ISBN 978-951-51-1243-9
UNIGRAFIA
HELSINKI 2015

JIE XIONG | PREDICTIVE CLASSIFICATION AND BAYESIAN INFERENCE

UNIVERSITY OF HELSINKI
FACULTY OF SCIENCE

PREDICTIVE CLASSIFICATION AND BAYESIAN INFERENCE

JIE XIONG

Department of Mathematics and Statistics

Predictive Classification and Bayesian Inference

Jie Xiong

Academic dissertation

To be presented, with the permission of the Faculty of Science of the University of Helsinki, for public examination in Päärakennus, Auditorium XII, on June 15th, 2015, at 12 o'clock noon.

University of Helsinki
Finland

Supervisor
Jukka Corander, University of Helsinki, Finland

Pre-examiners
Daniel Thorburn, Stockholms Universitet, Sweden
Mattias Villani, Linköpings Universitet, Sweden

Opponent
Arnoldo Frigessi, University of Oslo, Norway

Custos
Jukka Corander, University of Helsinki, Finland

Contact information

Department of Mathematics and Statistics
P.O. Box 68 (Gustaf Hällströmin katu 2b)
FI-00014 University of Helsinki
Finland

Email address: mathstat-info@helsinki.fi
URL: http://www.math.helsinki.fi/
Telephone: +358-02941-51506, +358-02941-51502

ISBN 978-951-51-1243-9 (paperback)
ISBN 978-951-51-1244-6 (PDF)
Helsinki 2015
Unigrafia

Predictive Classification and Bayesian Inference

Jie Xiong

Department of Mathematics and Statistics
P.O. Box 68, FI-00014 University of Helsinki, Finland
Jie.Xiong@helsinki.fi

Helsinki, May 2015, 119 pages

Abstract

A general inductive probabilistic framework for clustering and classification is introduced using the principles of Bayesian predictive inference, such that all quantities are jointly modelled and the uncertainty is fully acknowledged through the posterior predictive distribution. Several learning rules have been considered and the theoretical results are extended to acknowledge complex dependencies within the datasets. Multiple probabilistic models have been developed for analysing data from a wide variate of fields of applications. State-of-art algorithms are introduced and developed for the model optimization.

Acknowledgements

I am grateful for the funding provided by the Finnish Doctoral Programme in Stochastics and Statistics (FDPSS) and the Finnish Center of Excellence in Computational Inference Search(COIN).

I would like to express my most special appreciation and thanks to my supervisor Professor Jukka Corander, who has also been a tremendous mentor for me. I would like to thank you for your guiding and encouraging which help me finally finish this thesis. Your advices have been priceless. I also wish to thank Sirkka-Liisa Varvio for the kind help in MBI programme and introducing me to the BSG research group.

I wish to express my gratitude to all my co-authors: Väino, Henrik, Paul, Johan, Yaqiong, Jing and Prof. Timo Koski. I also want to thank present and former members of our research group: Lu C.,Lu W., Alberto, Mikhail, Jukka S., Jukka K., Elina, Minna, Hailin and others. Without your assistance, my PhD journey could be much longer and more tough. I want to thank Swee Chong Wong, Hongyu Su, Chengyu Liu, Zitong Li and all the other friends for sharing experience in science and life, and for drinking with me to go through the Finnish winter.

Special thanks to my parents for your love and all of the sacrifices that you have made for me. Finally, I would like express appreciation to my beloved wife Yitian who always stay with me and support me to go through all the difficult moments.

Helsinki, May 2015

Jie Xiong

List of original articles

This thesis consists of five articles and an introductory part. We refer to the articles by Roman numerals I-V.

I. Jukka Corander, Jie Xiong, Yaqiong Cui, and Timo Koski. Optimal Viterbi Bayesian predictive classification for data from finite alphabets. *Journal of Statistical Planning and Inference*, 143(2): 261-275, 2013.

II. Väinö Jääskinen, Jie Xiong, Jukka Corander, and Timo Koski. Sparse Markov chains for sequence data. *Scandinavian Journal of Statistics*, 41(3):639-655, 2013.

III. Henrik Nyman, Jie Xiong, Johan Pensar and Jukka Corander. Marginal and simultaneous predictive classification using stratified graphical models. *Advances in Data Analysis and Classification*, DOI: 10.1007/s11634-015-0199-5, 2015.

IV. Jie Xiong, Väinö Jääskinen, and Jukka Corander. Recursive learning for sparse Markov models. *Bayesian Analysis*, doi:10.1214/15-BA949, 2015.

V. Paul Blomstedt, Jing Tang, Jie Xiong, Christian Granlund, and Jukka Corander (2014). A Bayesian predictive model for clustering data of mixed discrete and continuous type. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3):489-498 2014.

Author's contribution to Articles I-IV

I. JX and JC shared the main responsibility in writing the article and the main responsibility for the implementation and empirical testing. JX also had the main responsibility for deriving the asymptotic results.

II. JX contributed mainly to developing the method, implementation and empirical testing, while VJ also contributed to these. VJ and JC had the main roles in writing the article.

III. JX contributed to developing the method and implementation. HN contributed mainly in designing the model, implementation and empirical results. HN and JC had the main roles in writing the article.

IV. JX had a main role in developing the method, implementation and empirical testing, while VJ also contributed to implementation and empirical testing. JX and JC had the main role in writing the article.

V. JX participated in implementation and empirical testing, while PB and TJ had the main role. PB had the main responsibility for developing the methods, implementation, and empirical testing. PB and JC contributed mainly to writing the article.

Contents

Chapter 1

Introduction

Machine learning became a popular topic in the 90's due to the development of computational resources and the interaction of Computer Science and Statistics research. There are multitudinous clustering and classification principles existing with their own different applications. Among these methods, Bayesian predictive inference methods have attracted considerable popularity in multiple studies with their versatility both from the theoretical and applied perspective(Huo et al., 1997; Nadas, 1985; Jiang et al., 1998; Huo and Lee, 1997; Watanabe et al., 2004). In the contemporary world, the amount of data for analysing is growing exponentially in a wide variate of fields of science and industry(Hilbert and López, 2011). To deal with this data explosion, new methods shall be developed to quickly identify, analyse and validate complex information in Big Data.

Clustering and Classification are the most common tasks in machine learning and statistics, and there are multitudinous clustering and classification principles existing with their own different applications(Bishop, 2006; Duda et al., 2012; Hastie et al., 2009; Ripley, 1996). Among these methods, Bayesian predictive inference methods have attracted considerable popularity in multiple studies with their versatility both from the theoretical and applied perspective(Huo et al., 1997; Nadas, 1985; Jiang et al., 1998; Huo and Lee, 1997; Watanabe et al., 2004).

In this thesis, we focus on the Bayesian predictive learning princi-

ples based on generative probabilistic models, where the uncertainty is fully acknowledged through the posterior predictive distribution. Especially, we operationalize the idea of predictive learning introduced by Geisser (1993), and apply it with multiple probabilistic models for analysing data from different sources.

