+1)})$.
  3. $\alpha^{(t+1)} = \Lambda^{(t+1)} G_Y \big((\Lambda^{(t+1)} G_Y)^2 + \delta_T I_T\big)^{-1} \Lambda^{(t+1)} \mathbf{k}_Y(\tilde{y}_{t+1})$

conditional kernel mean $E[k(\cdot, X)|Y = \tilde{y}_1]$ if the prior $\pi(X_1)$ is not available. The computation for the sequential filtering is summarized in Table 1.2.

Applications of the KBR filter to artificial data and camera-angle estimation problems are shown in [11], which demonstrates favorable performance of the KBR filter in comparison with other methods.

1.4.2 Discussions

The matrix inversion $(G_X + T\varepsilon_T I_T)^{-1}$ can be computed only once before the testing phase, while $((\Lambda^{(t+1)} G_Y)^2 + \delta_T I_T)^{-1}$ must be computed every time step in the testing phase, since it depends on $\widehat{\mu}^{(t+1)}$. Direct matrix inversion would cost $O(T^3)$, which is not feasible for large T. Substantial reduction in computational cost can be achieved by low-rank matrix approximations such as incomplete Cholesky factorization. Given an approximation of rank r for the Gram matrices and transfer matrix, the Woodbury identity yields the computation costs just $O(Tr^2)$ for each time step. In fact, let $G_X \approx R_X R_X^T$, $G_Y \approx R_Y R_Y^T$, and $G_{XX_+} \approx A_X B_{X_+}^T$ be the low-rank approximations, where the rank of R_X, R_Y, A_X and B_{X_+} is r at most. It is easy to see from the Woodbury identity that

$$\beta^{(t+1)} \approx \frac{1}{T\varepsilon_T}\big\{R_X R_X^T \alpha^{(t)} - R_X(R_X^T R_X + T\varepsilon_T I_r)^{-1}(R_X^T R_X) R_X^T \alpha^{(t)}\big\},$$

$$\mu^{(t+1)} \approx \frac{1}{T\varepsilon_T}\big\{A_X B_{X_+}^T \beta^{(t+1)} - R_X(R_X^T R_X + T\varepsilon_T I_r)^{-1}(R_X^T A_X) B_{X_+}^T \beta^{(t+1)}\big\},$$

and

$$\alpha^{(t+1)} \approx \frac{1}{\delta_T}\big\{\Lambda^{(t+1)}R_Y R_Y^T \Lambda^{(t+1)}\mathbf{k}_Y(\tilde{y}_{t+1}) \\ - \Lambda^{(t+1)}R_Y H_Y\big(H_Y^2 + \delta_T I_r\big)^{-1}H_Y R_Y^T \Lambda^{(t+1)}\mathbf{k}_Y(\tilde{y}_{t+1})\big\},$$

where $H_Y = R_Y^T \Lambda^{(t+1)} R_Y$ $(r \times r)$. Since $\Lambda^{(t+1)}$ is a diagonal matrix, all of the above computation can be done at the cost $O(Tr^2)$.

For the nonparametric state-space models, where training data are given as in the KBR filter, an alternative method is the conditional density estimation, including kernel density estimation or partitioning of the space [25, 34]. It is known, however, that these estimators have low estimation accuracy if the dimensionality is more than several. Empirical studies have shown that the KBR approach gives better estimation accuracy than the density estimation approach for large-dimensional cases; see [11].

Another possible Bayesian method for the nonparametric setting is Gaussian processes. An advantage of Gaussian processes is that one can use standard techniques of Bayesian inference such as hyperparameter selection with the marginal likelihood. Also, direct computation of the posterior is possible. On the other hand, the obtained posterior is unimodal by the nature of Gaussian distribution so that it may not be suitable for problems where multimodal posteriors are essential [22, 23]. In addition, since Gaussian processes are basically a model with univariate response, it is difficult to handle the correlation among a large number of response variables.

A possible limitation of the KBR filter is the assumption that training data exist including the state variable. While one might think it unrealistic, there are indeed some problems where one can obtain training data. One of such cases is expensive measurement: although one can observe the state variable, the measurement is very expensive, and one wishes to use a limited number of training data for inference. For instance, in sensor-based localization problems, pairs of sensor and location data can be once measured with some expensive devises and used for location estimation based solely on new sensor information [18, 27]. Another situation is that states are observed with considerable time delay. In this case, we can use the observed state variables for training, but the current state variable is not known and to be estimated.

It is true that performance of any kernel methods depends on the choice of a kernel. Additionally, in the KBR there are two regularization parameters to be chosen as hyperparameters. In the KBR filter, since we have training data for state variables, we can evaluate the prediction accuracy and thus use the validation approach by dividing the training data into the data for training and evaluation. This method for hyperparameter choice has been successfully used in the filtering applications of KBR in [11, 20].

This article discusses only the fully nonparametric setting of state-space models; both of the state transition and observation model are unknown and estimated nonparametrically. There are, however, semiparametric situations, where one of them is known. Consider vision-based robot localization problems, where the state x_t is the location and orientation of a robot, while the observation y_t is a movie image taken by video camera mounted on the robot. In this case, it is easy to provide a reasonable parametric model for the dynamics of robot move. On the other hand, the observation model from the location/orientation to the image is too complex and environment-dependent. It is thus preferable to apply a nonparametric method based on data for this observation model. Since the kernel method is purely a nonparametric method expressing the information with Gram matrices, it is not straightforward to combine the kernel Bayesian approach with parametric models. Reference [20] has proposed the kernel Monte Carlo filter, which is a combination of sampling and KBR method for the semiparametric situation, and demonstrated the preferable performance of the proposed method for the vision-based robot localization problem.

1.5 Conclusions

This article has provided detailed explanations of recently proposed kernel mean approach to Bayesian inference. The basic ideas, intuitions, and implementation issues have been discussed in details. A new result on the convergence rate of the estimator of conditional kernel mean has been also presented. As an application of the KBR approach, nonparametric state-space models are discussed focusing the algorithm and efficient computation.

Acknowledgments The author has been supported in part by MEXT Grant-in-Aid for Scientific Research on Innovative Areas 25120012.

Appendix: Proof of Theorem 1.3.2

First, we show a lemma to derive a convergence rate of conditional kernel mean.

Lemma 1.5.1 *Assume that the kernels are measurable and bounded. Let* $N(\varepsilon) := \mathrm{Tr}[C_{YY}(C_{YY} + \varepsilon I)^{-1}]$ *and* ε_n *be a constant such that* $\varepsilon_n \to 0$ *as* $n \to \infty$*. Then,*

$$\left\| (\widehat{C}_{YY}^{(n)} - C_{YY})(C_{YY} + \varepsilon_n I)^{-1} \right\| = O_p\left(\frac{1}{\varepsilon_n n} + \sqrt{\frac{N(\varepsilon_n)}{\varepsilon_n n}} \right)$$

and

$$\left\| (\widehat{C}_{XY}^{(n)} - C_{XY})(C_{YY} + \varepsilon_n I)^{-1} \right\| = O_p\left(\frac{1}{\varepsilon_n n} + \sqrt{\frac{N(\varepsilon_n)}{\varepsilon_n n}} \right)$$

as $n \to \infty$.

Proof The first result is shown in [4] (page 349). While the proof of the second one is similar, it is shown below for completeness.

Let ξ_{yx} be an element in $\mathscr{H}_{\mathscr{Y}} \otimes \mathscr{H}_{\mathscr{X}}$ defined by

$$\xi_{yx} := \big\{(C_{YY} + \varepsilon_n I)^{-1} k(\cdot, y)\big\} \otimes k(\cdot, x).$$

With identification between $Hy \otimes \mathscr{H}_{\mathscr{X}}$ and the Hilbert–Schmidt operators from $\mathscr{H}_{\mathscr{X}}$ to $\mathscr{H}_{\mathscr{Y}}$,

$$E[\xi_{YX}] = (C_{YY} + \varepsilon_n I)^{-1} C_{YX}.$$

Take $a > 0$ such that $k(x, x) \le a^2$ and $k(y, y) \le a^2$. It follows from $\|f \otimes g\| = \|f\|\,\|g\|$ and $\|(C_{YY} + \varepsilon_n I)^{-1}\| \le 1/\varepsilon_n$ that

$$\|\xi_{yx}\| = \big\|(C_{YY} + \varepsilon_n I)^{-1} k(\cdot, y)\big\| \, \big\|k(\cdot, x)\big\| \le \frac{1}{\varepsilon_n} \|k(\cdot, y)\|\, \|k(\cdot, x)\| \le \frac{a^2}{\varepsilon_n},$$

and

$$\begin{aligned}
E\|\xi_{YX}\|^2 &= E\big\|\{(C_{YY} + \varepsilon_n I)^{-1} k(\cdot, Y)\} \otimes k(\cdot, X)\big\|^2 \\
&= E\|k(\cdot, X)\|^2 \, \big\|(C_{YY} + \varepsilon_n I)^{-1} k(\cdot, Y)\big\|^2 \\
&\le a^2 E\big\|(C_{YY} + \varepsilon_n I)^{-1} k(\cdot, Y)\big\|^2 \\
&= a^2 E\big\langle (C_{YY} + \varepsilon_n I)^{-2} k(\cdot, Y), k(\cdot, Y)\big\rangle \\
&= a^2 E \mathrm{Tr}\big[(C_{YY} + \varepsilon_n I)^{-2} (k(\cdot, Y) \otimes k(\cdot, Y)^*)\big] \\
&= a^2 \mathrm{Tr}\big[(C_{YY} + \varepsilon_n I)^{-2} C_{YY}\big] \\
&\le \frac{a^2}{\varepsilon_n} \mathrm{Tr}\big[(C_{YY} + \varepsilon_n I)^{-1} C_{YY}\big] = \frac{a^2}{\varepsilon_n} N(\varepsilon_n).
\end{aligned}$$

Here $k(\cdot, Y)^*$ is the dual element of $k(\cdot, Y)$ and $k(\cdot, Y) \otimes k(\cdot, Y)^*$ is regarded as an operator on $\mathscr{H}_{\mathscr{Y}}$. In the last inequality, $(C_{YY} + \varepsilon_n I)^{-1}$ in the trace is replaced by its upper bound $\varepsilon_n^{-1} I$. Since $\frac{1}{n} \sum_{i=1}^{n} (C_{YY} + \varepsilon_n I)^{-1} \xi_{Y_i X_i} = (C_{YY} + \varepsilon_n I)^{-1} \widehat{C}_{YX}^{(n)}$, it follows from Proposition 2 in [4] that for all $n \in \mathbb{N}$ and $0 < \eta < 1$

$$\Pr\Bigg(\Big\| (C_{YY} + \varepsilon_n I)^{-1} \widehat{C}_{YX}^{(n)} - (C_{YY} + \varepsilon_n I)^{-1} C_{YX} \Big\| \\ \ge 2\left(\frac{2a^2}{n\varepsilon_n} + \sqrt{\frac{a^2 N(\varepsilon_n)}{\varepsilon_n n}} \right) \log \frac{2}{\eta} \Bigg) \le \eta,$$

which proves the assertion. □

Proof of Theorem 1.3.2 First, we have

$$
\begin{aligned}
&\big\|\widehat{C}^{(n)}_{XY}(\widehat{C}^{(n)}_{YY}+\varepsilon_n I)^{-1}k_{\mathscr{Y}}(\cdot,y_0)-E[k_{\mathscr{X}}(\cdot,X)|Y=y_0]\big\|_{\mathscr{H}_{\mathscr{X}}}\\
&\le \big\|\widehat{C}^{(n)}_{XY}(\widehat{C}^{(n)}_{YY}+\varepsilon_n I)^{-1}k_{\mathscr{Y}}(\cdot,y_0)-C_{XY}(C_{YY}+\varepsilon_n I)^{-1}k_{\mathscr{Y}}(\cdot,y_0)\|_{\mathscr{H}_{\mathscr{X}}} \qquad (1.19)\\
&\quad+\big\|C_{XY}(C_{YY}+\varepsilon_n I)^{-1}k_{\mathscr{Y}}(\cdot,y_0)-E[k_{\mathscr{X}}(\cdot,X)|Y=y_0]\big\|_{\mathscr{H}_{\mathscr{X}}}. \qquad (1.20)
\end{aligned}
$$

Using the general formula $A^{-1}-B^{-1}=A^{-1}(B-A)B^{-1}$ for any invertible operators A, B, the first term in the right-hand side of the above inequality is upper bounded by

$$
\begin{aligned}
&\big\|(\widehat{C}^{(n)}_{XY}-C_{XY})(\widehat{C}^{(n)}_{YY}+\varepsilon_n I)^{-1}k_{\mathscr{Y}}(\cdot,y_0)\big\|_{\mathscr{H}_{\mathscr{X}}}\\
&\qquad+\big\|C_{XY}(C_{YY}+\varepsilon_n I)^{-1}(C_{YY}-\widehat{C}^{(n)}_{YY})(\widehat{C}^{(n)}_{YY}+\varepsilon_n I)^{-1}k_{\mathscr{Y}}(\cdot,y_0)\big\|_{\mathscr{H}_{\mathscr{X}}}\\
&\le\big\|(\widehat{C}^{(n)}_{XY}-C_{XY})(\widehat{C}^{(n)}_{YY}+\varepsilon_n I)^{-1}\big\|\,\big\|k_{\mathscr{Y}}(\cdot,y_0)\big\|_{\mathscr{H}_{\mathscr{Y}}}\\
&\qquad+\frac{1}{\sqrt{\varepsilon_n}}\|C_{XX}\|^{1/2}\big\|(\widehat{C}^{(n)}_{YY}-C_{YY})(\widehat{C}^{(n)}_{YY}+\varepsilon_n I)^{-1}\big\|\,\big\|k_{\mathscr{Y}}(\cdot,y_0)\big\|_{\mathscr{H}_{\mathscr{Y}}},
\end{aligned}
$$

where in the second inequality the decomposition $C_{XY}=C_{XX}^{1/2}W_{XY}C_{YY}^{1/2}$ with some $W_{XY}:\mathscr{H}_{\mathscr{Y}}\to\mathscr{H}_{\mathscr{X}}$ ($\|W_{XY}\|\le 1$) [2] is used. It follows from Lemma 1.5.1 that

$$
\begin{aligned}
&\big\|\widehat{C}^{(n)}_{XY}(\widehat{C}^{(n)}_{YY}+\varepsilon_n I)^{-1}k_{\mathscr{Y}}(\cdot,y_0)-C_{XY}(C_{YY}+\varepsilon_n I)^{-1}k_{\mathscr{Y}}(\cdot,y_0)\|_{\mathscr{H}_{\mathscr{X}}}\\
&\qquad=O_p\left(\varepsilon_n^{-1/2}\left\{\frac{1}{\varepsilon_n n}+\sqrt{\frac{N(\varepsilon_n)}{\varepsilon_n n}}\right\}\right),
\end{aligned}
$$

as $n\to\infty$. It is known (Proposition 3, [4]) that, under the assumption on the decay rate of the eigenvalues, $N(\varepsilon)\le\frac{b\beta}{b-1}\varepsilon^{-1/b}$ holds with some $\beta\ge 0$. Since $\varepsilon_n^{-3/2}n^{-1}\ll\varepsilon_n^{-1-\frac{1}{2b}}n^{-1/2}$ for $b>1$ and $n\varepsilon_n\to\infty$, we have

$$
\begin{aligned}
&\big\|\widehat{C}^{(n)}_{XY}(\widehat{C}^{(n)}_{YY}+\varepsilon_n I)^{-1}k_{\mathscr{Y}}(\cdot,y_0)-C_{XY}(C_{YY}+\varepsilon_n I)^{-1}k_{\mathscr{Y}}(\cdot,y_0)\|_{\mathscr{H}_{\mathscr{X}}}\\
&\qquad=O_p\left(\varepsilon_n^{-1-\frac{1}{2b}}n^{-1/2}\right), \qquad (1.21)
\end{aligned}
$$

as $n\to\infty$.

For the second term of Eq. (1.19), let $\Theta:=E[k(X,\tilde{X})|Y=\cdot,\tilde{Y}=*]\in\mathscr{R}(C_{YY}\otimes C_{YY})$. Note that for any $\varphi\in\mathscr{H}_{\mathscr{Y}}$ we have

$$
\begin{aligned}
\langle C_{XY}\varphi,C_{XY}\varphi\rangle&=E[k(X,\tilde{X})\varphi(Y)\varphi(\tilde{Y})]\\
&=E\big[E[k(X,\tilde{X})|Y,\tilde{Y}]\varphi(Y)\varphi(\tilde{Y})\big]=\langle(C_{YY}\otimes C_{YY})\Theta,\varphi\otimes\varphi\rangle_{\mathscr{H}_{\mathscr{Y}}\otimes\mathscr{H}_{\mathscr{Y}}}.
\end{aligned}
$$

Similarly,

$$\begin{aligned}\langle C_{XY}\varphi, E[k(\cdot,X)|Y=y_0]\rangle_{\mathcal{H}_\mathcal{X}} &= \langle E[k(X,\tilde{X})|Y=y_0,\tilde{Y}=*], C_{YY}\varphi\rangle_{\mathcal{H}_\mathcal{Y}} \\ &= \langle (I\otimes C_{YY})\Theta, k(\cdot,y_0)\otimes\varphi\rangle_{\mathcal{H}_\mathcal{Y}\otimes\mathcal{H}_\mathcal{Y}}.\end{aligned}$$

It follows form these equalities with $\varphi = (C_{YY}+\varepsilon_n I)^{-1}k_\mathcal{Y}(\cdot,y_0)$ that

$$\begin{aligned}&\big\|C_{XY}(C_{YY}+\varepsilon_n I)^{-1}k_\mathcal{Y}(\cdot,y_0) - E[k_\mathcal{X}(\cdot,X)|Y=y_0]\big\|^2_{\mathcal{H}_\mathcal{X}} \\ &= \big\langle\big\{(C_{YY}+\varepsilon_n I)^{-1}C_{YY}\otimes(C_{YY}+\varepsilon_n I)^{-1}C_{YY} - I\otimes(C_{YY}+\varepsilon_n I)^{-1}C_{YY} \\ &\quad - (C_{YY}+\varepsilon_n I)^{-1}C_{YY}\otimes I + I\otimes I\big\}\Theta, k_\mathcal{Y}(\cdot,y_0)\otimes k_\mathcal{Y}(*,y_0)\big\rangle_{\mathcal{H}_\mathcal{Y}\otimes\mathcal{H}_\mathcal{Y}}.\end{aligned}$$

From the assumption $\Theta \in \mathscr{R}(\mathbb{C}_{YY}\otimes C_{YY})$, there is $\Psi \in \mathcal{H}_\mathcal{Y}\otimes\mathcal{H}_\mathcal{Y}$ such that $\Theta = (C_{YY}\otimes C_{YY})\Psi$. Let $\{\phi_i\}$ be the eigenvectors of C_{YY} with eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots 0$. Since the eigenvectors and eigenvalues of $C_{YY}\otimes C_{YY}$ are given by $\{\phi_i\otimes\phi_j\}_{ij}$ and $\lambda_i\lambda_j$, respectively, with the fact $(C_{YY}+\varepsilon_n I)^{-1}C_{YY}^2\phi_i = (\lambda_i^2/(1+\lambda_i))\phi_i$ and Parseval's theorem we have

$$\begin{aligned}&\big\|\big\{(C_{YY}+\varepsilon_n I)^{-1}C_{YY}\otimes(C_{YY}+\varepsilon_n I)^{-1}C_{YY} - I\otimes(C_{YY}+\varepsilon_n I)^{-1}C_{YY} \\ &\quad - (C_{YY}+\varepsilon_n I)^{-1}C_{YY}\otimes I + I\otimes I\big\}\Theta\big\|^2_{\mathcal{H}_\mathcal{Y}\otimes\mathcal{H}_\mathcal{Y}} \\ &= \sum_{i,j}\left\{\frac{\lambda_i^2}{\lambda_i+\varepsilon_n}\frac{\lambda_j^2}{\lambda_j+\varepsilon_n} - \frac{\lambda_i^2\lambda_j}{\lambda_i+\varepsilon_n} - \frac{\lambda_i\lambda_j^2}{\lambda_j+\varepsilon_n} + \lambda_i\lambda_j\right\}^2\langle\phi_i\otimes\phi_j,\Psi\rangle^2_{\mathcal{H}_\mathcal{X}\otimes\mathcal{H}_\mathcal{X}} \\ &= \varepsilon_n^4\sum_{i,j}\left\{\frac{\lambda_i\lambda_j}{(\lambda_i+\varepsilon_n)(\lambda_j+\varepsilon_n)}\right\}^2\langle\phi_i\otimes\phi_j,\Psi\rangle^2_{\mathcal{H}_\mathcal{X}\otimes\mathcal{H}_\mathcal{X}} \leq \varepsilon_n^4\|\Psi\|^2_{\mathcal{H}_\mathcal{X}\otimes\mathcal{H}_\mathcal{X}},\end{aligned}$$

which shows

$$\big\|C_{XY}(C_{YY}+\varepsilon_n I)^{-1}k_\mathcal{Y}(\cdot,y_0) - E[k_\mathcal{X}(\cdot,X)|Y=y_0]\big\|_{\mathcal{H}_\mathcal{X}} = O(\varepsilon_n). \qquad (1.22)$$

By balancing Eqs. (1.21) and (1.22), the assertion is obtained with $\varepsilon_n = n^{-b/(4b+1)}$.

□

References

1. Aronszajn, N.: Theory of reproducing kernels. Trans. Am. Math. Soc. **68**(3), 337–404 (1950)
2. Baker, C.: Joint measures and cross-covariance operators. Trans. Am. Math. Soc. **186**, 273–289 (1973)
3. Berlinet, A., Thomas-Agnan, C.: Reproducing kernel Hilbert Spaces in Probability and Statistics. Kluwer Academic Publisher (2004)
4. Caponnetto, A., De Vito, E.: Optimal rates for regularized least-squares algorithm. Found. Comput. Math. **7**(3), 331–368 (2007)

5. Doucet, A., Freitas, N.D., Gordon, N.: Sequential Monte Carlo Methods in Practice. Springer (2001)
6. Fine, S., Scheinberg, K.: Efficient SVM training using low-rank kernel representations. J. Mach. Learn. Res. **2**, 243–264 (2001)
7. Fukumizu, K., Bach, F., Jordan, M.: Dimensionality reduction for supervised learning with reproducing kernel Hilbert spaces. J. Mach. Learn. Res. **5**, 73–99 (2004)
8. Fukumizu, K., Bach, F., Jordan, M.: Kernel dimension reduction in regression. Ann. Stat. **37**(4), 1871–1905 (2009)
9. Fukumizu, K., Gretton, A., Sun, X., Schölkopf, B.: Kernel measures of conditional dependence. In: Advances in Neural Information Processing Systems 20, pp. 489–496. MIT Press (2008)
10. Fukumizu, K., R.Bach, F., Jordan, M.I.: Kernel dimension reduction in regression. Technical Report 715, Department of Statistics, University of California, Berkeley (2006)
11. Fukumizu, K., Song, L., Gretton, A.: Kernel Bayes' rule: Bayesian inference with positive definite kernels. J. Mach. Learn. Res. **14**, 3753–3783 (2013)
12. Fukumizu, K., Sriperumbudur, B.K., Gretton, A., Schölkopf, B.: Characteristic kernels on groups and semigroups. Adv. Neural Inf. Proc. Syst. **20**, 473–480 (2008)
13. Gretton, A., Borgwardt, K., Rasch, M., Schölkopf, B., Smola, A.: A kernel method for the two-sample-problem. In: Advances in Neural Information Processing Systems 19, pp. 513–520. MIT Press (2007)
14. Gretton, A., Borgwardt, K.M., Rasch, M.J., Schölkopf, B., Smola, A.: A kernel two-sample test. J. Mach. Learn. Res. **13**, 723–773 (2012)
15. Gretton, A., Fukumizu, K., Harchaoui, Z., Sriperumbudur, B.: A fast, consistent kernel two-sample test. Adv. Neural Inf. Process. Syst. **22**, 673–681 (2009)
16. Gretton, A., Fukumizu, K., Sriperumbudur, B.: Discussion of: brownian distance covariance. Ann. Appl. Stat. **3**(4), 1285–1294 (2009)
17. Gretton, A., Fukumizu, K., Teo, C.H., Song, L., Schölkopf, B., Smola, A.: A kernel statistical test of independence. In: Advances in Neural Information Processing Systems 20, pp. 585–592. MIT Press (2008)
18. Haeberlen, A., Flannery, E., Ladd, A.M., Rudys, A., Wallach, D.S., Kavraki, L.E.: Practical robust localization over large-scale 802.11 wireless networks. In: Proceedings of 10th International Conference on Mobile computing and networking (MobiCom '04), pp. 70–84 (2004)
19. Kanagawa, M., Fukumizu, K.: Recovering distributions from gaussian rkhs embeddings. J. Mach. Learn. Res. W&CP **3**, 457–465 (2014)
20. Kanagawa, M., Nishiyama, Y., Gretton, A., Fukumizu, K.: Monte carlo filtering using kernel embedding of distributions. In: Proceedings of 28th AAAI Conference on Artificial Intelligence (AAAI-14), pp. 1987–1903 (2014)
21. Kwok, J.Y., Tsang, I.: The pre-image problem in kernel methods. IEEE Trans. Neural Networks **15**(6), 1517–1525 (2004)
22. McCalman, L.: Function embeddings for multi-modal bayesian inference. Ph.D. thesis. School of Information Technology. The University of Sydney (2013)
23. McCalman, L., O'Callaghan, S., Ramos, F.: Multi-modal estimation with kernel embeddings for learning motion models. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 2845–2852 (2013)
24. Mika, S., Schölkopf, B., Smola, A., Müller, K.R., Scholz, M., Rätsch, G.: Kernel PCA and de-noising in feature spaces. In: Advances in Neural Information Pecessing Systems 11, pp. 536–542. MIT Press (1999)
25. Monbet, V., Ailliot, P., Marteau, P.: l^1-convergence of smoothing densities in non-parametric state space models. Stat. Infer. Stoch. Process. **11**, 311–325 (2008)
26. Moulines, E., Bach, F.R., Harchaoui, Z.: Testing for homogeneity with kernel Fisher discriminant analysis. In: Advances in Neural Information Processing Systems 20, pp. 609–616. Curran Associates, Inc. (2008)
27. Quigley, M., Stavens, D., Coates, A., Thrun, S.: Sub-meter indoor localization in unmodified environments with inexpensive sensors. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2010), pp. 2039 – 2046 (2010)

28. Schölkopf, B., Smola, A.: Learning with Kernels. MIT Press (2002)
29. Song, L., Fukumizu, K., Gretton, A.: Kernel embeddings of conditional distributions: a unified kernel framework for nonparametric inference in graphical models. IEEE Sig. Process. Mag. **30**(4), 98–111 (2013)
30. Song, L., Huang, J., Smola, A., Fukumizu, K.: Hilbert space embeddings of conditional distributions with applications to dynamical systems. In: Proceedings of the 26th International Conference on Machine Learning (ICML2009), pp. 961–968 (2009)
31. Sriperumbudur, B.K., Fukumizu, K., Lanckriet, G.: Characteristic kernels and rkhs embedding of measures. J. Mach. Learn. Res. Universality **12**, 2389–2410 (2011)
32. Sriperumbudur, B.K., Gretton, A., Fukumizu, K., Schölkopf, B., Lanckriet, G.: Hilbert space embeddings and metrics on probability measures. J. Mach. Learn. Res. **11**, 1517–1561 (2010)
33. Steinwart, I., Hush, D., Scovel, C.: Optimal rates for regularized least squares regression. Proc. COLT **2009**, 79–93 (2009)
34. Thrun, S., Langford, J., Fox, D.: Monte carlo hidden markov models: Learning non-parametric models of partially observable stochastic processes. In: Proceedings of International Conference on Machine Learning (ICML 1999), pp. 415–424 (1999)
35. Wan, E., and van der Merwe, R.: The unscented Kalman filter for nonlinear estimation. In: Adaptive Systems for Signal Processing, Communications, and Control Symposium (AS-SPCC 2000), pp. 153–158. IEEE (2000)
36. Widom, H.: Asymptotic behavior of the eigenvalues of certain integral equations. Trans. Am. Math. Soc. **109**, 278–295 (1963)
37. Widom, H.: Asymptotic behavior of the eigenvalues of certain integral equations II. Arch. Ration. Mech. Anal. **17**, 215–229 (1964)
38. Williams, C.K.I., Seeger, M.: Using the Nyström method to speed up kernel machines. In: Advances in Neural Information Processing Systems, vol. 13, pp. 682–688. MIT Press (2001)

Chapter 2
How to Utilize Sensor Network Data to Efficiently Perform Model Calibration and Spatial Field Reconstruction

Gareth W. Peters, Ido Nevat and Tomoko Matsui

Abstract This chapter provides a tutorial overview of some modern applications of the statistical modeling that can be developed based upon spatial wireless sensor network data. We then develop a range of new results relating to two important problems that arise in spatial field reconstructions from wireless sensor networks. The first new result allows one to accurately and efficiently obtain a spatial field reconstruction which is optimal in the sense that it is the Spatial Best Linear Unbiased Estimator for the field reconstruction. This estimator is obtained under three different system model configurations that represent different types of heterogeneous and homogeneous wireless sensor networks. The second novelty presented in this chapter relates to development of a framework that allows one to incorporate multiple sensed modalities from related spatial processes into the spatial field reconstruction. This is of practical significance for instance, if there are *d* spatial physical processes that are all being monitored by a wireless sensor network and it is believed that there is a relationship between the variability in the target spatial process to be reconstructed and the other spatial processes being monitored. In such settings it should be beneficial to incorporate these other spatial modalities into the estimation and spatial reconstruction of the target process. In this chapter we develop a spatial covariance regression framework to provide such estimation functionality. In addition, we develop a highly efficient estimation procedure for the model parameters via an Expectation Maximization algorithm. Results of the estimation and spatial field reconstructions are provided for two different real-world applications related to modeling the spatial relationships between coastal wind speeds and ocean height bathymetry measurements based on sensor network observations.

G.W. Peters (✉)
Department of Statistical Science, University College London, London, UK
e-mail: gareth.peters@ucl.ac.uk

I. Nevat
Institute for Infocomm Research, A*STAR, Singapore, Singapore
e-mail: idonevat@gmail.com

T. Matsui
The Institute of Statistical Mathematics, Tokyo, Japan
e-mail: tmatsui@ism.ac.jp

G.W. Peters and T. Matsui (eds.), *Modern Methodology and Applications in Spatial-Temporal Modeling*, JSS Research Series in Statistics,
DOI 10.1007/978-4-431-55339-7_2

2.1 Introduction to Wireless Sensor Networks

Wireless sensor networks (WSN) are composed of a large numbers of low-cost, low-power, densely distributed, and possibly heterogeneous sensors. WSN increasingly attract considerable research attention due to the large number of applications, such as environmental monitoring [36], weather forecasts [14, 15, 20, 35, 36], surveillance [39], health care [22], structural safety and building monitoring [9], and home automation [3, 15]. We consider WSN which consist of a set of spatially distributed low-cost sensors that have limited resources, such as energy and communication bandwidth. These sensors monitor a spatial physical phenomenon containing some desired attributes (e.g., pressure, temperature, concentrations of substance, sound intensity, radiation levels, pollution concentrations, etc.) and regularly communicate their observations to a Fusion Center (FC) in a wireless manner (for example, as in [4, 5, 12, 24, 38, 42]). The FC collects these observations and fuses them in order to reconstruct the signal of interest, based on which effective actions are made [3].

The majority of recent research on WSN consider problems related to addressing estimation of a single point source, such as source localization [23, 31, 32, 46, 47], or source detection (i.e., hypothesis testing) [11, 19, 26] class of problems. In [23, 31, 32, 46], location estimation algorithms of a scalar point source were developed, and in [47] the Posterior Cramér-Rao lower bound (PCRLB) for single target tracking in WSN with quantization was approximated via particle filters. In [11, 19], decision fusion algorithms for a single source detection were developed, and in [26] a vector-valued quantity of a single source was estimated in WSN with censoring and quantization.

In this chapter we explain how one can utilize the entire set of sensor data to not just obtain estimation of a given point source localization but instead to reconstruct the entire spatial field under a statistical model. Hence, we move beyond the estimation of a single location parameter by developing models to reconstruct the entire spatial random field which exhibits spatial dependency structure that we capture via either a homogeneous or nonhomogeneous spatial covariance function, depending on the statistical properties of the observed spatial field.

In general the following two fundamental problems naturally arise within this context, and they are the general focus of this chapter:

1. ***Spatial field model calibration and selection***: the task is to determine the best-fitting statistical model for the characterization of the spatial process and to perform the model parameter estimation and then model selection.
2. ***Spatial field reconstruction***: the task is to accurately estimate and predict the intensity of a spatial random field, not only at the locations of the sensors, but at a variety of other out-of-sample locations.

We consider in this chapter to model the physical phenomenon being monitored by the WSN according to a Gaussian random field (GRF) with a spatial correlation structure [4, 16, 28, 42]. More generally, examples of GRFs include wireless channels [2], speech processing [33], natural phenomena (temperature, rainfall intensity, etc.) [14, 20], and recently in models developed in [27, 29, 30, 34].

The simplest form of Gaussian process model would typically assume that the spatial field observed is only corrupted by additive Gaussian noise. For example, in [16] a linear regression algorithm for GRF reconstruction in mobile wireless sensor networks was presented, but relied on the assumption of only Additive White Gaussian Noise (AWGN); in [45] an algorithm was developed to learn the parameters of nonstationary spatiotemporal GRFs again assuming AWGN; and in [21] an algorithm for choosing sensor locations in GRF assuming AWGN was developed.

In practical WSN deployments, two deviations from these simplified modeling assumptions arise and can be important in practice to consider: these include the presence of heterogeneous sensor types, i.e., sensors may have different degrees of accuracy throughout the field of spatial monitoring; and secondly quite often the sensors may employ some form of energy conservation such as quantization of analog measurements to digital for efficient and low-power wireless transmission to the FC. To further elaborate these points, one may for instance consider the scenario in which high-quality sensors may be deployed by government agencies (e.g., weather stations). These are sparsely deployed due to their high costs, limited space constraints, high power consumption, etc. Then in order to improve the coverage of the WSN, low-quality cheap sensors perhaps employing quantization may be deployed to augment the higher quality analog sensor network [36]. For instance, battery operated low-cost sensors can be deployed and use simple wireless transmission techniques for data aggregation to the FC [43]. The low-quality sensors considered in this chapter transmit a single bit for every analog observation they obtain, making them very energy efficient. The FC then receives a vector of observations which are mixed continuous (high quality) and discrete (low-quality 1-bit values). This makes the data fusion a very complex inference problem.

Hence, the consequence of this type of practical framework is that the observations are heterogeneous and generally non-Gaussian distributed as the quantization procedure introduces a nonlinear transformation of the observations. The sensors transmit their quantized measurements to a FC over wireless channels, which introduce further distortion, due to bandwidth and power constraints. Such practical WSN were considered in [26, 31, 32, 44]. However, these works only considered the estimation of a point source and not of the entire spatial random field, the recent works of [27, 29, 30, 34] extend these frameworks to the entire field reconstruction problem, it is these frameworks that are summarized in this chapter. The intention of the chapter is to highlight and survey recent results that may be obtained for such modeling frameworks.

Notation: random variables are denoted by upper case letters and their realizations by lower case letters. In addition, bold will be used to denote a vector or matrix quantity, and lower subscripts refer to the element of a vector or matrix. We denote $N\left(x;\mu,\sigma^2\right)=\phi\left(x;\mu,\sigma^2\right)$ as the probability density function (PDF) of a random normal (Gaussian) variable with mean μ and variance σ^2. Its cumulative distribution function (CDF) is denoted by $\Phi\left(\lambda,\mu,\sigma^2\right)=\int_{-\infty}^{\lambda}\phi\left(x;\mu,\sigma^2\right)\mathrm{d}x$. We also define $\delta\left(a,b,c,d\right):=\phi\left(a;c,d\right)-\phi\left(b;c,d\right)$ and $\Delta\left(a,b,c,d\right):=\Phi\left(a;c,d\right)-\Phi\left(b;c,d\right)$. We will utilize throughout the chapter the following notations:

- $\mathbf{x}_{\mathcal{N}}$ is the physical location (in terms of $[x, y]$ coordinates) of the N sensors deployed in the field, comprised of N_A analog and N_D digital or quantised sensors such that $N = N_A + N_D$.
- $\mathbf{Y}_{\mathcal{N}} = \{Y_1, \ldots, Y_N\} \in \mathbb{R}^{N \times 1}$ is the collection of observations from all sensors (both analog and binary) at the fusion center.
- $\mathbf{Y}_{\mathcal{A}} \subseteq \mathbf{Y}_{\mathcal{N}}$ is the collection of observations from all N_A analog sensors at the fusion center which are located at points $\mathbf{x}_i \in \mathcal{X}^A$ such that $\text{Card}(\mathcal{X}^A) = N_A$.
- $\mathbf{Y}_{\mathcal{D}} \subseteq \mathbf{Y}_{\mathcal{N}}$ is the collection of observations from all N_D lower quality quantized or digital sensors at the fusion center which are located at points $\mathbf{x}_i \in \mathcal{X}^D$ such that $\text{Card}(\mathcal{X}^D) = N_D$.
- $\mathbf{f}_{\mathcal{N}} = \{f_1, \ldots, f_N\} \in \mathbb{R}^{N \times 1}$ is the realization of the random spatial field being monitored $f(\cdot)$ at the sensors located at $\mathbf{x}_{\mathcal{N}}$.
- $\mathbf{f}_{\mathcal{A}} \subseteq \mathbf{f}_{\mathcal{N}}$ is the realization of the random spatial field being monitored $f(\cdot)$ at the analog sensors, located at $\mathbf{x}_{\mathcal{A}} \subseteq \mathbf{x}_{\mathcal{N}}$.
- $\mathbf{f}_{\mathcal{D}} \subseteq \mathbf{f}_{\mathcal{N}}$ is the realization of the random spatial field being monitored $f(\cdot)$ at the digital sensors, located at $\mathbf{x}_{\mathcal{D}} \subseteq \mathbf{x}_{\mathcal{N}}$.
- $\mathbf{x}_{\mathcal{N}\setminus \text{n}} := \left[\mathbf{x}_1, \ldots, \mathbf{x}_{n-1}, \mathbf{x}_{n+1}, \ldots, \mathbf{x}_N\right]$.

Furthermore, we generically denote a location in space $\mathbf{x}_*$ for which the lower script $*$ indicates that a sensor is not located at this point to make a measurement but for which one wishes to reconstruct the spatial process $f(\mathbf{x}_*) = f_*$.

2.2 Introduction to Spatial Gaussian Random Fields

We consider a generic system model where wireless sensors are deployed in the field. The sensors monitor a spatial physical phenomenon which is observed with measurement error, quantization error, and incomplete sampling of the spatial field. These quantized measurements are transmitted over imperfect wireless channels to the fusion center (FC) to obtain an estimate of the spatial phenomenon at any point of interest in space. We first provide a formal definition of the spatial random Gaussian field followed by detailed WSN assumptions.

We assume that the observed phenomenon can be adequately modeled by a spatially dependent continuous process with a spatial correlation structure. The degree of the spatial correlation in the process increases with the decrease of the separation between two observing locations and can be accurately modeled as a Gaussian random field.[1] A Gaussian process (GP) defines a distribution over a space of functions and it is completely specified by the equivalent of sufficient statistics for such a process, and is formally defined as follows:

Definition 2.1 (*Gaussian process* [1, 37]): Let $\mathcal{X} \subset \mathbb{R}^D$ be some bounded domain of a d-dimensional real-valued vector space. Denote by $f(\mathbf{x}) : \mathcal{X} \mapsto \mathbb{R}$ a stochastic process parametrized by $\mathbf{x} \in \mathcal{X}$. Then, the random function $f(\mathbf{x})$ is a Gaussian

[1] We use Gaussian Process and Gaussian random field interchangeably.

process if all its finite-dimensional distributions are Gaussian, where for any $m \in \mathbb{N}$, the random variables $(f(\mathbf{x}_1), \dots, f(\mathbf{x}_m))$ are normally distributed.

A GP in this chapter is formally defined by the following class of random functions:

$$\begin{aligned}\mathcal{F} := \{f(\cdot) : \mathcal{X} \mapsto \mathbb{R} \text{ s.t. } f(\cdot) \sim \mathcal{GP}(\mu(\cdot; \boldsymbol{\Theta}), \mathcal{C}(\cdot, \cdot; \boldsymbol{\Omega})), \text{ with}\\ \mu(\mathbf{x}; \boldsymbol{\Theta}) := \mathbb{E}[f(\mathbf{x})] : \mathcal{X} \mapsto \mathbb{R},\\ \mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\Omega}\right) := \mathbb{E}\left[(f(\mathbf{x}_i) - \mu(\mathbf{x}_i; \boldsymbol{\Theta}))\left(f\left(\mathbf{x}_j\right) - \mu\left(\mathbf{x}_j; \boldsymbol{\Theta}\right)\right)\right] : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}\big\},\end{aligned}$$

where at each point the mean of the function is $\mu(\cdot; \boldsymbol{\Theta}) : \mathcal{X} \mapsto \mathbb{R}$, parameterised by $\boldsymbol{\Theta}$, and the spatial dependence between any two points is given by the covariance function (Mercer kernel) $\mathcal{C}(\cdot, \cdot; \boldsymbol{\Omega}) : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$, parameterised by $\boldsymbol{\Omega}$, see detailed discussion in [37].

It will be useful to make the following notational definitions:

$$\begin{aligned}k(\mathbf{x}_*, \mathbf{x}_{\mathcal{N}}) &:= \mathbb{E}[f(\mathbf{x}_*)\ f(\mathbf{x}_{\mathcal{N}})] \in \mathbb{R}^{1 \times N}\\ \mathcal{K}(\mathbf{x}_{\mathcal{N}}, \mathbf{x}_{\mathcal{N}}) &:= \begin{bmatrix} \mathcal{C}(\mathbf{x}_1, \mathbf{x}_1) & \cdots & \mathcal{C}(\mathbf{x}_1, \mathbf{x}_n) \\ \vdots & \ddots & \vdots \\ \mathcal{C}(\mathbf{x}_n, \mathbf{x}_1) & \cdots & \mathcal{C}(\mathbf{x}_n, \mathbf{x}_n) \end{bmatrix} \in \mathcal{S}^+\left(\mathbb{R}^n\right),\end{aligned}$$

with $\mathcal{S}^+(\mathbb{R}^n)$ is the manifold of symmetric positive definite matrices. To proceed with an understanding of this class of statistical model we need to consider the choice of the kernel functions available as these will have important implications for the ability of the GP model to capture the variability of the observed process over space.

2.2.1 Model Choices for Spatial Covariance Functions

In this section we discuss a few parametric family of kernels which characterize the covariance function in the Guassian process. A kernel, also called a covariance function, a kernel function, or a covariance kernel, is a positive definite function of two input vectors, for instance locations in space $\boldsymbol{x}_i \in \mathbb{R}^2$ and $\boldsymbol{x}_j \in \mathbb{R}^2$. There are many possible choices of covariance function that one may consider, sometimes the choice is based upon a known physical structure for the spatial processing being monitored, and other times the choice of kernel is obtained based on a statistical model selection procedure. In this section we briefly note some common choices considered in practice and the resulting properties of their implied covariance structure.

In many settings, it may be suitable to make a simplifying assumption such as assuming a spatially isotropic covariance structure in which the spatial covariance kernel may be modeled, for instance via the popular radial basis or the squared exponential function kernel given by

$$\mathbb{C}\text{ov}\left(f(\boldsymbol{x}), f(\boldsymbol{x}')\right) = \mathcal{C}_{\boldsymbol{\Omega}}(\mathbf{x}, \mathbf{x}') = \sigma^2 \exp\left(-\frac{||\mathbf{x}-\mathbf{x}'||_2^2}{2l^2}\right), \tag{2.1}$$

for parameter vector $\boldsymbol{\Omega} = (\sigma^2, l)$, with σ the magnitude of the covariance and l defining the characteristic length scale.

The second more flexible family of isotropic covariance function is recommended and used in a variety of application domains, see discussion in [25, 37], the Matern family of Mercer kernels which is characterized by covariance functions given by

$$\mathbb{C}\text{ov}\left(f(\boldsymbol{x}), f(\boldsymbol{x}')\right) = \mathcal{C}_{\boldsymbol{\Omega}}(\mathbf{x}, \mathbf{x}') = \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}||\mathbf{x}-\mathbf{x}'||_2}{l}\right)^{\nu} K_{\nu}\left(\frac{\sqrt{2\nu}||\mathbf{x}-\mathbf{x}'||_2}{l}\right), \tag{2.2}$$

for $\boldsymbol{\Omega} = (\nu, l)$ with $\nu > 0, l > 0$ and the modified Bessel function given by

$$K_{\nu}(x) = \int_0^{\infty} \exp\left(-x \cosh t\right) \cosh\left(\nu t\right) dt. \tag{2.3}$$

Other possible kernel choices widely used in practice include cases in which there is a periodic structure such as characterized by the kernel,

$$\mathbb{C}\text{ov}\left(f(\boldsymbol{x}), f(\boldsymbol{x}')\right) = \mathcal{C}_{\boldsymbol{\Omega}}(\mathbf{x}, \mathbf{x}') = \sigma^2 \exp\left(-\frac{2}{l^2}\sin^2\left(\pi\frac{\boldsymbol{x}-\boldsymbol{x}'}{p}\right)\right), \tag{2.4}$$

where $\boldsymbol{\Omega} = (\sigma, l, p)$.

Though these kernels presented above are widely used, it has been argued in [37, 40] that in many problems such restrictive isotropic and smoothness assumptions may not be appropriate for modeling realistic processes, in which case one may resort to an alternative class of covariance kernel which is less restrictive in terms of their spatial symmetries. For instance, one may consider the class of quadrant symmetric kernels that make less restrictive assumptions regarding the isotropy and involves selecting a kernel choice that satisfies the ‘even’ condition for each component given by

$$\mathcal{C}(x_1, \ldots, x_k, \ldots, x_n) = \mathcal{C}(x_1, \ldots, -x_k, \ldots, x_n). \tag{2.5}$$

where, quadrant symmetry implies homogeneity in the weak sense, see discussions in [41].

Another class of kernels one may consider is given by the anisotropic family of dot product “regression” kernels in which one considers the basic regression structure $\sigma_0^2 + \mathbf{x}^t\mathbf{x}$ and generalizes it with a covariance matrix and positive powers to obtain for strictly positive $\sigma > 0$ an inhomogeneous family. Typically, one considers one of three kernels in this family for the spatial covariance given by,

Linear Kernel:

$$\mathbb{C}\text{ov}\left(f(\boldsymbol{x}), f(\boldsymbol{x}')\right) = \mathcal{C}_{\boldsymbol{\Omega}}\left(\boldsymbol{x}, \boldsymbol{x}'\right) = \left(\sigma^2 + \mathbf{x}^T \Sigma_1 \mathbf{x}'\right),$$

Quadratic Kernel:

$$\mathbb{C}\text{ov}\left(f(\boldsymbol{x}), f(\boldsymbol{x}')\right) = \mathcal{C}_{\boldsymbol{\Omega}}\left(\boldsymbol{x}, \boldsymbol{x}'\right) = \left(\sigma^2 + \mathbf{x}^T \Sigma_2 \mathbf{x}'\right)^2, \tag{2.6}$$

Cubic Kernel:

$$\mathbb{C}\text{ov}\left(f(\boldsymbol{x}), f(\boldsymbol{x}')\right) = \mathcal{C}_{\boldsymbol{\Omega}}\left(\boldsymbol{x}, \boldsymbol{x}'\right) = \left(\sigma^2 + \mathbf{x}^T \Sigma_3 \mathbf{x}'\right)^3,$$

with $\boldsymbol{\Omega} = (\sigma, \Sigma_i, p_i)$. The linear covariance kernel can also be utilized under an alternative parameterization, it will prove to be beneficial in the context of the estimation developed in the following sections. Under this choice, one assumes that the spatial variability in the process $f(\cdot)$ can be explained by a set of covariates, denoted generically by $\mathbf{w} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{N^A}]$, where at each of a set of N^A sensors, a set of covariates $\mathbf{w}_i \in \mathbb{R}^{(q\times 1)}$ are observed and provide the following explanatory structure for the spatial processes correlations, given by

$$\mathbb{C}\text{ov}\left[f\left(\boldsymbol{x}_i\right), f\left(\boldsymbol{x}_j\right)|\mathbf{w}\right] = \mathcal{C}_{\boldsymbol{\Omega}}\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \left(\sigma^2 + \boldsymbol{b}_i^T \mathbf{w}\mathbf{w}^T \boldsymbol{b}_j\right),$$

where the parameters of association of the spatial field for each local point are defined by vectors $\boldsymbol{b}_i \in \mathbb{R}^{(q\times 1)}$ for $i \in \left\{1, \ldots, N^A\right\}$.

Having formally specified the semi-parametric class of Gaussian process models, we proceed with presenting the system model.

2.3 Wireless Sensor Network System Model

We now present the WSN system with practical quantization and imperfect wireless channels:

1. Consider a random spatial phenomenon to be monitored over a 2-dimensional space $\mathcal{X} \in \mathbb{R}^2$. The mean response of the physical process is a smooth continuous spatial function $f(\cdot) : \mathcal{X} \mapsto \mathbb{R}$, and is modeled as a Gaussian Process (GP) according to

$$f\left(\cdot\right) \sim \mathcal{GP}\left(\mu\left(\cdot; \boldsymbol{\Theta}\right), \mathcal{C}\left(\cdot, \cdot; \boldsymbol{\Omega}\right)\right).$$

2. Let N be the number of sensors that are deployed over a 2-D region $\mathcal{X} \subseteq \mathbb{R}^2$, with $\mathbf{x}_n \in \mathcal{X}, n = \{1, \ldots, N\}$, the physical location of the nth sensor, assumed known by the FC. The number of analog (high quality) and digital (lower quality) senors is N_{A} and N_{D}, respectively, so that $N = N_{\text{A}} + N_{\text{D}}$.
3. **Sensors measurement model**: each sensor collects a noisy observation of the spatial phenomenon $f\left(\cdot\right)$. At the nth sensor, the observation is expressed as:

$$Z\left(\mathbf{x}_n\right) = f\left(\mathbf{x}_n\right) + V_n^s,\ n = \{1, \ldots, N\}$$

where V_n^s are i.i.d. Gaussian noise terms, i.e., $V_n^s \sim N\left(0, \sigma_{\text{S}}^2\right)$.
4. **Analog (high quality) sensors processing and communication model**: each of the analog high-quality N_{A} sensors transmits its noisy observation to the FC over

AWGN channels, as follows:

$$Y_n^A = Z(\mathbf{x}_n) + V_n^A, \quad n = \{1, \ldots, N_{\mathrm{A}}\},$$

where V_n^A is i.i.d. Gaussian noise $V_n^A \sim N\left(0, \sigma_{\mathrm{A}}^2\right)$.

5. **Digital quantized (lower quality) sensors processing**: each of the N_{D} digital sensors first performs a thresholding-based decision based on its noisy observations. This step is summarized as follows for two common settings: first for the case in which an L-bit quantizer is assumed to operate at the sensor; second for the case in which a simple binary thresholding decision is performed.

   Setting 1—Low-Quality Power/Bandwidth Constrained Sensors:
   the quantizer explicitly maps its input $Z_n = Z(\mathbf{x}_n)$ to the output B_n through a mapping or encoder $B_n : \mathbb{R} \mapsto \{0, \ldots, L-1\}$, as follows:

$$B_n = \mathcal{Q}[Z_n] := \begin{cases} 0, & \lambda_0 \le Z_n < \lambda_1 \\ 1, & \lambda_1 \le Z_n < \lambda_2 \\ \vdots & \vdots \\ L-2, & \lambda_{L-2} \le Z_n < \lambda_{L-1} \\ L-1, & \lambda_{L-1} \le Z_n < \lambda_L, \end{cases}$$

   where $\lambda_0 = -\infty$ and $\lambda_L = \infty$.

   Setting 2—Basic Thresholding:
   the sensor simply thresholds via a binary decision rule (a special case of the $L = 1$ quantizer), with the binary decision rule given by:

$$B_n = \begin{cases} 1, & Z(\mathbf{x}_n) > \lambda \\ 0, & Z(\mathbf{x}_n) \le \lambda. \end{cases} \tag{2.7}$$

   where λ is a predefined threshold. We denote the thresholding operation by $\mathcal{Q}[\cdot]$.

6. **Digital quantized (lower quality) sensors communication model**: each of the N_{D} digital sensors, having first performed the quantization or thresholding-based decision on its noisy observations, then transmits the L-bit decision over imperfect wireless channels [11, 19, 26]. The decision $B_n = B(\mathbf{x}_n)$ is transmitted to the FC over imperfect binary wireless channels, as in [32]. Under this model, the statistic B_n is transmitted to the FC over imperfect wireless channels for which the conditional probability mass function (PMF) of the quantized/encoded observation from the nth sensor can be expressed, for all $m \in \{0, \ldots, L-1\}$, as:

$$\mathbb{P}\left(Y_n = m \middle| f(\mathbf{x}_n)\right) = \sum_{l=0}^{L-1} \mathbb{P}\left(Y_n = m \middle| B_n = l\right) \mathbb{P}\left(B_n = l \middle| f(\mathbf{x}_n)\right),$$

where $\mathbb{P}\left(Y_n = m \middle| B_n = l\right)$ represents the channels statistics (e.g., probability of making an error).

7. **Additional modalities sensed by high-quality analog sensors**: it is assumed that for the N_A analog sensors they are capable of making observations of additional spatial covariates, related to the physical process being monitored. At the nth analog sensor location the vector of additional spatial covariates is denoted $\widetilde{\mathbf{W}}_n \in \mathbb{R}^{(q\times 1)}$. The analog sensor then transmits this vector of additional covariates to the FC over AWGN channels, as follows:

$$\mathbf{W}_n = \widetilde{\mathbf{W}}_n + \mathbf{V}_n^C, \quad n = \{1, \ldots, N_\mathrm{A}\},$$

where $\mathbf{V}_n^C$ is i.i.d Gaussian noise $V_n^C \sim N(\mathbf{0}, \Sigma)$. In the remainder of this chapter we consider to stack all the N_A sensor covariates into a matrix $\boldsymbol{W} = \left[\mathbf{W}_1, \ldots, \mathbf{W}_{N_A}\right]$ for which we denote the realization by the matrix $\mathbf{w} \in \mathbb{R}^{(q\times N_A)}$.

2.3.1 Homogeneous and Heterogeneous WSNs

Hence, having specified this system model, we now consider two classes of WSN, the first will be termed the **"homogeneous sensor networks"** in which we assume each sensor performs processing of the sensed observed spatial phenomenon via the L-bit quantization before transmission to the FC for spatial field reconstruction. We note that in the ideal case $L \to \infty$ one would obtain from such a network the optimal estimation, in the sense of information content in the reconstruction of the spatial field. The second class of WSN we consider is the **"heterogeneous sensor networks"** in which a subset of sensors have capability, wireless transmission bandwidth and battery power, to transmit unquantized observations to the FC, whilst the remainder of cheaper sensors are bandwidth constrained, battery constrained, or inaccurate enough to only transmit L-bit quantized observations to the FC. In practice these lower quality sensors typically may even be simple binary quantizations of the analog sensed signal, in such cases one has $L = 1$ binary thresholding of the observed spatial field.

It is also not unreasonable to assume that the higher quality sensors which are not battery or bandwidth constrained may have additional capabilities to also observe or sense other spatial attributes in the monitoring environment. For instance, one may be interested in monitoring wind speed as the primary target spatial process, however, these higher quality sensors may also monitor other potentially related spatial physical attributes such as barometric pressure, temperature, humidity, and bathymetry. In general these other processes being monitored will be termed alternative modalities, and these modalities can often be very informative of the spatial structure and dynamics of the target physical process that one wishes to reconstruct the spatial field for based on the sensor observations. For this reason, we demonstrate next how to incorporate such other sensed modality information into the covariance structure of the target spatial process as part of a specialized form of spatial covariance regression.

In the case of the heterogeneous WSN model, we assumed that one may wish to incorporate alternative sensed modalities, i.e., exogenous spatial covariates which are observed jointly at the analog (high quality) sensors. This can be achieved in two standard ways in the regression model, through the trend (mean of the GP) or through the volatility in the sensed spatial process model (covariance function of the GP). In this manuscript we focus on incorporation of the spatial covariates into explaining helping to explain the spatial variability of the target spatial process with respect to both spatial structure as well as variability in these other sensed modalities. This creates a powerful class of models that has both spatial features and explanatory power derived from incorporation of related local spatial processes that should improve the accuracy of spatial reconstructions.

In this case one may consider to develop a spatial covariance kernel comprised of the following structure for any two locations $\mathbf{x}_i$ and $\mathbf{x}_j$:

$$\begin{aligned}\widetilde{\mathcal{C}}\left(\mathbf{x}_i,\mathbf{x}_j\right) &= \mathbb{E}\left[f\left(\mathbf{x}_i\right)\ f\left(\mathbf{x}_j\right)\right]\\ &= \mathcal{C}_{\boldsymbol{\Omega}}(\mathbf{x}_i,\mathbf{x}_j) + \left(\zeta_i^2 + \boldsymbol{b}_i^T\mathbf{w}\mathbf{w}^T\boldsymbol{b}_i\right)\mathbb{I}\left[\mathbf{x}_i \in \mathcal{X}^A, \mathbf{x}_j \notin \mathcal{X}^A\right]\\ &\quad + \left(\zeta_j^2 + \boldsymbol{b}_j^T\mathbf{w}\mathbf{w}^T\boldsymbol{b}_j\right)\mathbb{I}\left[\mathbf{x}_i \notin \mathcal{X}^A, \mathbf{x}_j \in \mathcal{X}^A\right]\\ &\quad + \left(\zeta_{i,j}^2 + \boldsymbol{b}_{ij}^T\mathbf{w}\mathbf{w}^T\boldsymbol{b}_{ij}\right)\mathbb{I}\left[\mathbf{x}_i \in \mathcal{X}^A, \mathbf{x}_j \in \mathcal{X}^A\right]\end{aligned} \tag{2.8}$$

where we denote the set of locations of the analog sensors in the WSN by the subset $\mathcal{X}^A \subseteq \mathcal{X}$. In this structure the first functional form $\mathcal{C}_{\boldsymbol{\Omega}}(\mathbf{x}_i,\mathbf{x}_j)$ represents the parameterization, via a kernel, for the spatial dependence of the target spatial field. The remaining three terms correspond to incorporated information in the spatial covariance regression structures arising from realizations of the additional modalities characterized by vector $\boldsymbol{w}$ which are only available at the analog sensor locations. This is quite a generic structure since many possible choices may be made for what would go into $\boldsymbol{w}$.

The validity of construction of the spatial kernel in this manner utilizes the fact that in general the linear combination of two kernels given by

$$k_{12}\left(\mathbf{x}_i,\mathbf{x}_j\right) = c_1 k_1\left(\mathbf{x}_i,\mathbf{x}_j\right) + c_2 k_2\left(\mathbf{x}_i,\mathbf{x}_j\right) \tag{2.9}$$

is a valid Mercer kernel and will construct a covariance matrix which will be symmetric and positive definite so long as $c_1, c_2 > 0$ and kernels k_1 and k_2 are Mercer kernels.

The construction of the covariance kernel in this manner admits two different types of interpretation of the resulting spatial model. The first is based on a linear combination of two GPs, the second is based on a hybrid model which involves a linear combination of a GP and a Gaussian graphical model (GMM) of [18]. In the remainder of this chapter we adopt the first approach.

When we interpret the spatial process model as a linear combination of two Gaussian processes, then this would be like thinking that theoretically the sensed additional modalities being utilized as covariates, which can be observed over the entire spatial domain and that they have a smooth functional relationship spatially. In this case the resulting GP model would be:

$$f(\cdot) = h(\cdot) + g(\cdot) \sim \mathcal{GP}\left(\mu\left(\cdot;\boldsymbol{\theta}_h\right) + \mu\left(\cdot;\boldsymbol{\theta}_g\right), \mathcal{C}_h\left(\cdot,\cdot\right) + \mathcal{C}_g\left(\cdot,\cdot\right)\right). \tag{2.10}$$

Here, we associate $\mathcal{C}_h(\cdot,\cdot)$ to $\mathcal{C}_{\boldsymbol{\Omega}}(\cdot,\cdot)$ and we interpret the Gaussian process $h(\cdot)$ as the baseline spatial process model and we associate $\mathcal{C}_g(\cdot,\cdot)$ with the additional spatial covariate terms from the additionally sensed modalities giving the spatial covariance function for the secondary, independent spatial Gaussian process $g(\cdot)$ given by manipulating (2.8) as follows:

$$\mathcal{C}_g(\cdot,\cdot) := \widetilde{\mathcal{C}}\left(\mathbf{x}_i, \mathbf{x}_j\right) - \mathcal{C}_{\boldsymbol{\Omega}}(\mathbf{x}_i, \mathbf{x}_j) \tag{2.11}$$

We note that if it is not suitable to make a smooth spatial relationship (potentially nonstationary in space) for the additional covariates variability in space, then in this case it would be more beneficial to think of the resulting model under the second interpretation of a hybrid GP and GGM model.

2.4 Model Calibration for WSN Spatial Models

For the different classes of WSN system models developed above we will require the ability to evaluate the spatial cross-correlation between observations of the target spatial process. This will be useful for both calibration purposes as well as spatial field estimation purposes.

Hence, we first consider the covariance matrix of the spatially distributed observations, given by $\mathbb{E}_{\mathbf{Y}_{\mathcal{N}}}\left[\mathbf{Y}_{\mathcal{N}}\mathbf{Y}_{\mathcal{N}}^T\right]$. The expression for the individual covariance terms in the covariance matrix need to be considered under one of three possible cases:

- Case 1 with $\mathbf{x}_i \in \mathcal{X}^A$ and $\mathbf{x}_j \in \mathcal{X}^A$, i.e., both sensors are high-quality analog sensors;
- Case 2 with $\mathbf{x}_i \in \mathcal{X}^A$ and $\mathbf{x}_j \in \mathcal{X}^D$, i.e., one sensor is analog and one sensor is a cheaper quantized sensor; and
- Case 3 in which $\mathbf{x}_i \in \mathcal{X}^D$ and $\mathbf{x}_j \in \mathcal{X}^D$. The resulting covariance matrix results for the (i, j)th components are specified in Theorem 2.1.

Theorem 2.1 (Covariance between Spatial Observations) *The (i, j)th term of* $\mathbb{E}_{\mathbf{Y}_{\mathcal{N}}}\left[\mathbf{Y}_{\mathcal{N}}\mathbf{Y}_{\mathcal{N}}^T\right]$ *is given by one of the following three cases where we define throughout the notation*

$$
\begin{aligned}
c_2 &:= \frac{\mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_j\right)}{\mathcal{C}\left(\mathbf{x}_j, \mathbf{x}_j\right)}, \\
c_1 &:= \mu\left(\mathbf{x}_i\right) - c_2 \mu\left(\mathbf{x}_j\right), \\
G_1(a, b; m, s) &:= \{\Phi(a; m, s) - \Phi(b; m, s)\} \\
G_2(a, b; m, s) &:= \{\phi(a; m, s) - \phi(b; m, s)\}.
\end{aligned}
$$

Case 1: $\mathbf{x}_i \in \mathcal{X}^{\mathbf{A}}$ ***and*** $\mathbf{x}_j \in \mathcal{X}^A$
In this case one has two high-quality analog sensors resulting in the cross-correlation given by

$$
\begin{aligned}
\mathbb{E}_{Y_i,Y_j}\left[Y_i Y_j\right] &= \mathbb{E}_{f_i,f_j}\left[\mathbb{E}_{Y_i,Y_j}\left[Y_i Y_j \middle| f_i, f_j\right]\right] = \mathbb{E}_{f_i,f_j}\left[f_i f_j\right] \\
&= \mathcal{C}_g\left(\mathbf{x}_i, \mathbf{x}_j\right) + \mathcal{C}_h\left(\mathbf{x}_i, \mathbf{x}_j\right).
\end{aligned}
$$

Case 2: $\mathbf{x}_i \in \mathcal{X}^A$ ***and*** $\mathbf{x}_j \in \mathcal{X}^D$
In this case one has a high-quality analog sensor and a lower quality L-bit quantized sensor resulting in the cross-correlation given by

$$
\begin{aligned}
\mathbb{E}_{Y_i,Y_j}\left[Y_i Y_j\right] &= \mathbb{E}_{f_i,f_j}\left[\mathbb{E}_{Y_i,Y_j}\left[Y_i Y_j \middle| f_i, f_j\right]\right] \\
&= \mathbb{E}_{f_j}\left[\left(c_1 + c_2 f_j\right) \sum_{l=0}^{L-1} \sum_{k=0}^{L-1} l \mathbb{P}r\left(Y_j = l | B_j = k\right) \left[G_1\left(\lambda_{k+1}, \lambda_k; f_j, \sigma_A^2\right)\right]\right] \\
&= c_1 \sum_{l=0}^{L-1} \sum_{k=0}^{L-1} l \mathbb{P}r\left(Y_j = l | B_j = k\right) \mathbb{E}_{f_j}\left[G_1\left(\lambda_{k+1}, \lambda_k; f_j, \sigma_A^2\right)\right] \\
&\quad + \ c_2 \sum_{l=0}^{L-1} \sum_{k=0}^{L-1} l \mathbb{P}r\left(Y_j = l | B_j = k\right) \mathbb{E}_{f_j}\left[f_j G_1\left(\lambda_{k+1}, \lambda_k; f_j, \sigma_A^2\right)\right],
\end{aligned}
\tag{2.12}
$$

where we obtain for the first integral the closed form expression

$$
\begin{aligned}
&\mathbb{E}_{f_j}\left[G_1\left(\lambda_{k+1}, \lambda_k; f_j, \sigma_A^2\right)\right] \\
&\quad = \frac{1}{\sigma_{f_j}^2}\left[\Phi\left(\frac{\mu_{f_j} - \lambda_{k+1}}{\sqrt{1 + \frac{\sigma_{f_j}^2}{\sigma_A^2}}}\right) - \Phi\left(\frac{\mu_{f_j} - \lambda_k}{\sqrt{1 + \frac{\sigma_{f_j}^2}{\sigma_A^2}}}\right)\right].
\end{aligned}
\tag{2.13}
$$

and the second integral the closed form expression

$$
\begin{aligned}
&\mathbb{E}_{f_j}\left[f_j G_1\left(\lambda_{k+1}, \lambda_k; f_j, \sigma_A^2\right)\right] \\
&\quad = \left(\frac{\sigma_{f_j}^2 + \mu_{f_j}}{\sigma_A^2 \sqrt{1 + \frac{\sigma_{f_j}^2}{\sigma_A^2}}}\right)\left[\Phi\left(\frac{\mu_{f_j} - \lambda_{k+1}}{\sqrt{1 + \frac{\sigma_{f_j}^2}{\sigma_A^2}}}\right) - \Phi\left(\frac{\mu_{f_j} - \lambda_k}{\sqrt{1 + \frac{\sigma_{f_j}^2}{\sigma_A^2}}}\right)\right].
\end{aligned}
\tag{2.14}
$$

***Case 3**: $\mathbf{x}_i \in \mathcal{X}^D$ **and** $\mathbf{x}_j \in \mathcal{X}^D$*
In this case one has two lower quality L-bit quantized sensors resulting in the cross-correlation given by

$$\begin{aligned}\mathbb{E}_{Y_i,Y_j}\left[Y_iY_j\right] = \sum_{k=0}^{L-1}\sum_{l=0}^{L-1} kl \sum_{m=0}^{L-1}\sum_{n=0}^{L-1} &\mathbb{P}\left(Y_i = k|B_i = m\right)\mathbb{P}\left(Y_j = l|B_j = n\right) \\ &\times\ \mathbb{E}_{f_i,f_j}\left[G_1\left(\lambda_{m+1},\lambda_m;\ f_i,\sigma_A^2\right)G_1\left(\lambda_{n+1},\lambda_n;\ f_j,\sigma_A^2\right)\right].\end{aligned} \tag{2.15}$$

Note: the approximation of the expectations in case 3 will be provided in detail in Sect. 2.5.1.2 where an efficient specialized form of quadrature rule will be developed based on the discrete cosine transform, known as the Clenshaw–Curtis quadrature rule.

2.4.1 Proof of Theorem 2.1

Using the law of total expectation, the (i, j)th term of $\mathbb{E}_{\mathbf{Y}_{\mathcal{N}}}\left[\mathbf{Y}_{\mathcal{N}}\mathbf{Y}_{\mathcal{N}}^T\right]$ is expressed as $\mathbb{E}_{Y_i,Y_j}\left[Y_iY_j\right] = \mathbb{E}_{f_i,f_j}\left[\mathbb{E}_{Y_i,Y_j}\left[Y_iY_j|f_i, f_j\right]\right]$. Deriving this quantity for Case 1 is trivial, so we focus on Case 2 and Case 3 below.

Case 2: $\mathbf{x}_i \in \mathcal{X}^A$ **and** $\mathbf{x}_j \in \mathcal{X}^D$
In this case one has a high-quality analog sensor and a lower quality L-bit quantized sensor resulting in the cross-correlation given by

$$\begin{aligned}\mathbb{E}_{Y_i,Y_j}\left[Y_iY_j\right] &= \mathbb{E}_{f_i,f_j}\left[\mathbb{E}_{Y_i,Y_j}\left[Y_iY_j|\, f_i, f_j\right]\right] \\ &= \mathbb{E}_{f_i,f_j}\left[\int\sum_{l=0}^{L-1} y_i l\Pr\left(Y_j = l|f_j\right) f_{Y_i}\left(y_i|f_i\right)dy_i\right] \\ &= \mathbb{E}_{f_i,f_j}\left[\int y_i\sum_{l=0}^{L-1}\sum_{k=0}^{L-1} l\Pr\left(Y_j = l|B_j = k\right)\Pr\left(B_j = k|f_j\right) f_{Y_i}\left(y_i|f_i\right)dy_i\right] \\ &= \mathbb{E}_{f_i,f_j}\left[f_i\sum_{l=0}^{L-1}\sum_{k=0}^{L-1} l\Pr\left(Y_j = l|B_j = k\right)\left[G_1\left(\lambda_{k+1},\lambda_k;\, f_j,\sigma_A^2\right)\right]\right] \\ &= \mathbb{E}_{f_j}\left[\int f_i f_{f_i|f_j}(f_i)df_i\sum_{l=0}^{L-1}\sum_{k=0}^{L-1} l\Pr\left(Y_j = l|B_j = k\right)\left[G_1\left(\lambda_{k+1},\lambda_k;\, f_j,\sigma_A^2\right)\right]\right] \\ &= \mathbb{E}_{f_j}\left[\left(c_1 + c_2 f_j\right)\sum_{l=0}^{L-1}\sum_{k=0}^{L-1} l\Pr\left(Y_j = l|B_j = k\right)\left[G_1\left(\lambda_{k+1},\lambda_k;\, f_j,\sigma_A^2\right)\right]\right]\end{aligned}$$

We may now work out these integrals for case 2 given generically by

$$\begin{aligned}
\mathbb{E}_{Y_i,Y_j}\left[Y_iY_j\right] &= \mathbb{E}_{f_i,f_j}\left[\mathbb{E}_{Y_i,Y_j}\left[Y_iY_j\middle|\, f_i, f_j\right]\right]\\
&= \mathbb{E}_{f_j}\left[\left(c_1+c_2f_j\right)\sum_{l=0}^{L-1}\sum_{k=0}^{L-1} l\mathbb{P}\mathrm{r}\left(Y_j=l|B_j=k\right)\left[G_1\left(\lambda_{k+1},\lambda_k;\, f_j,\sigma^2_{\mathrm{A}}\right)\right]\right]\\
&= c_1\sum_{l=0}^{L-1}\sum_{k=0}^{L-1} l\mathbb{P}\mathrm{r}\left(Y_j=l|B_j=k\right)\mathbb{E}_{f_j}\left[G_1\left(\lambda_{k+1},\lambda_k;\, f_j,\sigma^2_{\mathrm{A}}\right)\right]\\
&\quad + c_2\sum_{l=0}^{L-1}\sum_{k=0}^{L-1} l\mathbb{P}\mathrm{r}\left(Y_j=l|B_j=k\right)\mathbb{E}_{f_j}\left[f_jG_1\left(\lambda_{k+1},\lambda_k;\, f_j,\sigma^2_{\mathrm{A}}\right)\right],
\end{aligned}\tag{2.16}$$

and we need to evaluate the two expectations given by $\mathbb{E}_{f_j}\left[G_1\left(\lambda_{k+1},\lambda_k;\, f_j,\sigma^2_{\mathrm{A}}\right)\right]$ and $\mathbb{E}_{f_j}\left[f_jG_1\left(\lambda_{k+1},\lambda_k;\, f_j,\sigma^2_{\mathrm{A}}\right)\right]$. We start by considering the first integral

$$\begin{aligned}
&\mathbb{E}_{f_j}\left[G_1\left(\lambda_{k+1},\lambda_k;\, f_j,\sigma^2_{\mathrm{A}}\right)\right]\\
&= \mathbb{E}_{f_j}\left[\Phi\left(\lambda_{k+1};\, f_j,\sigma^2_{\mathrm{A}}\right)-\Phi\left(\lambda_k;\, f_j,\sigma^2_{\mathrm{A}}\right)\right]\\
&= \mathbb{E}_{f_j}\left[\Phi\left(f_j;\,\lambda_{k+1},\sigma^2_{\mathrm{A}}\right)-\Phi\left(f_j;\,\lambda_k,\sigma^2_{\mathrm{A}}\right)\right]\\
&= \mathbb{E}_{f_j}\left[\Phi\left(\frac{f_j-\lambda_{k+1}}{\sigma^2_{\mathrm{A}}}\right)-\Phi\left(\frac{f_j-\lambda_k}{\sigma^2_{\mathrm{A}}}\right)\right]\\
&= \int_{-\infty}^{\infty}\left\{\phi\left(\frac{f_j-\mu_{f_j}}{\sigma^2_{f_j}}\right)\Phi\left(\frac{f_j-\lambda_{k+1}}{\sigma^2_{\mathrm{A}}}\right)-\phi\left(\frac{f_j-\mu_{f_j}}{\sigma^2_{f_j}}\right)\Phi\left(\frac{f_j-\lambda_k}{\sigma^2_{\mathrm{A}}}\right)\right\}df_j.
\end{aligned}\tag{2.17}$$

Now denote $x=\frac{f_j-\mu_{f_j}}{\sigma_{f_j}}$ with $dx=\frac{1}{\sigma_{f_j}}df_j$ and

$$\begin{aligned}
&\mathbb{E}_{f_j}\left[G_1\left(\lambda_{k+1},\lambda_k;\, f_j,\sigma^2_{\mathrm{A}}\right)\right]\\
&= \frac{1}{\sigma^2_{f_j}}\int_{-\infty}^{\infty}\left\{\phi\left(x\right)\Phi\left(\frac{\sigma^2_{f_j}x+\mu_{f_j}-\lambda_{k+1}}{\sigma^2_{\mathrm{A}}}\right)-\phi\left(x\right)\Phi\left(\frac{\sigma^2_{f_j}x+\mu_{f_j}-\lambda_k}{\sigma^2_{\mathrm{A}}}\right)\right\}dx.
\end{aligned}\tag{2.18}$$

Now we can use the identity given by

$$\int_{-\infty}^{\infty}\phi(x)\Phi(a+bx)dx=\Phi\left(\frac{a}{\sqrt{1+b^2}}\right),\tag{2.19}$$

to obtain for the first expectation

$$\begin{aligned}
&\mathbb{E}_{f_j}\left[G_1\left(\lambda_{k+1},\lambda_k;\,f_j,\sigma_{\mathrm{A}}^2\right)\right]\\
&=\frac{1}{\sigma_{f_j}^2}\int_{-\infty}^{\infty}\left\{\phi\left(x\right)\Phi\left(\frac{\sigma_{f_j}x+\mu_{f_j}-\lambda_{k+1}}{\sigma_{\mathrm{A}}^2}\right)-\phi\left(x\right)\Phi\left(\frac{\sigma_{f_j}^2x+\mu_{f_j}-\lambda_k}{\sigma_{\mathrm{A}}^2}\right)\right\}dx\\
&=\frac{1}{\sigma_{f_j}^2}\left[\Phi\left(\frac{\mu_{f_j}-\lambda_{k+1}}{\sqrt{1+\frac{\sigma_{f_j}^2}{\sigma_A^2}}}\right)-\Phi\left(\frac{\mu_{f_j}-\lambda_k}{\sqrt{1+\frac{\sigma_{f_j}^2}{\sigma_A^2}}}\right)\right].
\end{aligned} \tag{2.20}$$