The background of the five articles are provided and the key issues are highlighted in the following chapters. In Chapter 2, the approach of making Bayesian predictive inference is introduced together with the clustering and classification rules. In Chapter 3, the models developed and utilized in the articles are presented briefly. The algorithms designed for the learning tasks are described in Chapter 4. Finally, the comparison between the models and algorithms, as well as the directions for future research are discussed in Chapter 7.

Chapter 2

Bayesian Predictive Inference

Statistical analysis generally laid the emphasis on inferences or decisions about parameters of statistical models until de Finetti proposed an observabilistic view on inference (Geisser, 1993; de Finetti, 1974). This view is based on understanding that direct inference about observables would better serve the purpose of statistical modelling in many different contexts.

However, in most of the cases, such an ultimate predictive approach becomes a difficult and tedious task. A conventional Bayesian approach models the distribution of data based on parameters, which are integrated out in the final model by introducing prior densities for the parameters. The final model should have a capability to provide calculable probability for the observed data and a predictive probability distribution for future data.

Example 2.0.1. Thumbtack tossing Consider tossing a metal thumbtack with a round curved head onto a soft surface. We keep track of whether the thumbtack ends up with the point up or down.

In Example 2.0.1, without any information to distinguish tosses, it is reasonable to model the outcomes of the individual tosses as independent and identically distributed (i.i.d.) Bernoulli random quantities with $X_i = 1$ indicating the ith toss is point up and $X_i = 0$

meaning that toss is point down.

In the frequentist framework, one is interested in a parameter, say θ, which has an unknown fixed value in $[0, 1]$ and claim that the tosses X_i are i.i.d. with $P(X_i = 1) = \theta$. Such an model of a sequence of n tosses is represented by the following likelihood function

$$P(X_1 = x_1, \ldots, X_n = x_n) = \prod_{i=1}^{n} \theta^{x_i}(1-\theta)^{1-x_i}, \tag{2.1}$$

whose value is maximized when θ taking the value as the relative frequency of observing tosses point up, that is $\sum_{i=1}^{n} \frac{x_i}{n}$.

Unlike in the above frequentist approach, a predictive Bayesian model aims at encapsulating a certain form of dependence among the observations. In this example, such a model can be constructed based on the following assumptions:

- The tosses X_i are judged to be independent, Bernoulli random quantities conditional on a random parameter θ.
- θ is associated with a probability distribution $p(\theta)$.
- According to the strong law of large numbers, $\theta = lim_{n\to\infty} y_n/n$, $y_n = \sum_{i=1}^{n} x_i$.

The prior predictive probability of the data sequence, also known as the marginal likelihood function, can be then expressed as

$$P(X_1 = x_1, \ldots, X_n = x_n) = \int_0^1 \prod_{i=1}^{n} \theta^{x_i}(1-\theta)^{1-x_i} p(\theta) d\theta, \tag{2.2}$$

In the above model, the quantities $x_1, \ldots, x_n$ are a random sample from Bernoulli distribution with parameter θ which corresponds to the joint conditional distribution in (2.1), where the parameter θ is treated as random variable assigned the prior distribution $p(\theta)$. This construction is generally referred to as a *prior predictive* model.

For making inferences about future observations based on the already observed data, a similar representation as in the *prior predictive* model is used. However, the prior distribution $p(\theta)$ for θ has

to be updated by the information in the acquired observations. In the thumbtack tossing example, for future tosses $x_{n+1}, \ldots, x_m$, the conditional probability function $P(X_{n+1} = x_{n+1}, \ldots, X_m = x_m | X_1 = x_1, \ldots, X_n = x_n)$ given $x_1, \ldots, x_n$ has the form

$$\int_0^1 \prod_{i=n+1}^{m} \theta^{x_i}(1-\theta)^{1-x_i} p(\theta|x_1, \ldots, x_n) d\theta, \tag{2.3}$$

where

$$p(\theta|x_1, \ldots, x_n) = \frac{\prod_{i=1}^{n} \theta^{x_i}(1-\theta)^{1-x_i} p(\theta) d\theta}{\int_0^1 \prod_{i=1}^{n} \theta^{x_i}(1-\theta)^{1-x_i} p(\theta) d\theta}. \tag{2.4}$$

This is called a *posterior predictive* model, which provides the basis for deriving the conditional predictive distribution of a generic random quantity defined in terms of the future observations.

2.1 Bayesian Clustering

Clustering is a typical task in unsupervised learning(Bishop, 2006) and can be interpreted as *prior predictive* inference in the Bayesian approach. In Articles II, IV, V, Bayesian clustering methods are implemented for learning the hidden structure underlying data. In Articles II and IV, the primary aim is to optimize Markovian models for sequence data, while in Article V we aim to find the optimal assignment for data items into a non-predetermined set of classes.

Given the observed data $\mathbf{X}$, let $\mathcal{S}$ denote a set of all the possible partitions of $\mathbf{X}$. The task of clustering is finding an optimal partition $S \in \mathcal{S}$ to represent the structure of $\mathbf{X}$ according to certain criteria. In Bayesian clustering, S is typically treated as a latent variable and we are mostly interested in the posterior distribution $P(S|\mathbf{X})$ of S according to the Bayes' rule:

$$P(S|\mathbf{X}) = \frac{P(\mathbf{X}|S)P(S)}{P(\mathbf{X})}, \tag{2.5}$$

where the likelihood function $P(\mathbf{X}|S)$ is specified by a *prior predictive* model

$$P(\mathbf{X}|S) = \int P(\mathbf{X}|S, \Theta) P(\Theta|S) d\Theta, \tag{2.6}$$

and

$$P(\mathbf{X}) = \sum_{S} P(\mathbf{X}|S)P(S) \tag{2.7}$$

Assuming a simple zero-one loss function, the optimal clustering solution can be obtained by identifying the maximum a *posteriori* (MAP) estimate of the partition S. Let $\mathbf{x}^{(n)}$ denote a dataset contains n items, $\mathbf{s}^{(n)} = (s_1, s_2, \ldots, s_n)$ the labels assigned to the items and $\mathcal{S}^{(n)}$ represent the set of all possible assignments. The MAP estimate is defined as:

$$\begin{aligned}
\hat{\mathbf{s}}^{(n)} &= \arg \max_{\mathbf{s}^{(n)} \in \mathcal{S}^{(n)}} p(\mathbf{s}^{(n)}|\mathbf{x}^{(n)}) \\
&= \arg \max_{\mathbf{s}^{(n)} \in \mathcal{S}^{(n)}} p(\mathbf{x}^{(n)}|\mathbf{s}^{(n)}) p(\mathbf{s}^{(n)}) && (2.8) \\
&= \arg \max_{\mathbf{s}^{(n)} \in \mathcal{S}^{(n)}} \int p(\mathbf{x}^{(n)}|\mathbf{s}^{(n)}, \theta) p(\theta|\mathbf{s}^{(n)}) d\theta \int p(\mathbf{s}^{(n)}|\phi) p(\phi) d\phi && (2.9)
\end{aligned}$$

In combinatorial mathematics, Bell number $B(n)$ counts the number of possible partitions in $\mathcal{S}^{(n)}$. As illustrated in Figure 2.1, the number increases faster than exponential as a function of n (Cameron, 1990). As a consequence, in most of the cases it is infeasible to evaluate $p(\mathbf{s}^{(n)}|\mathbf{x}^{(n)})$ for all possible partitions in $\mathcal{S}^{(n)}$. However, there are various stochastic and deterministic algorithms that can in principle estimate the posterior distribution and thus approximate the MAP estimate. Markov chain Monte Carlo algorithms are adopted in Article I. Stochastic greedy searching algorithms are used in Article II, III and V. A deterministic recursive algorithm is introduced in Article IV. The details of these algorithms are discussed in Chapter 4.