Now we consider the second integral $\mathbb{E}_{f_j}\left[f_jG_1\left(\lambda_{k+1},\lambda_k;\,f_j,\sigma_{\mathrm{A}}^2\right)\right]$ which can be rewritten as

$$\begin{aligned}
&\mathbb{E}_{f_j}\left[f_jG_1\left(\lambda_{k+1},\lambda_k;\,f_j,\sigma_{\mathrm{A}}^2\right)\right]\\
&=\int_{-\infty}^{\infty}f_j\left\{\phi\left(\frac{f_j-\mu_{f_j}}{\sigma_{f_j}^2}\right)\Phi\left(\frac{f_j-\lambda_{k+1}}{\sigma_{\mathrm{A}}^2}\right)-\phi\left(\frac{f_j-\mu_{f_j}}{\sigma_{f_j}^2}\right)\Phi\left(\frac{f_j-\lambda_k}{\sigma_{\mathrm{A}}^2}\right)\right\}df_j.
\end{aligned} \tag{2.21}$$

Now denote $x=\frac{f_j-\mu_{f_j}}{\sigma_{f_j}}$ with $dx=\frac{1}{\sigma_{f_j}}df_j$ and

$$\begin{aligned}
&\mathbb{E}_{f_j}\left[f_jG_1\left(\lambda_{k+1},\lambda_k;\,f_j,\sigma_{\mathrm{A}}^2\right)\right]\\
&=\frac{1}{\sigma_{f_j}^2}\int_{-\infty}^{\infty}\left(\sigma_{f_j}^2x+\mu_{f_j}\right)\left\{\phi\left(x\right)\Phi\left(\frac{\sigma_{f_j}^2x+\mu_{f_j}-\lambda_{k+1}}{\sigma_{\mathrm{A}}^2}\right)-\phi\left(x\right)\Phi\left(\frac{\sigma_{f_j}^2x+\mu_{f_j}-\lambda_k}{\sigma_{\mathrm{A}}^2}\right)\right\}dx\\
&=\int_{-\infty}^{\infty}x\left\{\phi\left(x\right)\Phi\left(\frac{\sigma_{f_j}^2x+\mu_{f_j}-\lambda_{k+1}}{\sigma_{\mathrm{A}}^2}\right)-\phi\left(x\right)\Phi\left(\frac{\sigma_{f_j}^2x+\mu_{f_j}-\lambda_k}{\sigma_{\mathrm{A}}^2}\right)\right\}dx\\
&\quad+\frac{\mu_{f_j}}{\sigma_{f_j}^2}\int_{-\infty}^{\infty}\left\{\phi\left(x\right)\Phi\left(\frac{\sigma_{f_j}^2x+\mu_{f_j}-\lambda_{k+1}}{\sigma_{\mathrm{A}}^2}\right)-\phi\left(x\right)\Phi\left(\frac{\sigma_{f_j}^2x+\mu_{f_j}-\lambda_k}{\sigma_{\mathrm{A}}^2}\right)\right\}dx.
\end{aligned} \tag{2.22}$$

Now we utilize the identity in Eq. 2.19 and the following additional identity

$$\int_{-\infty}^{\infty}x\phi(x)\Phi(a+bx)dx=\frac{b}{\sqrt{1+b^2}}\phi\left(\frac{a}{\sqrt{1+b^2}}\right), \tag{2.23}$$

to obtain the result

$$\begin{aligned}
&\mathbb{E}_{f_j}\left[f_jG_1\left(\lambda_{k+1},\lambda_k;\,f_j,\sigma_{\mathrm{A}}^2\right)\right]\\
&=\left(\frac{\sigma_{f_j}^2+\mu_{f_j}}{\sigma_A^2\sqrt{1+\frac{\sigma_{f_j}^2}{\sigma_A^2}}}\right)\left[\Phi\left(\frac{\mu_{f_j}-\lambda_{k+1}}{\sqrt{1+\frac{\sigma_{f_j}^2}{\sigma_A^2}}}\right)-\Phi\left(\frac{\mu_{f_j}-\lambda_k}{\sqrt{1+\frac{\sigma_{f_j}^2}{\sigma_A^2}}}\right)\right].
\end{aligned} \tag{2.24}$$

Case 3: $\mathbf{x}_i \in \mathcal{X}^D$ **and** $\mathbf{x}_j \in \mathcal{X}^D$
The conditional expectation, $\mathbb{E}_{Y_i,Y_j}\left[Y_i Y_j | f_i, f_j\right]$, can be expressed as:

$$\begin{aligned}
\mathbb{E}_{Y_i,Y_j}\left[Y_i Y_j | f_i, f_j\right] &= \sum_{k=0}^{L-1}\sum_{l=0}^{L-1} kl\mathbb{P}\left(Y_i = k, Y_j = l | f_i, f_j\right) \\
&= \sum_{k=0}^{L-1}\sum_{l=0}^{L-1} kl \sum_{m=0}^{L-1} \mathbb{P}\left(Y_i = k | B_i = m\right) G_1\left(\lambda_{m+1}, \lambda_m; f_i, \sigma_{\mathrm{A}}^2\right) \\
&\qquad \times \sum_{n=0}^{L-1} \mathbb{P}\left(Y_j = l | B_j = n\right) G_1\left(\lambda_{m+1}, \lambda_m; f_j, \sigma_{\mathrm{A}}^2\right).
\end{aligned}$$

Next we derive the unconditional expectation of $\mathbb{E}_{f_i,f_j}\left[Y_i Y_j | f_i, f_j\right]$:

$$\begin{aligned}
\mathbb{E}_{Y_i,Y_j}\left[Y_i Y_j\right] &= \mathbb{E}_{f_i,f_j}\Bigg[\sum_{k=0}^{L-1}\sum_{l=0}^{L-1} kl \sum_{m=0}^{L-1}\left(\mathbb{P}\left(Y_i = k | B_i = m\right) G_1\left(\lambda_{m+1}, \lambda_m; f_i, \sigma_{\mathrm{A}}^2\right)\right) \\
&\qquad \times \sum_{n=0}^{L-1}\left(\mathbb{P}\left(Y_j = l | B_j = n\right) G_1\left(\lambda_{n+1}, \lambda_n; f_j, \sigma_{\mathrm{A}}^2\right)\right)\Bigg] \\
&= \sum_{k=0}^{L-1}\sum_{l=0}^{L-1} kl \sum_{m=0}^{L-1}\sum_{n=0}^{L-1} \mathbb{P}\left(Y_i = k | B_i = m\right) \mathbb{P}\left(Y_j = l | B_j = n\right) \\
&\quad \times \mathbb{E}_{f_i,f_j}\left[G_1\left(\lambda_{m+1}, \lambda_m; f_i, \sigma_{\mathrm{A}}^2\right) G_1\left(\lambda_{n+1}, \lambda_n; f_j, \sigma_{\mathrm{A}}^2\right)\right].
\end{aligned}$$

■

Having derived the spatial cross-correlation between the observations, our next goal is to consider model calibrations. In the context of the spatial WSN models developed this will correspond to addressing the issue of parameter estimation, in particular hyperparameter estimation of the parameters in the covariance kernel functions given the observed data. To achieve this we will consider calibration based on the high-quality sensor information, given in Case 1.

To achieve the model calibration for the kernel parameters in an efficient manner we will develop a special representation of the problem in the form of a regression model through the introduction of an additional auxiliary variable for each observation, i.e., per sensor location. In doing this it will allow us to avoid directly trying to perform maximum likelihood estimation in the models, which can be very difficult, especially when it comes to the matrices of parameters given by each $\boldsymbol{b}_i$ for each analog sensor location. Instead, through the use of auxiliary variables we may write a random effects regression model, which preserves the conditional covariance structure developed above, whilst admitting an efficient estimation procedure comprised of simple expectation and maximization stages of the EM algorithm. In the models considered we will see that the expectation stage is closed form and analytic and the maximization stage is simply a least squares problem after a change of parameterization. Making estimation both guaranteed to converge to a maxima and highly computationally efficient.

2.4.2 Random Effects WSN Spatial Model Reinterpretation

We begin with the scenario in which the majority of the sensor are analog, i.e., the spatial distribution of such high-quality sensors is distributed in some manner over the entire field of interest when performing spatial field reconstruction. These high-quality sensors can be sparse and will still be supplemented by cheaper sensors as discussed above, however, in this stage we will concentrate on the calibration of the model based solely on the high-quality analog sensors.

The advantage of this approach is that we will be able to utilize an interesting result for the estimation of the model parameters which is based on results known for covariance regressions, see [17]. Consider the observation covariance at the analog sensors given in this case by

$$\begin{aligned}
\mathbb{C}\mathrm{ov}\left[\mathbf{Y}_{\mathcal{N}}|\mathbf{w}\right] &= \mathbb{E}_{\mathbf{Y}_{\mathcal{N}}}\left[\mathbf{Y}_{\mathcal{N}}\mathbf{Y}_{\mathcal{N}}^{T}|\mathbf{w}\right] \\
&= \mathbb{E}_{\boldsymbol{h}}\left[\boldsymbol{h}\boldsymbol{h}^{T}\right] + \mathbb{E}_{\boldsymbol{g}}\left[\boldsymbol{g}\boldsymbol{g}^{T}|\mathbf{w}\right] + \mathrm{diag}\left(\sigma_{\mathrm{A}}^{2}, \sigma_{\mathrm{A}}^{2}, \ldots, \sigma_{\mathrm{A}}^{2}\right) \\
&= \begin{bmatrix} \mathcal{C}_h(\mathbf{x}_1,\mathbf{x}_1) & \cdots & \mathcal{C}_h(\mathbf{x}_1,\mathbf{x}_n) \\ \vdots & \ddots & \vdots \\ \mathcal{C}_h(\mathbf{x}_n,\mathbf{x}_1) & \cdots & \mathcal{C}_h(\mathbf{x}_n,\mathbf{x}_n) \end{bmatrix} + \begin{bmatrix} \mathcal{C}_g(\mathbf{x}_1,\mathbf{x}_1) & \cdots & \mathcal{C}_g(\mathbf{x}_1,\mathbf{x}_n) \\ \vdots & \ddots & \vdots \\ \mathcal{C}_g(\mathbf{x}_n,\mathbf{x}_1) & \cdots & \mathcal{C}_g(\mathbf{x}_n,\mathbf{x}_n) \end{bmatrix} + \begin{bmatrix} \sigma_{\mathrm{A}}^{2} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_{\mathrm{A}}^{2} \end{bmatrix} \\
&= K_h + B\mathbf{w}\mathbf{w}^{T}B^{T} + \mathrm{diag}\left(\sigma_{\mathrm{A}}^{2}, \sigma_{\mathrm{A}}^{2}, \ldots, \sigma_{\mathrm{A}}^{2}\right).
\end{aligned} \tag{2.25}$$

We note that under our model formulations, typically we would select ζ_i^2, ζ_j^2 and $\zeta_{i,j}^2$ all to zero, since we already have a baseline covariance function given by the independent spatial GP $h(\cdot)$.

We may now reinterpret the model covariance as a form of covariance regression which admits a representation as a random effects model, making it an extension of the framework proposed in [17]. The random effects representation is given for m realizations of the spatial process, i.e., $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m$ with $\boldsymbol{y}_k = \boldsymbol{y}_{1:N^A,k} = [y_k(\mathbf{x}_1), \ldots, y_k(\mathbf{x}_{N^A})]$ and $\boldsymbol{\mu}_g = \boldsymbol{\mu}_{g,1:N^A} = \left[\mu_g(\mathbf{x}_1), \ldots, \mu_g(\mathbf{x}_{N^A})\right]$ is the spatial mean function of the first baseline spatial GP $g(\cdot)$, for each of the analog sensor locations $\mathbf{x}_i \in \mathcal{X}^A$. This then gives the random effect model given by

$$\boldsymbol{Y}_k = \boldsymbol{\mu}_g + B\mathbf{w}_k\Gamma_k + \boldsymbol{U}_k, \tag{2.26}$$

where one defines

$$\begin{aligned}
&\mathbb{E}\left[\boldsymbol{U}_i\right] = \mathbf{0}, \quad \mathbb{E}\left[\Gamma_i\boldsymbol{U}_i\right] = \mathbf{0}, \quad \mathbb{E}\left[\Gamma_i\right] = 0, \quad \mathbb{V}\mathrm{ar}\left[\Gamma_i\right] = 1, \\
&\mathbb{C}\mathrm{ov}\left[\boldsymbol{U}_i\right] = K_h + \mathrm{diag}\left(\sigma_{\mathrm{A}}^{2}, \sigma_{\mathrm{A}}^{2}, \ldots, \sigma_{\mathrm{A}}^{2}\right).
\end{aligned} \tag{2.27}$$

To see that this random effects formulation of the spatial model indeed produces the correct spatial covariance structure we consider the following:

$$\begin{aligned}&\mathbb{E}\left[\left(\boldsymbol{Y}_k-\boldsymbol{\mu}_g\right)\left(\boldsymbol{Y}_k-\boldsymbol{\mu}_g\right)^T\right]\\&=\mathbb{E}\left[\gamma_k^2 B\mathbf{w}\mathbf{w}^T B^T+\gamma_k\left(B\mathbf{w}\boldsymbol{u}_k^T+\boldsymbol{u}_k\mathbf{w}^T B^T\right)+\boldsymbol{u}_k\boldsymbol{u}_k^T\right]\\&=B\mathbf{w}\mathbf{w}^T B^T+K_h+\operatorname{diag}\left(\sigma_{\mathrm{A}}^2,\sigma_{\mathrm{A}}^2,\ldots,\sigma_{\mathrm{A}}^2\right)\end{aligned}\tag{2.28}$$

2.4.3 Random Effects WSN Spatial Model Estimation via EM Algorithm

We can now perform the estimation of the spatial field using the information from the high-quality analog sensors to make estimation via an EM algorithm using the reinterpreted random effects model form from Sect. 2.4.2. To achieve this, we make the following additional statistical assumptions regarding the random effects reinterpretation, in particular we assume that the regression errors are Guassian random vectors, independent of the Gaussian random variables for the random effect:

$$\begin{aligned}&\boldsymbol{u}_k\overset{iid}{\sim}N(\mathbf{0},A),\forall k\in\{1,\ldots,m\}\\&\Gamma_k\overset{iid}{\sim}N(0,1),\end{aligned}\tag{2.29}$$

with $A:=K_h+\operatorname{diag}\left(\sigma_{\mathrm{A}}^2,\sigma_{\mathrm{A}}^2,\ldots,\sigma_{\mathrm{A}}^2\right)$.

The resulting log-likelihood of the random effects model can be rewritten by subtracting the mean from the observations to obtain the matrix of mean adjusted residuals, given by $E=\left(\boldsymbol{e}_1^T,\ldots,\boldsymbol{e}_m^T\right)^T$, with residual vectors for the kth spatial map observation given by $\boldsymbol{e}_k=\left[\boldsymbol{Y}_k-\widehat{\boldsymbol{\mu}}_g\right]$. This results in the following log-likelihood for the model parameter matrices A and B, given the observation matrix of residuals E and covariate matrix W from the other sensed modalities, producing for a constant c the log-likelihood:

$$\begin{aligned}l(A,B;E,W)=c&-\frac{1}{2}\sum_{k=1}^{m}\log\left|A+B\mathbf{w}_k\mathbf{w}_k^T B^T\right|\\&-\frac{1}{2}\sum_{k=1}^{m}\operatorname{tr}\left[\left(A+B\mathbf{w}_k\mathbf{w}_k^T B^T\right)^{-1}\boldsymbol{e}_k\boldsymbol{e}_k^T\right]\end{aligned}\tag{2.30}$$

It is clear that direct maximization of this log-likelihood with respect to the matrices A and B will be a very challenging non-convex optimization problem. This arises since the matrix A must be optimized with respect to constraints that ensure that it remains symmetric and positive definite in order for it to be a well-defined covariance matrix.

Therefore, instead of attempting this difficult direct likelihood-based inference, we will adopt an alternative two-stage expectation maximization (EM) algorithm-based approach. The EM algorithm developed will be even more efficient and numerically robust, since both the expectation and maximization stages will be obtainable in closed form. In addition, we can be sure that such a procedure will find an optimum.

The ability to obtain a closed form expression for the expectation stage of the EM algorithm arises from the structure of the random effects model specified and the distributional assumptions made. One can show the following result given in Lemma 2.1 for the conditional distribution of the auxiliary random effects variable, conditional on the observations and covariates (other sensed modalities at each analog sensor location). Deriving this conditional distribution is important for the expectation step of the EM algorithm.

Lemma 2.1 (Conditional Distribution of the Random Effects) *The conditional distribution of the random effects given the data and covariates according to*

$$\left[\Gamma_k | \boldsymbol{y}_1, \ldots, \boldsymbol{y}_m, \mathbf{w}_1, \ldots, \mathbf{w}_m, A, B\right] \sim N\left(m_i, v_i\right) \tag{2.31}$$

with $\mathbf{w}_i \in \mathbb{R}^{(q\times 1)}$ *and*

$$\begin{aligned} v_i &= \left(1 + \mathbf{w}_i^T B^T A^{-1} B \mathbf{w}_i\right)^{-1} \in \mathbb{R}^+, \\ m_i &= v_i \left(\boldsymbol{y}_i - \boldsymbol{mu}\right)^T A^{-1} B \mathbf{w}_i \in \mathbb{R}. \end{aligned}$$

Proof The derivation of this conditional distribution for the random effect follows trivially from the standard multivariate Gaussian properties since the joint distribution for the N^A auxiliary variables and observations is multivariate Guassian:

$$p\left(\gamma_1, \ldots, \gamma_m, \boldsymbol{y}_1, \ldots, \boldsymbol{y}_m \middle| \mathbf{w}_1, \ldots, \mathbf{w}_m, A, B\right) = N\left(\boldsymbol{m}, C\right) \tag{2.32}$$

with $\boldsymbol{m} = [\boldsymbol{m}_1, \boldsymbol{m}_2]$ where $\boldsymbol{m}_1$ is a vector of m zeros and $\boldsymbol{m}_2$ a $1 \times mN^A$ vector given by $\boldsymbol{m}_2 = [\boldsymbol{\mu}_g, \ldots, \boldsymbol{\mu}_g]$; and $C = \oplus_{i=1}^2 C_i$ where C_1 is a $m \times m$ matrix $C_1 = \text{diag}(1, \ldots, 1)$ and $C_2 = \oplus_{j=1}^m A$. Then one can use the following properties of a multivariate normal to obtain the conditional distribution, where if μ and Σ are the mean and covariance of a Guassian random vector, which is partitioned as follows:

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix} \tag{2.33}$$

with sizes $q \times 1$ and $(N - q) \times 1$ and

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix} \tag{2.34}$$

with sizes $q \times q, q \times (N - q), (N - q) \times q$ and $(N - q) \times (N - q)$ then, the distribution of x_1 conditional on $x_2 = a$ is multivariate normal $(x1|x2 = a) \sim N(\mu, \Sigma)$

where

$$\bar{\boldsymbol{\mu}} = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\mathbf{a} - \boldsymbol{\mu}_2) \tag{2.35}$$

and covariance matrix

$$\overline{\boldsymbol{\Sigma}} = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}. \tag{2.36}$$

This decomposition completes the required proof. ■

It will be assumed for now that the mean process is already estimated and given by $\widehat{\boldsymbol{\mu}}_g$. In this section we discuss the more challenging aspect of estimation of the hyperparameters that make up the specifications of the covariance functions for process $g(\cdot)$ and $h(\cdot)$. This is achieved by the EM algorithm as follows:

We first write the complete data log-likelihood $\ln p\,(E|A, B, W, \gamma_{1:m})$ with respect to the matrix of $p \times m$ residuals E and random effects $\gamma_1, \ldots, \gamma_m$ as follows:

$$l(A, B; E, W, \gamma_{1:m}) = -\frac{1}{2}\left(mp\ln(2\pi) + m\ln|A| + \sum_{k=1}^{m}(e_i - \gamma_i B\boldsymbol{w}_i)^T A^{-1}(e_i - \gamma_i B\boldsymbol{w}_i).\right) \tag{2.37}$$

Then from the complete data likelihood we consider the expectation step with respect to the random effect (nuisance parameters) as obtained in Lemma 2.2.

Lemma 2.2 (Integrated Complete Data Likelihood) *The following conditional expectation of the complete data likelihood with respect to the conditional distribution of the random effects nuisance parameters is obtained:*

$$\begin{aligned} &-2\mathbb{E}_{\gamma_{1:m}}\left[l(A, B; E, W, \gamma_{1:m})|\widehat{A}, \widehat{B}\right] \\ &\quad = mp\ln(2\pi) + m\ln|A| + \sum_{k=1}^{m}(e_i - \widehat{m}_i B\boldsymbol{w}_i)^T A^{-1}(e_i - \widehat{m}_i B\boldsymbol{w}_i) \\ &\quad\quad + \sum_{k=1}^{m}\widehat{s}_i\boldsymbol{w}_i^T B^T A^{-1}B\boldsymbol{w}_i\widehat{s}_i, \end{aligned} \tag{2.38}$$

with $s_i = \sqrt{v_i}$ *and*

$$\begin{aligned} \widehat{v}_i &= \left(1 + \mathbf{w}_i^T\widehat{B}^T\widehat{A}^{-1}\widehat{B}\mathbf{w}_i\right)^{-1}, \\ \widehat{m}_i &= \widehat{v}_i\left(\boldsymbol{y}_i - \boldsymbol{mu}\right)^T\widehat{A}^{-1}\widehat{B}\mathbf{w}_i. \end{aligned}$$

Proof Here the previous estimates for the target model parameters, denoted $\widehat{A}$, $\widehat{B}$ are conditioned upon in the expectation in the sense that they are used to calculate the sufficient statistics for the distribution of the random effects $\gamma_{1:m}$ given by

$$\begin{aligned} \widehat{v}_i &= \left(1 + \mathbf{w}_i^T\widehat{B}^T\widehat{A}^{-1}\widehat{B}\mathbf{w}_i\right)^{-1} \\ \widehat{m}_i &= \widehat{v}_i\left(\boldsymbol{y}_i - \boldsymbol{mu}\right)^T\widehat{A}^{-1}\widehat{B}\mathbf{w}_i. \end{aligned}$$

One takes the conditional expectations of the complete data likelihood as follows:

$$\begin{aligned}
&-2\mathbb{E}_{\gamma_{1:m}}\left[l(A, B; E, W, \gamma_{1:m})|\widehat{A}, \widehat{B}\right] \\
&= mp\ln(2\pi) + m\ln|A| + \sum_{k=1}^{m}\mathbb{E}_{\gamma_{1:m}}\left[(e_i - \gamma_i B\boldsymbol{w}_i)^T A^{-1}(e_i - \gamma_i B\boldsymbol{w}_i)|\widehat{A}, \widehat{B}\right]
\end{aligned} \tag{2.39}$$

Next observe that γ_i's are i.i.d. hence we can consider the individual expectations

$$\begin{aligned}
&\mathbb{E}_{\gamma_i}\left[(e_i - \gamma_i B\boldsymbol{w}_i)^T A^{-1}(e_i - \gamma_i B\boldsymbol{w}_i)|\widehat{A}, \widehat{B}\right] \\
&= \mathbb{E}_{\gamma_i}\left[e_i^T A^{-1}e_i - \gamma_i^2\boldsymbol{w}_i^T B^T A^{-1}B\boldsymbol{w}_i|\widehat{A}, \widehat{B}\right] \\
&= \mathbb{E}_{\gamma_i}\left[e_i^T A^{-1}e_i|\widehat{A}, \widehat{B}\right] - \mathbb{E}_{\gamma_i}\left[\gamma_i^2\boldsymbol{w}_i^T B^T A^{-1}B\boldsymbol{w}_i|\widehat{A}, \widehat{B}\right] \\
&= (e_i - \widehat{m}_i B\boldsymbol{w}_i)^T A^{-1}(e_i - \widehat{m}_i B\boldsymbol{w}_i) + \widehat{s}_i\boldsymbol{w}_i^T B^T A^{-1}B\boldsymbol{w}_i\widehat{s}_i.
\end{aligned} \tag{2.40}$$

Then one simply rewrites the expression using this mean and variance expressions to complete the proof. ■

Having obtained a closed form expression for the expectation step, we next need to obtain the maximization step of the EM algorithm which involves the maximization of

$$\begin{aligned}
&\arg\min_{A,B} -2\mathbb{E}_{\gamma_{1:m}}\left[l(A, B; E, W, \gamma_{1:m})|\widehat{A}, \widehat{B}\right] \\
&= \arg\min_{A,B}\left\{mp\ln(2\pi) + m\ln|A| + \sum_{k=1}^{m}(e_i - \widehat{m}_i B\boldsymbol{w}_i)^T A^{-1}(e_i - \widehat{m}_i B\boldsymbol{w}_i)\right. \\
&\left. + \sum_{k=1}^{m}\widehat{s}_i\boldsymbol{w}_i^T B^T A^{-1}B\boldsymbol{w}_i\widehat{s}_i\right\}.
\end{aligned} \tag{2.41}$$

Finally, one observes that this maximization can be easily implemented through a least squares solution by rewriting the argument in the form of a single quadratic with a change of representation given by constructing:

- $\widetilde{W}$ as a $2m \times q$ matrix with ith row given by $m_i\boldsymbol{w}_i$ and whose $(n+i)$th row is given by $s_i\boldsymbol{w}_i$;
- $\widetilde{E}$ as a $2m \times p$ matrix of residuals given by $[E^T, \mathbf{0}]$ with the matrix of $\mathbf{0}$ the same dimension as matrix E, i.e., $m \times p$.

This produces the new argument for the optimization as follows:

$$\begin{aligned}
&\arg\min_{A,B} -2\mathbb{E}_{\gamma_{1:m}}\left[l(A, B; E, W, \gamma_{1:m})|\widehat{A}, \widehat{B}\right] \\
&= \arg\min_{A,B}\left\{mp\ln(2\pi) + m\ln|A| + \operatorname{tr}\left[(\widetilde{E} - B\widetilde{W})(\widetilde{E} - B\widetilde{W})^T A^{-1}\right]\right\}.
\end{aligned} \tag{2.42}$$

Rewriting the problem in this manner makes it appear directly as a least squares optimization problem which admits a solution given by:

$$\begin{aligned} \widehat{B} &= \widetilde{E}^T \widetilde{W} \left(\widetilde{W}^T \widetilde{W} \right)^{-1}, \\ \widehat{A} &= \frac{1}{n} \left(\widetilde{E} - \widetilde{W} \widehat{B} \right)^T \left(\widetilde{E} - \widetilde{W} \widehat{B} \right) \end{aligned} \tag{2.43}$$

The EM algorithm proceeds as follows:

- Initialize the parameters making matrices $\widehat{A}$ and $\widehat{B}$, where A is comprised of kernel hyperparameters and noise variance terms.
- Calculate the conditional estimators:

$$\begin{aligned} m_i &= \mathbb{E} \left[\Gamma_i | \, \widehat{A}, \widehat{B}, \boldsymbol{e}_i \right] \\ v_i &= \mathbb{V}\mathrm{ar} \left[\Gamma_i | \, \widehat{A}, \widehat{B}, \boldsymbol{e}_i \right] \end{aligned} \tag{2.44}$$

- Construct new matrices $\widetilde{W}$ and $\widetilde{E}$ based on the data $\boldsymbol{y}_{1:m}$ and covariates $\mathbf{w}_{1:m}$.
- Evaluate the updated model parameters via the following least squares solutions for updated $\widehat{A}$ and $\widehat{B}$ according to

$$\begin{aligned} \widehat{B} &= \widetilde{E}^T \widetilde{W} \left(\widetilde{W}^T \widetilde{W} \right)^{-1} \\ \widehat{A} &= \frac{1}{n} \left(\widetilde{E} - \widetilde{W} \widehat{B} \right)^T \left(\widetilde{E} - \widetilde{W} \widehat{B} \right) \end{aligned} \tag{2.45}$$

  where matrix $\widetilde{E}$ is the $2m \times 1$ matrix given by $\left(E^T, 0 \times E^T \right)^T$ and $\widetilde{W}$ is a $2m \times d$ matrix with ith row given by $m_i \mathbf{w}_i$ and whose $(m+i)$th is $\sqrt{v_i} \mathbf{w}_i$.
- Having solved for the matrix $\widehat{A}$, one then solves the system of equations given by

$$\begin{bmatrix} \widehat{A}_{11} & \cdots & \widehat{A}_{1N^A} \\ \vdots & \ddots & \vdots \\ \widehat{A}_{N^A 1} & \cdots & \widehat{A}_{N^A N^A} \end{bmatrix} = \begin{bmatrix} \mathcal{C}_h(\mathbf{x}_1, \mathbf{x}_1) & \cdots & \mathcal{C}_h(\mathbf{x}_1, \mathbf{x}_n) \\ \vdots & \ddots & \vdots \\ \mathcal{C}_h(\mathbf{x}_n, \mathbf{x}_1) & \cdots & \mathcal{C}_h(\mathbf{x}_n, \mathbf{x}_n) \end{bmatrix} + \begin{bmatrix} \sigma_{\mathrm{A}}^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_{\mathrm{A}}^2 \end{bmatrix} \tag{2.46}$$

  for the variance and hyperparameter terms in the kernels.
- repeat the above procedure until convergence

Having outlined an estimation procedure, the remainder of the chapter focuses on what can be done for spatial field reconstruction given an estimated model.

2.5 Spatial Field Reconstruction: Analytic Solutions

In this section we address the estimation problem known as *spatial field reconstruction* in the case of either a homogeneous WSN sensor model with all sensors performing an L-bit quantization, which was recently studied in [29, 30], and then in the second case based on a heterogeneous WSN framework with a mixture of analog and binary sensors. In general the target criterion in developing the spatial estimator of the field reconstruction is achieve the minimum mean squared error (MMSE) in the estimation. This involved the following distortion metric:

$$D\left(\widehat{f_*}, f_*\right) := \mathbb{E}\left[\left(f_* - \widehat{f_*}\right)^2\right]. \tag{2.47}$$

Under this framework, one may develop two closed form approximations for the estimators of the spatial field, in [27, 30] approximate series expansions based on saddle point and Laplace types were developed for nonlinear estimators which are optimal in the sense of minimizing the distortion metric in (2.47) [6].

In this chapter we wish to emphasize a computationally very efficient alternative class of estimators that we denote the spatial best linear unbiased estimators (S-BLUE) linear Bayes estimators. Such estimators are characterized by the following formal estimation objective (Objective 1):

Objective 1: spatial field reconstruction via best linear unbiased (S-BLUE) estimate, given by the solution to the following problem:

$$\widehat{f_*} := \widehat{a} + \widehat{\mathbf{B}}\mathbf{Y}_{\mathcal{N}} = \arg\min_{a,\mathbf{B}} \mathbb{E}\left[(f_* - (a + \mathbf{B}\mathbf{Y}_{\mathcal{N}}))^2\right], \tag{2.48}$$

where $\widehat{a} \in \mathbb{R}$ and $\widehat{\mathbf{B}} \in \mathbb{R}^{1\times N}$.

The S-BLUE estimators are optimal in the sense that it achieves minimum variance among all linear estimators and have the desirable properties of being unbiased and efficient.

Theorem 2.2 (Spatial Best Linear Unbiased Estimator (S-BLUE)) *The optimal linear estimator of the spatial field $\widehat{f_*}$ at location $\mathbf{x}_*$ in the class of all linear estimators taking the form $\widehat{f_*} = a + \mathbf{B}\mathbf{Y}_{\mathcal{N}}$ for some scalar $a \in \mathbb{R}$, vector $\mathbf{B} \in \mathbb{R}^{1\times N}$ at a location $\mathbf{x}_*$ that solves (2.48) is given by*

$$\widehat{f_*} = \mu\left(\mathbf{x}_*\right) + \mathbb{E}_{f_*,\mathbf{Y}_{\mathcal{N}}}\left[f_*\mathbf{Y}_{\mathcal{N}}^T\right]\mathbb{E}_{\mathbf{Y}_{\mathcal{N}}}^{-1}\left[\mathbf{Y}_{\mathcal{N}}\mathbf{Y}_{\mathcal{N}}^T\right]\left(\mathbf{Y}_{\mathcal{N}} - \mathbb{E}_{\mathbf{Y}_{\mathcal{N}}}\left[\mathbf{Y}_{\mathcal{N}}\right]\right). \tag{2.49}$$

In addition, one may derive the estimation accuracy of the S-BLUE in closed form according to the result in Corollary [27, 30].

Corollary 2.1 *The associated MSE of the S-BLUE is given by*

$$\sigma_*^2 = \mathcal{C}\left(\mathbf{x}_*, \mathbf{x}_*\right) - \mathbb{E}_{f_*,\mathbf{Y}_\mathcal{N}}\left[f_*\mathbf{Y}_\mathcal{N}^T\right]\mathbb{E}_{\mathbf{Y}_\mathcal{N}}^{-1}\left[\mathbf{Y}_\mathcal{N}\mathbf{Y}_\mathcal{N}^T\right]\mathbb{E}_{f_*,\mathbf{Y}_\mathcal{N}}\left[\mathbf{Y}_\mathcal{N} f_*\right]. \tag{2.50}$$

The following sequence of algorithmic steps are then required to perform estimation of the S-BLUE, see Algorithm 1.

Algorithm 1 S-BLUE Field reconstruction

```
Input: $\mathbf{Y}_\mathcal{N}, \mathbf{x}_\mathcal{N}, \mathbf{x}_*, \sigma_\mathrm{A}^2, \sigma^2, \mu(\cdot)$
Output: $\widehat{f_*}$
1: Calculate the cross-correlation vector, $\mathbb{E}_{f_*,\mathbf{Y}_\mathcal{N}}\left[f_*\mathbf{Y}_\mathcal{N}^T\right]$, where its $i$th element is implemented according to Lemma 2.3.
2: Calculate the covariance matrix, $\mathbb{E}_{\mathbf{Y}_\mathcal{N}}\left[\mathbf{Y}_\mathcal{N}\mathbf{Y}_\mathcal{N}^T\right]$, where its $(i, j)$th element is implemented according to Proposition 2.1 and the Clenshaw–Curtis coefficients in (2.57).
3: Calculate the S-BLUE of the intensity of the spatial field at a location $\mathbf{x}_*$ as follows:
   $\widehat{f_*} = \mu\left(\mathbf{x}_*\right) + \mathbb{E}_{f_*,\mathbf{Y}_\mathcal{N}}\left[f_*\mathbf{Y}_\mathcal{N}^T\right]\mathbb{E}_{\mathbf{Y}_\mathcal{N}}^{-1}\left[\mathbf{Y}_\mathcal{N}\mathbf{Y}_\mathcal{N}^T\right]\left(\mathbf{Y}_\mathcal{N} - \mathbb{E}_{\mathbf{Y}_\mathcal{N}}\left[\mathbf{Y}_\mathcal{N}\right]\right).$
```

The key components of the S-BLUE estimator that must be obtained for any form of WSN design involve the following components:

- the cross-correlation $\mathbb{E}_{f_*,\mathbf{Y}_\mathcal{N}}\left[f_*\mathbf{Y}_\mathcal{N}^T\right]$; and
- the covariance $\mathbb{E}_{\mathbf{Y}_\mathcal{N}}\left[\mathbf{Y}_\mathcal{N}\mathbf{Y}_\mathcal{N}^T\right]$, (detailed in Lemma 2.1).

In the following set of results we will derive these quantities and subsequently the S-BLUE estimators for two classes of WSN: the L-bit homogeneous quantized/digitized WSN; and the heterogeneous L-bit digital/quantized and analog WSN. We begin with a detailed account of the result for the heterogeneous case.

Note: the covariance matrix for case 1 is derived in Lemma 2.1, however, for case 2 and case 3 we provide in Sect. 2.5.1.2 an accurate and efficient approximation for the expectations.

2.5.1 S-BLUE Spatial Field Estimator for Heterogeneous L-bit and Analog WSNs

In this section we consider the development of the S-BLUE class of spatial field reconstruction estimator to the Heterogeneous WSN setting in which we incorporate also additional sensed modalities, included as regressors into the spatial covariance structure through the kernel developed in (2.8) for the analog sensors.

2.5.1.1 Deriving the Cross-Correlation $\mathbb{E}_{f_*,\mathbf{Y}_\mathcal{N}}\left[f_*\mathbf{Y}_\mathcal{N}^T\right]$

We now derive the cross-correlation between the spatial phenomenon predictive response, f_* at $\mathbf{x}_*$, and the observation vector $\mathbf{Y}_\mathcal{N}$. We prove that this quantity can be obtained exactly in closed form in the following Lemma 2.3.

Lemma 2.3 (Cross-Correlation between Spatial Process and Observations) *The ith element of $\mathbb{E}_{f_*,\mathbf{Y}_\mathcal{N}}\left[f_*\mathbf{Y}_\mathcal{N}^T\right]$ is given by one of two cases.*

Case 1 *- $\mathbf{x}_i \in \mathcal{X}^D$: where one has Cross-Correlation terms given by*

$$\begin{aligned}
&\mathbb{E}_{f_*,Y_i}\left[f_*Y_i\right]\\
&= c_1 \sum_{l=0}^{L-1} l \sum_{j=0}^{L-1} \mathbb{P}\left(Y_i = l | B_i = j\right) G_1\left(\lambda_{j+1}, \lambda_j; \mu\left(\mathbf{x}_i\right), \sigma_A^2 + \mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_i\right)\right)\\
&\quad + c_2 \sum_{l=0}^{L-1} l \sum_{j=0}^{L-1} \mathbb{P}\left(Y_i = l | B_i = j\right) \Big\{\mu\left(\mathbf{x}_i\right) G_1\left(\lambda_{j+1}, \lambda_j; \mu\left(\mathbf{x}_i\right), \sigma_A^2 + \mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_i\right)\right)\\
&\qquad\qquad - \mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_i\right) G_2\left(\lambda_{j+1}, \lambda_j; \mu\left(\mathbf{x}_i\right), \sigma_A^2 + \mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_i\right)\right)\Big\}.
\end{aligned}$$

where $c_2 := \frac{\mathcal{C}(\mathbf{x}_*,\mathbf{x}_i)}{\mathcal{C}(\mathbf{x}_i,\mathbf{x}_i)}$, $c_1 := \mu\left(\mathbf{x}_*\right) - c_2\mu\left(\mathbf{x}_i\right)$ and

$$\begin{aligned}
G_1\left(a, b; m, s\right) &= \left\{\Phi\left(a; m, s\right) - \Phi\left(b; m, s\right)\right\}\\
G_2\left(a, b; m, s\right) &= \left\{\phi\left(a; m, s\right) - \phi\left(b; m, s\right)\right\}.
\end{aligned}$$

Case 2 *- $\mathbf{x}_i \in \mathcal{X}^A$: where one has Cross-Correlation terms given by*

$$\mathbb{E}_{f_*,Y_i}\left[f_*Y_i\right] = c_1\mu\left(\mathbf{x}_i\right) + c_2\left[\mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_i\right) - \mu\left(\mathbf{x}_i\right)^2\right],$$

Proof To make the proof we consider the ith term of $\mathbb{E}_{f_*,\mathbf{Y}_\mathcal{N}}\left[f_*\mathbf{Y}_\mathcal{N}^T\right]$ which has its expectation decomposed via the tower property as follows:

$$\mathbb{E}_{f_*,Y_i}\left[f_*Y_i\right] = \mathbb{E}_{f_i}\left[\mathbb{E}_{f_*,Y_i}\left[f_*Y_i | f_i\right]\right]. \tag{2.51}$$

We then consider for each of the possible cases, i.e., Case 1 $\mathbf{x}_i \in \mathcal{X}^D$ and Case 2 $\mathbf{x}_i \in \mathcal{X}^A$, the analytic calculation of this cross-correlation. It will be useful to first make the following definitions used throughout the proof:

$$\begin{aligned}
c_2 &:= \frac{\mathcal{C}\left(\mathbf{x}_*, \mathbf{x}_i\right)}{\mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_i\right)},\\
c_1 &:= \mu\left(\mathbf{x}_*\right) - c_2\mu\left(\mathbf{x}_i\right),\\
G_1\left(a, b; m, s\right) &:= \left\{\Phi\left(a; m, s\right) - \Phi\left(b; m, s\right)\right\}\\
G_2\left(a, b; m, s\right) &:= \left\{\phi\left(a; m, s\right) - \phi\left(b; m, s\right)\right\}.
\end{aligned}$$

Case 1:
The conditional expectation, $\mathbb{E}_{f_*,Y_i}\left[f_* Y_i | f_i\right]$, can be expressed as:

$$\begin{aligned}\mathbb{E}_{f_*,Y_i}\left[f_* Y_i | f_i\right] &= \int_{-\infty}^{\infty}\sum_{l=0}^{L-1} f_* l p\left(f_*, Y_i = l | f_i\right)\mathrm{d}f_* \\ &= \int_{-\infty}^{\infty} f_*\phi\left(f_*; c_1 + c_2 f_i, \sigma^2_{\mathbf{f}_{\mathcal{N}}|\mathbf{Y}_{\mathcal{N}}}\right)\sum_{l=0}^{L-1} l \sum_{j=0}^{L-1}\left(\mathbb{P}\left(Y_i = l | B_i = j\right)\mathbb{P}\left(B_i = j | f_i\right)\right)\mathrm{d}f_* \\ &= \left(c_1 + c_2 f_i\right)\sum_{l=0}^{L-1} l \sum_{j=0}^{L-1}\mathbb{P}\left(Y_i = l | B_i = j\right) G_1\left(\lambda_{j+1}, \lambda_j; f_i, \sigma^2_{\mathrm{A}}\right),\end{aligned}$$

The expectation with respect to f_i of the first term is given by

$$\begin{aligned}&\mathbb{E}_{f_i}\left[c_1\sum_{l=0}^{L-1} l \sum_{j=0}^{L-1}\left(\mathbb{P}\left(Y_i = l | B_i = j\right) G_1\left(\lambda_{j+1}, \lambda_j; f_i, \sigma^2_{\mathrm{A}}\right)\right)\right] \\ &= c_1\sum_{l=0}^{L-1} l \sum_{j=0}^{L-1}\left(\mathbb{P}\left(Y_i = l | B_i = j\right)\mathbb{E}_{f_i}\left[G_1\left(\lambda_{j+1}, \lambda_j; f_i, \sigma^2_{\mathrm{A}}\right)\right]\right) \qquad (2.52) \\ &= c_1\sum_{l=0}^{L-1} l \sum_{j=0}^{L-1}\left(\mathbb{P}\left(Y_i = l | B_i = j\right) G_1\left(\lambda_{j+1}, \lambda_j; \mu\left(\mathbf{x}_i\right), \sigma^2_{\mathrm{A}} + \mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_i\right)\right)\right).\end{aligned}$$

The expectation of the second term is given by:

$$\begin{aligned}&\mathbb{E}_{f_i}\left[c_2 f_i\sum_{l=0}^{L-1} l \sum_{j=0}^{L-1}\left(\mathbb{P}\left(Y_i = l | B_i = j\right) G_1\left(\lambda_{j+1}, \lambda_j; f_i, \sigma^2_{\mathrm{A}}\right)\right)\right] \\ &= c_2\sum_{l=0}^{L-1} l \sum_{j=0}^{L-1}\left(\mathbb{P}\left(Y_i = l | B_i = j\right)\mathbb{E}_{f_i}\left[\left(f_i\Phi\left(\lambda_{j+1}, f_i, \sigma^2_{\mathrm{A}}\right) - f_i\Phi\left(\lambda_j, f_i, \sigma^2_{\mathrm{A}}\right)\right)\right)\right] \\ &= c_2\sum_{l=0}^{L-1} l \sum_{j=0}^{L-1}\Bigg(\mathbb{P}\left(Y_i = l | B_i = j\right)\Bigg(\int_{-\infty}^{\infty}\int_{-\infty}^{\lambda_j} f_i\phi\left(a; f_i, \sigma^2_{\mathrm{A}}\right)\phi\left(f_i; \mu\left(\mathbf{x}_i\right), \mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_i\right)\right)\mathrm{d}a\mathrm{d}f_i \\ &\quad - \int_{-\infty}^{\infty}\int_{-\infty}^{\lambda_j} f_i\phi\left(a; f_i, \sigma^2_{\mathrm{A}}\right)\phi\left(f_i; \mu\left(\mathbf{x}_i\right), \mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_i\right)\right)\mathrm{d}a\mathrm{d}f_i\Bigg)\Bigg) \qquad (2.53) \\ &= c_2\sum_{l=0}^{L-1} l \sum_{j=0}^{L-1}\mathbb{P}\left(Y_i = l | B_i = j\right) \\ &\quad\times\Big\{\mu\left(\mathbf{x}_i\right)\Phi\left(\lambda_{j+1}, \mu\left(\mathbf{x}_i\right), \sigma^2_{\mathrm{A}} + \mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_i\right)\right) - \mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_i\right)\phi\left(\lambda_{j+1}, \mu\left(\mathbf{x}_i\right), \sigma^2_{\mathrm{A}} + \mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_i\right)\right) \\ &\quad - \left(\mu\left(\mathbf{x}_i\right)\Phi\left(\lambda_j, \mu\left(\mathbf{x}_i\right), \sigma^2_{\mathrm{A}} + \mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_i\right)\right) - \mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_i\right)\phi\left(\lambda_j, \mu\left(\mathbf{x}_i\right), \sigma^2_{\mathrm{A}} + \mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_i\right)\right)\right)\Big\}.\end{aligned}$$

Combining (2.52) and (2.53), we obtain that the ith term of $\mathbb{E}_{f_*,\mathbf{Y}_\mathcal{N}}\left[f_*\mathbf{Y}_\mathcal{N}^T\right]$ is expressed as:

$$\begin{aligned}\mathbb{E}_{f_*,Y_i}\left[f_*Y_i\right] &= c_1\sum_{l=0}^{L-1} l \sum_{j=0}^{L-1}\mathbb{P}\left(Y_i=l|B_i=j\right)G_1\left(\lambda_{j+1},\lambda_j;\mu\left(\mathbf{x}_i\right),\sigma_\mathrm{A}^2+\mathcal{C}\left(\mathbf{x}_i,\mathbf{x}_i\right)\right)\\ &+c_2\sum_{l=0}^{L-1} l \sum_{j=0}^{L-1}\mathbb{P}\left(Y_i=l|B_i=j\right)\Big\{\mu\left(\mathbf{x}_i\right)G_1\left(\lambda_{j+1},\lambda_j;\mu\left(\mathbf{x}_i\right),\sigma_\mathrm{A}^2+\mathcal{C}\left(\mathbf{x}_i,\mathbf{x}_i\right)\right)\\ &\quad-\mathcal{C}\left(\mathbf{x}_i,\mathbf{x}_i\right)G_2\left(\lambda_{j+1},\lambda_j;\mu\left(\mathbf{x}_i\right),\sigma_\mathrm{A}^2+\mathcal{C}\left(\mathbf{x}_i,\mathbf{x}_i\right)\right)\Big\}.\end{aligned}$$

Case 2:
The conditional expectation, $\mathbb{E}_{f_*,Y_i}\left[f_*Y_i|f_i\right]$, can be expressed as:

$$\begin{aligned}\mathbb{E}_{f_*,Y_i}\left[f_*Y_i|f_i\right] &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_*y_i p_{Y_i|f_i}\left(y_i|f_i\right)p_{f_*|f_i}\left(f_*|f_i\right)\mathrm{d}y_i\mathrm{d}f_*\\ &= f_i\left(c_1+c_2f_i\right),\end{aligned}$$

The expectation with respect to f_i is then given by

$$\mathbb{E}_{f_i}\left[f_i\left(c_1+c_2f_i\right)\right]=c_1\mu\left(\mathbf{x}_i\right)+c_2\left[\mathcal{C}\left(\mathbf{x}_i,\mathbf{x}_i\right)-\mu\left(\mathbf{x}_i\right)^2\right] \tag{2.54}$$

Hence in Case 2 one obtains that the ith term of $\mathbb{E}_{f_*,\mathbf{Y}_\mathcal{N}}\left[f_*\mathbf{Y}_\mathcal{N}^T\right]$ is expressed as:

$$\mathbb{E}_{f_*,Y_i}\left[f_*Y_i\right]=c_1\mu\left(\mathbf{x}_i\right)+c_2\left[\mathcal{C}\left(\mathbf{x}_i,\mathbf{x}_i\right)-\mu\left(\mathbf{x}_i\right)^2\right]$$

■

2.5.1.2 Deriving the Covariance Matrix $\mathbb{E}_{\mathbf{Y}_\mathcal{N}}\left[\mathbf{Y}_\mathcal{N}\mathbf{Y}_\mathcal{N}^T\right]$ Estimators

We have already derived the covariance matrix, $\mathbb{E}_{\mathbf{Y}_\mathcal{N}}\left[\mathbf{Y}_\mathcal{N}\mathbf{Y}_\mathcal{N}^T\right]$ completely in case one and case two, what remains is the expectations in case three. Recall, Case 1 involved $\mathbf{x}_i\in\mathcal{X}^A$ and $\mathbf{x}_j\in\mathcal{X}^A$, i.e., both sensors are high-quality analog sensors; Case 2 with $\mathbf{x}_i\in\mathcal{X}^A$ and $\mathbf{x}_j\in\mathcal{X}^D$, i.e., one sensor is analog and one sensor is a cheaper quantized sensor; and Case 3 in which $\mathbf{x}_i\in\mathcal{X}^D$ and $\mathbf{x}_j\in\mathcal{X}^D$. These were given in Case 3 up to an expectation which would need to be approximated. We briefly explain in this section an efficient manner to perform such approximation using a form of quadrature.

Case 3: $\mathbf{x}_i \in \mathcal{X}^D$ **and** $\mathbf{x}_j \in \mathcal{X}^D$

$$\begin{aligned}\mathbb{E}_{Y_i,Y_j}\left[Y_i Y_j\right] &= \sum_{k=0}^{L-1}\sum_{l=0}^{L-1} kl \sum_{m=0}^{L-1}\sum_{n=0}^{L-1} \mathbb{P}\left(Y_i = k | B_i = m\right) \mathbb{P}\left(Y_j = l | B_j = n\right)\\ &\times \mathbb{E}_{f_i,f_j}\left[\left(\Phi\left(\lambda_{m+1}, f_i, \sigma_{\mathrm{A}}^2\right) - \Phi\left(\lambda_m, f_i, \sigma_{\mathrm{A}}^2\right)\right)\left(\Phi\left(\lambda_{n+1}, f_j, \sigma_{\mathrm{A}}^2\right) - \Phi\left(\lambda_n, f_j, \sigma_{\mathrm{A}}^2\right)\right)\right].\end{aligned}$$

This involves an intractable integral which we solve via an efficient numerical procedure, based on the Clenshaw–Curtis quadrature rule [10]. We begin by solving the first integral with respect to f_i, thus reducing the dimension of the problem:

$$\begin{aligned}&\mathbb{E}_{f_i,f_j}\left[\left(\Phi\left(\lambda_{m+1}, f_i, \sigma_{\mathrm{A}}^2\right) - \Phi\left(\lambda_m, f_i, \sigma_{\mathrm{A}}^2\right)\right)\left(\Phi\left(\lambda_{n+1}, f_j, \sigma_{\mathrm{A}}^2\right) - \Phi\left(\lambda_n, f_j, \sigma_{\mathrm{A}}^2\right)\right)\right]\\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} p\left(f_i, f_j\right)\left(\Phi\left(\lambda_{m+1}, f_i, \sigma_{\mathrm{A}}^2\right) - \Phi\left(\lambda_m, f_i, \sigma_{\mathrm{A}}^2\right)\right)\left(\Phi\left(\lambda_{n+1}, f_j, \sigma_{\mathrm{A}}^2\right)\right.\\ &\quad \left.-\Phi\left(\lambda_n, f_j, \sigma_{\mathrm{A}}^2\right)\right) \mathrm{d}f_i \mathrm{d}f_j\\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} p\left(f_i | f_j\right) p\left(f_j\right)\left(\Phi\left(\lambda_{m+1}, f_i, \sigma_{\mathrm{A}}^2\right) - \Phi\left(\lambda_m, f_i, \sigma_{\mathrm{A}}^2\right)\right)\left(\Phi\left(\lambda_{n+1}, f_j, \sigma_{\mathrm{A}}^2\right)\right.\\ &\quad \left.-\Phi\left(\lambda_n, f_j, \sigma_{\mathrm{A}}^2\right)\right) \mathrm{d}f_i \mathrm{d}f_j\\ &= \mathbb{E}_{f_j}\left[\Delta\left(\lambda_m, \lambda_{m+1}, c_1 + c_2 f_j, \sigma_{\mathrm{A}}^2 + \sigma^2\right)\Delta\left(\lambda_n, \lambda_{n+1}, f_j, \sigma_{\mathrm{A}}^2\right)\right], \qquad (2.55)\end{aligned}$$

where we define $c_2 := \frac{\mathcal{C}(\mathbf{x}_*, \mathbf{x}_i)}{\mathcal{C}(\mathbf{x}_i, \mathbf{x}_i)}$, and $c_1 := \mu\left(\mathbf{x}_*\right) - c_2 \mu\left(\mathbf{x}_i\right)$.

This integral with respect to f_j does not admit a closed form representation, and we utilize a numerical procedure to solve it. We now develop an efficient numerical solution via the Clenshaw–Curtis quadrature [10].

The Clenshaw–Curtis quadrature only works on finite integral domains, while (2.55) has infinite support. We shall first use a generic coordinate transformation which will transform the integral in (2.55) from an infinite interval into a finite one, presented in Lemma 2.4 and then utilize the Clenshaw–Curtis quadrature in Lemma 2.6 and finally calculate the covariance matrix in Proposition 2.1.

Lemma 2.4 (Generic Coordinate Transformation for Integration on Infinite Intervals) *Consider the generic coordinate transformation for the integrand and terminals via the mapping* $x = \frac{t}{1-t^2}$ *giving the mapped definite integral*

$$\int_{-\infty}^{+\infty} f(x)dx = \int_{-1}^{+1} f\left(\frac{t}{1-t^2}\right)\frac{1+t^2}{(1-t^2)^2}dt.$$

When Lemma 2.4 is applied to (2.55), one obtains

$$\begin{aligned}&\mathbb{E}_{f_j}\left[\Delta\left(\lambda_m, \lambda_{m+1}, c_1 + c_2 f_j, \sigma_{\mathrm{A}}^2 + \sigma^2\right)\Delta\left(\lambda_n, \lambda_{n+1}, f_j, \sigma_{\mathrm{A}}^2\right)\right]\\ &= \int_{-\infty}^{\infty}\Delta\left(\lambda_m, \lambda_{m+1}, c_1 + c_2 f_j, \sigma_{\mathrm{A}}^2 + \sigma^2\right)\Delta\left(\lambda_n, \lambda_{n+1}, f_j, \sigma_{\mathrm{A}}^2\right) p\left(f_j\right)\mathrm{d}f_j\end{aligned}$$

$$\overset{\left(f_j := \frac{t}{1-t^2}\right)}{=} \int_{-1}^{1} \Delta\left(\lambda_m, \lambda_{m+1}, c_1 + c_2 \frac{t}{1-t^2}, \sigma_{\mathrm{A}}^2 + \sigma^2\right) \Delta\left(\lambda_n, \lambda_{n+1}, \frac{t}{1-t^2}, \sigma_{\mathrm{A}}^2\right) \tag{2.56}$$

$$\times\, p\left(\frac{t}{1-t^2}\right) \frac{1+t^2}{\left(1-t^2\right)^2} \mathrm{d}t.$$

Next, we solve this integral via the Clenshaw–Curtis Quadrature rule.

Lemma 2.5 (Clenshaw–Curtis Quadrature Rule [10]) *Consider the closed form approximation of the integral*

$$\int_0^{\pi} g(\cos\theta)\sin(\theta)\, d\theta \simeq a_0 + \sum_{k=1}^{M/2-1} \frac{2a_{2k}}{1-(2k)^2} + \frac{a_M}{1-M^2}.$$

which involves finding a subset of the coefficients $\{a_k\}_{k\geq 0}$ *given by* a_{2k}*, due to aliasing arguments in [8]. These coefficients are solution to integrals involving periodic functions* $f(\cos\theta)$*, then the Fourier series can be computed efficiently and accurately up to Nyquist frequency* $k = M$*, through a* $(M+1)$ *equally spaced and equally weighted points* $\theta_m = m\pi/M$ *for* $m = 0, \ldots, M$*. At the endpoints of the domain the weights are given by* $1/2$ *to ensure double-counting is avoided. This is equivalent to a discrete cosine transform (DCT) approximation given by*

$$a_k = \frac{2}{M}\left[\frac{g(1)}{2} + \frac{g(-1)}{2}(-1)^k + \sum_{m=1}^{M-1} g(\cos[n\pi/M])\cos(mk\pi/M)\right],\ \forall k \in \{0, \ldots, M\}. \tag{2.57}$$

We now apply the Clenshaw–Curtis quadrature rule to our integral in (2.55).

Lemma 2.6 *The expectation in (2.55) can be evaluated by applying the Clenshaw–Curtis quadrature to the transformed integral in (2.56), as follows:*

$$\begin{aligned}
&\mathbb{E}_{f_j}\left[\Delta\left(\lambda_m, \lambda_{m+1}, c_1 + c_2 f_j, \sigma_A^2 + \sigma^2\right)\Delta\left(\lambda_n, \lambda_{n+1}, f_j, \sigma_A^2\right)\right] \\
&= \int_{-1}^{1} \underbrace{\Delta\left(\lambda_m, \lambda_{m+1}, c_1 + c_2\frac{t}{1-t^2}, \sigma_A^2 + \sigma^2\right)\Delta\left(\lambda_n, \lambda_{n+1}, \frac{t}{1-t^2}, \sigma_A^2\right) p\left(\frac{t}{1-t^2}\right)\frac{1+t^2}{\left(1-t^2\right)^2}}_{:=g(t)} dt \\
&\simeq a_0 + \sum_{k=1}^{M/2-1}\frac{2a_{2k}}{1-(2k)^2} + \frac{a_M}{1-M^2},
\end{aligned}$$

with a_k *defined in (2.57).*

Now that we have evaluated the expectation term, we derive the (i, j)th term of the covariance matrix.

Proposition 2.1 *The* (i, j)*th term of* $\mathbb{E}_{\mathbf{Y}_{\mathcal{N}}}\left[\mathbf{Y}_{\mathcal{N}}\mathbf{Y}_{\mathcal{N}}^{T}\right]$ *can be approximated as:*

$$\mathbb{E}_{Y_i,Y_j}\left[Y_iY_j\right] \simeq \sum_{k=0}^{L-1}\sum_{l=0}^{L-1} kl \sum_{m=0}^{L-1}\sum_{n=0}^{L-1} \mathbb{P}\left(Y_i = k|B_i = m\right)\mathbb{P}\left(Y_j = l|B_j = n\right)$$
$$\times\left(a_0 + \sum_{k=1}^{M/2-1}\frac{2a_{2k}}{1-(2k)^2} + \frac{a_M}{1-M^2}\right).$$

2.6 Simulations

In this section we consider two studies, the first is based on synthetic data generated from a known model. We use this controlled scenario to demonstrate the properties of our estimation methods and illustrate how accurate they will be in different settings. Then we study a real data application which involves analysis of wind speed data with the application in mind related to storm surge modeling in Europe, under the class of weather events known in insurance modeling as wind storms or storm surge insurance storms. This type of application is of direct relevance for both safety assessment and insurance pricing purposes, see [7, 13].

2.6.1 Synthetic Example

To evaluate the performance of the proposed algorithms and the improvement they provide we generated 2-D realizations from a Gaussian process with the following attributes: the mean is $\mu\left(\mathbf{x}\right) = 0$ and the kernel is a radial basis function with length scale, $l = 2$.

$$\mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\Omega}\right) := \exp\left(-\frac{\left\|\mathbf{x}_i - \mathbf{x}_j\right\|}{l}\right). \tag{2.58}$$

A realization from the GP is shown in Fig. 2.1. In this example we placed 10 high-quality sensors which are marked by the black markers. We then tested the field reconstruction algorithm for various system configurations, changing the number of low-quality sensors, the SNR and the probabilities of incorrect wireless channels transmission, denoted p_e. To obtain the same measure of SNR for both types of sensors, we set $\sigma_v^2 = 0$ and define $\mathrm{SNR} = 10\log\sigma_w^2$. The prediction mean squared errors (PMSE) are presented in the right side of Fig. 2.1. The results show that substantial improvements can be obtained by adding low-quality sensors. This is especially true in the cases of high SNR and perfect wireless channels communications, where the PMSE of the heterogeneous network is roughly 1/3 of the PMSE based only on high-quality sensors.

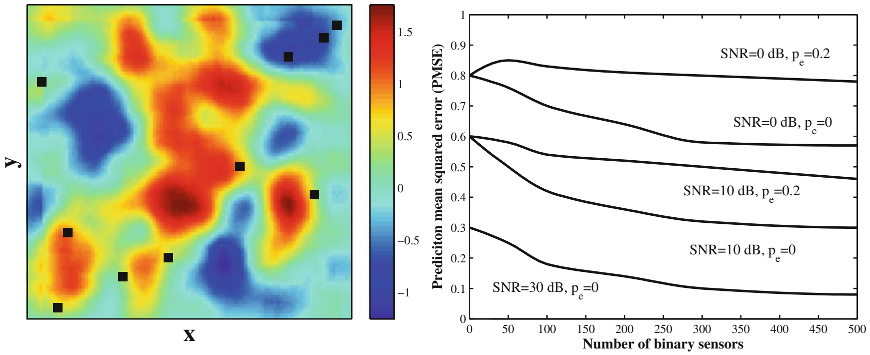

Fig. 2.1 Realization from a 2-d Gaussian process. The *black* markers denote the locations of the 10 high-quality sensors

2.6.2 Sensor Networks for Insurance: Wind Speed and Insurance Storms

In this study we use a publicly available insurance storm surge database known as the Extreme Wind Storms Catalogue.[2] The data is available for research as the XWS Datasets: (c) Copyright Met Office, University of Reading and University of Exeter. Licensed under Creative Commons CC BY 4.0 International License. This database is comprised of 23 storms which caused high insurance losses known as 'insurance storms' and 27 storms which were selected because they are the top 'noninsurance' storms as ranked by the storm severity index, see details on the web site.

The data provided is comprehensive and provides features such as the footprint of the observations on a location grid with a rotated pole at longitude = 177.5°, latitude = 37.5°. As discussed in the data description provided with the dataset, this is a standard technique used to ensure that the spacing in km between grid points remains relatively consistent. The footprints are on a regular grid in the rotated coordinate system, with horizontal grid spacing 0.22°. The data for each of the storms provide a list of grid number and maximum 3-s gust speed in meters per second. The true locations (longitude and latitude) of the grid points are given in grid locations file. We selected two storms to analyze, the first is known as Dagmar (Patrick or Tapani) which took place on 26/12/2011 and affected are Finland and Norway; and the second was the storm known as Ulli taking place on 03/01/2012 which affected the UK.

To understand the significance of this analysis, we note that the Dagmar-Patrick storm is reported to have caused damage worth 40 Million USD and reached a maximum wind speed of 70 mph over land. The storm Ulli is reported to have

[2] http://www.met.reading.ac.uk/~extws/database/dataDesc.

caused even more damage of 200 Million USD and reached a wind speed of 87 mph. In our study, we reconstruct the spatial map of these wind speeds for a given instant of time.

2.6.2.1 Model Calibration Wind Speed Data

To calibrate the model we first fit the hyperparameters of the model via maximum-likelihood estimation (MLE) procedure. We used a 2-D radial basis function of the following form:

$$\mathcal{C}\left(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\Omega}\right) := \sigma_x^2 \exp\left(-\frac{\|x_i - x_j\|}{l_x}\right) \times \sigma_y^2 \exp\left(-\frac{\|y_i - y_j\|}{l_y}\right), \tag{2.59}$$

thus decomposing the kernel into orthogonal coordinates which we found provided a much more accurate fit. The reason for this is it allows for inhomogeneity through differences in spatial dependence in vertical and horizontal directions, which is highly likely to occur in the types of wind speed data studied. The MLE of the length and scale parameters obtained are given by:

- Dagmar-Patrick storm: $\sigma_x^2 = 0.1, l_x = 0.5$ and $\sigma_y^2 = 10, l_y = 0.1$.
- Ulli storm: $\sigma_x^2 = 0.5, l_x = 0.1$ and $\sigma_y^2 = 1, l_y = 0.1$.

We note that details on how to estimate the GP hyperparameters can be found in [Chap. 5] [37]. The covariance function estimates are presented in Fig. 2.4 for the Dagmar-Patrick (left panel) and Ulli (right panel) wind storms. These plots show the spatial dependence over UK and Europe between wind speeds during the peak of the storm fronts as they transited across different regions of the English channel. It is clear that the correlation of the Dagmar-Patrick storm is much stronger than of the Ulli storm in both axes. This should have an impact on the quality of the field reconstruction estimation that we will demonstrate next.

2.6.2.2 Wind Field Intensity Estimation for Insurance Wind Storms

We performed wind field intensity estimation using our algorithm and compared it to the case where only high-quality sensors are utilized. The results are presented in Figs. 2.2 and 2.3, for the Dagmar-Patrick and Ulli storms, respectively. We set the region of interest (ROI) as shown in the upper left of Figs. 2.2 and 2.3. We then chose 15 locations to place high-quality sensors. These locations are depicted with black square markers. The actual wind speed field intensity is shown in the upper right figures. The lower left figures show the estimated field based only on the 15 high-quality sensors. The lower right figures show the estimated field based on the 15 high-quality and 100 low-quality sensors. To illustrate the impact of adding low-quality sensors make, we set the error probability of the wireless channels to zero. The

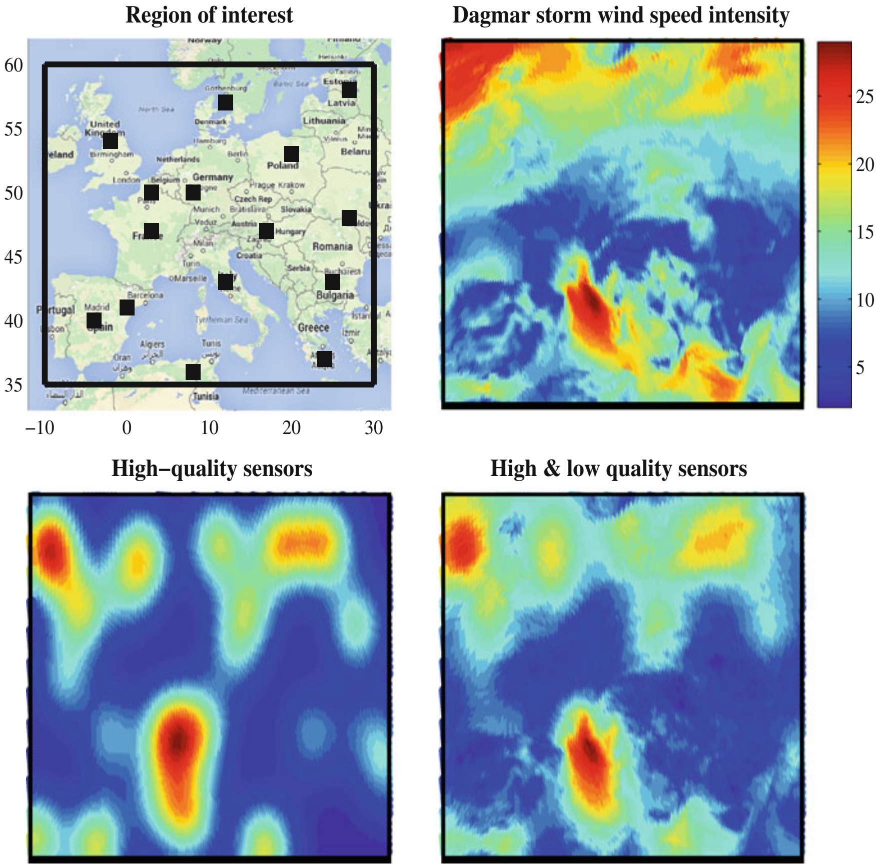

Fig. 2.2 Wind speed prediction of the Dagmar-Patrick storm. The *rectangular* in the *upper left* figure represents the region of interest which contains 15 high-quality sensors. The *upper right* figure represents the "true" data wind speed intensities (m/s). The *lower left* figure shows the field reconstruction based solely on the 15 high-quality sensors via Gaussian Process regression. The *lower right* figure shows the field reconstruction of our algorithm based on the heterogeneous network with 15 high-quality sensors and 100 low-quality sensors. The normalized mean squared error based on the high-quality sensors is 0.67 and based on both high- and low-quality sensors is 0.25

figures show that a significant improvement can be obtained by augmenting the high-quality sensor network with many cheap low-quality sensors. The field reconstruction (the ROI contains 14006 spatial points) prediction mean squared error for the two storms is given in Table 2.1. As expected the prediction performance for the Dagmar-Patrick storm is better than for the Ulli storm. This can be explained by the higher spatial correlation exhibited by the Dagmar-Patrick storm as shown in Fig. 2.4.

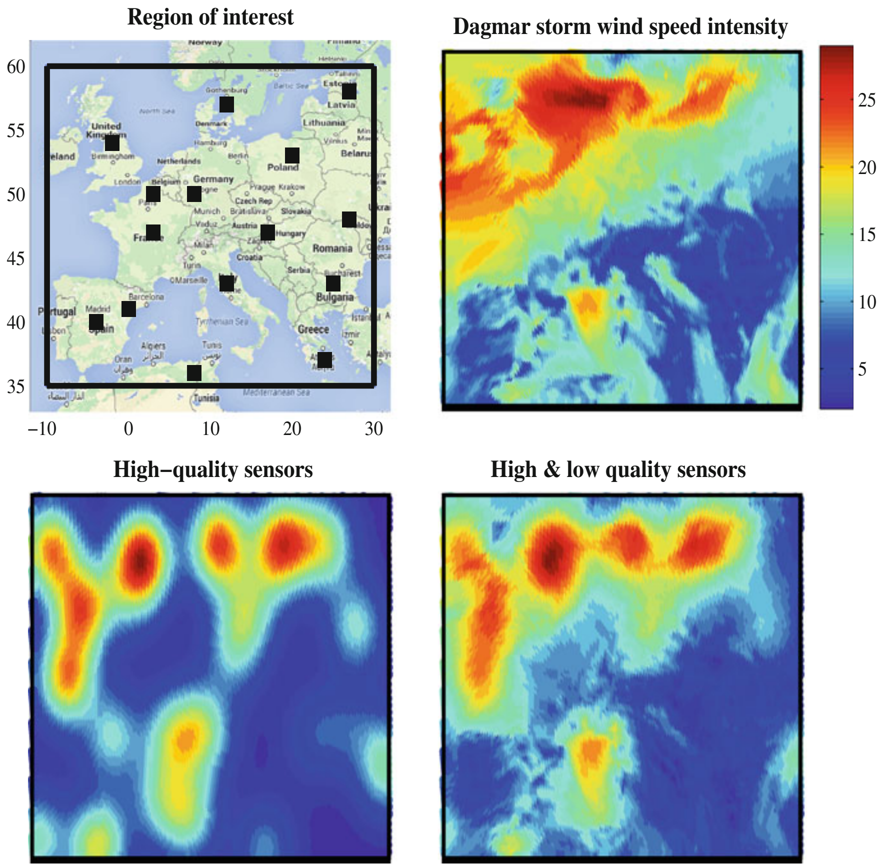

Fig. 2.3 Wind speed prediction of the Ulli storm. The *rectangular* in the *upper left* figure represents the region of interest which contains 15 high-quality sensors. The *upper right* figure represents the "true" data wind speed intensities (m/s). The *lower left* figure shows the field reconstruction based solely on the 15 high-quality sensors via Gaussian process regression. The *lower right* figure shows the field reconstruction of our algorithm based on the heterogeneous network with 15 high-quality sensors and 100 low-quality sensors. The normalized mean squared error based on the high-quality sensors is 0.85 and based on both high- and low-quality sensors is 0.37

Table 2.1 Field reconstruction performance for the two storms

Normalized prediction mean squared error		
Reconstruction method	Dagmar-Patrick storm	Ulli storm
15 high-quality sensors	0.67	0.85
15 high-quality and 100 low-quality sensors	0.25	0.37

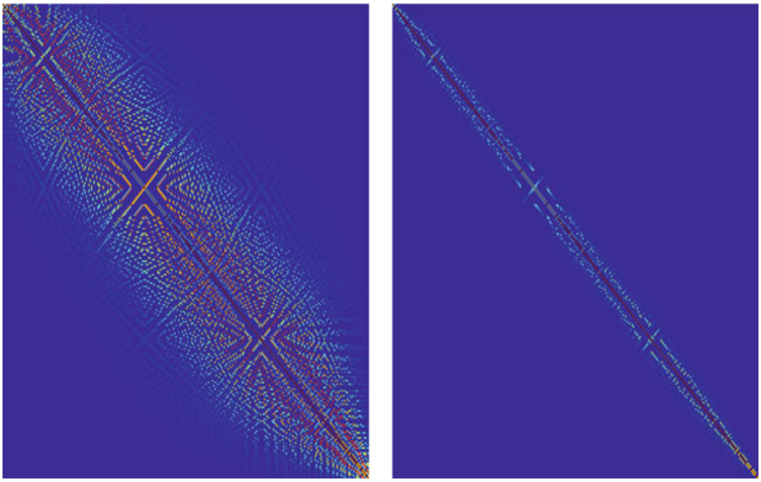

Fig. 2.4 The covariance function estimation of the Dagmar-Patrick (*left panel*) and Ulli (*right panel*) storms. These results show that the spatial correlation of the Dagmar-Patrick storm is larger than the Ulli storm

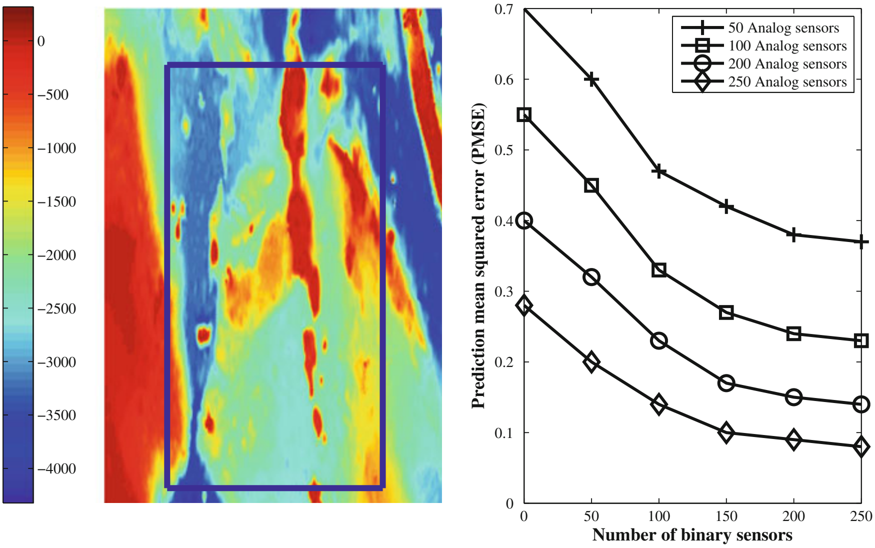

Fig. 2.5 Ocean depth estimation results. The *rectangular* in the *left figure* represents the region of interest and the intensity represents the true ocean's depth. The *right figure* presents the prediction MSE of our algorithm based on the heterogeneous network with 50, 100, 200, 250 high-quality sensors and varying number of low-quality sensors

2.6.3 Bathymetry Example

In this example we use the heterogeneous sensor network to estimate the spatial field for the depth of the ocean floor based on measurements known as Bathymetry. This type of analysis is also directly relevant to the wind speed and storm surge modeling done in the firs example, as bathymetric measurements are known to vary significantly during storm surge events and cyclones. This makes the spatial field reconstruction of such a feature directly relevant to modeling practically important spatial features. The ocean depth can help to provide an indication of the likely event of a flooding event from a storm front.

We use a publicly available database known as the eSurge.[3] We selected to analyze a square region in the north-east corner of Australia at the South-Pacific ocean presented in the left panel of Fig. 2.5. This region is known to be frequently hit by cyclones which cause a change in the topography of the ocean floor.

We performed the depth estimation using our algorithm and compared it to the case where only high-quality sensors are utilized. In each simulation the sensors were deployed on a regular grid and we changed the number of high-quality and low-quality sensors deployed. We then calculated the prediction mean squared error (PMSE) which is presented in the right panel of Fig. 2.5. Similarly to the synthetic example in Sect. 2.6.1, there was a diminishing improvement when the number of low-quality sensors was above 200.