2.2 Semi-Supervised and Supervised Predictive Classification

Classification is a typical task in supervised learning(Bishop, 2006). To make inference about the partition S for the observed data $\mathbf{X}$, a set of training data $\mathbf{Z}$ is provided with *a priori* specified partition label set T. In Article I, the following two distinct settings for classification are considered: supervised classification, where the universe of all the possible labels is *a priori* given, and semi-supervised classification,

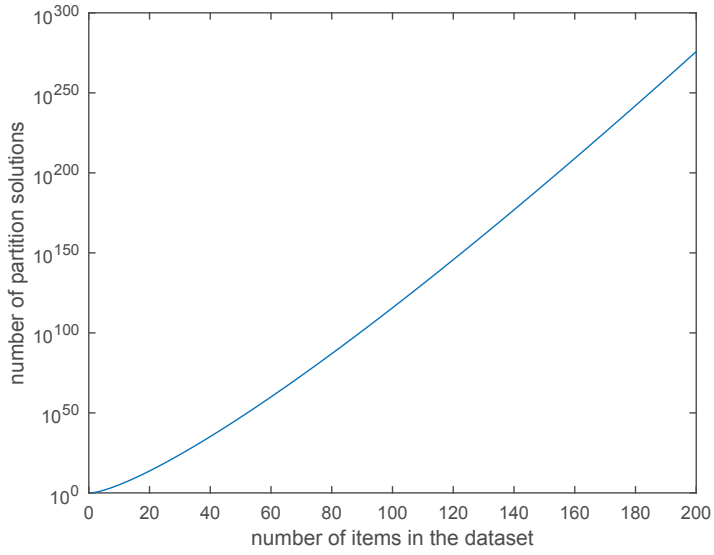

Figure 2.1: The Bell number increases exponentially as a function of size of the dataset.

where only part of the universe of possible labels is *a priori* defined, such that items are allowed to form previously unknown groups during the classification task. The term semi-supervised classification is generally used in various distinct ways. Apart from the definition used in this thesis, it may refer to an unsupervised clustering task with additional constraints (Basu, 2005). The classification task is similar to clustering in terms of finding the best allocation of the data $\mathbf{X}$ and the posterior distribution of S has the following form:

$$P(S|\mathbf{X},\mathbf{Z},T) = \frac{P(\mathbf{X}|S,\mathbf{Z},T)P(S|\mathbf{Z},T)}{P(\mathbf{X}|\mathbf{Z},T)}. \tag{2.10}$$

However, in supervised classification, as seen from the above expression, prior information $P(\Theta|S)$ of the parameter is updated by the training dataset $\mathbf{Z}$ and label set T, and therefore can be referred to as the *posterior predictive* inference. Therefore in supervised classification

$$P(\mathbf{X}|S,\mathbf{Z},T) = \int P(\mathbf{X}|S,\Theta)P(\Theta|S,\mathbf{Z},T)d\Theta. \tag{2.11}$$

and

$$P(S|T) = \int P(S|\Phi)P(\Phi|T)d\Phi, \tag{2.12}$$

where Θ and Φ denote the generating parameters of the model for features and labels, respectively.

In semi-supervised classification, only part of the information about model parameters can be updated in the light of the training data, which leads to a combination of prior and posterior predictive inferences. The corresponding distributions have the form

$$P(\mathbf{X}|S, \mathbf{Z}, T) = \int P(\mathbf{X}^T|S^T, \Theta)P(\mathbf{X}^U|S^U, \Theta)P(\Theta|S^T, S^U, \mathbf{Z}, T))d\Theta. \tag{2.13}$$

and

$$P(S|T) = \int P(S^T, S^U|\Phi)P(\Phi|T)d\Phi, \tag{2.14}$$

where $\mathbf{X}^T$ and S^T represent the part of data and the labels for which information has been updated using training data, and $\mathbf{X}^U$ and S^U represent the part of data and its partition for which information is not available from the training data.

Marginal Classifier

In classification tasks, it is conventional to classify the each item in $\mathbf{X}$ individually and independently of the other candidate items. This approach is based on the generating i.i.d. assumption, where the features of any item are independent of the features of any any other item conditional on the fixed generative probability measure. In the Bayesian framework, this assumption leads to the standard marginal predictive classifier (Ripley, 1996).

Definition 2.2.1. *Supervised predictive marginal classifier. Let* $\mathbf{x}^{(n)} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$ *denote a dataset containing* n *items, and* $\mathbf{s}^{(n)} = (s_1, s_2, \ldots, s_n)$ *represent the labels of the items. Given a training dataset* $\mathbf{z}^{(m)}$ *with* m *items and its corresponding labels* $\mathbf{t}^{(m)} = (t_1, t_2, \ldots, t_m)$*, a supervised*

predictive marginal classifier yields the following posterior distribution for the label s_i *of the* $\mathbf{x}_i$ *in* $\mathbf{x}^{(n)}$.

$$p(s_i|\mathbf{x}_i, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) = \frac{p(\mathbf{x}_i|s_i, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})p(s_i|\mathbf{t}^{(m)})}{\sum_{s_i \in \mathcal{T}} p(\mathbf{x}_i|s_i, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})p(s_i|\mathbf{t}^{(m)})}, \tag{2.15}$$

where $p(\mathbf{x}_i|s_i, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) = \int p(\mathbf{x}_i|s_i, \theta)p(\theta|s_i, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})d\theta$ *is the predictive probability distribution of* $\mathbf{x}_i$ *with label* s_i*, and* $p(s_i|\mathbf{t}^{(m)}) = \int p(s_i|\phi)p(\phi|\mathbf{t}^{(m)})d\phi$ *is the prior probability of label* s_i *and* $\mathcal{T}$ *is the set of possible values of labels defined in* $\mathbf{t}^{(m)}$.

The optimal solution of the classification task corresponding to zero-one loss can be obtained by finding the mode of the posterior distribution in 2.15:

$$\begin{aligned} \hat{s_i} &= \arg\max_{s_i \in \mathcal{T}} p(s_i|\mathbf{x}_i, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) \\ &= \arg\max_{s_i \in \mathcal{T}} p(\mathbf{x}_i|s_i, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})p(s_i|\mathbf{t}^{(m)}). \end{aligned} \tag{2.16}$$

A predictive marginal classifier that assigns the label s_i with value $\hat{s_i}$ by applying the MAP estimate minimizes the averaged risk of misclassification (Ripley, 1996). This is proven in Nadas (1985) in an application to speech recognition.

For semi-supervised marginal classification, besides $\mathcal{T}$, s_i can take a value c^* referring to unknown classes which are not present in $\mathbf{t}^{(m)}$ and therefore we have semi-supervised predictive marginal classifier defined as follows:

Definition 2.2.2. *Semi-supervised predictive marginal classifier. Let* $\mathbf{x}^{(n)} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$ *denote a dataset containing* n *items, and* $\mathbf{s}^{(n)} = (s_1, s_2, \ldots, s_n)$ *represent the labels of the items. Given a training dataset* $\mathbf{z}^{(m)}$ *with* m *items and its corresponding labels* $\mathbf{t}^{(m)} = (t_1, t_2, \ldots, t_m)$*, a semi-supervised predictive marginal classifier yields the following posterior distribution for the label* s_i *of the* $\mathbf{x}_i$ *in* $\mathbf{x}^{(n)}$.

$$p(s_i|\mathbf{x}_i, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) = \frac{p(\mathbf{x}_i|s_i, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})p(s_i|\mathbf{t}^{(m)})}{\sum_{s_i \in \{\mathcal{T}, c^*\}} p(\mathbf{x}_i|s_i, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})p(s_i|\mathbf{t}^{(m)})}, \tag{2.17}$$

where $p(\mathbf{x}_i|s_i, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})p(s_i|\mathbf{z}^{(m)}, \mathbf{t}^{(m)}) = \int p(\mathbf{x}_i|s_i, \theta)p(\theta|s_i)d\theta \int p(s_i|\phi)p(\phi)d\phi$ *when* $s_i = c^*$*, since no predictive information can be retrieved from the training data if the item is assigned to an unknown class.*

The optimal solution of the classification can be obtained by the following MAP estimate:

$$\begin{aligned}\hat{s}_i &= \arg \max_{s_i \in \{\mathcal{T}, c^*\}} p(s_i|\mathbf{x}_i, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) \\ &= \arg \max_{s_i \in \{\mathcal{T}, c^*\}} p(\mathbf{x}_i|s_i, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})p(s_i|\mathbf{t}^{(m)}). \end{aligned} \tag{2.18}$$

Simultaneous Classifier

In classification, the items in $\mathbf{X}$ can be considered independent with each other only conditionally on a given generative probability measure, which is not in practice exactly known. Therefore, marginal dependence exists between the test items in the predictive probability distribution and inductive learning theory based on predictive modelling implies that items should be treated simultaneously.

Definition 2.2.3. *Supervised predictive simultaneous classifier. Let* $\mathbf{x}^{(n)} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$ *denote a dataset contains* n *items, and* $\mathbf{s}^{(n)} = (s_1, s_2, \ldots, s_n)$ *represent the labels of the items. Given a training dataset* $\mathbf{z}^{(m)}$ *with* m *items and its corresponding labels* $\mathbf{t}^{(m)} = (t_1, t_2, \ldots, t_m)$*, a supervised predictive simultaneous classifier yields the following joint posterior probability* $p(\mathbf{s}^{(n)}|\mathbf{x}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})$ *for the labels* $\mathbf{s}^{(n)}$ *of all items in* $\mathbf{x}^{(n)}$*.*

$$p(\mathbf{s}^{(n)}|\mathbf{x}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) = \frac{p(\mathbf{x}^{(n)}|\mathbf{s}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})p(\mathbf{s}^{(n)}|\mathbf{t}^{(m)})}{\sum_{\mathbf{s}^{(n)} \in \mathcal{T}^{(n)}} p(\mathbf{x}^{(n)}|\mathbf{s}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})p(\mathbf{s}^{(n)}|\mathbf{t}^{(m)})}, \tag{2.19}$$

and

$$p(\mathbf{x}^{(n)}|\mathbf{s}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) = \int p(\mathbf{x}^{(n)}|\mathbf{s}^{(n)}, \theta)p(\theta|\mathbf{s}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})d\theta \tag{2.20}$$

$$p(\mathbf{s}^{(n)}|\mathbf{t}^{(m)}) = \int p(\mathbf{s}^{(n)}|\phi)p(\phi|\mathbf{t}^{(m)})d\phi \tag{2.21}$$

where $\mathcal{T}^{(n)}$ represent the set of all the possible solutions of label assignment for items in $\mathbf{x}^{(n)}$, with the values of labels defined in $\mathbf{t}^{(m)}$.

The optimal solution $\hat{\mathbf{s}}^{(n)}$ can be obtained by finding the MAP estimator:

$$\begin{aligned}\hat{\mathbf{s}}^{(n)} &= \arg \max_{\mathbf{s}^{(n)} \in \mathcal{T}^{(n)}} p(\mathbf{s}^{(n)} | \mathbf{x}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) \\ &= \arg \max_{\mathbf{s}^{(n)} \in \mathcal{T}^{(n)}} p(\mathbf{x}^{(n)} | \mathbf{s}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) p(\mathbf{s}^{(n)} | \mathbf{t}^{(m)}). \end{aligned} \tag{2.22}$$

A simultaneous predictive classifier which assigns the labels $\mathbf{s}^{(n)}$ with value $\hat{\mathbf{s}}^{(n)}$ by maximizing the posterior (2.19), is optimal under the 0-1 loss function rewarding only decisions where all predicted labels are correct (Ripley, 1996). Ripley (1996) claimed that marginal classifier is less accurate when $n > 1$, since one can benefit from further learning about the uncertainty of the model parameters by using other items in $\mathbf{x}^{(n)}$, when the values of parameters are unknown.

For a semi-supervised simultaneous classifier, items in $\mathbf{x}^{(n)}$ are allowed to form unknown groups which are not defined in $\mathbf{t}^{(m)}$, and therefore each element in $\mathbf{s}^{(n)}$ can take values other than those in $\mathcal{T}$. These values for potential unknown groups are denoted as $\mathcal{C}$, and $\{\mathcal{T}, \mathcal{C}\}^{(n)}$ represents all the possible solutions for the semi-supervised simultaneous classification.