References

1. Adler, R., Taylor, J.: Random Fields and Geometry, vol. 115. Springer, New York (2007)
2. Agrawal, P., Patwari, N.: Correlated link shadow fading in multi-hop wireless networks. IEEE Trans. Wirel. Commun. **8**(8), 4024–4036 (2009)
3. Akyildiz, I., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: a survey. Comput. Netw. **38**(4), 393–422 (2002)
4. Akyildiz, I., Vuran, M., Akan, O.: On exploiting spatial and temporal correlation in wireless sensor networks. In: Proceedings of WiOpt'04: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks pp. 71–80 (2004)
5. Anastasi, G., Conti, M., Di Francesco, M., Passarella, A.: Energy conservation in wireless sensor networks: a survey. Ad Hoc Netw. **7**(3), 537–568 (2009)
6. Berger, J.: Statistical Decision Theory and Bayesian Analysis. Springer, New York (1985)
7. Berz, G.: Windstorm and storm surges in Europe: loss trends and possible counter-actions from the viewpoint of an international reinsurer. Philos. Trans. R. Soc. A: Math. Phys. Eng. Sci. **363**(1831), 1431–1440 (2005)
8. Boyd, J.: Chebyshev and Fourier Spectral Methods. Dover Publications, New York (2001)
9. Chintalapudi, K., Fu, T., Paek, J., Kothari, N., Rangwala, S., Caffrey, J., Govindan, R., Johnson, E., Masri, S.: Monitoring civil structures with a wireless sensor network. IEEE Internet Comput. **10**(2), 26–34 (2006)
10. Clenshaw, C., Curtis, A.: A method for numerical integration on an automatic computer. Numer. Math. **2**(1), 197–205 (1960)

[3] http://www.storm-surge.info/project.

11. Cohen, K., Leshem, A.: Energy-efficient detection in wireless sensor networks using likelihood ratio and channel state information. IEEE J. Sel. Areas Commun. **29**(8), 1671–1683 (2011)
12. Fazel, F., Fazel, M., Stojanovic, M.: Random access sensor networks: field reconstruction from incomplete data. In: IEEE Information Theory and Applications Workshop (ITA), pp. 300–305 (2012)
13. Flather, R., Smith, J., Richards, J., Bell, C., Blackman, D.: Direct estimates of extreme storm surge elevations from a 40-year numerical model simulation and from observations. Global Atmos. Ocean Syst. **6**(2), 165–176 (1998)
14. Fonseca, C., Ferreira, H.: Stability and contagion measures for spatial extreme value analyses. arXiv:1206.1228 (2012)
15. French, J.P., Sain, S.R.: Spatio-Temporal Exceedance Locations and Confidence Regions. Annals of Applied Statistics. Prepress (2013)
16. Gu, D., Hu, H.: Spatial Gaussian process regression with mobile sensor networks. IEEE Trans. Neural Netw. Learn. Syst. **23**(8), 1279–1290 (2012)
17. Hoff, P.D., Niu, X.: A Covariance Regression Model. arXiv:1102.5721 (2011)
18. Højsgaard, S., Edwards, D., Lauritzen, S.: Gaussian graphical models. In: Graphical Models with R, pp. 77–116. Springer, New York (2012)
19. Katenka, N., Levina, E., Michailidis, G.: Local vote decision fusion for target detection in wireless sensor networks. IEEE Trans. Signal Process. **56**(1), 329–338 (2008)
20. Kottas, A., Wang, Z., Rodriguez, A.: Spatial modeling for risk assessment of extreme values from environmental time series: a Bayesian nonparametric approach. Environmetrics **23**(8), 649–662 (2012). doi:10.1002/env.2177
21. Krause, A., Singh, A., Guestrin, C.: Near-optimal sensor placements in gaussian processes: theory, efficient algorithms and empirical studies. J. Mach. Learn. Res. **9**, 235–284 (2008)
22. Lorincz, K., Malan, D.J., Fulford-Jones, T.R., Nawoj, A., Clavel, A., Shnayder, V., Mainland, G., Welsh, M., Moulton, S.: Sensor networks for emergency response: challenges and opportunities. IEEE Pervasive Comput. **3**(4), 16–23 (2004)
23. Masazade, E., Niu, R., Varshney, P., Keskinoz, M.: Energy aware iterative source localization for wireless sensor networks. IEEE Trans. Signal Process. **58**(9), 4824–4835 (2010)
24. Matamoros, J., Fabbri, F., Antón-Haro, C., Dardari, D.: On the estimation of randomly sampled 2D spatial fields under bandwidth constraints. IEEE Trans. Wirel. Commun. **10**(12), 4184–4192 (2011)
25. Matern, B.: Spatial variation. meddelanden fraan statens skogsforskningsinstitut, **49**(5), 1–144. Also appeared as Lecture Notes in Statistics, vol. 36 (1986)
26. Msechu, E., Giannakis, G.: Sensor-centric data reduction for estimation with WSNs via censoring and quantization. IEEE Trans. Signal Process. **60**(1), 400–414 (2012)
27. Nevat, I., Peters, G., Collings, I.: Location-aware cooperative spectrum sensing via Gaussian processes. In: IEEE Australian Communications Theory Workshop (AusCTW), pp. 19–24 (2012)
28. Nevat, I., Peters, G.W., Collings, I.B.: Location-aware cooperative spectrum sensing via gaussian processes. In: Communications Theory Workshop (AusCTW), 2012 Australian, pp. 19–24. IEEE (2012)
29. Nevat, I., Peters, G.W., Collings, I.B.: Estimation of correlated and quantized spatial random fields in wireless sensor networks. In: 2013 IEEE International Conference on Communications (ICC), pp. 1931–1935. IEEE (2013)
30. Nevat, I., Peters, G.W., Collings, I.B.: Random field reconstruction with quantization in wireless sensor networks. IEEE Trans. Signal Process. **61**, 6020–6033 (2013)
31. Niu, R., Varshney, P.K.: Target location estimation in sensor networks with quantized data. IEEE Trans. Signal Process. **54**(12), 4519–4528 (2006)
32. Ozdemir, O., Niu, R., Varshney, P.K.: Channel aware target localization with quantized data in wireless sensor networks. IEEE Trans. Signal Process. **57**(3), 1190–1202 (2009)
33. Park, S., Choi, S.: Gaussian processes for source separation. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1909–1912 (2008)

34. Peters, G., Nevat, I., Lin, S., Matsui, T.: Modelling threshold exceedence levels for spatial stochastic processes observed by sensor networks. In: 2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), pp. 1–7. IEEE (2014)
35. Rajasegarar, S., Havens, T.C., Karunasekera, S., Leckie, C., Bezdek, J.C., Jamriska, M., Gunatilaka, A., Skvortsov, A., Palaniswami, M.: High-resolution monitoring of atmospheric pollutants using a system of low-cost sensors. IEEE Trans. Geosci. Remote Sens. **52**, 3823–3832 (2014)
36. Rajasegarar, S., Zhang, P., Zhou, Y., Karunasekera, S., Leckie, C., Palaniswami, M.: High resolution spatio-temporal monitoring of air pollutants using wireless sensor networks. In: 2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), pp. 1–6. IEEE (2014)
37. Rasmussen, C., Williams, C.: Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning). The MIT Press (2005)
38. Schabenberger, O., Pierce, F.J.: Contemporary Statistical Models for the Plant and Soil Sciences. CRC Press, Boca Raton (2002)
39. Sohraby, K., Minoli, D., Znati, T.: Wireless Sensor Networks: Technology, Protocols, and Applications. Wiley, Hoboken (2007)
40. Stein, M.L.: Interpolation of Spatial Data: Some Theory for Kriging. Springer, New York (1999)
41. Vanmarcke, E.: Random Fields: Analysis and Synthesis. World Scientific Publishing Company Inc., Singapore (2010)
42. Vuran, M.C., Akan, O.B., Akyildiz, I.F.: Spatio-temporal correlation: theory and applications for wireless sensor networks. Comput. Netw. J., Elsevier **45**, 245–259 (2004)
43. Werner-Allen, G., Lorincz, K., Ruiz, M., Marcillo, O., Johnson, J., Lees, J., Welsh, M.: Deploying a wireless sensor network on an active volcano. IEEE Internet Comput. **10**(2), 18–25 (2006)
44. Wu, T., Cheng, Q.: Distributed estimation over fading channels using one-bit quantization. IEEE Trans. Wirel. Commun. **8**(12), 5779–5784 (2009)
45. Xu, Y., Choi, J.: Adaptive sampling for learning Gaussian processes using mobile sensor networks. Int. J. Sens. **11**(3), 3051–3066 (2011)
46. Zheng, Y., Niu, R., Varshney, P.: Closed-form performance for location estimation based on quantized data in sensor networks. In: 13th Conference on Information Fusion (FUSION), pp. 1–7. IEEE (2010)
47. Zhou, Y., Li, J., Wang, D.: Posterior cramér-rao lower bounds for target tracking in sensor networks with quantized range-only measurements. IEEE Signal Process. Lett. **17**(2), 157–160 (2010)

Chapter 3
Speech and Music Emotion Recognition Using Gaussian Processes

Konstantin Markov and Tomoko Matsui

Abstract Gaussian Processes (GPs) are Bayesian nonparametric models that are becoming more and more popular for their superior capabilities to capture highly nonlinear data relationships in various tasks ranging from classical regression and classification to dimension reduction, novelty detection and time series analysis. Here, we introduce Gaussian processes for the task of human emotions recognition from emotionally colored speech as well as estimation of emotions induced by listening to a piece of music. In both cases, first, specific features are extracted from the audio signal, and then corresponding GP-based models are learned. We consider both static and dynamic emotion recognition tasks, where the goal is to predict emotions as points in the emotional space or their time trajectory, respectively. Compared to the current state-of-the-art modeling approaches, in most cases, GPs show better performance.

3.1 Introduction

Emotions play an important role in human-to-human communication. Expressed both by speech and body language, they convey a lot of nonlinguistic information making human interaction inherently "natural." That is why it is important to study and model emotions in order to achieve as natural as possible human–computer communication. The first and foremost task is to accurately identify the emotional state of a person. This would benefit current speech recognition and translation systems, facilitate development of new human centric applications, and also help diagnose and prevent mental health disorders such as depression which exhibit specific emotional patterns.

K. Markov (✉)
The University of Aizu, Fukushima, Japan
e-mail: markov@u-aizu.ac.jp

T. Matsui
The Institute of Statistical Mathematics, Tokyo, Japan
e-mail: tmatsui@ism.ac.jp

G.W. Peters and T. Matsui (eds.), *Modern Methodology and Applications in Spatial-Temporal Modeling*, JSS Research Series in Statistics,
DOI 10.1007/978-4-431-55339-7_3

On the other hand, a lot of music data have become available recently either locally or over the Internet and in order for users to benefit from them, an efficient music information retrieval (MIR) technology is necessary. Although users are more likely to use genres or artists names when searching or categorizing music, the main power of music is in its ability to communicate and trigger emotions in listeners. Thus, determining computationally the emotional content of music is also an important task.

There are two approaches to represent emotions in computer systems: categorical and dimensional [3, 24]. Categorical approach involves finding emotional descriptors, usually adjectives, which can be arranged into groups. Given the perceptual nature of human emotion, it is difficult to come up with an intuitive and coherent set of adjectives and their specific grouping. Depending on the research objectives, the number of emotion categories and their names can vary greatly. A popular choice is the set of so-called "primary" emotions [6] which includes joy, sadness, fear, anger, surprise, and disgust. Other emotions can be produced by "mixing" primary emotions like colors in a color palette. To alleviate the challenge of ensuring consistent interpretation of emotion categories, some studies propose to describe emotions using continuous multidimensional metrics defined on low-dimensional spaces. Most widely accepted is the two-dimensional *Valence–Arousal* (V–A) affect space [45, 48] where emotions are represented by points in the V–A plane. Figure 3.1 shows the space where some regions are associated with distinct emotion categories. An extension to three-dimensional affect space which includes additional *Dominance* (D) axes has also been proposed [14]. It can be argued that emotions are not necessarily constant, but can vary within utterances or during the course of a song. This variation in time can be represented by a trajectory in the emotional space. Here, we assume that in the case of static emotions, the task is to automatically find the point in the V–A or V–A–D space which corresponds to the speaker affect state or emotion induced by a given music piece. For dynamic emotions, the task would be to estimate or track the emotion trajectory in the affect space.

An important problem in emotion recognition is how to extract features that efficiently and compactly characterize different emotions. One aspect of this problem is the analysis window used for feature extraction. A standard approach in audio signal

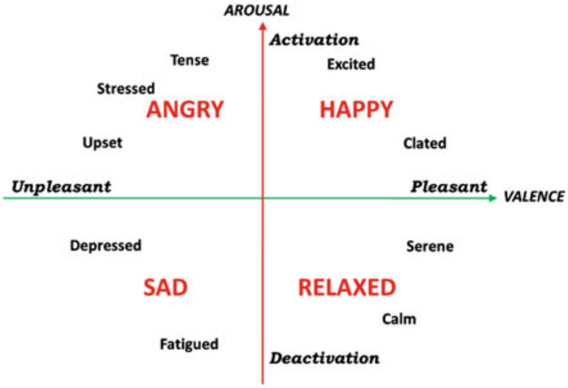

Fig. 3.1 Two-dimensional (Valence-Arousal) affective space of emotions. Different regions correspond to different categorical emotions

processing is to divide the signal into small intervals called frames from which local feature vectors are extracted. This is justified for quickly changing targets. Emotions, however, vary slowly and the analysis interval may be as long as few seconds. Common approach is to obtain some statistics such as mean, variance, etc. of the local features for each interval and stack them into one vector. This technique is well suited for the case of dynamic emotion recognition. For the static emotions case, analysis interval is usually extended to cover the whole utterance or song and global statistics are calculated.

There is a strong evidence that prosodic features such as pitch and energy are closely related to the emotional content of an utterance. Overall energy and its distribution across frequencies as well as duration of pauses are directly affected by the arousal state of the speaker [5]. Spectral-based features commonly used in speech recognition, i.e., MFCC and LPCC, have also shown good performance, though the log frequency power coefficients (LFPC) have been found to perform better [40]. When data from other modalities such as video are available, features extracted from facial expressions can be combined with the acoustic features which may lead to an improved recognition accuracy [22].

Prior studies focused on searching for emotion-specific music features have not found any dominant single one [64], so the most commonly used are those utilized in the other MIR tasks as well. Conventional features can be divided into “low-level” features including timbre (zero-crossing rate, spectral centroid, flux, roll-off, MFCC, and others) and temporal (amplitude modulation or autoregressive coefficients) features, as well as “mid-level” features, such as rhythm, pitch, and harmony [16]. On the other hand, it is also possible to apply unsupervised learning methods to find some “high level” representations of the “low-level” features, and then use them as a new type of features. This can be accomplished using non-negative matrix factorization (NMF), sparse coding [34], or deep neural networks (DNN) [29].

For categorical emotions, both speech and music emotion recognition tasks can be cast as a classification problem, so the same models can be used. This holds for the dimensional emotions as well since the task is actually a regression problem. Hidden markov models (HMM), Gaussian mixture models (GMM), support vector machine (SVM), and neural networks have been used to classify emotions [12]. Regression models, such as multiple linear regression (MLR), support vector regression (SVR), or Adaboost.RT, as well as multi-level least-squares or regression trees [3] have been successfully applied for dimensional emotion estimation. Model learning is usually supervised and requires labeled training data. Finding consistent emotion labels in terms of V–A or V–S–D values is even more challenging than obtaining category labels because emotion interpretation can be very subjective and varies among listeners. It requires data annotation by multiple experts which are expensive, time consuming, and labor intensive [1]. Especially, problematic is the collection of ground truth labels for time-continuous emotions, because the reaction lag of evaluators also needs to be taken into account [33].

Gaussian processes have been known as nonparametric Bayesian models for quite some time, but just recently have attracted attention of researchers from other fields than statistics and applied mathematics. After the work of Rasmussen and

Williams [44] which introduced GPs for the machine learning tasks of classification and regression, many researchers have utilized GPs in various practical applications. As SVMs, they are also based on kernel functions and Gram matrices, and can be used as their plug-in replacement. The advantage of GPs with respect to SVMs is that their predictions are truly probabilistic and that they provide a measure of the output uncertainty. Another big plus is the availability of algorithms for their hyperparameter learning. The downside is that the GP training complexity is $\mathcal{O}(n^3)$, which makes them difficult to use in large-scale tasks. Several sparse approximation methods have been proposed [7, 53], but this problem has not yet been fully solved and is a topic of an ongoing research.

Evaluation of the emotion recognition systems is usually performed in terms classification accuracy for categorical emotions. In the case of dimensional emotions, Pearson correlation coefficient and/or root-mean-squared error measures (RMSE) are used and often applied for each affect dimension separately. However, recently there have been discussions about the usefulness of the correlation coefficient from practical point of view. The analysis given in [22] shows that in order to achieve high correlation, coarse trajectory estimation is enough, while close frame-wise matching of up to 90 % of the trajectory can still result in much lower correlation. There are also different opinions on how to treat cases when correlation coefficient is negative.

In the next section, various existing emotion recognition systems are reviewed and compared. Brief introduction of the Gaussian processes and their implementation in regression tasks is given in Sects. 3.3 and 3.4. GP regression models can be used for static emotion estimation in a straightforward way. During training, they learn the nonlinear mapping between the feature vectors and the corresponding affect dimensions values. Thus, separate GP models are trained for each arousal and valence (and Dominance) dimension. Dynamic emotion trajectories can be considered as a time series data, so methods from statistical time series analysis would be applicable to ensure that not only feature-emotion mapping, but also temporal evolution of emotions is taken into account. One such method is Bayesian filtering by state-space models (SSMs). It is briefly described in Sect. 3.5. A widely used SSM based on linear functions is the Kalman filter (KF) [18] which is explained in Sect. 3.6. Linearity assumptions of KF, however, are significant drawback. On the other hand, particle filters (PF) allow for nonlinear functions to be used such as GPs. Section 3.7 describes the PF basics and its implementation using Gaussian processes. How to build emotion recognition systems using GPs for both static and dynamic emotions and some evaluation results on speech and music data are presented in Sect. 3.8. The last section contains some discussion and conclusions.

3.2 Related Studies

There are many studies on speech emotion recognition and most of them take the categorical approach to emotion representation. Various types of classifiers have been used such as HMM, GMM, SVM, ANN, k-mean, and others. The most popular is a fully connected HMM using prosodic features [39, 52]. In [40], a discrete HMM with MFCC, LPCC, and LFPC vectors was used and up to 75.5 % accuracy was obtained over the set of "primary" emotions. For dimensional dynamic emotion recognition, however, there are just a few studies. This task has been facilitated by the audio-visual emotion challenge (AVEC) series of evaluations. The 2013 winner [38] uses MFCC and other spectral low-level descriptors as features and partial least-squares (PLS) regression. However, this approach fails to capture dynamics information. This problem is solved in [60] using long short-term RNN to capture the time dependencies in emotion trajectories.

In one of the earliest studies on music emotion recognition, features representing timbre, rhythm, and pitch were used in SVM-based system to classify music into 13 mood categories [30]. With 499 hand-labeled 30 s clips, an accuracy of 45 % was achieved. In 2007, music emotion classification was included in the MIR evaluation exchange (MIREX) benchmarks and the best performance of 61.5 % was again achieved using SVM classifier [56]. However, recent studies have suggested that regression approaches using continuous mood representation can perform better than categorical classifiers [63]. SVR was applied in [64] to map music clips, each represented by a single feature vector, into two-dimensional V–A space. After principal component analysis (PCA)-based feature dimensionality reduction, this system achieved R^2 scores of 0.58 and 0.28 for arousal and valence, respectively. Later, this approach was extended by representing perceived emotion of a clip as a probability distribution in the emotion plane [62]. It also is possible to combine categorical and continuous emotion representations by quantizing the V–A space and apply emotion cluster classification using SVM [51], or another regression model, trained for each cluster [11].

For dynamic emotions, one approach is to divide a piece of music into segments short enough to assume that emotion does not change within each segment, and then use standard classification techniques [32]. Another study [49] considers arousal and valence as latent states of a linear dynamical system and applies KF to recover emotion dynamics over time. However, KF is a linear system and has its limitations. There exist nonlinear SSMs such as the extended Kalman filter (EKF) and unscented Kalman filter (UKF), but they put certain constraints on the SSM state and measurement functions and often suffer from stability issues. Another approach is to consider the fact that for some time intervals, emotion depends on the past and future system inputs. This suggests that context-sensitive or recurrent models can be applied. One such model is the conditional random field (CRF), but for its direct implementation the emotion space needs to be discretized [50]. However, recently proposed CRF extension allows to overcome this drawback [20]. Another model which has gained popularity lately is the long short-term memory (LSTM) recurrent neural network. It has been successfully applied for dynamic music emotion recognition and has shown state-of-the-art performance [59, 61].

Although Gaussian processes have become popular in machine learning community and have been used in such tasks as object categorization in computer vision [23] or economics and environmental studies [46], there are still few GP applications in the field of signal processing. In one such application, GP regression model is applied to time-domain voice activity detection and speech enhancement [41]. In [31], using GP, researchers estimate speakers likability given recordings of their voices. Another recent study employs GPs for head-related transfer function (HRTF) estimation in acoustic scene analysis [26]. Finally, several extensions and new models based on GPs have been developed. For example, Gaussian process latent variable model (GP-LVM) was introduced for nonlinear dimensionality reduction [27], but have also been applied to image reconstruction [54] and human motion modeling [28]. Another promising extension is the Gaussian process dynamic model (GPDM) [58]. It is a nonlinear dynamical system which can learn the mapping between two continuous variables spaces. One of the first applications of GPDM in audio signal processing was for speech phoneme classification [42]. Although the absolute classification accuracy of the GPDM was not high, in certain conditions, they outperformed the conventional hidden Markov model (HMM). In [19], GPDM is used as a model for nonparametric speech representation and speech synthesis.

Some previous studies [35–37] have shown that GPs can be a feasible alternative to SVMs both for music genre classification and static emotion recognition. For the varying emotion case, as mentioned earlier, a state-space models are well suited. A number of GP-based state-space models (GP-SSM) have been proposed recently. GP-BayesFilters [25] use GPs as nonlinear functions and derive GP particle filter, GP-EKF, and GP-UKF algorithms using Monte Carlo (MC) sampling. In [8, 9], an analytic filtering approximation algorithm is presented, but lacks an analytic approach to GP-SSM parameter learning. An attempt to derive such algorithm is done in [55] which, however, has some stability problems. A Particle Markov Chain Monte Carlo (PMCMC) training method is described in [15], but it suffers from slowly converging MC sampling techniques. The problem of training GP-based state-space models parameters can be made much easier if true observations of the latent state process are available. This way, the state dynamics parameters can be learned separately from the parameters of the measurement function. In the KF case, training of the corresponding matrices and noise variances can be done using multivariate linear regression. For the GP-SSM, similar approach is applicable. The difference is that since GP output is scalar and separate GP models have to be trained for each state or observation vector dimension. Models parameters can be obtained using GP regression model learning as explained in Sect. 3.4.

3.3 Gaussian Processes

Gaussian processes are used to describe distributions over functions. Formally, the GP is defined as a collection of random variables any finite number of which has a joint Gaussian distribution [44]. It is completely specified by its mean and covariance functions. For a real process $f(\boldsymbol{x})$, the mean function $m(\boldsymbol{x})$, and the covariance function $k(\boldsymbol{x}, \boldsymbol{x}')$ are defined as

$$\begin{aligned} m(\boldsymbol{x}) &= \mathbb{E}[f(\boldsymbol{x})] \\ k(\boldsymbol{x}, \boldsymbol{x}') &= \mathbb{E}[(f(\boldsymbol{x}) - m(\boldsymbol{x}))(f(\boldsymbol{x}') - m(\boldsymbol{x}'))]. \end{aligned} \tag{3.1}$$

Thus, the GP can be written as

$$f(\boldsymbol{x}) \sim \mathscr{GP}(m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}')). \tag{3.2}$$

A GP prior over function $f(\boldsymbol{x})$ implies that for any finite number of inputs $\boldsymbol{X} = \{\boldsymbol{x}_i\} \in \mathbb{R}^d, i = 1, \dots, n$, the vector of function values $\boldsymbol{f} = [f(\boldsymbol{x}_1), \dots, f(\boldsymbol{x}_n)]^T = [f_1, \dots, f_n]^T$ has a multivariate Gaussian distribution

$$\boldsymbol{f} \sim \mathscr{N}(\boldsymbol{\mu}, \boldsymbol{K}) \tag{3.3}$$

where the mean $\boldsymbol{\mu}$ is often assumed to be zero. The covariance matrix $\boldsymbol{K}$ has the following form:

$$\boldsymbol{K} = \begin{bmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}_1) & \dots & k(\boldsymbol{x}_1, \boldsymbol{x}_n) \\ k(\boldsymbol{x}_2, \boldsymbol{x}_1) & \dots & k(\boldsymbol{x}_2, \boldsymbol{x}_n) \\ \vdots & & \vdots \\ k(\boldsymbol{x}_n, \boldsymbol{x}_1) & \dots & k(\boldsymbol{x}_n, \boldsymbol{x}_n) \end{bmatrix} \tag{3.4}$$

and characterizes the correlation between different points in the process. For $k(\boldsymbol{x}, \boldsymbol{x}')$, any kernel function which produces symmetric and semi-definite covariance matrix can be used.

3.4 Gaussian Process Regression

Given input data vectors $\boldsymbol{X} = \{\boldsymbol{x}_i\}, i = 1, \dots, n$ and their corresponding target values $\boldsymbol{y} = \{y_i\}$, in the simplest regression task, y and $\boldsymbol{x}$ are related as

$$y = f(\boldsymbol{x}) + \varepsilon \tag{3.5}$$

where the latent function $f(\boldsymbol{x})$ is unknown and ε is often assumed to be a zero mean Gaussian noise, i.e., $\varepsilon \sim \mathscr{N}(0, \sigma_n^2)$. Putting a GP prior over $f(\boldsymbol{x})$ allows us to marginalize it out, which means that we do not need to specify its form and parameters. This makes our model very flexible and powerful since $f(\boldsymbol{x})$ can be any nonlinear function of unlimited complexity.

In practice, targets y_i are assumed to be conditionally independent given f_i, so that the likelihood can be factorized as

$$p(\boldsymbol{y}|\boldsymbol{f}) = \prod_1^n p(y_i|f_i) \tag{3.6}$$

where $p(y_i|f_i) = \mathcal{N}(y_i|f_i, \sigma_n^2)$, according to our observation model Eq. (3.5). Since $\boldsymbol{f}$ has normal distribution, i.e., $\boldsymbol{f}|\boldsymbol{X} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{K})$, it follows that $\boldsymbol{y}$ is also a Gaussian random vector

$$p(\boldsymbol{y}|\boldsymbol{X}) = \mathcal{N}(\boldsymbol{y}|\mathbf{0}, \boldsymbol{K} + \sigma_n^2 \boldsymbol{I}). \tag{3.7}$$

Given some new (test) input $\boldsymbol{x}_*$, we can now estimate the unknown target y_* and, more importantly, its distribution. Graphically, the relationship between all involved variables can be represented as shown in Fig. 3.2. To find y_*, we first obtain the joint probability of training targets $\boldsymbol{y}$ and $f_* = f(\boldsymbol{x}_*)$, which is Gaussian

$$p(\boldsymbol{y}, f_*|\boldsymbol{x}_*, \boldsymbol{X}) = \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \boldsymbol{K} + \sigma_n^2 \boldsymbol{I} & \boldsymbol{k}_* \\ \boldsymbol{k}_*^T & k(\boldsymbol{x}_*, \boldsymbol{x}_*) \end{bmatrix}\right) \tag{3.8}$$

where $\boldsymbol{k}_*^T = [k(\boldsymbol{x}_1, \boldsymbol{x}_*), \ldots, k(\boldsymbol{x}_n, \boldsymbol{x}_*)]$. Then, from this distribution, it is easy to obtain the conditional $p(f_*|\boldsymbol{y}, \boldsymbol{x}_*, \boldsymbol{X})$, which is also Gaussian

$$p(f_*|\boldsymbol{y}, \boldsymbol{x}_*, \boldsymbol{X}) = \mathcal{N}(f_*|\mu_{f_*}, \sigma_{f_*}^2) \tag{3.9}$$

with mean and variance

$$\mu_{f_*} = \boldsymbol{k}_*^T (\boldsymbol{K} + \sigma_n^2 \boldsymbol{I})^{-1} \boldsymbol{y}, \tag{3.10}$$

$$\sigma_{f_*}^2 = k(\boldsymbol{x}_*, \boldsymbol{x}_*) - \boldsymbol{k}_*^T (\boldsymbol{K} + \sigma_n^2 \boldsymbol{I})^{-1} \boldsymbol{k}_* \tag{3.11}$$

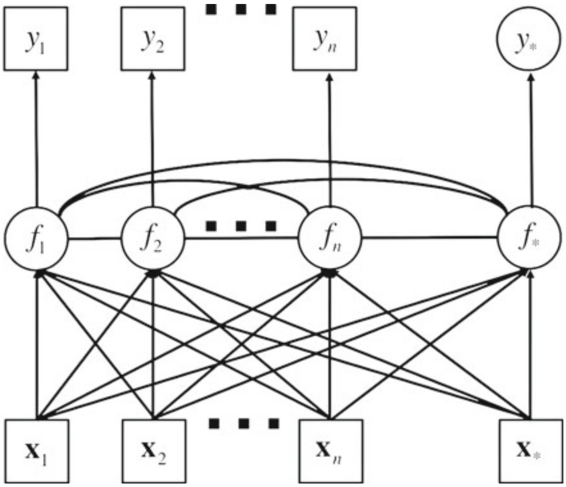

Fig. 3.2 Graphical representation of observable $\boldsymbol{x}$, y, (enclosed in *squares*), latent f, and unobservable y_* (enclosed in *circles*) variable relationships in Gaussian process-based regression task

It is worth noting that the mean μ_{f_*} is a linear combination of the observed targets $\boldsymbol{y}$. It can also be viewed as a linear combination of the kernel functions $k(\boldsymbol{x}_*, \boldsymbol{x}_i)$. On the other hand, the variance $\sigma^2_{f_*}$ depends only on inputs $\boldsymbol{X}$.

To find out the predictive distribution of y_*, we marginalize out f_*

$$\begin{aligned} p(y_*|\boldsymbol{y}, \boldsymbol{x}_*, \boldsymbol{X}) &= \int p(y_*|f_*)p(f_*|\boldsymbol{y}, \boldsymbol{x}_*, \boldsymbol{X})df_* \\ &= \mathcal{N}(y_*|\mu_{y_*}, \sigma^2_{y_*}) \end{aligned} \tag{3.12}$$

where it is easy to show that for homoscedastic likelihood, as in our case, the predictive mean and variance are [43]

$$\mu_{y_*} = \mu_{f_*}, \text{ and} \tag{3.13}$$

$$\sigma^2_{y_*} = \sigma^2_{f_*} + \sigma^2_n. \tag{3.14}$$

Making this mean our predicted target, $y_{\text{pred}} = \mu_{y_*}$ will minimize the risk for a squared loss function $(y_{\text{true}} - y_{\text{pred}})^2$. The variance $\sigma^2_{y_*}$, on the other hand, shows the model uncertainty about y_{pred}.

Parameter learning

Until now, we have considered fixed covariance function $k(\boldsymbol{x}, \boldsymbol{x}')$, but in general, it is parameterized by some parameter vector $\boldsymbol{\theta}$. This introduces *hyper-parameters* to GP, which are unknown and, in practice, very little information about them is available. A Bayesian approach to their estimation would require a *hyper-prior* $p(\boldsymbol{\theta})$ and evaluation of the following posterior:

$$p(\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{X}) = \frac{p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\boldsymbol{y}|\boldsymbol{X})} = \frac{p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta})p(\boldsymbol{\theta})}{\int p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}} \tag{3.15}$$

where the likelihood $p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta})$ is actually the GP marginal likelihood over function values $\boldsymbol{f}$

$$p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}) = \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{X}, \boldsymbol{\theta})d\boldsymbol{f}. \tag{3.16}$$

However, the evaluation of the integral in Eq. (3.15) can be difficult and as an approximation we may directly maximize Eq. (3.16) w.r.t. the hyperparameters $\boldsymbol{\theta}$. This is known as maximum likelihood II (ML-II) type hyperparameter estimation. Since both the GP prior $\boldsymbol{f}|\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{K})$ and the likelihood $\boldsymbol{y}|\boldsymbol{f} \sim \mathcal{N}(\boldsymbol{f}, \sigma^2_n \boldsymbol{I})$ are Gaussians, the logarithm of Eq. (3.16) can be obtained analytically

$$\log p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}) = -\frac{1}{2}\boldsymbol{y}^T \boldsymbol{K}_y^{-1} \boldsymbol{y} - \frac{1}{2}\log|\boldsymbol{K}_y| - \frac{n}{2}\log 2\pi \tag{3.17}$$

where $\boldsymbol{K}_y = \boldsymbol{K} + \sigma_n^2 \boldsymbol{I}$ is the covariance matrix of the noisy targets $\boldsymbol{y}$. Hyperparameters $\boldsymbol{\theta} = \{\sigma_n^2, \boldsymbol{\theta}_k\}$ include the noise variance and parameters of the kernel function. Those which maximize Eq. (3.17) can be found using gradient-based optimization method. Partial derivatives for each θ_i are found from

$$\begin{aligned}\frac{\partial \log p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta})}{\partial \theta_i} = &- \frac{1}{2}\boldsymbol{y}^T \boldsymbol{K}_y^{-1} \frac{\partial \boldsymbol{K}_y}{\partial \theta_i} \boldsymbol{K}_y^{-1} \boldsymbol{y} \\ &- \frac{1}{2}\mathtt{tr}(\boldsymbol{K}_y^{-1} \frac{\partial \boldsymbol{K}_y}{\partial \theta_i})\end{aligned} \tag{3.18}$$

where for $\theta_i = \sigma_n^2$ we have

$$\frac{\partial \boldsymbol{K}_y}{\partial \sigma_n^2} = \sigma_n^2 \boldsymbol{I}. \tag{3.19}$$

Usually, kernel function parameters are all positive, which would require constrained optimization. In practice, this problem is easily solved by optimizing with respect to the logarithm of the parameters, so simple unconstrained optimization algorithms can be used.

3.5 State-Space Models

There are many ways to define a state-space model. Here, we consider an SSM given by

$$\boldsymbol{x}_t = f(\boldsymbol{x}_{t-1}) + \boldsymbol{u}_{t-1}, \quad \boldsymbol{x}_t \in \mathscr{R}^d, \tag{3.20}$$

$$\boldsymbol{y}_t = g(\boldsymbol{x}_t) + \boldsymbol{v}_t \qquad \boldsymbol{y}_t \in \mathscr{R}^e, \tag{3.21}$$

where $f()$ and $g()$ are the unknown functions governing temporal state dynamics and state-to-measurement mapping, respectively. System and observation noises, $\boldsymbol{u}_t \sim \mathscr{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_u)$ and $\boldsymbol{v}_t \sim \mathscr{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_v)$, are both Gaussian with uncorrelated dimensions. The same SSM can be written in terms of probability distributions as

$$p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathscr{N}(\boldsymbol{x}_t; f(\boldsymbol{x}_{t-1}), \boldsymbol{\Sigma}_u), \tag{3.22}$$

$$p(\boldsymbol{y}_t|\boldsymbol{x}_t) = \mathscr{N}(\boldsymbol{y}_t; g(\boldsymbol{x}_t), \boldsymbol{\Sigma}_v). \tag{3.23}$$

Figure 3.3 shows the SSM as a graphical model with arrows denoting dependencies between variables. The initial state $\boldsymbol{x}_0$ is assumed to have known Gaussian distribution $p(\boldsymbol{x}_0) = \mathscr{N}(\boldsymbol{\mu}_0^x, \boldsymbol{\Sigma}_0^x)$. For a sequence of T measurements, the task of filtering is to find approximations to the posterior distribution $p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t})$, where for any sequence $\{z_n\}_{n>0}$ and any $i < j$, $z_{i:j} = z_i, \ldots, z_j$. Often, the task is defined as to find the marginal distribution $p(\boldsymbol{x}_t|\boldsymbol{y}_{1:t})$ [2].

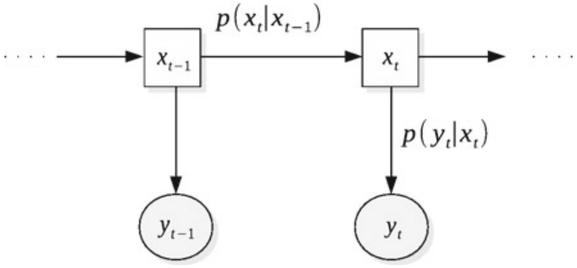

Fig. 3.3 Graphical representation of state-space model. States $\boldsymbol{x}_t$ are continuous latent variables and measurements $\boldsymbol{y}_t$ are observable vectors. *Arrows* show the probabilistic relationship between variables

Following a Bayesian approach, the distribution of interest can be decomposed as follows:

$$p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t}) = \frac{p(\boldsymbol{x}_{1:t}, \boldsymbol{y}_{1:t})}{p(\boldsymbol{y}_{1:t})} \tag{3.24}$$

$$= \frac{p(\boldsymbol{x}_{1:t-1}, \boldsymbol{y}_{1:t-1})p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})p(\boldsymbol{y}_t|\boldsymbol{x}_t)}{p(\boldsymbol{y}_t, \boldsymbol{y}_{1:t-1})} \tag{3.25}$$

$$= p(\boldsymbol{x}_{1:t-1}|\boldsymbol{y}_{1:t-1})\frac{p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})p(\boldsymbol{y}_t|\boldsymbol{x}_t)}{p(\boldsymbol{y}_t|\boldsymbol{y}_{1:t-1})} \tag{3.26}$$

where

$$p(\boldsymbol{y}_t|\boldsymbol{y}_{1:t-1}) = \int p(\boldsymbol{x}_{t-1}|\boldsymbol{y}_{1:t-1})p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})p(\boldsymbol{y}_t|\boldsymbol{x}_t)d\boldsymbol{x}_{t-1:t}. \tag{3.27}$$

This allows $p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t})$ to be obtained recursively starting from $p(\boldsymbol{x}_0|\boldsymbol{y}_0) = p(\boldsymbol{x}_0)$ and moving forward one step at a time. Similarly, for the marginal distribution $p(\boldsymbol{x}_t|\boldsymbol{y}_{1:t})$, we can find that

$$p(\boldsymbol{x}_t|\boldsymbol{y}_{1:t}) = \frac{p(\boldsymbol{y}_t|\boldsymbol{x}_t)p(\boldsymbol{x}_t|\boldsymbol{y}_{1:t-1})}{p(\boldsymbol{y}_t|\boldsymbol{y}_{1:t-1})} \tag{3.28}$$

where

$$p(\boldsymbol{x}_t|\boldsymbol{y}_{1:t-1}) = \int p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})p(\boldsymbol{x}_{t-1}|\boldsymbol{y}_{1:t-1})d\boldsymbol{x}_{t-1}. \tag{3.29}$$

Commonly, Eqs. (3.28) and (3.29) are referred to update and prediction steps. However, most particle filtering methods do not use these steps, but numerically approximate Eq. (3.26) [10].