Definition 2.2.4. *Semi-supervised predictive simultaneous classifier. Let $\mathbf{x}^{(n)} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$ denote a dataset containing n items, and $\mathbf{s}^{(n)} = (s_1, s_2, \ldots, s_n)$ represent the labels of the items. Given a training dataset $\mathbf{z}^{(m)}$ with m items and its corresponding labels $\mathbf{t}^{(m)} = (t_1, t_2, \ldots, t_m)$, a supervised predictive simultaneous classifier yields the following joint posterior probability $p(\mathbf{s}^{(n)} | \mathbf{x}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})$ for the labels $\mathbf{s}^{(n)}$ of all items in $\mathbf{x}^{(n)}$.*

$$\begin{aligned} &p(\mathbf{s}^{(n)} | \mathbf{x}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) = \\ &\frac{p(\mathbf{x}^{(n)} | \mathbf{s}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) p(\mathbf{s}^{(n)} | \mathbf{t}^{(m)})}{\sum_{\mathbf{s}^{(n)} \in \{\mathcal{T}, \mathcal{C}\}^{(n)}} p(\mathbf{x}^{(n)} | \mathbf{s}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) p(\mathbf{s}^{(n)} | \mathbf{t}^{(m)})}, \end{aligned} \tag{2.23}$$

where

$$p(\mathbf{x}^{(n)}|\mathbf{s}^{(n)},\mathbf{z}^{(m)},\mathbf{t}^{(m)}) = \int p(\mathbf{x}^{(n)^T}|\mathbf{s}^{(n)^T},\theta)p(\mathbf{x}^{(n)^U}|\mathbf{s}^{(n)^U},\theta) \quad (2.24)$$
$$p(\theta|\mathbf{s}^{(n)^T},\mathbf{s}^{(n)^U},\mathbf{z}^{(m)},\mathbf{t}^{(m)})d\theta$$

and

$$p(\mathbf{s}^{(n)}|\mathbf{t}^{(m)}) = \int p(\mathbf{s}^{(n)^T},\mathbf{s}^{(n)^U}|\phi)p(\phi|\mathbf{t}^{(m)})d\phi \quad (2.25)$$

The optimal solution $\hat{\mathbf{s}}^{(n)}$ can be obtained by finding the MAP estimate:

$$\begin{aligned}\hat{\mathbf{s}}^{(n)} &= \arg \max_{\mathbf{s}^{(n)}\in\{\mathcal{T},\mathcal{C}\}^{(n)}} p(\mathbf{s}^{(n)}|\mathbf{x}^{(n)},\mathbf{z}^{(m)},\mathbf{t}^{(m)}) \\ &= \arg \max_{\mathbf{s}^{(n)}\in\{\mathcal{T},\mathcal{C}\}^{(n)}} p(\mathbf{x}^{(n)}|\mathbf{s}^{(n)},\mathbf{z}^{(m)},\mathbf{t}^{(m)})p(\mathbf{s}^{(n)}|\mathbf{t}^{(m)}). \quad (2.26)\end{aligned}$$

The above simultaneous classifier is an optimal rule under the zero-one loss function

$$L_1(\mathbf{s}^{(n)},\mathbf{s}^{(n)*}) = \begin{cases} 0 & \text{if } \mathbf{s}^{(n)} = \mathbf{s}^{(n)*} \\ 1 & \text{if } \mathbf{s}^{(n)} \neq \mathbf{s}^{(n)*}, \end{cases} \quad (2.27)$$

which imposes a constant loss on all label sequences $\mathbf{s}^{(n)}$ apart from the sequence of true label $\mathbf{s}^{(n)*}$(Bernardo and Smith, 1994). However, Ripley (1991) showed that in the context of image analysis, the simultaneous MAP rule is not optimal under a more pragmatic loss function

$$L_2(\mathbf{s}^{(n)},\mathbf{s}^{(n)*}) = \sum_{i=1}^{n} I(s_i \neq s_i^*) \quad (2.28)$$

which aims at ensuring that the rate of incorrect labels is minimized.

Marginalized Classifier

The optimal classifier under the loss function L_2 can be established by considering the marginal distribution each s_i from the joint posterior distribution derived in predictive simultaneous classifier. Such a

principle is referred to as a marginalized classifier, in contrast to the standard marginal classifier which treats all test items independent of each other. Under marginalized predictive framework, all the labels of remaining items in $\mathbf{x}^{(n)}$ denoted as $\mathbf{s}_{-i}^{(n)}$ is considered as nuisance parameters in the classification task for assigning label s_i for each item $\mathbf{x}_i$.

Definition 2.2.5. *Supervised predictive marginalized classifier. Let* $\mathbf{x}^{(n)} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$ *denote a dataset containing* n *items, and* $\mathbf{s}^{(n)} = (s_1, s_2, \ldots, s_n)$ *represent the labels of the items. Given a training dataset* $\mathbf{z}^{(m)}$ *with* m *items and its corresponding labels* $\mathbf{t}^{(m)} = (t_1, t_2, \ldots, t_m)$*, a supervised predictive marginalized classifier yields the posterior distribution of* s_i *by marginalization of the joint posterior distribution in (2.19) over all the possible classification solutions of remaining items in* $\mathbf{x}^{(n)}$

$$p(s_i|\mathbf{x}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) = \frac{p(\mathbf{x}^{(n)}|\mathbf{s}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})p(\mathbf{s}^{(n)}|\mathbf{t}^{(m)})}{\sum_{s_i \in \mathcal{T}} \sum_{\mathbf{s}_{-i}^{(n-1)} \in \mathcal{T}_{-i}^{(n-1)}} p(\mathbf{x}^{(n)}|\mathbf{s}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})p(\mathbf{s}^{(n)}|\mathbf{t}^{(m)})}, \tag{2.29}$$

where $\mathbf{s}_{-i}^{n-1} = (s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n)$ *denote the labels of all the items in* $\mathbf{x}^{(n)}$ *except* $\mathbf{x}_i$*, and* $\mathcal{T}_{-i}^{(n-1)}$ *represent a set of all the possible values of* $\mathbf{s}_{-i}^{n-1}$ *defined by* $\mathbf{t}^{(m)}$*.*

The optimal classification solution $\hat{\mathbf{s}}_i$ can be obtained by the MAP estimate:

$$\begin{aligned} \hat{s_i} &= \arg\max_{s_i \in \mathcal{T}} p(s_i|\mathbf{x}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) \\ &= \arg\max_{s_i \in \mathcal{T}} \sum_{\mathbf{s}_{-i}^{(n-1)} \in \mathcal{T}^{(n-1)}} p(\mathbf{x}^{(n)}|\mathbf{s}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})p(\mathbf{s}^{(n)}|\mathbf{t}^{(m)}). \end{aligned} \tag{2.30}$$

The marginal uncertainty about the classification assignment of $\mathbf{s}_i$ represented by marginalized classifier follows from the application of the law of total probability on the simultaneous classifier. The marginalization operation applied to the simultaneous classifier aims at quantifying the total evidence in the data for supporting any particular origin of $\mathbf{x}_i$. This can be of particular interest in applications

where classification is related to some sensitive information about the items (e.g. in forensics applications) and classifiers need be based on a cautious strategy, which eventually prevents classification in the presence of a too large uncertainty about the origin of a particular item.

A semi-supervised marginalized Classifier can be derived analogically from the posterior distribution (2.23) from semi-supervised predictive simultaneous classier.

Definition 2.2.6. *Semi supervised predictive marginalized classifier. Let* $\mathbf{x}^{(n)} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$ *denote a dataset containing* n *items, and* $\mathbf{s}^{(n)} = (s_1, s_2, \ldots, s_n)$ *represent the labels of the items. Given a training dataset* $\mathbf{z}^{(m)}$ *with* m *items and its corresponding labels* $\mathbf{t}^{(m)} = (t_1, t_2, \ldots, t_m)$*, a supervised predictive marginalized classifier yields the posterior distribution of* s_i *by marginalization of the joint posterior distribution in (2.19) over all the possible classification solutions of remaining items in* $\mathbf{x}^{(n)}$

$$p(s_i|\mathbf{x}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) = \frac{p(\mathbf{x}^{(n)}|\mathbf{s}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})p(\mathbf{s}^{(n)}|\mathbf{t}^{(m)})}{\sum_{s_i \in \{\mathcal{T},\mathcal{C}\}} \sum_{\mathbf{s}_{-i}^{(n)} \in \{\mathcal{T},\mathcal{C}\}_{-i}^{(n-1)}} p(\mathbf{x}^{(n)}|\mathbf{s}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})p(\mathbf{s}^{(n)}|\mathbf{t}^{(m)})}, \quad (2.31)$$

The optimal solution $\hat{\mathbf{s}}_i$ can be obtained by the MAP estimate:

$$\begin{aligned} \hat{s_i} &= \arg\max_{s_i \in \mathcal{T}} p(s_i|\mathbf{x}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) \\ &= \arg\max_{s_i \in \{\mathcal{T},\mathcal{C}\}} \sum_{s_{-i} \in \{\mathcal{T},\mathcal{C}\}^{(n-1)}} p(\mathbf{x}^{(n)}|\mathbf{s}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})p(\mathbf{s}^{(n)}|\mathbf{t}^{(m)}). \quad (2.32) \end{aligned}$$

Chapter 3

Graphical Models for Bayesian Learning

In Chapter 2, different Bayesian unsupervised and supervised learning principles were introduced. However, all these principles are model based, i.e. the likelihood function $P(X|H)$ and the prior distribution $P(H)$ need be specified. In this chapter, we consider statistical models for representing sequence data.

3.1 Sparse Markov Models

One of the simplest models to encode sequence data is the Markov chain. Given finite alphabet set with J symbols, we label the symbols with integers and denote the alphabet set as $\mathcal{X} = 1, \ldots . J$.

Definition 3.1.1. *Markov chain (MC). Let* $\{X_t\}_{t=0}^{\infty}$ *be a sequence of random variables. If for all* $t \geq 1$ *and* $j_0, j_1, \ldots, j_t \in \mathcal{X}$*,*

$$P(X_t = j_t | X_{t-1} = j_{t-1}, \ldots, X_0 = j_0) = P(X_t = j_t | X_{t-1} = j_{t-1}), \tag{3.1}$$

then $\{X_t\}_{i=0}^{\infty}$ *is called a Markov chain.*

The possible values of X_t form a countable set $\mathcal{X}$ called the state space of the chain. The Markov property in Definition 3.1 indicates that the conditional probability distribution for the current variable X_t of the sequence depends only on the state of the previous variable X_{t-1},

and not additionally on the state of earlier variables. In a *time homogeneous* Markov chain, the transition probabilities $P(X_t = i|X_{t-1} = j), a, b \in \mathcal{X}$ are independent of t and therefore can be represented by the following transition Matrix Θ with $p_{i|j} = P(X_t = j|X_{t-1} = i)$:

$$\Theta = \begin{pmatrix} p_{1|1} & \dots & p_{1|J} \\ \vdots & \dots & \vdots \\ p_{J|1} & \dots & p_{J|J} \end{pmatrix} \tag{3.2}$$

For modelling sequence data having a more complicated dependence structure, higher order Markov chains can be considered:

Definition 3.1.2. *Markov chain (MC) of mth order. Let $\{X_t\}_{n=0}^{\infty}$ be a sequence of random variables. If for all $t \geq m$ and $j_0, j_1, \dots, j_t \in \mathcal{X}$,*

$$\begin{aligned} P(X_t = j_t|X_{t-1} = j_{t-1}, \dots, X_0 = j_0) = \\ P(X_t = j_t|X_{t-1} = j_{t-1}, \dots, X_{t-m} = j_{t-m}), \end{aligned} \tag{3.3}$$

for a positive integer m, then $\{X_n\}_{n=0}^{\infty}$ is called a Markov chain of mth order.

For a *time homogenous* Markov Chain of mth order, the size of transition Matrix Θ is $|J|^m \times J$. The number of free parameters of the model grows exponentially with the order m. Therefore, estimating a Markov chain with large m requires a large amount of data and may imply substantial computational cost. However, in a higher order Markov model, the effective length of dependence is not necessarily a constant. By pruning any redundant dependency, model complexity can be significantly reduced. Such approaches are termed as Variable order Markov models, with pioneering work introduced in Rissanen et al. (1983)

Definition 3.1.3. *Variable length Markov chain (VLMC). Let $\{X_t\}_{t=0}^{\infty}$ be a time homogeneous Markov chain of mth order. Denote by $c_{\text{pre}} : \mathcal{X}^m \to \cup_{j=0}^{m} \mathcal{X}^j$ a function which maps $x_{t-1}, \dots, x_{t-m} \to x_{t-1}, \dots, x_{t-l}$ where*

$$\begin{aligned} &l = l(x_{t-1}, \dots, x_{t-m}) = \\ &\min\{k \in \mathbb{Z}_0^+; P(X_t = j_t|X_{t-1} = x_{t-1}, \dots, X_{t-m} = x_{t-m}) = \\ &P(X_t = j_t|X_{t-1} = x_{t-1}, \dots, X_{t-k} = x_{t-k}) \text{ for all } j_t \in \mathcal{X}\}, \end{aligned}$$

such that $l = 0$ corresponds to independence. Then l is a variable length memory and $c_{\text{pre}}(\cdot)$ is the preliminary context function. Final context function $c(\cdot)$ is obtained by lumping together some of the values of $c_{\text{pre}}(\cdot)$ that share the second to last symbol. A Markov chain of mth order with a variable length memory l is called a Variable length Markov chain of order p where p is the smallest integer such that $l(x_{n-1}, \ldots, x_{n-m}) \leq p \leq m$ for all $x_{n-1}, \ldots, x_{n-m} \in \mathcal{X}^m$.

In Article II, we consider modelling data sequences using Bayesian inference on sparse Markov chain, which need not correspond to a hierarchical representation of contexts used in variable length Markov chains. The sparse Markov models are more general and can further reduce the parameter space of the original Markov model and therefore lead to attractive properties on multiple applications, e.g. prediction and data compression.