As a by-product of the sequential filtering distribution estimation, the marginal likelihood $p(\boldsymbol{y}_{1:t})$ can be easily obtained from

$$p(\boldsymbol{y}_{1:t}) = p(y_1) \prod_{k=2}^{t} p(\boldsymbol{y}_k | \boldsymbol{y}_{1:k-1}) \tag{3.30}$$

When we apply an SSM for continuous emotion recognition, states $\boldsymbol{x}_t$ would represent the unknown affect vector in the V–A(–D) space, and $\boldsymbol{y}_t$ would correspond to feature vectors extracted from the audio signal. When observations of the state variable are available during training, $f()$ and $g()$ can be learned independently which makes the SSM parameter estimation simpler.

3.6 Kalman Filter

As we already mentioned, when state dynamics and measurement functions are linear, such as $f(\boldsymbol{x}) = \boldsymbol{F}\boldsymbol{x}$ and $g(\boldsymbol{x}) = \boldsymbol{G}\boldsymbol{x}$ with matrix parameters $\boldsymbol{F}$ and $\boldsymbol{G}$, an analytic solution can be easily obtained [47]. It can be shown that all distributions of interest are Gaussian:

$$p(\boldsymbol{x}_t | \boldsymbol{y}_{1:t-1}) = \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{\mu}_t^p, \boldsymbol{\Sigma}_t^p) \tag{3.31}$$

$$p(\boldsymbol{x}_t | \boldsymbol{y}_{1:t}) = \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t) \tag{3.32}$$

$$p(\boldsymbol{y}_t | \boldsymbol{y}_{1:t-1}) = \mathcal{N}(\boldsymbol{y}_t; \boldsymbol{G}\boldsymbol{\mu}_t^p, \boldsymbol{S}_t) \tag{3.33}$$

with means and covariances which can be computed from the prediction step

$$\boldsymbol{\mu}_t^p = \boldsymbol{F}\boldsymbol{\mu}_{t-1}, \tag{3.34}$$

$$\boldsymbol{\Sigma}_t^p = \boldsymbol{F}\boldsymbol{\Sigma}_{t-1}\boldsymbol{F}^T + \boldsymbol{\Sigma}_u, \tag{3.35}$$

and the update step

$$\boldsymbol{\mu}_t = \boldsymbol{\mu}_t^p + \boldsymbol{K}_t(\boldsymbol{y}_t - \boldsymbol{G}\boldsymbol{\mu}_t^p), \tag{3.36}$$

$$\boldsymbol{\Sigma}_t = \boldsymbol{\Sigma}_t^p - \boldsymbol{K}_t\boldsymbol{S}_t\boldsymbol{K}_t^T, \tag{3.37}$$

$$\boldsymbol{S}_t = \boldsymbol{G}\boldsymbol{\Sigma}_t^p\boldsymbol{G}^T + \boldsymbol{\Sigma}_v, \tag{3.38}$$

$$\boldsymbol{K}_t = \boldsymbol{\Sigma}_t^p\boldsymbol{G}^T\boldsymbol{S}_t^{-1}. \tag{3.39}$$

This is an optimal filtering solution given that linearity assumption holds and that noises are indeed Gaussian. In practice, however, most often neither is true.

In general, when there are no ground truth observations of the latent state variables, estimation of $\boldsymbol{F}$ and $\boldsymbol{G}$ as well as the noise variances $\boldsymbol{\Sigma}_u$ and $\boldsymbol{\Sigma}_v$ can be done using likelihood maximization via expectation–maximization algorithm [18]. However, when they are available, simple multivariate linear regression can be used to obtain the necessary parameters.

3.7 Particle Filters

Using nonlinear functions for $f()$ and $g()$ would greatly increase the expressiveness of the state-space model, but introduces two problems—what kind of nonlinearity is suitable for the task at hand and how to estimate its parameters. Gaussian processes allow to eliminate the first problem and, when state observations are available, provide solution to the second.

However, filtering with SSM when $f()$ and $g()$ are described by GPs is not straightforward. There are just a few studies on this problem and no common and efficient algorithm exists yet. Here, we utilize a particle filter-based approximation similar to the one proposed in [25].

Particle filters are a class of Monte Carlo algorithms which are based on sampling methods for density function approximations. Thus, the filtering distribution of interest can be approximated by

$$p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t}) \approx \frac{1}{N}\sum_{i=1}^{N}\delta(\boldsymbol{x}_{1:t} - \boldsymbol{x}^{i}_{1:t}) \tag{3.40}$$

where samples, called particles, $\boldsymbol{x}^{i}_{1:t}$, $i = 1, \ldots, N$ are independently drawn from the distribution. However, in practice, often it is impossible to generate samples directly from $p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t})$. The importance sampling (IS) method solves this problem by introducing the so-called *importance distribution*, $q()$, from which samples can be easily obtained, i.e.,

$$\boldsymbol{x}^{i}_{1:t} \sim q(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t}) \tag{3.41}$$

and then we get the approximation as

$$p(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t}) \approx \sum_{i=1}^{N} w^{i}_{t}\delta(\boldsymbol{x}_{1:t} - \boldsymbol{x}^{i}_{1:t}) \tag{3.42}$$

where

$$w^{i}_{t} \propto \frac{p(\boldsymbol{x}^{i}_{1:t}|\boldsymbol{y}_{1:t})}{q(\boldsymbol{x}^{i}_{1:t}|\boldsymbol{y}_{1:t})}. \tag{3.43}$$

For sequential distribution approximation, it would be useful to have an importance density which can be factorized as

$$q(\boldsymbol{x}_{1:t}|\boldsymbol{y}_{1:t}) = q(\boldsymbol{x}_t|\boldsymbol{x}_{1:t-1}, \boldsymbol{y}_{1:t})q(\boldsymbol{x}_{1:t-1}|\boldsymbol{y}_{1:t-1}). \tag{3.44}$$

This way, taking into account Eq. (3.26), the weights become

$$w_t^i \propto \frac{p(\boldsymbol{x}_{1:t-1}^i|\boldsymbol{y}_{1:t-1})p(\boldsymbol{x}_t^i|\boldsymbol{x}_{t-1}^i)p(\boldsymbol{y}_t|\boldsymbol{x}_t^i)}{q(\boldsymbol{x}_t^i|\boldsymbol{x}_{1:t-1}^i, \boldsymbol{y}_{1:t})q(\boldsymbol{x}_{1:t-1}^i|\boldsymbol{y}_{1:t-1})}, \tag{3.45}$$

$$= w_{t-1}^i \frac{p(\boldsymbol{x}_t^i|\boldsymbol{x}_{t-1}^i)p(\boldsymbol{y}_t|\boldsymbol{x}_t^i)}{q(\boldsymbol{x}_t^i|\boldsymbol{x}_{1:t-1}^i, \boldsymbol{y}_{1:t})}. \tag{3.46}$$

Often, it is convenient to simplify the importance distribution from the denominator to $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{y}_{1:t})$ which makes it possible to keep only the current samples $\boldsymbol{x}_t^i$ instead of the whole histories $\boldsymbol{x}_{1:t}^i$. Thus, the sequential importance sampling (SIS) algorithm involves iteration of two main steps: sampling from the importance distribution, $\boldsymbol{x}_t^i \sim q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}^i, \boldsymbol{y}_{1:t})$ and weights update according to Eq. (3.46). However, the SIS algorithm suffers from the so-called "degeneracy" problem where after several iterations, all but few or even single particle will have negligible weights. A common solution is to "resample" with replacement N samples from the $p(\boldsymbol{x}_t|\boldsymbol{y}_{1:t})$ approximated by the pool of particles so that $Pr(\boldsymbol{x}_t^{i*} = \boldsymbol{x}_t^j) = w_t^j$ and then reset the weights to $1/N$.

In many cases, it is convenient to choose the importance distribution to be the SSM's dynamic model

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{y}_{1:t}) = p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}). \tag{3.47}$$

Then, assuming that "resampling" is performed at each step, the weights become simply

$$w_t^i \propto p(\boldsymbol{y}_t|\boldsymbol{x}_t). \tag{3.48}$$

This particular particle filter setting is known as bootstrap filter [17]. In the next section, we describe the bootstrap filter algorithm when Gaussian processes are used as SSM dynamics and measurements models.

3.7.1 Particle Filter with GP

In order to implement a bootstrap filter, it is necessary to be able to sample from $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ and to calculate $p(\boldsymbol{y}_t|\boldsymbol{x}_t)$. They, according to Eqs. (3.22) and (3.23) are Gaussians so it is easy to do it. Means of these distributions are obtained from the GPs output and variances $\boldsymbol{\Sigma}_u$ and $\boldsymbol{\Sigma}_v$ are learned during GP parameter estimation (see Sect. 3.4). One feature of the GP is that its output is actually a Gaussian distribution, and therefore, the output variance will have to be added to the corresponding dimension of $\boldsymbol{\Sigma}_u$ or $\boldsymbol{\Sigma}_v$.

Algorithm 1 provides the steps of the GP particle filter. It is assumed that GP parameters $\boldsymbol{\theta}_x$ and $\boldsymbol{\theta}_y$ for each target dimension are already obtained.

Algorithm 1 GP Particle filter

```
Input: $N, T, \boldsymbol{y}_{1:T}, \boldsymbol{\theta}_x, \boldsymbol{\theta}_y, \boldsymbol{\mu}_0^x, \boldsymbol{\Sigma}_0^x$,     Output: $\hat{\boldsymbol{x}}_{1:T}$
1. for $i = 1, \ldots, N$
2.    $\boldsymbol{x}_0^i \sim \mathcal{N}(\boldsymbol{\mu}_0^x, \boldsymbol{\Sigma}_0^x)$      $\Rightarrow$ initialize particle $i$
3.    $w_0^i = 1/N$          $\Rightarrow$ initialize weight $i$
4. end
5. for $t = 1, \ldots, T$
6.    Resample particles $\boldsymbol{x}_t^i$ according to weights $w_t^i$
7.    for $i = 1, \ldots, N$
8.       $\boldsymbol{f}_t^i, \boldsymbol{\Sigma}_{x,t}^i = GP(\boldsymbol{x}_{t-1}^i | \boldsymbol{\theta}_x)$
9.       $\boldsymbol{x}_t^i \sim \mathcal{N}(\boldsymbol{f}_t^i, \boldsymbol{\Sigma}_{x,t}^i + \boldsymbol{\Sigma}_u)$   $\Rightarrow$ propagate particle $i$
10.      $\boldsymbol{g}_t^i, \boldsymbol{\Sigma}_{y,t}^i = GP(\boldsymbol{x}_t^i | \boldsymbol{\theta}_y)$
11.      $w_t^i = \mathcal{N}(\boldsymbol{y}_t; \boldsymbol{g}_t^i, \boldsymbol{\Sigma}_{y,t}^i + \boldsymbol{\Sigma}_v)$   $\Rightarrow$ update weight $i$
12.   end
13.   $w_t^i = w_t^i / \sum_i w_t^i$      $\Rightarrow$ normalize weights
14.   $\hat{\boldsymbol{x}}_t = \sum_i w_t^i \boldsymbol{x}_t^i$    $\Rightarrow$ estimated mean of $p(\boldsymbol{x}_t | \boldsymbol{y}_{1:t})$
15. end
16. return $\hat{\boldsymbol{x}}_{1:T}$
```

The computational complexity of this algorithm is $\mathcal{O}(NT(d+e)n^2)$, where n is the number of training vectors, because for each particle at each time t algorithm evaluates GP d times in step 8 and e times in step 10.

3.8 System Evaluation

Although from a practical point of view it might be better to have an emotion recognition system which has a categorical output, i.e., recognizes emotions in terms of textual descriptors, here, we assume that the task is to estimate the V–A(–D) point or trajectory in the affect space as accurately as possible. After that, categorical emotions can be easily obtained by affect space clustering. As a performance evaluation measures, we adopt the Pearson correlation coefficient (R) and the root-mean-square error (RMSE) which are widely used in regression tasks.

For the systems implementation, where possible, we used open-source software packages such as the GPML toolbox [43] for Gaussian processes models and the EKF/UKF toolbox [21] for Kalman filtering.

As explained in Sect. 3.4, GP covariance function parameters can be estimated via optimization procedures, but the type of the covariance function as well as the mean function which can be other than zero are system parameters to be set heuristically. The most common choices for covariance function include

- Linear (Lin) with parameter l

$$k(\boldsymbol{x}, \boldsymbol{x}') = (\boldsymbol{x}^T \boldsymbol{x}' + 1)/l^2 \tag{3.49}$$

- Squared exponential (Exp) with parameters σ and l

$$k(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 \exp(-\frac{1}{2l^2}(\boldsymbol{x} - \boldsymbol{x}')^T(\boldsymbol{x} - \boldsymbol{x}')) \tag{3.50}$$

- Matérn (Mat) of degree 3 with parameters σ and l

$$\begin{aligned} k(\boldsymbol{x}, \boldsymbol{x}') &= \sigma^2(1 + r)\exp(-r), \\ r &= \sqrt{\frac{3}{l^2}(\boldsymbol{x} - \boldsymbol{x}')^T(\boldsymbol{x} - \boldsymbol{x}')} \end{aligned} \tag{3.51}$$

As for the mean function, previous experimental studies [36] showed that constant mean may be a better choice.

3.8.1 Speech Emotion Estimation Experiments

The database used in these experiments has been released as part of the Audio/Visual Emotion Challenge and Workshop (AVEC 2014) [57]. It consists of recordings from 84 subjects. There are 100 recordings for training and as many for testing. Duration ranges from 6 to 248 s. Each recording is annotated using three affective dimensions: arousal, valence, and dominance. The AVEC 2014 database includes speech features extracted using the openSMILE toolkit [13]. The feature set consists of 32 energy and spectral related low-level descriptors (LLD) and 6 voicing related LLDs. These features are aggregated in windows of 3 s with 1 s overlap and various statistics such as mean, standard deviation, flatness, skewness, and kurtosis are calculated for each window.

Since the original feature dimension is too high, two subsets of features were used. The first one includes only the LLD means. In the second one, LLDs delta coefficients (ΔLLDs) are included as well. Table 3.1 compares the performance of two GP-based particle filter systems with the KF. Results are given as average over all affect dimensions (V–A–D) and all 100 test samples. We have to note that for considerable number of test samples, the correlation coefficient showed negative values resulting in reduced total average.[1]

[1] These results are not directly comparable with the official AVEC'2014 results because they have been computed using the absolute R value which boosts them to the 0.5–0.6 range. We, however, believe that this approach masks system errors which are the reason for negative R values.

Table 3.1 Comparison between Kalman filter and GP-based particle filters using Linear (Lin) and squared exponential (Exp) covariance functions

Feature set		KF		GP-PF (Lin)		GP-PF (Exp)	
	# dims	R	$RMSE$	R	$RMSE$	R	$RMSE$
LLD	38	0.0350	0.1598	0.1219	0.1303	0.1417	0.0850
LLD+ΔLLD	76	0.0881	0.1691	0.1631	0.1430	0.1642	0.0890

As can be expected, the GP-based particle filter systems outperform the KF significantly. They are able to better capture the complex relationship between acoustic features and emotion representation. Increased data dimension improves the correlation measure R, but also worsens to some extend the root-mean-square error.

3.8.2 Music Emotion Estimation Experiments

For the music emotion estimation experiments, the "MediaEval'2014" database [1] was used. It consists of 1744 clips (each 45 s long) taken at random locations from 1744 different songs. They belong to various genres which can be grouped into the following eight groups: Blues, Electronic, Rock, Classical, Folk, Jazz, Country, and Pop. For training, we selected randomly 500 clips making sure that they are uniformly distributed across genre groups. In a similar way, another 500 clips were selected for testing. Each clip has a static arousal and valence annotation with score on a 9-point scale. Dynamic V–A annotations at 2 Hz rate are also available.

As feature vectors we adopted the features released by the "MediaEval'2014" organizers which include loudness, roughness, hcdf, spectral flux, and zero-crossing rate calculated at the same 2 Hz rate.

Static emotion

In order to obtain a single vector representation of each clip, two level statistics of the original feature vectors were computed. First, mean and standard deviation were taken from sliding windows of 6 vectors, which corresponds to 3 s of signal. Then, same statistics were calculated from the widow level data over the whole clip. Thus, the total dimension of the feature vectors is 20.

For the static emotion estimation case, the "MediaEval'2014" evaluation procedure was followed. It includes the R^2 as well as the RMSE measures. R^2 is commonly used to describe the goodness of fit of a statistical model and is defined as

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \overline{y})^2} \tag{3.52}$$

Table 3.2 Performance comparison between GP and SVM regression-based emotion estimation systems in terms of R^2 and RMSE measures

System	Arousal		Valence		Average	
	R^2	$RMSE$	R^2	$RMSE$	R^2	$RMSE$
SVR (Lin)	0.6801	0.1014	0.3612	0.1002	0.5207	0.1008
SVR (Rbf)	0.6869	0.0997	0.3713	0.0996	0.5291	0.0997
GP (Lin)	0.6747	0.1013	0.3604	0.1013	0.5176	0.1013
GP (Exp)	0.6986	0.0972	0.3594	0.1002	0.5290	0.0987
GP (Mat)	0.6973	0.0969	0.3536	0.1007	0.5255	0.0988

Table 3.3 Dynamic motion emotion recognition results using Kalman filter (KF) and GP-based particle filter (GP-PF) with several different covariance functions

System	Arousal		Valence		Average	
	R	$RMSE$	R	$RMSE$	R	$RMSE$
KF	0.1309	0.2862	0.0864	0.3048	0.1087	0.2955
GP-PF (Lin)	0.2504	0.2184	0.1328	0.2863	0.1916	0.2524
GP-PF (Exp)	0.2753	0.2166	0.1361	0.2718	0.2057	0.2442
GP-PF (Mat)	0.2821	0.2215	0.1295	0.2809	0.2058	0.2512

where y_i are the reference values, $\overline{y}$ is their mean, and $\hat{y}_i$ are the corresponding estimates. R^2 takes values in the range [0, 1][2] with $R^2 = 1$ meaning a perfect data fit.

For comparison, an SVM regression-based system with linear (Lin) and RGB (Rbf) kernel functions was built using the LIBSVM toolkit [4]. The cost parameter was optimized manually using a grid search. The other parameters were set to their defaults. Table 3.2 shows the GPR and SVR results for arousal and valence separately as well as the average score.

There is negligible difference in the GPR and SVR results, especially when exponential covariance and kernel functions are used which is the best case for both models. This to some extend confirms some previous results on the same task [36], but with different features, that GPR shows same or better performance than SVR. On the other hand, selected features may be too simple reveal the full potential of the models.

Dynamic emotion

For dynamic music emotion recognition, the original feature vectors were used to learn the GP parameters for the GP-PF system with various covariance functions. For comparison, KF system was also trained. Table 3.3 summarizes the emotion estimation results using these two systems. As in the speech emotion case, the GP-PF clearly outperforms the KF in both correlation and root-mean-squared error measures.

[2] In practice, it can take values outside this range, which would indicate estimation failure.

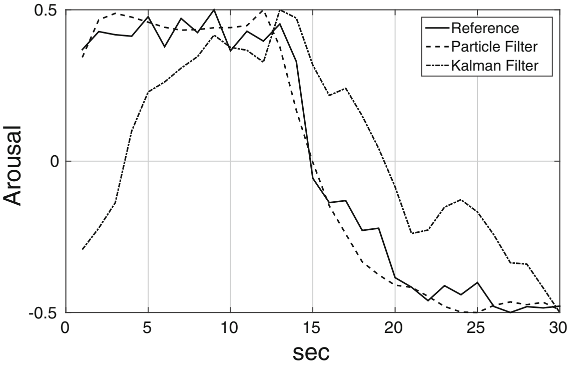

Fig. 3.4 Example of successful estimation of the arousal trajectory. The *solid curve* shows the reference arousal change and the other two are the GP particle filter and KF estimates with correlation coefficient of 0.988 and 0.698, respectively. All *curves* are scaled to fit in the [–0.5, 0.5] range

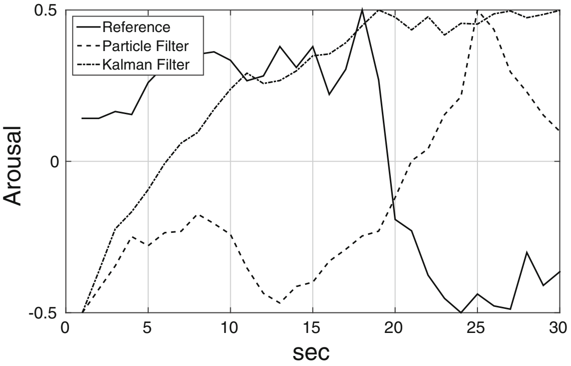

Fig. 3.5 Example of failed arousal trajectory estimation. The GP particle filter result ($R = -0.869$) exhibits opposite behavior, i.e., in contrast to the reference, at the beginning it is low and then goes up, while Kalman filter ($R = -0.460$) fails to capture the change and increases gradually

Examples of successful and failed estimation of the arousal trajectory are presented in Figs. 3.4 and 3.5, respectively. In each figure, there are three curves corresponding to the reference trajectory and the estimated trajectories from the GP particle and Kalman filters. As can be seen, even in the failed case, GP-PF was able to capture the change in the trajectory, although in the opposite direction.

3.9 Discussion and Conclusions

In this chapter, we introduced the Gaussian processes for the task of speech and music emotion recognition. For static emotion, i.e., when single point in the affect space has to be estimated for one utterance or music clip, GP regression can be used. Compared to the current state-of-the-art SVM regression, GPs perform on par or better than SVM as other studies have also shown.

The GP and SVM have many common characteristics. They are both nonparametric, kernel-based models, and their implementation and usage as regressors is very similar. However, GPs are probabilistic Bayesian predictors which in contrast to SVM produce Gaussian distributions as their output. Another GP advantage is the possibility of parameter learning from the training data. On the other hand, SVM provides a sparse solution, i.e., only "support" vectors are used for the inference, which can be a plus when working with large amount of data.

Although the same regression approach can be applied to the case of dynamic emotion recognition, capturing the characteristics of the emotion evolution in time greatly benefits the estimation performance. Thus, state-space models are well suited for such cases. The Kalman filter is a widely used linear state-space model which has been thoroughly studied and is fast and efficient model when data relationships are close to linear. When these relationships are highly nonlinear, however, the KF performance drops significantly. Nonlinear extensions, such as EKF or UKF, lessen the linearity restrictions; however, they require some prior knowledge about the form of the nonlinear functions and often suffer from stability issues.

The main advantage of Gaussian processes is that they do not require any knowledge or assumptions about the data relationships. As shown in Sect. 3.4, the mapping function $f()$ is marginalized out during the inference and can be any function with unlimited degree of nonlinearity. This leads to an improved system performance and as the above evaluations show, can be as much as two times better than the one of a linear system. Compared to other powerful nonlinear models such as Continuous Conditional Random Fields [20] or LSTM neural networks [61], the GP-based system has the advantage of being nonparametric. Thus, there is no need to choose explicit nonlinear (feature) functions as in the case of CRF or to train huge number of parameters (weights) for the NNs. Another advantage is the fully probabilistic nature of the GPs, which allows meaningful interpretation of their outputs. However, as with all nonparametric models, GPs scale poorly and for large tasks are computationally expensive.

Gaussian processes quickly penetrate many research fields and application areas which are currently dominated by the support vector machines or neural networks and show impressive performance on par or often better than the state-of-the-art approaches. Of course, there are some issues with GPs which need further improvement such as high computational complexity and storage requirements, but the current active research on GP theory will hopefully solve these problems in the near future.

References

1. Aljanaki, A., Yang, Y.H., Soleymani, M.: Emotion in music task at MediaEval 2014. In: MediaEval 2014 Workshop. Barcelona, Spain (2014)
2. Arulampalam, M.S., Maskell, S., Gordon, N., Clapp, T.: A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. IEEE Trans. Sig. Process. **50**(2), 174–188 (2002)
3. Barthed, M., Fazekas, G., Sandler, M.: Multidisciplinary perspectives on musicemotion recognition: implications for content and context-based models. In: Proceedings of the 9th Symposium on Computer Music Modeling and Retrieval (CMMR), pp. 492–507 (2012)

4. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**, 27:1–27:27 (2011). http://www.csie.ntu.edu.tw/~cjlin/libsvm
5. Cowie, R., Cornelius, R.R.: Describing the emotional states that are expressed in speech. Speech Commun. **40**(1), 5–32 (2003)
6. Cowie, R., Douglas-Cowie, E., Tsapatsoulis, N., Votsis, G., Kollias, S., Fellenz, W., Taylor, J.: Emotion recognition in human-computer interaction. IEEE Sig. Process. Mag. **18**(1), 32–80 (2001)
7. Csat, L., Opper, M.: Sparse on-line gaussian processes. Neural Comput. **14**(3), 641–668 (2002)
8. Deisenroth, M., Huber, M., Hanebeck, U.: Analytic moment-based gaussian process filtering. In: Proceedings of the 26th Annual International Conference on Machine Learning, ICML '09, pp. 225–232 (2009)
9. Deisenroth, M., Turner, R., Huber, M., Hanebeck, U., Rasmussen, C.: Robust filtering and smoothing with gaussian processes. IEEE Trans. Autom. Control **57**(7), 1865–1871 (2012)
10. Doucet, A., Johansen, A.M.: A tutorial on particle filtering and smoothing: fifteen years later. Handb. nonlinear Filtering **12**, 656–704 (2009)
11. Eerola, T., Lartillot, O., Toiviainen, P.: Prediction of multidimensional emotional ratings in music from audio using multivariate regression models. In: ISMIR, pp. 621–626 (2009)
12. El Ayadi, M., Kamel, M.S., Karray, F.: Survey on speech emotion recognition: features, classification schemes, and databases. Pattern Recognit. **44**(3), 572–587 (2011)
13. Eyben, F., Wöllmer, M., Schuller, B.: Opensmile: the munich versatile and fast open-source audio feature extractor. In: Proceedings of the International Conference on Multimedia, pp. 1459–1462. ACM (2010)
14. Fontaine, J.R., Scherer, K.R., Roesch, E.B., Ellsworth, P.C.: The world of emotions is not two-dimensional. Psychol. Sci. **18**(12), 1050–1057 (2007)
15. Frigola, R., Lindsten, F., Schon, T., Rasmussen, C.: Bayesian inference and learning in gaussian process state-space models with particle MCMC. In: Advances in Neural Information Processing Systems, pp. 3156–3164 (2013)
16. Fu, Z., Lu, G., Ting, K.M., Zhang, D.: A survey of audio-based music classification and annotation. IEEE Trans. Multimedia **13**(2), 303–319 (2011)
17. Gordon, N.J., Salmond, D.J., Smith, A.F.: Novel approach to nonlinear/non-gaussian bayesian state estimation. IEEE Proc. Radar Sig. Process. **140**, 107–113 (1993)
18. Haykin, S. (ed.): Kalman Filtering and Neural Networks. Wiley (2001)
19. Henter, G., Frean, M., Kleijn, W.: Gaussian process dynamical models for nonparametric speech representation and synthesis. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4505–4508 (2012)
20. Imbrasaite, V., Baltrusaitis, T., Robinson, P.: Emotion tracking in music using continuous conditional random fields and relative feature representation. In: 2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), pp. 1–6 (2013). doi:10.1109/ICMEW.2013.6618357
21. Jouni, H., Simo, S.: Optimal filtering with kalman filters and smoothers. manual for matlab toolbox ekf/ukf. Helsinki University of Technology, Department of Biomedical Engineering and Computational Science (2008)
22. Kächele, M., Schels, M., Schwenker, F.: Inferring depression and affect from application dependent meta knowledge. In: Proceedings of the 4th International Workshop on Audio/Visual Emotion Challenge, AVEC '14, pp. 41–48. ACM (2014)
23. Kapoor, A., Grauman, K., Urtasun, R., Darrell, T.: Gaussian processes for object categorization. Int. J. Comput. Vis. **88**(2), 169–188 (2010)
24. Kim, E., Schmidt, E., Mingeco, R., Morton, B., Richardson, P., Scott J. Spec, J., Turnbull, D.: Music emotion recognition: a state of the art review. In: Proceedings of the International Society for Music Information Retrieval Conference (ISMIR), pp. 255–266 (2010)
25. Ko, J., Fox, D.: GP-Bayes filters: bayesian filtering using gaussian process prediction and observation models. Auton. Robots **27**(1), 75–90 (2009)
26. Komatsu, T., Nishino, T., Peters, G., Matsui, T., Takeda, K.: Modeling head-related transfer functions via spatial-temporal gaussian process. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 301–305 (2013)

27. Lawrence, N.: Probabilistic non-linear principal component analysis with gaussian process latent variable models. J. Mach. Learn. Res. **6**, 1783–1816 (2005)
28. Lawrence, N., Moore, A.: Hierarchical gaussian process latent variable models. In: Proceedings of the 24th International Conference on Machine Learning, pp. 481–488. ACM (2007)
29. Lee, H., Pham, P., Largman, Y., Ng, A.Y.: Unsupervised feature learning for audio classification using convolutional deep belief networks. In: Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, A. Culotta (eds.) Advances in Neural Information Processing Systems, vol. 22, pp. 1096–1104 (2009)
30. Li, T., Ogihara, M.: Detecting emotion in music. ISMIR **3**, 239–240 (2003)
31. Lu, D., Sha, F.: Predicting likability of speakers with gaussian processes. In: Proceedings of the 13th Annual Conference of the International Speech Communication Association (2012)
32. Lu, L., Liu, D., Zhang, H.J.: Automatic mood detection and tracking of music audio signals. IEEE Trans. Audio, Speech, Lang. Process. **14**(1), 5–18 (2006)
33. Mariooryad, S., Busso, C.: Correcting time-continuous emotional labels by modeling the reaction lag of evaluators. IEEE Trans. Affect. Comput. (2014). doi:10.1109/TAFFC.2014.2334294
34. Markov, K., Matsui, T.: High level feature extraction for the self-taught learning algorithm. EURASIP J. Audio, Speech, Music Process. **2013**(1), 6 (2013)
35. Markov, K., Matsui, T.: Music genre classification using gaussian process models. In: Proceedings of the IEEE Workshop on Machine Learning for Signal Processing (MLSP) (2013)
36. Markov, K., Matsui, T.: Music genre and emotion recognition using gaussian processes. IEEE Access **2**, 688–697 (2014)
37. Markov, K., Iwata, M., Matsui, T.: Music emotion recognition using gaussian processes. In: Proceedings of the ACM Multimedia 2013 Workshop on Crowdsourcing for Multimedia, CrowdMM. ACM, ACM, Barcelona, Spain (2013)
38. Meng, H., Huang, D., Wang, H., Yang, H., AI-Shuraifi, M., Wang, Y.: Depression recognition based on dynamic facial and vocal expression features using partial least square regression. In: Proceedings of the 3rd ACM International Workshop on Audio/Visual Emotion Challenge, AVEC '13, pp. 21–30. ACM (2013)
39. Nogueiras, A., Moreno, A., Bonafonte, A., Mariño, J.B.: Speech emotion recognition using hidden markov models. In: INTERSPEECH, pp. 2679–2682 (2001)
40. Nwe, T.L., Foo, S.W., De Silva, L.C.: Speech emotion recognition using hidden markov models. Speech Commun. **41**(4), 603–623 (2003)
41. Park, S., Choi, S.: Gaussian process regression for voice activity detection and speech enhancement. In: Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN), pp. 2879–2882 (2008)
42. Park, H., Yun, S., Park, S., Kim, J., Yoo, C.: Phoneme classification using constrained variational gaussian process dynamical system. Adv. Neural Inf. Process. Syst. **25**, 2015–2023 (2012)
43. Rasmussen, C., Nickisch, H.: Gaussian processes for machine learning (GPML) toolbox. J. Mach. Learn. Res. **11**, 3011–3015 (2010)
44. Rasmussen, C., Williams, C.: Gaussian Processes for Machine Learning. Adaptive Computation and Machine Learning. The MIT Press, Cambridge (2006)
45. Russell, J.: A circumplex model of affect. J. Pers. Soc. Psychol. **39**(6), 1161–1178 (1980)
46. Saatçi, Y., Turner, R., Rasmussen, C.: Gaussian process change point models. In: Proceedings 27th Annual International Conference on Machine Learning, pp. 927–934 (2010)
47. Särkkä, S.: Bayesian filtering and smoothing, vol. 3. Cambridge University Press (2013)
48. Scherer, K.R.: What are emotions? and how can they be measured? Soc. Sci. Inf. **44**(4), 695–729 (2005). doi:10.1177/0539018405058216
49. Schmidt, E., Kim, Y.: Prediction of time-varying musical mood distributions using kalman filtering. In: 2010 Ninth International Conference on Machine Learning and Applications (ICMLA), pp. 655–660 (2010)
50. Schmidt, E.M., Kim, Y.E.: Modeling musical emotion dynamics with conditional random fields. In: ISMIR, pp. 777–782 (2011)
51. Schmidt, E.M., Turnbull, D., Kim, Y.E.: Feature selection for content-based, time-varying musical emotion regression. In: Proceedings of the International Conference on Multimedia Information Retrieval, pp. 267–274. ACM (2010)

52. Schuller, B., Rigoll, G., Lang, M.: Hidden markov model-based speech emotion recognition. In: 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03), vol. 2, pp. II–1. IEEE (2003)
53. Snelson, E., Ghahramani, Z.: Sparse gaussian processes using pseudo-inputs. In: Advances in Neural Information Processing Systems, pp. 1257–1264. MIT press, Cambridge (2006)
54. Titsias, M., Lawrence, N.: Bayesian gaussian process latent variable model. In: Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (2010)
55. Turner, R., Deisenroth, M., Rasmussen, C.: State-space inference and learning with gaussian processes. In: Proceedings of the 13th Internatioanl Conference on Artificial Intelligence and Statistics (AISTATS), pp. 868–875 (2010)
56. Tzanetakis, G.: Marsyas submissions to mirex 2007. Music Information Retrieval Evaluation eXchange (MIREX) (2007)
57. Valstar, M., Schuller, B., Smith, K., Almaev, T., Eyben, F., Krajewski, J., Cowie, R., Pantic, M.: AVEC 2014 – 3D dimensional affect and depression recognition challenge. In: Proceedings 4th ACM International Workshop on Audio/visual Emotion Challenge (2014)
58. Wang, J., Fleet, D., Hertzmann, A.: Gaussian process dynamical models for human motion. IEEE Trans.Pattern Anal. Mach. Intell. **30**(2), 283–298 (2008)
59. Weninger, F., Eyben, F., Schuller, B.: On-line continuous-time music mood regression with deep recurrent neural networks. In: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5412–5416 (2014). doi:10.1109/ICASSP.2014.6854637
60. Wollmer, M., Eyben, F., Reiter, S., Schuller, B., Cox, C., Douglas-Cowie, E., Cowie, R.: Abandoning emotion classes-towards continuous emotion recognition with modelling of long-range dependencies. Proc. INTERSPEECH **2008**, 597–600 (2008)
61. Wollmer, M., Kaiser, M., Eyben, F., Schuller, B., Rigoll, G.: LSTM-modeling of continuous emotions in an audiovisual affect recognition framework. Image Vis. Comput. **31**(2), 153–163 (2013)
62. Yang, Y.H., Chen, H.: Prediction of the distribution of perceived music emotions using discrete samples. IEEE Trans. Audio, Speech, Lang. Proces. **19**(7), 2184–2196 (2011)
63. Yang, Y.H., Chen, H.: Machine recognition of music emotion: a review. ACM Trans. Intell. Syst. Technol. **3**(3), 40:1–40:30 (2012)
64. Yang, Y.H., Lin, Y.C., Su, Y.F., Chen, H.: A regression approach to music emotion recognition. IEEE Trans. Audio, Speech, Lang. Proces. **16**(2), 448–457 (2008)

Chapter 4
Topic Modeling for Speech and Language Processing

Jen-Tzung Chien

Abstract In this chapter, we present state-of-art machine learning approaches for speech and language processing with highlight on topic models for structural learning and temporal modeling from unlabeled sequential patterns. In general, speech and language processing involves extensive knowledge of statistical models. We require designing a flexible, scalable, and robust system to meet heterogeneous and nonstationary environments in the era of big data. This chapter starts from an introduction of unsupervised speech and language processing based on factor analysis and independent component analysis. Unsupervised learning is then generalized to a latent variable model which is known as the topic model. The evolution of topic models from latent semantic analysis to hierarchical Dirichlet process, from non-Bayesian parametric models to Bayesian nonparametric models, and from single-layer model to hierarchical tree model is investigated in an organized fashion. The inference approaches based on variational Bayesian and Gibbs sampling are introduced. We present several case studies on topic modeling for speech and language applications including language model, document model, segmentation model, and summarization model.

4.1 Unsupervised Learning in General

Machine learning is generally categorized into supervised learning and unsupervised learning. Supervised learning aims to find a function mapping from observations to their classes, while the unsupervised learning has a broad goal of extracting salient features and discovering structural information from the given data. In the era of big data, an enormous amount of multimedia data is available in Internet which contains speech, text, image, music, video, social network, and many other specialized technical data. It is challenging to extract reliable features and explore latent structure

J.-T. Chien (✉)
Department of Electrical and Computer Engineering, National Chiao Tung University,
Hsinchu, Taiwan
e-mail: jtchien@nctu.edu.tw

G.W. Peters and T. Matsui (eds.), *Modern Methodology and Applications in Spatial-Temporal Modeling*, JSS Research Series in Statistics,
DOI 10.1007/978-4-431-55339-7_4

from these abundant heterogeneous data which are prone to be noisy, mismatched, mislabeled, misaligned, and ill-posed. In addition, the probabilistic learning models may be improperly assumed, overestimated, or underestimated. The issue of model regularization plays an important role in machine learning.

In general, we need some statistical models or tools for modeling, analyzing, searching, recognizing, and understanding real-world data. Such modeling should faithfully represent the uncertainty in model structure and parameters. The noise condition in observation data should be sufficiently reflected. The learning method should be automatic and adaptive to unknown environments and scalable for large amount of data. The uncertainty in heterogeneous data may be expressed by a prior distribution or even a prior process. We aim to construct a learning machine which provides the ways to organize, understand, search, and summarize a large amount of electronic archives automatically. It is attractive to learn such a model in an unsupervised manner which discovers the hidden themes or topics that pervade data collection. This model can be used to annotate any kinds of documents according to their latent themes. With these annotations, we can organize, summarize, search, and predict for future data.