Definition 3.1.4. *Sparse Markov chain (SMC) Consider a time homogeneous Markov chain $MC(m)$ of order m. Let $S = \{s_1, \ldots, s_k\}$ be a partition of $\mathcal{X} = \{1, \ldots, J\}^m$ such that $\mathbf{p}_{i|\cdot} = \mathbf{p}_{j|\cdot} = \boldsymbol{\theta}_c$ for all pairs of $\{i, j\}$, $i, j \in s_c$, $c = 1, \ldots, k$, and $\mathcal{P} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_k\}$ is the set of k distinct transition probability vectors. The pair $(S, \mathcal{P})$ forms a sparse Markov chain model from $MC(m)$.*

Bayesian Representation for Sparse Markov Chain

Consider an SMC model defined by the pair $(S, \mathcal{P})$, where we have k vectors of parameters $\{\mathbf{p}_{c|\cdot} : c = 1, ..., k\}$. An analytical expression for the marginal likelihood of observed sequence data given S was derived in Article II. Let $\theta \in \Theta$ denote collectively the set of quantitative parameters of an SMC model. A conjugate multivariate Dirichlet prior for the matrix of transition probabilities (see e.g. Koski, 2001) has the expression

$$p(\theta|\alpha, q) = \prod_{c=1}^{k} \left[\frac{\Gamma(\alpha)}{\prod_{j=1}^{J} \Gamma(\alpha q_j)} \prod_{j=1}^{J} p_{c|j}^{\alpha q_j - 1} \right], \alpha > 0, q_j > 0, \sum_{j=1}^{J} q_j = 1 \tag{3.4}$$

The likelihood of an observed data sequence $\mathbf{x} = x_0 x_1 \cdots x_n$ under the SMC model can be expressed as

$$p(\mathbf{x}|\theta, S) \propto \prod_{i \in \mathcal{X}^m} \prod_{j=1}^{J} p_{i|j}^{n_{i|j}} = \prod_{c=1}^{k} \prod_{j=1}^{J} p_{c|j}^{n_{c|j}}, \tag{3.5}$$

where $n_{i|j}$ is the observed count of transitions from the state i to j in $\mathbf{x}$ and $n_{c|j} = \sum_{i \in s_c} n_{i|j}$, and the initial distribution is not of interest in the model and thus omitted.

Consequently, the marginal likelihood $p(\mathbf{x}|S)$ of $\mathbf{x}$ is available analytically by applying the properties of Dirichlet distribution, such that

$$\begin{aligned} p(\mathbf{x}|S) &\propto \int_{\theta \in \Theta} p(\mathbf{x}|\theta, S) p(\theta|\alpha, q) d\theta \\ &\propto \int_{\theta \in \Theta} \left[\prod_{c=1}^{k} \frac{\Gamma(\alpha)}{\prod_{j=1}^{J} \Gamma(\alpha q_j)} \prod_{j=1}^{J} p_{c|j}^{\alpha q_j - 1} \prod_{j=1}^{J} p_{c|j}^{n_{c|j}} \right] d\theta \\ &\propto \prod_{c=1}^{k} \frac{\Gamma(\alpha)}{\prod_{j=1}^{J} \Gamma(\alpha q_j)} \frac{\prod_{j=1}^{J} \Gamma(n_{c|j} + \alpha q_j)}{\Gamma((\sum_{j=1}^{J} n_{c|j}) + \alpha)}, \end{aligned} \tag{3.6}$$

where $\Gamma(\cdot)$ is the Gamma function. Inference about S can be done using the posterior distribution

$$p(S|\mathbf{x}) \propto p(\mathbf{x}|S) p(S). \tag{3.7}$$

For simplicity, a uniform prior $p(S) = 1/B(|\mathcal{X}|^m)$ can be assigned over the space of all possible partitions of $\mathcal{X}^m$ to obtain the posterior probability of S, where $B(n)$ is the Bell number of n. However, alternative priors could also be adopted. For example, a Dirichlet process (DP) prior assigns the following probability on the partitions

$$p(S) \propto \beta^k \prod_{c=1}^{k} \Gamma(|s_c|), \tag{3.8}$$

where β is a concentration parameter governing the implied probability mass over possible values of k. A uniform prior on k, given an

upper limit $K \leq |\mathcal{X}|^m$, distributes evenly the same probability mass $1/K$ over all partitions with a given number of classes k. Since the number of ways of partitioning a set of $|\mathcal{X}|^m$ elements into k non-empty subsets is given by the Stirling number of the second kind , the probability of a partition S with k clusters equals: $p(S) = K^{-1}\left\{ {|\mathcal{X}|^m \atop k} \right\}^{-1}$. Both of these alternative priors imply penalties for any particular partition when k increases and therefore favour the partition with smaller k. It should nevertheless be noted that the Dirichlet prior for the transition probability vectors already imposes an increasing penalty as a function of the number of clusters in the partition.

3.2 Hidden Markov Models

Hidden Markov models (HMM) are applied with particular success to deal with multiple types of time-ordered data, e.g. speech recognition (Huang et al., 1990; Huo and Lee, 1997; Huo et al., 1997; Huo and Lee, 2000) and image recognizing problems (Yamato et al., 1992), where different HMMs are used to construct the model for the feature. Given a Markov chain, a hidden Markov model associates each state of the Markov chain with a probability distribution to generate an observation, while the state of the underlying Markov chain remains unobserved. In article I, Hidden Markov models are introduced to model the predictive classification framework for sequential data.

Definition 3.2.1. *Let* $\mathbf{x}^{(n)} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$ *denote a dataset containing* n *items, and* $\mathbf{s}^{(n)} = (s_1, s_2, \ldots, s_n)$ *represent the labels of the items. A standard hidden Markov model for classification structure is constructed as follows:*

(A) The label sequence $\mathbf{s}^{(n)} = \{s_t\}_{t=1}^n$ *follows a Markov chain with a finite state space* $C = \{1, ..., k\}$ *with* k *states. The transition probabilities*

$$\psi_{i|j} = p(s_t = j | s_{t-1} = i), t \geqslant 2, i, j \in C, \tag{3.9}$$

are assumed to be time-homogeneous. Thus, the transition matrix is represented by

$$\underline{\boldsymbol{\psi}} = (\psi_{i|j})_{i=1,j=1}^{k,k}, \psi_{i|j} \geqslant 0, \sum_{j=1}^{k} \psi_{i|j} = 1. \tag{3.10}$$

The initial state s_1 *at time* 1 *is specified by an initial distribution*

$$\psi^{Init} = (\psi_1^{Init}, ..., \psi_k^{Init}), \tag{3.11}$$

where $\psi_i^{Init} = p(s_1 = i)$.

(B) The sequence of item vectors $\mathbf{x}^{(n)} = \{\mathbf{x}_t\}_{t=1}^n$ *with a finite state space* $\mathcal{X}$ *and their label sequence* $\mathbf{s}^{(n)} = \{s_t\}_{t=1}^n$, *at any time* t *are related by the conditional probability distributions*

$$p(\mathbf{x}_t | s_t = c, \boldsymbol{\theta}_c) = \prod_{j=1}^{d} \prod_{l=1}^{r_j} \theta_{cjl}^{\mathbf{1}_l(x_{tj})}. \tag{3.12}$$

where $\mathbf{1}_l(x_{tj})$ *is an indicator function which equals* 1, *if* $\mathbf{x}_t$ *has the value* l *on the dimension* j *and otherwise* 0.