In this chapter, we first survey a series of unsupervised models in Sect. 4.1.1 and address the history and the evolution of different topic models in Sect. 4.1.2. We then focus on topic model based on the latent Dirichlet allocation (LDA) [7] in Sect. 4.1.3. We introduce the inference procedures of LDA including the approximate inference based on variational inference and Gibbs sampling. Section 4.2 addresses the issue of model selection and its solution based on Bayesian nonparametrics (BNP). We briefly survey BNP approaches to topic models including hierarchical Dirichlet process, the nested Dirichlet process and hierarchical Pitman–Yor process in Sect. 4.2. Section 4.3 presents some advances in topic models especially for the applications of speech and language processing including language model in Sect. 4.3.1, document model in Sect. 4.3.2, segmentation model in Sect. 4.3.3, and summarization model in Sect. 4.3.4. Finally, the summary and future direction are provided in Sect. 4.4.

4.1.1 Unsupervised Models

There are many unsupervised learning approaches in the literature which are available to explore latent features of observation data. Principal component analysis (PCA) [30] is known as a statistical procedure that uses an orthogonal transformation to project a set of possibly correlated observation variables $\mathbf{x} \in \mathcal{R}^D$ into a set of linearly uncorrelated variables $\mathbf{z} \in \mathcal{R}^K$ where $K \ll D$. The projected variables are treated as a kind of latent variables which are also called the principal components. The projection is obtained by finding the eigenvalues and the corresponding eigenvectors of the covariance matrix of observation data. The maximal amount of variance is achieved by this linear projection.

Factor analysis (FA) [1] is closely related to PCA but with more domain-specific constraints on the underlying structure. FA uses the regression model for the error terms, while PCA is a descriptive statistical method for the variance. FA incorporates the common factors $\mathbf{z} \in \mathcal{R}^K$ with a factor loading matrix $\mathbf{W} \in \mathcal{R}^{D \times K}$ and a specific factor vector $\boldsymbol{\varepsilon}$ in order to represent the observed data via $\mathbf{x} = \mathbf{W}\mathbf{z} + \boldsymbol{\varepsilon}$. FA is seen as a latent variable model owing to the common factors $\mathbf{z}$ which are unseen in unsupervised learning procedure. FA model is constructed by imposing the following conditions. The common factors and specific factors are distributed by the zero-mean Gaussians with $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_K)$ and $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Psi})$, respectively, where $\mathbf{I}_K$ is an $K \times K$ identity matrix and $\boldsymbol{\Psi}$ is an $D \times D$ diagonal matrix. And, two sets of factors are uncorrelated by $\mathbb{E}[\mathbf{z}\boldsymbol{\varepsilon}^T] = 0$. The latent factors account for common variance in the data. Basically, PCA and FA are solved by eigen-analyzing different covariance matrices and accordingly correspond to the second-order approaches where the principal components in PCA and common factors in FA are Gaussian distributed.

Independent component analysis (ICA) [21] and blind source separation find a set of latent components that are non-Gaussian and mutually independent, i.e., a much stronger assumption. ICA assumes that the observation vector $\mathbf{x}$ is mixed from a set of independent components $\mathbf{z}$ by $\mathbf{x} = \mathbf{W}\mathbf{z}$ where $\mathbf{W}$ is an $D \times K$ mixing matrix. ICA discovers the independent components or latent sources by maximizing the statistical independence or non-Gaussianity of the estimated components which can be measured based on the information-theoretic criterion using mutual information [2] and the higher order statistics using kurtosis [28]. The demixing matrix is estimated by optimizing such a contrast function. The iterative learning solution to ICA is obtained accordingly. In general, ICA is known as a higher order approach to explore independent components for unsupervised learning which produces a tighter or stronger clustering than the uncorrelated components in PCA and the uncorrelated factors in FA.

PCA, FA, and ICA have been successfully developed as the unsupervised approaches to explore latent variables for a number of applications in speech and language processing. For example, PCA was employed in the technique called eigenvoice [33] which assumed that the supervector of acoustic parameters lay in a subspace spanned by a few eigenvectors or latent components. Speaker adapted acoustic model was obtained by estimating the coefficients of a linear expansion over the eigenvectors. FA was adopted to explore the common factors from acoustic features and apply them to build the streamed hidden Markov model [17] where the streaming regularity was governed by the correlation between speech features which was inherent in common factors. FA was also applied for subspace-based speech enhancement [16] where the principal subspace and minor subspace were constructed from common factors and partitioned according to the values of eigenvalues so that the representation of noisy speech was improved for estimation of clean speech. In addition, ICA was exploited for speech recognition where an unsupervised learning was performed to compensate the pronunciation variations in acoustic model via an ICA algorithm [12]. More recently, a convex divergence [15] was designed as a contrast function for ICA algorithm which improved the convergence speed for blind source separation of speech and music signals. In general, the unsupervised learning algorithms using PCA, FA, and ICA are useful to identify salient features or mixture sources **z** from continuous observations **x** based on a whole collection of observation vectors $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$.

4.1.2 Evolution of Topic Models

Latent variable model based on a whole set of continuous observation vectors could be extended to the one based on the *groups* of *discrete* observation data. This extension was originally developed to conduct a latent semantic analysis [22] and build a latent topic model using a set of grouped words from different documents $\mathcal{D} = \{\mathbf{w}_1, \ldots, \mathbf{w}_M\}$ where each document $\mathbf{w}_m = \{w_{mn}\}$ is composed of N_m words and each word is from a dictionary of $\mathcal{V}$ words. Topic model is developed as an unsupervised learning approach to discover latent features or semantic topics which are used to index or annotate the observed text documents. The annotations could be applied for information retrieval and many other applications. Beyond text annotations, the acoustic topic model was proposed for audio tag classification where the acoustic characteristics were represented by discrete symbols for estimation of latent acoustic topics [31]. In [35], topic model was developed to conduct audio mixture analysis where the acoustic data in time–frequency domain were treated as a bag of frequencies to find acoustic topics. A bag of spectrograms was created to build the convolutive topic model with shift-invariance property in both time and frequency. In the fields of computer vision [24], topic model was established as a Bayesian hierarchical model for scene classification where the image of a scene was seen as a collection of local regions or a bag of image features. Each image was automatically annotated with the themes determined by using topic model.

Topic models have been widely developed as a powerful tool for data analysis, annotation, regression, and classification. Figure 4.1 briefly illustrates the evolution and history of topic models. The earliest topic model called latent semantic analysis (LSA) was proposed by Deerwester et al. [22] in 1990. LSA was invented for automatic indexing and retrieval through a singular value decomposition (SVD) over a word-by-document matrix. The latent structure of words and documents was explored

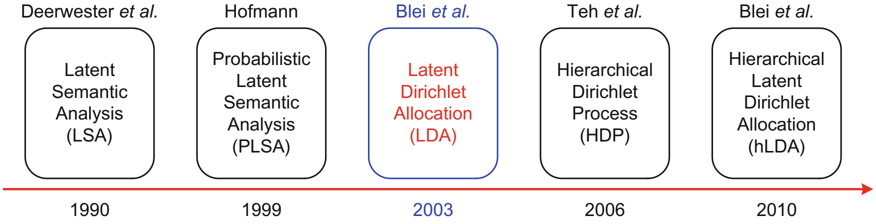
Deerwester et al.
Latent Semantic Analysis (LSA)
1990
Hofmann
Probabilistic Latent Semantic Analysis (PLSA)
1999
Blei et al.
Latent Dirichlet Allocation (LDA)
2003
Teh et al.
Hierarchical Dirichlet Process (HDP)
2006
Blei et al.
Hierarchical Latent Dirichlet Allocation (hLDA)
2010

Fig. 4.1 Evolution and history of topic models

from the decomposed matrices. The next milestone of topic model was achieved by the method called probabilistic latent semantic analysis (PLSA) proposed by Hofmann [27] in 1999. PLSA is a probabilistic framework of LSA where the parameters given latent semantic topics were estimated by maximum likelihood theory using the expectation maximization (EM) algorithm [23]. In 2003, Blei et al. proposed the latent Dirichlet allocation (LDA) [7] for text modeling, document classification, and collaborative filtering. LDA is known as the most popular topic model with the largest citations in the literature. LDA is an extended paradigm from PLSA by introducing a Dirichlet prior to represent the topic probabilities or topic proportions so that the unseen documents could be generalized from Bayesian perspective without greatly increasing the number of parameters. LDA parameters are inferred by maximizing the marginal likelihood over latent topics and topic proportions according to the variational Bayesian (VB) inference [7] and the Gibbs sampling inference [26].

In 2006, Teh et al. proposed the hierarchical Dirichlet process (HDP) [39] which relaxes the constraint of LDA that the number of topics should be known and fixed in topic model. A Bayesian nonparametric (BNP) approach was developed as an expressive probabilistic representation with less assumption-laden approach to inference. The prior process is introduced to conduct a flexible Bayesian learning with infinite topic representation. HDP was implemented by the stick-breaking process and inferred by using the Gibbs sampling procedure. However, topic models based on LDA and HDP assume that topics are independent. To incorporate the topic correlation or even the topic hierarchy into topic model, Blei et al. proposed the nested Chinese restaurant process (nCRP) and built the hierarchical LDA (hLDA) for document representation [3, 4] in 2010. Gibbs sampling was applied to sample a tree path and then sample a tree layer to represent a word w_{mn} in a target document $\mathbf{w}_m$. The tree layers in a tree path reflect different degrees of sharing in the estimated topic parameters. In this chapter, we focus on the topic model based on LDA and its inference procedures using VB-EM algorithm and Gibbs sampling in Sect. 4.1.3. The extensions to HDP and nCRP will be addressed in Sect. 4.2. Some advances in topic model for speech and language processing are described in Sect. 4.3. First of all, we address the early works on topic model based on LSA and PLSA.

Latent Semantic Analysis

Latent semantic analysis (LSA) [22] goes beyond the lexical level from a collection of text documents $\mathcal{D}$ and aims to reveal the latent semantic structure in low-dimensional data space. This algorithm first constructs a word-by-document matrix $\mathbf{W}$ with the element ω_{vm} representing the number of times of a word v occurring in document m. This $\mathcal{V} \times M$ matrix is then decomposed and approximated using the SVD method to produce $\mathbf{W} \approx \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$ where $\boldsymbol{\Sigma}$ is an $K \times K$ diagonal matrix with a reduced dimension $K < \min(\mathcal{V}, M)$, $\mathbf{U}$ is an $\mathcal{V} \times K$ matrix whose columns are the first K eigenvectors derived from word-by-word correlation matrix $\mathbf{W}\mathbf{W}^\top$, and $\mathbf{V}$ is an $M \times K$ matrix whose columns are the first K eigenvectors derived from the document-by-document correlation matrix $\mathbf{W}^\top\mathbf{W}$. Each column of $\boldsymbol{\Sigma}\mathbf{V}^\top$ characterizes the location of a particular document in the reduced K-dimensional semantic topic space. Based on this property, we measure the similarity between two documents m and m' by projecting the corresponding document vectors $\mathbf{v}_m$ and $\mathbf{v}_{m'}$ into the semantic topic space as $\boldsymbol{\Sigma}\mathbf{v}_m$ and $\boldsymbol{\Sigma}\mathbf{v}_{m'}$ and then calculating the cosine similarity between two K-dimensional vectors $\cos(\boldsymbol{\Sigma}\mathbf{v}_m, \boldsymbol{\Sigma}\mathbf{v}_{m'})$. Using this similarity, we accordingly conduct the information retrieval by finding the similarity between a query q and a reference document d_m based on $\cos(\boldsymbol{\Sigma}\mathbf{v}_q, \boldsymbol{\Sigma}\mathbf{v}_m)$ where the query vector in semantic topic space is calculated by $\mathbf{v}_q = \boldsymbol{\Sigma}^{-1}\mathbf{U}^\top\boldsymbol{\omega}_q$ using the vector $\boldsymbol{\omega}_q$ consisting of the number of occurrences of different words in query q.

Probabilistic Latent Semantic Analysis

LSA model was established by applying the SVD method which minimizes the approximation error by using the decomposed matrices. LSA is seen as a nonparametric method where there is no probabilistic distribution assumed in this topic model. The system performance and model generalization are constrained. Hofmann [27] introduced a probabilistic solution to LSA based on maximum likelihood (ML) theory. Figure 4.2 shows the graphical representation of the probabilistic LSA (PLSA). PLSA is seen as an aspect model which represents the co-occurrence data of words (denoted by w_n) and documents (denoted by d_m) associated with a topic or latent variable $z_n = k$. The generative model for co-occurrence w_n and d_m is expressed by the joint probability $p(w_n, d_m)$. Under this latent variable model, the joint likelihood function of training data $\mathcal{D} = \{w_n, d_m\}$ is formed by

$$p(\mathcal{D}|\Theta) = \prod_{m=1}^{M}\prod_{n=1}^{N_m}\sum_{k=1}^{K} p(w_n|z_n = k)p(z_n = k|d_m)p(d_m) \tag{4.1}$$

where PLSA parameters $\Theta = \{p(w_n = v|z_n = k), p(z_n = k|d_m)\}$ consist of two sets of topic-based multinomials with the number of parameters given by $\mathcal{V}K + KM$. ML estimation of PLSA parameters is performed by maximizing Eq. (4.1) with respect to Θ. However, such ML estimation suffers from the incomplete data problem due to the missing variable $z_n = k$ or simply z_k. EM algorithm is applied to resolve this problem by alternatively and iteratively performing the E step which calculates the auxiliary function $Q(\Theta'|\Theta) = \mathbb{E}_{(Z)}[\log p(\mathcal{D}, Z|\Theta')|\mathcal{D}, \Theta]$ and then the M step which maximizes $Q(\Theta'|\Theta)$ with respect to Θ'. Here, the auxiliary function $Q(\Theta'|\Theta)$ is calculated as an expectation of log likelihood function using new parameter estimate Θ' given the current estimate Θ. The expectation is performed over latent variables $Z = \{z_k\}$. After EM iterations, ML PLSA parameters are converged at the mode $\hat{\Theta}$.

By expanding the joint probability $p(w_v, d_m)$ where w_v implies $w_n = v$, we may bridge the connection between PLSA and LSA by defining $\mathbf{U} = \{p(w_v|z_k)\}_{v,k}$, $\mathbf{V} = \{p(d_m|z_k)\}_{m,k}$ and $\boldsymbol{\Sigma} = \text{diag}\{p(z_k)\}_k$. And, a matrix with likelihood entries is formed by $\mathbf{P} = \{p(w_v, d_m)\}_{v,m} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$. Basically, PLSA assumes that the estimated parameters for different topics are nonnegative, while the elements of the decomposed matrices in LSA, estimated from the eigen-analysis, are not guaranteed to be nonnegative. LSA may violate the nonnegative nature of word count. In addition, the Dirichlet priors for multinomial parameters $\{p(w_v|z_k)\}$ and $\{p(z_k|d_m)\}$ were introduced to conduct the maximum a posteriori (MAP) estimation with constraints $\sum_v p(w_v|z_k) = 1$ and $\sum_k p(z_k|d_m) = 1$. MAP PLSA model was developed for an adaptive topic model which adapted the PLSA parameters to fit the topic-changing domains [18].

4.1.3 Latent Dirichlet Allocation

There are three issues in PLSA topic model. First, the PLSA parameters estimated by ML theory are prone to be overtrained. Model generalization is not assured. Second, PLSA could not model the unseen documents. Third, the number of parameters is proportionally increased by the number of topics K and the number of documents M. To overcome these issues, latent Dirichlet allocation (LDA) [7] was proposed by introducing a Dirichlet prior with hyperparameters $\boldsymbol{\alpha}$ for document-dependent topic proportions $\boldsymbol{\theta}_m = \{p(z_k|d_m)\}$ over K topics as seen in the graphical representation in Fig. 4.2b. Each document is treated as a "random mixture" over latent topics. Topic model is generalized to unseen data through the shared prior distribution $p(\boldsymbol{\theta}_m|\boldsymbol{\alpha})$ with a common hyperparameter $\boldsymbol{\alpha} = \{\alpha_k\}$ where $\alpha_k > 0$. Model construction using LDA is described as follows:

1. For each document $\mathbf{w}_m = \{w_{mn}|n = 1, \ldots, N_m\}$
   a. Draw topic proportions $\boldsymbol{\theta}_m \sim \text{Dir}(\boldsymbol{\alpha})$
   b. For each word w_{mn}
      i. Choose a topic by $z_{mn} = k \sim \text{Mult}(\boldsymbol{\theta}_m)$
      ii. Choose a word by $w_{mn} = v|z_{mn} = k, \boldsymbol{\beta} \sim \text{Mult}(\beta_{vk})$

Here, $\boldsymbol{\beta} = \{\beta_{vk}\} = \{p(w_v|z_k)\}$ denotes the $\mathcal{V} \times K$ multinomial matrix consisting of conditional multinomials β_{vk} for different words under different topics. There are two latent variables in LDA including topic proportions $\boldsymbol{\theta} = \{\theta_{mk}\}$ and topic assignments $\mathbf{z} = \{z_{mn}\}$. LDA parameters $\{\boldsymbol{\alpha}, \boldsymbol{\beta}\}$ are estimated by maximizing the marginal likelihood over two latent variables

$$p(\mathcal{D}|\boldsymbol{\alpha}, \boldsymbol{\beta}) = \prod_{m=1}^{M} \int p(\boldsymbol{\theta}_m|\boldsymbol{\alpha}) \prod_{n=1}^{N_m} \sum_{k=1}^{K} p(z_{mn} = k|\boldsymbol{\theta}_m) p(w_{mn}|z_{mn} = k, \boldsymbol{\beta}) d\boldsymbol{\theta}_m. \tag{4.2}$$

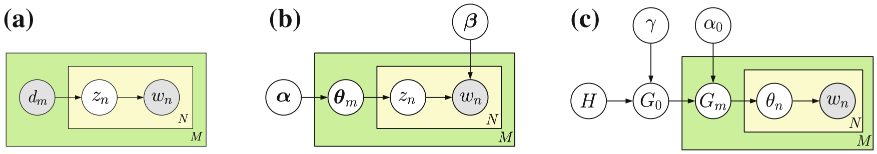

Fig. 4.2 Graphical representation for **a** PLSA, **b** LDA and **c** HDP

We can see that the number of parameters in LDA is $\mathcal{V}K + K$ which is much smaller than $\mathcal{V}K + KM$ for PLSA. A shared $\boldsymbol{\alpha}$ for all documents in LDA can be used to generalized to unseen data and keep a compact model complexity.

However, the exact solution to model inference based on Eq. (4.2) does not exist due to the coupling of multiple latent variables $\boldsymbol{\theta}$ and $\mathbf{z}$ in posterior distribution $p(\boldsymbol{\theta}, \mathbf{z}|\mathcal{D}, \boldsymbol{\alpha}, \boldsymbol{\beta})$. In what follows, we introduce the approximate inference procedures based on variational Bayesian and Gibbs sampling.

Inference by Variational Bayesian

Variational Bayesian (VB) inference is known as the deterministic approach to infer model parameters through a convexity-based variational procedure which is implemented by using the Jensen's inequality. VB aims to resolve the intractable posterior distribution $p(\boldsymbol{\theta}, \mathbf{z}|\mathcal{D}, \boldsymbol{\alpha}, \boldsymbol{\beta})$ by using a factorizable variational distribution

$$q(\boldsymbol{\theta}, \mathbf{z}|\boldsymbol{\gamma}, \boldsymbol{\phi}) = \prod_{m=1}^{M} q(\boldsymbol{\theta}_m|\boldsymbol{\gamma}_m) \prod_{n=1}^{N_m} q(z_{mn}|\phi_{mn}) \tag{4.3}$$

through maximizing a lower bound of the logarithm of marginal likelihood $\mathcal{L}(\boldsymbol{\gamma}, \boldsymbol{\phi}; \boldsymbol{\alpha}, \boldsymbol{\beta})$ where $\boldsymbol{\gamma}$ and $\boldsymbol{\phi}$ denote the variational Dirichlet and multinomial parameters, respectively. We have the relation

$$\log p(\mathcal{D}|\boldsymbol{\alpha}, \boldsymbol{\beta}) = \mathcal{L}(\boldsymbol{\gamma}, \boldsymbol{\phi}; \boldsymbol{\alpha}, \boldsymbol{\beta}) + \mathrm{KL}(q(\boldsymbol{\theta}, \mathbf{z}|\boldsymbol{\gamma}, \boldsymbol{\phi}) \| p(\boldsymbol{\theta}, \mathbf{z}|\mathcal{D}, \boldsymbol{\alpha}, \boldsymbol{\beta})). \tag{4.4}$$

Therefore, maximizing the lower bound $\mathcal{L}(\boldsymbol{\gamma}, \boldsymbol{\phi}; \boldsymbol{\alpha}, \boldsymbol{\beta})$ with respect to variational parameters $\{\boldsymbol{\gamma}, \boldsymbol{\phi}\}$ is equivalent to estimating the new variational distribution $q(\boldsymbol{\theta}, \mathbf{z}|\hat{\boldsymbol{\gamma}}, \hat{\boldsymbol{\phi}})$ which is closest to the true posterior $p(\boldsymbol{\theta}, \mathbf{z}|\mathcal{D}, \boldsymbol{\alpha}, \boldsymbol{\beta})$ with the smallest Kullback–Leibler divergence $\mathrm{KL}(\cdot\|\cdot)$. Basically, finding the approximate posterior distribution $q(\boldsymbol{\theta}, \mathbf{z}|\hat{\boldsymbol{\gamma}}, \hat{\boldsymbol{\phi}})$ is seen as an expectation step (also called VB-E step) in VB-EM algorithm. Then, in VB-M step, we upgrade the lower bound using the new variational parameters $\mathcal{L}(\hat{\boldsymbol{\gamma}}, \hat{\boldsymbol{\phi}}; \boldsymbol{\alpha}, \boldsymbol{\beta})$ and maximize the updated lower bound with respect to the model parameters $\{\boldsymbol{\alpha}, \boldsymbol{\beta}\}$ so as to estimate the new LDA parameters $\{\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}\}$. VB-EM algorithm is run to upgrade the variational distribution and increase the lower bound, and accordingly improve the marginal likelihood using the continuously updated model parameters. LDA parameters are finally estimated with convergence after VB-EM iterations as detailed in [7]. Notably, since the Dirichlet distribution in LDA is seen as the conjugate prior for the multinomial likelihood of the observed words, the solutions to variational Dirichlet parameter vector $\hat{\boldsymbol{\gamma}} = \{\hat{\gamma}_k\}$, variational multinomial parameters $\hat{\boldsymbol{\phi}} = \{\hat{\phi}_{nk}\}$, and conditional multinomial distributions $\hat{\boldsymbol{\beta}} = \{\hat{p}(w_v|z_k)\}$ are derived in the closed form. Only the solution to Dirichlet model parameters $\hat{\boldsymbol{\alpha}}$ is calculated by the Newton–Raphson algorithm. Importantly, the variational Dirichlet parameters $\hat{\boldsymbol{\gamma}}$ are seen as the surrogate of the Dirichlet model parameters $\hat{\boldsymbol{\alpha}}$ which sufficiently reflect the topic proportions $\boldsymbol{\theta}$. The variational lower bound $\mathcal{L}(\hat{\boldsymbol{\gamma}}, \hat{\boldsymbol{\phi}}; \boldsymbol{\alpha}, \boldsymbol{\beta})$ is treated as a tractable surrogate for the intractable log marginal likelihood $\log p(\mathcal{D}|\boldsymbol{\alpha}, \boldsymbol{\beta})$.

Inference by Gibbs Sampling

Griffiths and Steyvers [26] presented a Markov chain Monte Carlo (MCMC) inference solution to LDA topic model. MCMC provides another realization of approximate inference which fulfills the full Bayesian perspective. Different from the deterministic approximation based on VB, MCMC is known as a stochastic approximation. MCMC uses the numerical sampling computation rather than solving the integral and expectation analytically. MCMC provides highly flexible models without limitation of any specific distribution and can be used to infer the infinite-dimensional topic models based on HDP and nCRP which will be addressed in Sect. 4.2. MCMC is computationally expensive without convergence guaranteed. The asymptotically exact solution can be found. However, VB never generates the exact solution but guarantees convergence and fast implementation. The strengths and weaknesses using VB and MCMC are complementary.

Gibbs sampling is a simple and widely applicable realization of MCMC algorithm. Every single state of a Markov chain is seen as an outcome of a latent variable in a variable sequence $\mathbf{z} = \{z_1, \ldots, z_K\}$. Each step of the Gibbs sampling procedure replaces the value for one of the variables z_k by a value drawn from the distribution of that variable conditioned on the values of the remaining states $\mathbf{z}_{-k}$ (i.e., $\mathbf{z} = \{z_k, \mathbf{z}_{-k}\}$) including the preceding states $z_{1:(k-1)}^{(\tau+1)}$ in new iteration $\tau + 1$ and the succeeding states $z_{k+1:K}^{(\tau)}$ in current iteration τ

$$z_k^{(\tau+1)} \sim p\left(z_k \middle| z_{1:(k-1)}^{(\tau+1)}, z_{(k+1):K}^{(\tau)}\right). \tag{4.5}$$

The sampling procedure is repeated with T iterations by cycling through the variables in a particular order or in a random order with some distribution.

Using Gibbs sampling procedure for LDA, we sample the topic assignment z_k according to the predictive posterior distribution $p(z_{mn} = k|\mathbf{z}_{-(mn)}, \mathcal{D})$ given by

$$\begin{aligned} p(w_{mn} &= v|z_{mn} = k, \mathbf{z}_{-(mn)}, \mathbf{w}_{-(mn)})p(z_{mn} = k|\mathbf{z}_{-(mn)}) \\ &= \mathbb{E}[\beta_{vk}|\mathbf{z}_{-(mn)}, \mathbf{w}_{-(mn)}]\,\mathbb{E}[\theta_{mk}|\mathbf{z}_{-(mn)}] \\ &= \frac{\eta+\sum_{m=1}^{M}\sum_{i=1,i\neq n}^{N_m} z_{mi}^k w_{mi}^v}{\mathcal{V}\eta+\sum_{m=1}^{M} N_m-1}\;\frac{\alpha+\sum_{i=1,i\neq n}^{N_m} z_{mi}^k}{K\alpha+N_m-1} \end{aligned} \tag{4.6}$$

where η is the Dirichlet parameter of β_{vk}, $w_{mn} = v$ is expressed by $w_{mn}^v = 1$ and $z_{mn} = k$ is written by $z_{mn}^k = 1$. Here, we use the property of predictive multinomial

$$p(z_k|\mathbf{z}_{-k}) = \int p(z_k|\theta)p(\theta|\mathbf{z}_{-k})d\theta = \mathbb{E}[\theta|\mathbf{z}_{-k}]. \tag{4.7}$$

With a set of samples of topic assignments for different words and documents $\mathbf{z} = \{z_{mn}\}$, we can estimate the multinomial parameters for topics $\hat{\boldsymbol{\theta}} = \{\hat{\theta}_{mk}\}$ and for words under different topics $\hat{\boldsymbol{\beta}} = \{\hat{\beta}_{vk}\}$ by using the expected value of multinomials as given in Eq. (4.7).

4.2 Bayesian Nonparametric Learning

Topic models based on LSA, PLSA, and LDA are constructed as a finite-dimensional mixture representation which assumes that (1) the number of topics is fixed and (2) different topics are independent. These assumptions constrain the flexibility and performance of topic model in presence of scalable data under heterogeneous condition. The topic models based on HDP [39] and nCRP [3, 4] were accordingly developed to resolve these two assumptions through Bayesian nonparametric (BNP) learning. In general, BNPs are used to characterize a big parameter space and construct the probability measure over this space. We setup a stochastic prior process on probability distributions which is a measure on function space. A Bayesian model on an infinite-dimensional parameter space is established. BNPs allow data representation to grow structurally when more data are collected. Number of clusters or topics (or model structure) is unknown a priori. In what follows, we describe BNP learning based on the Dirichlet process and the Pitman–Yor (PY) process. We then introduce the topic models produced by HDP and nCRP and the language model drawn from the hierarchical PY (HPY) process [38].

Dirichlet Process

Dirichlet process (DP) is realized to find the flexible data partitions and provide the nonparametric prior over the number of topics K via a distribution over probability measures $G \sim \text{DP}(\alpha_0, G_0)$ where $\alpha_0 > 0$ is a strength parameter and G_0 is a base measure over a probability space Ω with any partitions $A_1, \ldots, A_k \in \Omega$ as

$$(G(A_1), \ldots, G(A_k)) \sim \text{Dir}(\alpha_0 G_0(A_1), \ldots, \alpha_0 G_0(A_k)) \tag{4.8}$$

which is an infinite-dimensional generalization of Dirichlet distribution. The topic-based representation of a single document $\mathbf{w}$ is formed by drawing the probability measure θ_n for each word w_n using an DP G. The predictive multinomial for new parameter θ_{n+1} in partition A given the previous ones $\theta_{1:n}$ is obtained by Eq. (4.7) as

$$\begin{aligned} p(\theta_{n+1} \in A|\theta_{1:n}, \alpha_0, G_0) = \mathbb{E}[G(A)|\theta_{1:n}] &= \sum_{i=1}^{n} \frac{1}{\alpha_0 + n}\delta_{\theta_i}(A) + \frac{\alpha_0}{\alpha_0 + n} G_0(A) \\ &= \sum_{k=1}^{K} \frac{n_k}{\alpha_0 + n}\delta_{\phi_k}(A) + \frac{\alpha_0}{\alpha_0 + n} G_0(A) \end{aligned} \tag{4.9}$$

where $\phi_1, \ldots, \phi_K$ denote the distinct values from $\theta_{1:n}$. DP can be realized by using the stick-breaking process (SBP) and the Chinese restaurant process (CRP). Equation (4.9) can be explained as a metaphor of CRP with the existing K tables (or clusters). New customer θ_{n+1} enters a restaurant and chooses an occupied table k with probability $\frac{n_k}{\alpha_0+n}$ or a new table with probability $\frac{\alpha_0}{\alpha_0+n}$ where n_k denotes the number of customers who have seated in table ϕ_k. On the other hand, using the

SBP, we randomly break a unit-length stick into two segments and find the proportions $\boldsymbol{\pi} = \{\pi_k\} \sim \text{GEM}(\alpha_0)$ with constraint $\sum_k \pi_k = 1$ using the GEM distribution through a process of drawing beta variables $\{\pi'_k\}$. An DP, $G \sim \text{DP}(\alpha_0, G_0)$, is implemented by

$$\phi_k \sim G_0, \quad \pi'_k|\alpha_0 \sim \text{Beta}(1, \alpha_0), \quad \pi_k = \pi'_k \prod_{j=1}^{k-1}(1 - \pi'_j), \quad G = \sum_{k=1}^{\infty} \pi_k \delta_{\phi_k}. \quad (4.10)$$

Pitman–Yor Process

Pitman–Yor (PY) process [34], $\text{PY}(d_0, \alpha_0, G_0)$, is expressed as a three-parameter distribution over distributions where $0 \le d_0 < 1$ is a discount parameter which characterizes the power-law distribution in natural language, namely *many unique words are observed and most of them rarely*. Basically, d_0 controls the asymptotic growth of the number of unique words, while α_0 controls the overall number of unique words. When $d_0 = 0$, this PY process reverts to $\text{DP}(\alpha_0, G_0)$. When $d_0 > 0$, PY process draws a longer tail probability measure than the DP. Let $G_\emptyset = [G_\emptyset(w)]_{w \in \Omega_v}$ represent the vector of unigrams with empty context $\emptyset$ and $G_0(w) = \frac{1}{\mathcal{V}}$. The predictive unigram probability of a new word w is calculated by

$$\begin{aligned} p(w|\mathcal{D}, d_0, \alpha_0) &= \sum_{k=1}^{m.} \frac{n_k - d_0}{\alpha_0 + n.} \delta_{\phi_k}(w) + \frac{\alpha_0 + d_0 m.}{\alpha_0 + n.} G_0(w) \\ &= \frac{n_w - d_0 m_w}{\alpha_0 + n.} + \frac{\alpha_0 + d_0 m.}{\alpha_0 + n.} \frac{1}{|\mathcal{V}|} \end{aligned} \quad (4.11)$$

where $n. = \sum_k n_k$ is the total number of customers in different tables, m_w is the number of occupied tables labeled by word w, and $m. = \sum_w m_w$ is calculated over different words. Physical meaning of discounting scheme using d_0 is obvious in both terms of right-hand side of Eq. (4.11). The number of occurrences of the seen words is discounted and distributed for those of the unseen words in case of $n_w = m_w = 0$.

Hierarchical Dirichlet Process

HDP deals with the mixed membership representation for multiple documents or grouped data where each document $\mathbf{w}_m$ is associated with a mixture model which is drawn from an DP by $G_m \sim \text{DP}(\alpha_0, G_0)$. Data in different documents share a global mixture model drawn from a global DP by $G_0 \sim \text{DP}(\gamma, H)$ as seen in Fig. 4.2c. HDP can be expressed by the mixture models with the shared atoms $\{\phi_k\}_{k=1}^{\infty}$ but different weights or proportions $\boldsymbol{\beta} = \{\beta_k\}_{k=1}^{\infty}$ and $\boldsymbol{\pi}_m = \{\pi_{mk}\}_{k=1}^{\infty}$ so that we have $G_0 = \sum_k \beta_k \delta_{\phi_k}$ and $G_m = \sum_k \pi_{mk} \delta_{\phi_k}$ with constraints $\sum_k \beta_k = \sum_k \pi_{mk} = 1$. The HDP topic model is accordingly established through a stick-breaking process based on an GEM distribution

$$\begin{aligned} &\boldsymbol{\beta}|\gamma \sim \text{GEM}(\gamma), \quad \boldsymbol{\pi}_m|\alpha_0, \boldsymbol{\beta} \sim \text{DP}(\alpha_0, \boldsymbol{\beta}), \quad z_{mn}|\boldsymbol{\pi}_m \sim \text{Mult}(\boldsymbol{\pi}_m) \\ &\phi_k|H \sim H, \qquad w_{mn}|z_{mn}, \{\phi_k\}_{k=1}^{\infty} \sim \text{Mult}(\phi_{z_{mn}}) \end{aligned} \quad (4.12)$$

where the infinite-dimensional topic multinomials $\{\phi_k\}_{k=1}^{\infty}$ are incorporated. Importantly, a two-stage SBP was implemented to connect the relation between the topic proportions for words in corpus level $\boldsymbol{\beta}$ and in document level $\boldsymbol{\pi}_m$ [39].

The Nested Chinese Restaurant Process

The topic model based on LDA assumes that different topics are independent. To relax this restriction, the correlated topic model (CTM) [6] was proposed by introducing a multivariate logistic Gaussian distribution as a prior distribution to replace the Dirichlet prior distribution for topic proportions $\boldsymbol{\theta}$ in Sect. 4.1.3. Logistic Gaussian adopts a softmax transformation to impose the condition of summing the proportions to be one. The non-diagonal elements of the corresponding covariance matrix induce the dependencies between the transformed topic multinomials. However, CTM fixed the number of topics and did not consider the topic hierarchy.

Blei et al. proposed the nested Chinese restaurant process (nCRP) [4] and built the hierarchical LDA [3] to explore different levels of aspects for topic modeling without fixing the model structure. Figure 4.3a depicts an infinitely branching tree structure for nCRP representation of words (denoted by blue circles) and document (denote by yellow rectangle). Thick arrows denote a tree path $\mathbf{c}_m$ drawn from nine words of a document $\mathbf{w}_m$ or d_m. Each word w_{mn} is assigned by a topic parameter ϕ_k at a tree node along $\mathbf{c}_m$ using topic proportions $\boldsymbol{\pi}_m$.

1. For each node k in the infinite tree
   a. Draw a topic parameter $\phi_k | H \sim H$
2. For each document $\mathbf{w}_m = \{w_{mn} | n = 1, \ldots, N_m\}$
   a. Draw a tree path by $\mathbf{c}_m \sim \text{nCRP}(\alpha_0)$
   b. Draw topic proportions over layers of $\mathbf{c}_m$ by a stick-breaking process $\boldsymbol{\pi}_m \sim \text{GEM}(\gamma)$
   c. For each word w_{mn}
      i. Choose a layer or a topic by $z_{mn} = k \sim \boldsymbol{\pi}_m$

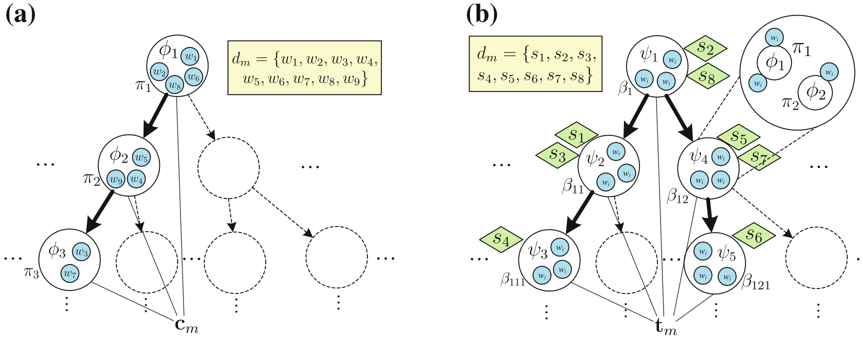

Fig. 4.3 Graphical representation for **a** nCRP and **b** sentence-based nCRP

ii. Choose a word based on topic $z_{mn} = k$ by $w_{mn}|z_{mn}, \mathbf{c}_m, \{\phi_k\}_{k=1}^{\infty} \sim \text{Mult}\left(\phi_{\mathbf{c}_m(z_{mn})}\right)$

In implementation of nCRP, Gibbs sampling is applied to sample the posterior tree path and word topic $\{\mathbf{c}_m, z_{mn}\}$ for M documents in $\mathcal{D} = \{\mathbf{w}_m\}$ with N_m words in each document according to the individual posterior probabilities of $\mathbf{c}_m$ and z_{mn} given $\mathcal{D}$ and the current values of all the other latent variables, i.e., $p(\mathbf{c}_m|\mathbf{c}_{-m}, \mathcal{D}, \mathbf{z}, \alpha_0, H)$ and $p(z_{mn}|\mathcal{D}, \mathbf{z}_{-(mn)}, \mathbf{c}_m, \gamma, H)$. Again, "−" denotes the self-exception. The tree path $\mathbf{c}_m$ is selected for each customer or document $\mathbf{w}_m$. The tree nodes along $\mathbf{c}_m$ imply a series of visits of this customer to different restaurants in different days. A hierarchical topic model is constructed with different degrees of sharing from root node (broad topic) to leaf nodes (specific topics).

Hierarchical Pitman–Yor Process

Teh [38] presented an BNP learning for language model (LM) to deal with the issue of data sparseness in higher order n-gram model. To cope with this issue, conventional method using the Kneser-Ney (KN) LM smoothing [32] was empirically developed by discounting the number of occurrences for seen n-gram events and distributing these occurrences to unseen n-gram events. Such discounting mechanism reflects the power-law property of natural language and does improve n-gram modeling. Interestingly, KN-LM can be interpreted as a hierarchical Bayesian framework according to the hierarchical Pitman–Yor (HPY) process. Similar to the style of hierarchical generative process based on HDP, HPY process conducts a hierarchical generation of PY processes to draw the discounted n-gram probabilities $p(w_i|w_{i-n+1}^{i-1})$ where the predictive probability of next word $w = w_i$ is based on a history or a context vector consisting of previous $n - 1$ words $\mathbf{u} = \{w_{i-n+1}, \ldots, w_{i-1}\} \triangleq w_{i-n+1}^{i-1}$. The HPY process is expressed by a recursive formula where the PY process $G_{\mathbf{u}}$ is formed with a nested base measure $G_{\pi(\mathbf{u})}$ of backoff context $\pi(\mathbf{u})$, which is also an PY process given by a base measure of doubly backoff context $\pi(\pi(\mathbf{u}))$ in a much lower order model. We have

$$G_{\mathbf{u}} \sim \text{PY}(d_{|\mathbf{u}|}, \alpha_{|\mathbf{u}|}, G_{\pi(\mathbf{u})}), \qquad G_{\pi(\mathbf{u})} \sim \text{PY}(d_{|\pi(\mathbf{u})|}, \alpha_{|\pi(\mathbf{u})|}, G_{\pi(\pi(\mathbf{u}))}) \tag{4.13}$$

where the parameters $d_{|\mathbf{u}|}$ and $\alpha_{|\mathbf{u}|}$ depend on the length of context $|\mathbf{u}|$. This is repeated until we reach to PY process for unigram model with empty context $\emptyset$, $G_{\emptyset} \sim \text{PY}(d_0, \alpha_0, G_0)$, as implemented in Eq. (4.11). A kind of linearly interpolated LM (called HPY-LM) is accordingly produced by using the HPY process which combines the mixture probability measures from the higher order statistics in the nth-order model from $\mathbf{u}$ and the lower order LM in the $(n-1)$th-order base measure from backoff context $G_{\pi(\mathbf{u})}$. The combination weights are formed from an PY process mixture model. In Sect. 4.3.1, we will present a new BNP inference procedure for topic-based LM.

4.3 Advanced Topic Models and Their Applications

We have surveyed the fundamental topic models based on the non-Bayesian parametric methods using LSA and PLSA, the Bayesian parametric method using LDA, and the Bayesian nonparametric methods using HDP and nCRP. Model structure has been extended from single-layer model (LSA, PLSA, LDA, HDP) to multiple-layer model (nCRP). Approximate inference algorithms using VB for LDA and Gibbs sampling for LDA, HDP, and nCRP have been addressed. In this section, we will present a series of advanced topic models for different applications including speech recognition, information retrieval, document classification, text segmentation, and document summarization. Here, we categorize these advanced topic models into different information models ranging from language model, document model, segmentation model to summarization model. Going beyond LDA topic model, some other issues are concerned and tackled to achieve flexible, scalable, and robust information systems for real-world applications.

4.3.1 Language Model

Speech recognition system is constructed with two essential models: acoustic model and language model (LM) which considerably affect the system performance. LM provides a prior word probability which characterizes the regularities in natural language. LM is not only useful for speech recognition but also for many other information systems including optical character recognition, spell correction, question answering, automatic summarization, information retrieval, etc. Basically, LM based on n-gram probability $p(w_i|w_{i-n+1}^{i-1})$ is constrained with two weaknesses: (1) lack of training data for higher order LM with large n and (2) lack of long-distance information due to the limitation of n-gram window. To deal with the sparseness of training data, HPY process [38] in Sect. 4.2 was presented to draw the smoothed LM with discounting scheme which was seen as Bayesian interpretation for the heuristic solution based on KN-LM [32]. Considering the issue of long-distance information, the topic-based LMs were proposed by merging the latent semantic information which relaxes the constraint of using short-term lexical information. In [25], PLSA topic model was incorporated into the construction of n-gram model. In addition, the LDA-LM was constructed by employing LDA-based topic information into LM training where the topic prediction was based on the hypothesis of either history words [36] or the words in a whole sentence [37]. In what follows, we introduce the extension of PLSA-LM and LDA-LM to the Dirichlet class LM [13] and the generalization of HPY-LM to the hierarchical Pitman–Yor-Dirichlet LM [9] where the topic models are taken into account.

Dirichlet Class Language Model

The key issue in LDA-LM [36, 37] is that topic information for word prediction is estimated from a set of training documents $\mathcal{D}$ which is treated as a bag of words.

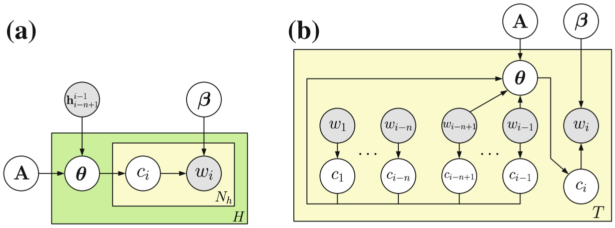

Fig. 4.4 Graphical representation for **a** DC-LM and **b** cache DC-LM

Such estimation did not consider the latent variables based on the *sequential order* of $n-1$ history words $\{w_{i-n+1}, \ldots, w_{i-1}\}$. Such ordering information is crucial for word prediction in natural language. Dirichlet class LM (DC-LM) [13] was proposed to deal with this issue through the representation of history words w_{i-n+1}^{i-1} by concatenating a sequence of $n-1$ history word vectors which are encoded by 1-of-$\mathcal{V}$ coding scheme. An $(n-1)\mathcal{V} \times 1$ supervector $\mathbf{h}_{i-n+1}^{i-1}$ is formed as the surrogate of w_{i-n+1}^{i-1} and then projected into an C-dimensional class space or topic space so that the class proportions are drawn from a Dirichlet prior $\boldsymbol{\theta} \sim \text{Dir}(\mathbf{A}^\top \mathbf{h}_{i-n+1}^{i-1})$. Graphical representation is shown in Fig. 4.4a. Here, the parameter $\mathbf{A} = \{\mathbf{a}_1, \ldots, \mathbf{a}_C\}$ in DC-LM plays a similar role to $\boldsymbol{\alpha}$ in LDA. The other parameters $\boldsymbol{\beta} = \{\beta_{vc}\}$ are seen as the class conditional multinomials for $\mathcal{V}$ words. In a corpus $\mathcal{D}$, there are H histories with N_h words predicted by each history. As a result, DC-LM is calculated by integrating over different classes c_i and proportions $\boldsymbol{\theta}$

$$\begin{aligned} p(w_i|\mathbf{h}_{i-n+1}^{i-1}, \mathbf{A}, \boldsymbol{\beta}) &= \sum_{c_i=1}^{C} p(w_i|c_i, \boldsymbol{\beta}) \int p(\boldsymbol{\theta}|\mathbf{h}_{i-n+1}^{i-1}, \mathbf{A}) p(c_i|\boldsymbol{\theta}) d\boldsymbol{\theta} \\ &= \sum_{c=1}^{C} \beta_{ic} \frac{\mathbf{a}_c^\top \mathbf{h}_{i-n+1}^{i-1}}{\sum_{j=1}^{C} \mathbf{a}_j^\top \mathbf{h}_{i-n+1}^{i-1}}. \end{aligned} \tag{4.14}$$

DC-LM parameters $\{\mathbf{A}, \boldsymbol{\beta}\}$ are estimated according to an VB-EM procedure [13]. DC-LM acts as a new Bayesian class LM which is a smoothed LM over the classes of histories. However, the long-distance information outside n-gram window was not characterized. For this concern, a cache DC-LM was proposed by incorporating the cache memory from all history words w_1^{i-1} into DC-LM as illustrated in Fig. 4.4. We can see that cache DC-LM is calculated through choosing the best class sequence $\hat{c}^{i-1}$ associated with each history word sequence $\hat{w}^{i-1}$. However, DC-LM is constructed with the fixed number of classes or topics C without considering power-law property.

Hierarchical Pitman–Yor-Dirichlet Language Model

Using HPY-LM, the predictive n-gram from $G_{\mathbf{u}}$ is inferred by marginalizing out the prior measure of backoff context $G_{\pi(\mathbf{u})}$. HPY-LM copes with the issue of data sparseness and holds the power-law property of natural language. But, topic information was not captured and accordingly the long-distance information was missed in HPY-LM. In [9], a hierarchical Pitman–Yor-Dirichlet LM (HPYD-LM) was proposed to achieve an BNP learning for the discounted topic-based LM which is seen as a flexible LM with power-law distributions and latent topics where the number of topics is unbounded. An HPYD process is constructed to draw the HPYD-LM. Different from the parametric topic mixture model

$$p(w_i|w_{i-n+1}^{i-1}) = \sum_{k=1}^{K} p(z_i = k|w_{i-n+1}^{i-1}) p(w_i|w_{i-n+1}^{i-1}, z_i = k) \tag{4.15}$$

HPYD process combines a prior process for drawing the topic-dependent smoothed n-gram $p(w_i|w_{i-n+1}^{i-1}, z_i = k)$ from an PY process, and a prior process for topic mixture probability $p(z_i = k|w_{i-n+1}^{i-1})$ from an DP. Starting from the uniform seed measure $H_0(w) = 1/\mathcal{V}$ for all words $w \in \Omega_v$, we draw a word measure from a global topic by $G_0 \sim \text{DP}(\gamma_0, H_0)$. The distribution of topic-dependent unigram $G_{\emptyset z_i}$ with empty context $\emptyset$ and topic assignment z_i is sampled by an PY process $G_{\emptyset z_i} \sim \text{PY}(d_1, \alpha_1, G_0)$ where G_0 is acted as a prior base measure. Next, $G_{\emptyset z_i}$ serves as a base measure for an DP to draw a distribution of unigrams $G_{w_i} \sim \text{DP}(\gamma_1, G_{\emptyset z_i})$. Using G_{w_i} as a prior measure, we draw the distribution of topic-dependent bigrams by using PY process $G_{w_{i-1}z_i} \sim \text{PY}(d_2, \alpha_2, G_{w_i})$. This measure is again acted as a prior basis for an DP to draw the distribution of bigrams $G_{w_{i-1}w_i} \sim \text{DP}(\gamma_2, G_{w_{i-1}z_i})$. Therefore, HPYD process is recursively realized by sampling the distribution of topic-dependent n-grams $p(w_i|w_{i-n+1}^{i-1}, z_i)$ from $G_{w_{i-n+1}^{i-1}z_i}$ and then that of n-grams $p(w_i|w_{i-n+1}^{i-1})$ from $G_{w_{i-n+1}^{i}}$ by

$$G_{w_{i-n+1}^{i-1}z_i} \sim \text{PY}\left(d_n, \alpha_n, G_{w_{i-n+1}^{i-1}}\right), \qquad G_{w_{i-n+1}^{i}} \sim \text{DP}\left(\gamma_n, G_{w_{i-n+1}^{i-1}z_i}\right). \tag{4.16}$$

A hierarchical Chinese restaurant process (HCRP) [9] was designed to implement the HPYD process and infer the HPYD-LM. Imagine that there are Chinese restaurants serving customers with infinite tables, infinite menus, and infinite dishes. For each restaurant with context $\mathbf{u}$, the first customer or word with parameter θ_1 enters the restaurant and chooses the first table in restaurant $\mathbf{u}$. He or she draws a shared menu for all customers seating with the same table and then orders a dish which is labeled by a distinct word $w_{\mathbf{u}1}$. Each customer θ_i only chooses one table and one dish from the single menu corresponding to that table. Each table has its own menu. Following this way, each customer chooses a table with a distinct menu and then draws a dish from that menu. Note that the menus in this HCRP are associated with the topics in HPYD-LM. The menus in restaurant $\mathbf{u}$ are obtained from two information sources: (1) the corresponding menus from the lower order or back off restaurant $\pi(\mathbf{u})$ and (2) the clustering information from the customers in higher order restaurant $\mathbf{u}$. The HPYD n-gram is determined by calculating the predictive or marginal probability of a test word w appearing after a context $\mathbf{u}$ given by a set of training data $\mathcal{D}$. The marginalization is performed over the arrangements of tables $\mathbf{t} = \{t_i, \mathbf{t}_{-i}\}$, menus $\mathbf{z} = \{z_i, \mathbf{z}_{-i}\}$, dishes $\mathbf{l} = \{l_i, \mathbf{l}_{-i}\}$ of all training words $\mathbf{w} = \{w_i, \mathbf{w}_{-i}\}$, and the hyperparameters $\boldsymbol{\lambda} = \{d_m, \alpha_m, \gamma_m | 1 \le m \le n\}$. A Gibbs sampling procedure was developed to draw the tables, the menus, and the dishes according to the corresponding posterior probabilities $p(t_i = t|\mathbf{t}_{-i}, \mathbf{z}, \boldsymbol{\lambda}, \mathbf{w}, \mathbf{u})$, $p(z_i = k|\mathbf{z}_{-i}, \mathbf{t}, \boldsymbol{\lambda}, \mathbf{w}, \mathbf{u})$, and $p(l_i = w|\mathbf{l}_{-i}, z_i = k, \mathbf{z}_{-i}, \boldsymbol{\lambda}, \mathbf{w}_{-i}, \mathbf{u})$, respectively [9]. At last, we realize the HPYD process and obtain the HPYD n-gram $p(w_i = w|\mathbf{w}_{-i}, \mathbf{z}, \boldsymbol{\lambda}, \mathbf{u})$.

4.3.2 Document Model

Some other advanced topic models are developed for robust document modeling by compensating the nonstationary condition or conducting the sparse representation.

Dynamic Topic Model

Blei and Lafferty [5] proposed a dynamic topic model (DTM) to analyze the time evolution of topics in a large document collection. The state space models using natural parameters of LDA topic model were implemented to provide a qualitative window over the content of a large data collection. In particular, the topics associated with time slice t evolve from the topics associated with slice $t - 1$. Accordingly, the conditional multinomials $\boldsymbol{\beta} = \{\boldsymbol{\beta}_k\}$ and the Dirichlet parameters $\boldsymbol{\alpha}$ are represented by the state space model with the evolution using Gaussians given by the isotropic covariance parameters σ^2 and δ^2

$$\boldsymbol{\beta}_{t,k}|\boldsymbol{\beta}_{t-1,k} \sim \mathcal{N}(\boldsymbol{\beta}_{t-1,k}, \sigma^2 I), \qquad \boldsymbol{\alpha}_t|\boldsymbol{\alpha}_{t-1} \sim \mathcal{N}(\boldsymbol{\alpha}_{t-1}, \delta^2 I). \tag{4.17}$$

Such time-dependent continuous variables are converted into the proportion variables to draw topics $\{z_{mnt}\}$ using $\boldsymbol{\alpha}_t$ and choose the corresponding words $\{w_{mnt}\}$ using $\boldsymbol{\beta}_{t,z}$ for each time slice t. DTM is an extension of LDA to meet nonstationary condition and has been successfully applied to analyze the evolution of topic words in the journal *Science* over 120 years [5].

Sparse Topic Model

The real-world text documents are usually contaminated with noises and redundancies. Sparse representation is helpful to establish a compact model which is robust to adverse conditions. Recently, a sparse Bayesian learning was introduced to perform sparse document representation using the sparse LDA (sLDA) [11]. Previous topic model based on LDA assumes that all of K topics are fully connected to each word w_{mn} in a document. The sLDA topic model aims to select salient features in LDA network by incorporating the spike-and-slab priors [29] into a Bayesian framework. The spike distribution is used to select salient features, while the slab distribution is applied to establish topic model based on the selected relevant topics. As addressed in Sect. 4.1.3, the connections between topics and words in LDA network are sufficiently reflected by the variational multinomial parameters $\{\hat{\phi}_{nk}\}$ which are introduced as the hyperparameters of the variational distributions of latent variables $\{z_{mn} = k\}$.

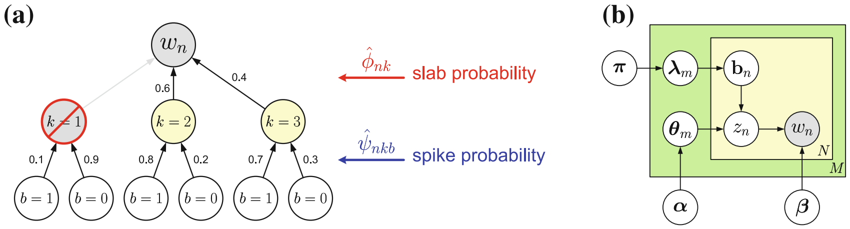

Fig. 4.5 **a** Illustration for feature selection using spike-and-slab priors. **b** Graphical representation for sparse LDA

Such connection is used to select salient features or topics for document representation. Figure 4.5a illustrates the feature selection using spike-and-slab priors. The variational parameter $\hat{\phi}_{nk}$ is treated as a slab probability which connects the representation of a target word w_{mn} using the relevant topics (here $k = 2$ and $k = 3$). This judgment is made from an indicator $b_{nmk} \sim \text{Bern}(\lambda_{mk})$ using a Bernoulli parameter drawn from a beta distribution $\lambda_{mk} \sim \text{Beta}(\boldsymbol{\pi})$. A word w_{mn} is chosen using the conditional multinomial where only the relevant topic k with $b_{nmk} = 1$ is merged, namely

$$w_{mn} = v | b_{nmk} = 1, z_{mn} = k \sim \text{Mult}(\beta_{vk}). \tag{4.18}$$

Graphical representation of sLDA is shown in Fig. 4.5b. An VB-EM procedure was developed to infer the sLDA parameters $\{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}\}$ by maximizing the marginal likelihood $p(\mathcal{D}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi})$ over four latent variables $\{\mathbf{z}, \boldsymbol{\theta}, \mathbf{b}, \boldsymbol{\lambda}\}$. Notably, marginal likelihood is only accumulated for all training samples $\{w_{mn\}}$ connected with their associated topics $z_{nm} = k$ with condition $b_{nmk} = 1$. The variational distributions with parameters $\{\boldsymbol{\phi}, \boldsymbol{\gamma}, \boldsymbol{\psi}, \boldsymbol{\eta}\}$ corresponding to latent variables $\{\mathbf{z}, \boldsymbol{\theta}, \mathbf{b}, \boldsymbol{\lambda}\}$ are estimated by maximizing the variational lower bound. Importantly, the variational binomial parameters $\hat{\boldsymbol{\psi}} = \{\hat{\psi}_{nkb}\}$ for binomial indicators $\mathbf{b} = \{b_{nmk}\}$ are estimated as the spike probabilities for feature selection, while the variational multinomial parameters $\hat{\boldsymbol{\phi}} = \{\hat{\phi}_{nk}\}$ for multinomial topics $\mathbf{z} = \{z_{nm}\}$ are calculated as the slab probabilities to model those selected features. In this illustration, the spike probability for topic $k = 1$ under $b_{nmk} = 1$ is too small to contribute the generation of a target word w_{mn}.

4.3.3 Segmentation Model

Sequential patterns in natural language usually appear without explicit boundaries but with the variations of temporal topics. Text segmentation aims to partition the text data into homogeneous processing units or semantically coherent chunks. This research horizon is crucial for many applications including language modeling, speech recognition, text categorization, retrieval and summarization, and also topic detection and tracking. However, in real world, the observed text stream is constructed by a set of heterogeneous documents, making it difficult to extract homogeneous topics. In what follows, we introduce how LDA topic model is extended to cope with the stream-level segmentation and the document-level segmentation [14]. In stream-level segmentation, the text stream is partitioned into topic-coherent documents. In document-level segmentation, the pseudodocument is further segmented into word-coherent paragraphs. Such a hierarchical segmentation makes it feasible to build a precise topic model to compensate the varying distributions of topics and words in nonstationary conditions. This idea can be applied to conduct automatic transcription for lecture speech where the discussion topics are changed by time. This is similar to the situation that the topics are moving between two concatenated documents.

Topic-Based Stream-Level Segmentation

Segmentation of a text stream can be treated as a task of detecting the boundary of documents according to the similarity between sentences $\mathbf{w}_{t-1}$ and $\mathbf{w}_t$ at each sentence time t which is measured by calculating the cosine distance between the corresponding topic proportions $s(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t)$. The sentence-dependent topic proportions $\boldsymbol{\theta}_t = \{\theta_{tk}\}$ are determined by using the MAP estimate of variational posteriors $\mathbb{E}[\theta_{tk}|\hat{\gamma}_{tk}]$. We draw a segmentation probability based on the beta distribution using this one-sided contextual similarity, i.e., $\omega \sim \text{Beta}(1 - \varepsilon_t, \varepsilon_t)$ where $\varepsilon_t = s(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t)$. The segmentation label c for each pair of sentences is then chosen by a binomial distribution $c \sim \text{Bin}(\boldsymbol{\omega})$. The segmentation boundary is detected when $c = 1$, otherwise this sentence is grouped into the previous segment. The number of segments is determined automatically. In this study, contextual topic information plays an important role for stream-level segmentation. In [14], the one-sided contextual similarity $s(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t)$ was improved by using the two-sided contextual similarity for beta parameter via $\varepsilon_t = \max\{s(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_t), s(\boldsymbol{\theta}_{t-1}, \boldsymbol{\theta}_{t+1})\}$. A smoothed boundary detection was performed. The segmentation error due to the non-topic sentences was alleviated. This stream-level segmentation is performed to compensate the variations of topic distributions $\boldsymbol{\theta}$ in a text stream.

Topic-Based Document-Level Segmentation

Furthermore, the variations of word distributions within a pseudodocument are treated in the document-level segmentation. It is because that the usage of the same words in a natural language system is gradually varied over different paragraphs or segments due to the composition style and document structure. Accordingly, we merge a Markov chain to characterize the dynamics of word distributions in LDA topic model. Figure 4.6a shows graphical representation of the resulting nonstationary LDA. Here, each word w_{mn} or simply w_n is generated due to both topic z_n and segment or state s_n. A left-to-right hidden Markov model topology without state skipping is implemented for document-level segmentation. The model parameters consist of $\{\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}\}$ where $\mathbf{A} = \{a_{s_{n-1}s_n}\}$ denotes the state transition probabilities.

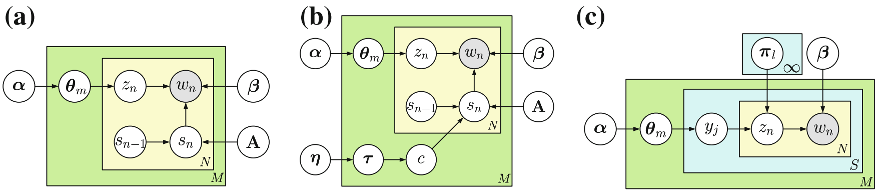

Fig. 4.6 Graphical representation for **a** nonstationary LDA, **b** adaptive and nonstationary LDA, and **c** sentenced-based LDA

Again, the VB-EM algorithm is applied to estimate model parameters $\{\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}\}$ by maximizing the marginal likelihood

$$\begin{aligned} p(\mathcal{D}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}) = \prod_{m=1}^{M} \int p(\boldsymbol{\theta}_m|\boldsymbol{\alpha}) \sum_{s} \prod_{n=1}^{N_m} \sum_{k=1}^{K} p(z_{mn} = k|\boldsymbol{\theta}_m) \\ \times p(w_{mn}|z_{mn} = k, s_{mn}, \boldsymbol{\beta}) p(s_{mn} = s|s_{m,n-1}, \mathbf{A}) d\boldsymbol{\theta}_m. \end{aligned} \tag{4.19}$$

This nonstationary LDA was constructed from spoken documents and merged into n-gram language model. The speech recognition results were rescored for spoken documents [19]. In [20], an adaptive segmentation model was proposed by introducing a style variable c which indicated the number of stylistic changes in a document as depicted in Fig. 4.6b. Style variable is modeled by a multinomial distribution $c \sim \text{Mult}(\boldsymbol{\tau})$ with the style proportions drawn from a Dirichlet prior $\boldsymbol{\tau} \sim \text{Dir}(\boldsymbol{\eta})$. The hybrid stream-level and document-level segmentation was successfully applied for topic detection and tracking in [14].

4.3.4 Summarization Model

Automatic summarization aims to extract the thematic contents or sentences from a large set of documents. A good summary is helpful for browsers to capture the themes and concepts from multiple documents in a very short time. Beyond document representation in word level and document level using LDA, the key issue in a summarization system is to conduct a hierarchical modeling over words, sentences, and documents in a corpus. Given the trained parameters, we can measure the similarity between a document and individual sentence and select the top-ranked sentences according to the Kullback–Leibler (KL) divergence. In a practical system, we usually observe heterogeneous documents where the topics are ambiguous, inconsistent, and diverse. A good summary should reflect the diversity of topics in documents and keep the redundancy to be minimum. In what follows, we survey two advanced topic models for document summarization. One is the parametric model based on the sentence-based LDA [8] and the other one is the nonparametric model based on the sentenced-based nested Chinese restaurant process [10].

Sentence-Based Latent Dirichlet Allocation

A simple extension to allow sentence modeling in LDA topic model is to introduce the sentence-level latent variable $y_j = l$ for each sentence s_j and connect it with the word-level latent variable $z_n = k$ for document representation. Different from the latent topics in word-level representation, we use another related concept called "themes" as the latent variables for sentence-level representation. A sentence-based LDA is constructed as depicted in Fig. 4.6c. Each word $w_n = v$ in sentence s_j $(1 \leq j \leq S)$ and document d_m is drawn by using a word-level multinomial parameter β_{vk} where the latent topic $z_n = k$ is determined by using a theme-dependent topic proportion π_{lk} with latent theme $y_j = l$ $(1 \leq l \leq L)$. This theme is drawn from a document-dependent theme proportion θ_{ml} which is governed by a Dirichlet prior with hyperparameters $\boldsymbol{\alpha} = \{\alpha_l\}$. Notably, each sentence is associated with a latent theme $y_j = l$. Each theme is used to draw the corresponding latent topic $z_n = k$ for representation of a target word $w_n = v$. As a result, document summarization is performed by calculating the KL divergence using the sentence-based unigram $p(w_n|s_j)$ and document-based unigram $p(w_n|d_m)$. The estimated model parameters $\{\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\pi}}\}$ and their variational parameters via VB-EM algorithm are used to calculate these two unigram probabilities $p(w_n|s_j)$ and $p(w_n|d_m)$.

Sentence-Based Nested Chinese Restaurant Process

Similar to what we have discussed in Sect. 4.2 for standard LDA, there are two limitations in the sentence-based LDA which constrain the performance of document representation and summarization. First, the number of themes L and the number of topics K are fixed in advance. Second, different themes l are assumed to be independent while different topics k are independent as well. A sentence-based nCRP was proposed to relax these two assumptions and apply for Bayesian nonparametric document summarization [10]. A metaphor for sentence-based nCRP (snCRP) is displayed in Fig. 4.3b. An infinitely branching tree structure is built for representation of words, sentences, and documents based on an nCRP compound HDP by a two-stage procedure. In the first stage, each sentence s_j of a document d_m is drawn from a document-dependent theme mixture model $G_{s,m}$ via an nCRP. In the second stage, each word w_n of a sentence s_j under a tree node is drawn from a theme-dependent topic mixture model $G_{w,l}$ via an HDP. The probability measures of two models and the relation between the measures of theme ψ_l and topic ϕ_k are expressed by

$$G_{s,m} = \sum_{l=1}^{\infty} \theta_{ml} \delta_{\psi_l}, \qquad G_{w,l} = \sum_{k=1}^{\infty} \pi_{lk} \delta_{\phi_k}, \qquad \psi_l \sim \sum_{k} \pi_{lk} \phi_k. \tag{4.20}$$

Here, the theme proportions $\boldsymbol{\theta}_m = \{\theta_{ml}\}$ and the topic proportions $\boldsymbol{\pi}_l = \{\pi_{lk}\}$ in sentence-based nCRP are similar to those in sentenced-based LDA.

Using this approach, the document-dependent theme mixture model $G_{s,m}$ is established under a sentence-based tree model with atoms $\{\psi_l\}_{l=1}^{\infty}$. Different from the word-based nCRP in Fig. 4.3a using a single tree path $\mathbf{c}_m$ for representation of words $\{w_n\}$ in a document d_m, the sentence-based nCRP in Fig. 4.3b represents the sentences

$\{s_j\}$ of a document d_m based on the theme parameters $\{\psi_l\}$ along the subtree path $\mathbf{t}_m \sim \text{snCRP}(\alpha_0)$. A wide coverage of thematic information in $\mathbf{t}_m$ is beneficial to compensate the thematic uncertainties or variations in the sentences from heterogeneous documents $\mathcal{D}$. Furthermore, the theme-dependent topic mixture model $G_{w,l}$ is constructed by treating the words of the sentences in a tree node l as the grouped data and modeling those grouped data in different tree nodes according to an HDP. The shared atoms $\{\phi_k\}_{k=1}^{\infty}$ are involved. Each word w_n in sentence s_j and document d_m is chosen by a multinomial distribution with parameter $\phi_{\mathbf{t}_m(y_j, z_n)}$ which is selected from the parameter of topic $z_n = k$ under a tree node of theme $y_j = l$ from a subtree path $\mathbf{t}_m$. The topic k and theme l are drawn from the topic proportions $\boldsymbol{\pi}$ and theme proportions $\boldsymbol{\theta}$, respectively. Importantly, the theme-dependent topic proportions are drawn by an GEM distribution $\boldsymbol{\pi}_l|\gamma_w \sim \text{GEM}(\gamma_w)$ using a word-level strength parameter γ_w through a stick-breaking processing (SBP). The document-dependent theme proportions are chosen by a treeGEM distribution $\boldsymbol{\theta}_m|\gamma_s \sim tree\text{GEM}(\gamma_s)$ using a sentence-level parameter γ_s through a tree SBP. In [10], a Gibbs sampling was developed to sample a document-dependent subtree branches $\mathbf{t}_m = \{t_{mj}\}$, document-dependent theme labels $\mathbf{y} = \{y_j\}$ and theme-dependent topic labels $\mathbf{z} = \{z_n\}$ according to the posterior probabilities $p(t_{mj} = t|\mathbf{t}_{m(-j)}, \mathcal{D}, \mathbf{y}, \alpha_0)$, $p(y_j = l|d_m, \mathbf{y}_{-j}, \mathbf{t}_m, \gamma_s)$ and $p(z_n = k|\mathcal{D}, \mathbf{z}_{-n}, y_j = l, \gamma_w)$, respectively. A document summarization system was established through a sentence selection procedure over the inferred tree model for sentences.

4.4 Summary and Future Direction

We have presented the theoretical background and surveyed some advances in topic models for speech and language processing. In theoretical background, we started from the general unsupervised learning methods using latent variable models based on FA and ICA and then moved to general topic models for natural language applications. We systematically addressed the evolution of topic models from the parametric models using LSA, PLSA, and LDA to the Bayesian nonparametric models using HDP and nCRP. The inference solutions to LDA based on VB and Gibbs sampling procedures were investigated. The Bayesian nonparametric learning methods via DP, PY process, HDP, and HPY process were introduced. From these theoretical surveys, we would like to move beyond baseline topic model using LDA toward building a flexible, hierarchical, adaptive, and scalable topic model to meet a variety of heterogeneous conditions in real-world information systems.

In the advanced studies, we presented a series of extended topic models which were developed and applied for speech recognition, document retrieval, text segmentation, and document summarization. We discussed different issues in LDA topic model including topic correlation, model complexity, topic structure, model smoothing, power-law property, temporal modeling, overtrained problem, sparse representation, nonstationary condition, and ill-posed condition. A variety of solutions were proposed to achieve finite-dimensional and infinite-dimensional topic-based language models, dynamic and sparse topic-based document models, topic-based stream-level and document-level segmentation, and sentence-based LDA and nCRP summarization models. The HPY compound HDP was developed for topic-based language model, while the nCRP compound HDP was exploited for sentence clustering and hierarchical modeling of words, sentences, and documents.

Some suggestions are provided for future direction. In the era of big data, we build an infinite model from heterogeneous data. We should think more seriously about the problems at hand, systematically extract the latent information, and carefully represent the model variations. We need to take care of some challenging issues including parallel processing in algorithm level as well as in system level, rapid inference algorithm and sequential MCMC algorithm and work on big learning for topic model. It is interesting to discover ubiquitous extensions and connections to the nonnegative matrix factorization, tensor decomposition and deep neural network and apply them to speech recognition, speaker recognition, speech synthesis, music classification, source separation, etc.

References

1. Basilevsky, A.: Statistical Factor Analysis and Related Methods—Theory and Applications. Wiley, New York (1994)
2. Bell, A.J., Sejnowski, T.J.: An information-maximization approach to blind separation and blind deconvolution. Neural Comput. **7**, 1129–1159 (1995)
3. Blei, D., Griffiths, T., Jordan, M., Tenenbaum, J.: Hierarchical topic models and the nested Chinese restaurant process. Adv. Neural Inf. Proc. Syst. **16**, 17–24 (2004)
4. Blei, D.M., Griffiths, T.L., Jordan, M.I.: The nested Chinese restaurant process and Bayesian nonparametric inference of topic hierarchies. J. ACM **57**(2) (2010). Article 7
5. Blei, D.M., Lafferty, J.D.: Dynamic topic models. In: Proceedings of International Conference on Machine Learning, pp. 113–120 (2006)
6. Blei, D.M., Lafferty, J.D.: A correlated topic model of science. Ann. Appl. Stat. **1**(1), 17–35 (2007)
7. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
8. Chang, Y.L., Chien, J.T.: Latent Dirichlet learning for document summarization. In: Proceedings of International Conference on Acoustics, Speech, and Signal Processing, pp. 1689–1692 (2009)
9. Chien, J.T., Chang, Y.L.: Hierarchical Pitman-Yor and Dirichlet process for language model. In: Proceedings of Annual Conference of International Speech Communication Association, pp. 2212–2216 (2013)
10. Chien, J.T., Chang, Y.L.: Hierarchical theme and topic model for summarization. In: Proceedings of IEEE International Workshop on Machine Learning for Signal Processing, pp. 1–6 (2013)
11. Chien, J.T., Chang, Y.L.: Bayesian sparse topic model. J. Signal Proc. Syst. **74**(3), 375–389 (2014)
12. Chien, J.T., Chen, B.C.: A new independent component analysis for speech recognition and separation. IEEE Trans. Audio Speech Lang. Process. **14**(4), 1245–1254 (2006)
13. Chien, J.T., Chueh, C.H.: Dirichlet class language models for speech recognition. IEEE Trans. Audio, Speech, Lang. Process. **19**(3), 482–495 (2011)

14. Chien, J.T., Chueh, C.H.: Topic-based hierarchical segmentation. IEEE Trans. Audio Speech Lang. Process. **20**(1), 55–66 (2012)
15. Chien, J.T., Hsieh, H.L.: Convex divergence ICA for blind source separation. IEEE Trans. Audio Speech Lang. Process. **20**(1), 290–301 (2012)
16. Chien, J.T., Ting, C.W.: Factor analyzed subspace modeling and selection. IEEE Trans. Audio Speech Lang. Process. **16**(1), 239–248 (2008)
17. Chien, J.T., Ting, C.W.: Acoustic factor analysis for streamed hidden Markov model. IEEE Trans. Audio Speech Lang. Process. **17**(7), 1279–1291 (2009)
18. Chien, J.T., Wu, M.S.: Adaptive Bayesian latent semantic analysis. IEEE Trans. Audio Speech Lang. Process. **16**(1), 198–207 (2008)
19. Chueh, C.H., Chien, J.T.: Nonstationary latent Dirichlet allocation for speech recognition. In: Proceedings of Annual Conference of International Speech Communication Association, pp. 372–375 (2009)
20. Chueh, C.H., Chien, J.T.: Adaptive segment model for spoken document retrieval. In: Proceedings of International Symposium on Chinese Spoken Language Processing, pp. 261–264 (2010)
21. Comon, P.: Independent component analysis, a new concept? Signal Process. **36**(3), 287–314 (1994)
22. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by latent semantic analysis. J. Am. Soc. Inf. Sci. **41**(6), 391–407 (1990)
23. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. R. Stat. Soc. B **39**(1), 1–38 (1977)
24. Fei-Fei, L., Perona, P.: A Bayesian hierarchical model for learning natural scene categories. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 524–531 (2005)
25. Gildea, D., Hofmann, T.: Topic-based language models using EM. In: Proceedings of European Conference on Speech Communication and Technology, pp. 2167–2170 (1999)
26. Griffiths, T.L., Steyvers, M.: Finding scientific topics. Proc. Nat. Acad. Sci. U.S.A. **101**(1), 5228–5235 (2004)
27. Hofmann, T.: Probabilistic latent semantic indexing. In: Proceedings of Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 50–57 (1999)
28. Hyvarinen, A.: Fast and robust fixed-point algorithms for independent component analysis. IEEE Trans. Neural Netw. **10**(3), 626–634 (1999)
29. Ishwaran, H., Rao, J.S.: Spike and slab variable selection: frequentist and Bayesian strategies. Ann. Stat. **33**(2), 730–773 (2005)
30. Jolliffe, I.T.: Principal Component Analysis. Springer, New York (1986)
31. Kim, S., Georgiou, P., Narayanan, S.: Latent acoustic topic models for unstructured audio classification. APSIPA Trans. Signal Inf. Process. **1** (2012). doi:10.1017/ATSIP.2012.7
32. Kneser, R., Ney, H.: Improved backing-off for *m*-gram language modeling. In: Proceedings of International Conference on Acoustics, Speech, and Signal Processing, pp. 181–184 (1995)
33. Kuhn, R., Junqua, J.C., Nguyen, P., Niedzielski, N.: Rapid speaker adaptation in eigenvoice space. IEEE Trans. Audio Speech Lang. Process. **8**(4), 695–707 (2000)
34. Pitman, J., Yor, M.: The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. Ann. Probab. **25**, 855–900 (1997)
35. Smaragdis, P., Shashanka, M., Raj, B.: Topic models for audio mixture analysis. In: Proceedings of NIPS Workshop on Applications for Topic Models: Text and Beyond (2009)
36. Tam, Y.C., Schultz, T.: Dynamic language model adaptation using variational Bayes inference. In: Proceedings of Annual Conference of International Speech Communication Association, pp. 5–8 (2005)
37. Tam, Y.C., Schultz, T.: Unsupervised language model adaptation using latent semantic marginals. In: Proceedings of Annual Conference of International Speech Communication Association (2006)

38. Teh, Y.W.: A hierarchical Bayesian language model based on Pitman-Yor processes. In: Proceedings of International Conference on Computational Linguistics and Annual Meeting of the Association for Computational Linguistics, pp. 985–992 (2006)
39. Teh, Y.W., Jordan, M.I., Beal, M.J., Blei, D.M.: Hierarchical Dirichlet processes. J. Am. Stat. Assoc. **101**(476), 1566–1581 (2006)