(C) For any sequence of labels $\mathbf{s}^{(n)} = \{s_t\}_{t=1}^n$, *the probability of observing item vector sequence* $\mathbf{x}^{(n)} = \{\mathbf{x}_t\}_{t=1}^n$ *is*

$$p(\mathbf{x}^{(n)} | \mathbf{s}^{(n)}, \boldsymbol{\theta}) = \prod_{t=1}^{n} p(\mathbf{x}_t | s_t = c, \boldsymbol{\theta}_c) = \prod_{t=1}^{n} \prod_{j=1}^{d} \prod_{l=1}^{r_j} \theta_{s_t jl}^{\mathbf{1}_l(x_{tj})} \tag{3.13}$$

3.3 Stratified Graphical Models

In general the features of observed data are not necessarily conditionally independent given a label and can even have more complicated dependent structure than the Markov chain form discussed previously. Madden (2009) showed that the Bayesian network model introduced by Friedman et al. (1997) has the potential to represent data structure more faithfully and outperform the naive Bayes model in classification task if proper parameter estimation is applied. In Article III, graphical models (GM) are introduced into the predictive classification framework and it is further developed to allow local context-specific independences on top of the conditional independences.

A graphical model (GM) for data $\mathbf{X}$ is defined by the undirected graph $G = G(\Delta, E)$ with a set of nodes Δ and of a set of undirected edges $E \subseteq (\Delta \times \Delta)$ and a joint distribution P_Δ over the variables X_Δ

satisfying a set of independences introduced by G. The outcome space for the variables $\mathbf{X}_A$, where $A \subseteq \Delta$, is denoted by $\mathcal{X}_A$ and a certain outcome is denoted by $x_A \in \mathcal{X}_A$. In Nyman et al. (2014), stratum is introduced to define the context-specific independences.

Definition 3.3.1. *Stratum Let the pair (G, P_Δ) be a GM for Δ. For all $\{\delta, \gamma\} \in E$, let $L_{\{\delta,\gamma\}}$ denote the set of nodes adjacent to both δ and γ. For a non-empty $L_{\{\delta,\gamma\}}$, define the stratum of the edge $\{\delta, \gamma\}$ as the subset $\mathcal{L}_{\{\delta,\gamma\}}$ of outcomes $x_{L_{\{\delta,\gamma\}}} \in \mathcal{X}_{L_{\{\delta,\gamma\}}}$ for which X_δ and X_γ are independent given $X_{L_{\{\delta,\gamma\}}} = x_{L_{\{\delta,\gamma\}}}$, i.e. $\mathcal{L}_{\{\delta,\gamma\}} = \{x_{L_{\{\delta,\gamma\}}} \in \mathcal{X}_{L_{\{\delta,\gamma\}}} : X_\delta \perp X_\gamma | X_{L_{\{\delta,\gamma\}}} = x_{L_{\{\delta,\gamma\}}}\}$*

The corresponding graphical models are called stratified graphical models(SGM).

Definition 3.3.2. *stratified graphical Model (SGM) A stratified graphical model is defined by the triple (G, L, P_Δ), where G is the underlying graph, L equals the joint collection of all strata $\mathcal{L}_{\{\delta,\gamma\}}$ for the edges of G, and P_Δ is a joint distribution on Δ which factorizes according to the restrictions imposed by G and L. The correspond graph is referred to as stratified graph (SG), and denoted by G_L.*

Consider an SG with a decomposable underlying graph G having the cliques $\mathcal{C}(G)$ and separators $S(G)$. The SG is decomposable if no strata are assigned to edges in any separator and all stratified edges in every clique have at least one node in common.

Definition 3.3.3. *Decomposable Stratified Graph Let (G, L) constitute an SG with G being Decomposable. Further, let E_L denote the set of all labelled edges, E_C the set of all edges in clique C, and E_S the set of all edges in the separators of G. The SG is defined as decomposable if*

$$E_L \cap E_S = \emptyset \tag{3.14}$$

and

$$E_L \cap E_C = \emptyset \text{ or } \bigcap_{\{\delta,\gamma\} \in E_L \cap E_C} \neq \emptyset \text{ for all } C \in \mathcal{C}(G) \tag{3.15}$$

Let $\mathbf{X}$ denote the data with $|\Delta|$ dimensions. For a given decomposable graphical model, we can define the likelihood of the dataset as

$$P(\mathbf{X}|G) = \frac{\prod_{C \in \mathcal{C}(G)} P_C(\mathbf{X}_C)}{\prod_{S \in \mathcal{S}(G)} P_S(\mathbf{X}_S)}, \tag{3.16}$$

where $\mathcal{C}(G)$ and $\mathcal{S}(G)$ are the cliques and separators, respectively, of the graph G. The analytic form of the predictive probability distribution is introduced in Article III and applied with the classifiers discussed in Chapter 2.

3.4 Hierarchical Modelling

Most model-based clustering algorithms assume that the features of the data are either continuous or discrete, and these types are not presented simultaneously within the same feature (Cheeseman et al., 1996; Morlini, 2012). In Article V, the data with mixed discrete and continuous type within the same feature are represented by a model with multiple level structure.

Consider the Bayesian predictive clustering frame work discussed in Chapter 2. Assume that the jth feature observed data X_j is associated with cumulative distribution function $F_j(x)$ for the observed value $x \in \mathbb{R}$, we define $\mathcal{D}_j := \{x : F_j(x) - F_j(x-) > 0\} \subset \mathbb{R}$, to be the set of discontinuity points with respect to F_j, where $F_j(x-)$ denotes the left-hand limit of F_j at x.By the properties of the distribution function and by virtue of $\mathcal{D}_j$ being countable, the type of X_j is discrete if

$$\sum_{x \in \mathbb{R}} [F_j(x) - F_j(x-)] = 1, \tag{3.17}$$

continuous if

$$\sum_{x \in \mathbb{R}} [F_j(x) - F_j(x-)] = 0, \tag{3.18}$$

and *mixed* if

$$0 < \sum_{x \in \mathbb{R}} [F_j(x) - F_j(x-)] < 1, \tag{3.19}$$

At the first level, we consider the observed value for random variable $boolean(X_j \in \mathcal{D}_j)$ with a probability distribution $b(x \in \mathcal{D}_j)$. The discrete observed values and continuous values are modelled separately by a discrete distribution f_d and a continuous distribution f_c, respectively. In Article V, multinomial distribution is adopted to model the categorical data, and continuous observations are described as an additional category and further modelled by Gaussian distribution.

Chapter 4

Algorithms for Bayesian Learning

4.1 EM algorithm

In Chapter 2, we concluded that the solutions of Bayesian clustering and classification can be obtained by finding maximum a *posteriori* (MAP) estimates of the predictive posterior distributions. The expectation-maximum (EM) algorithm introduced by Dempster et al. (1977) is an iterative approach for solving the maximum a *posteriori* (MAP) problems. In the case of Bayesian clustering introduced by Jääskinen et al. (2014), the EM algorithm is applied with a latent variable $\mathbf{s}^{(n)}$ and the unknown parameter θ to find the solution:

Data: input data: $\mathbf{x}^{(n)}, \epsilon$
Initialize θ_0 for θ and $k = 0$;
repeat
  Expectation Step: Calculate $Q(\theta|\theta_k) = E_{\mathbf{s}^{(n)}}(\ln p(\theta, \mathbf{s}^{(n)}|\mathbf{x}^{(\mathbf{n})}, \theta_k)) = \sum_{\mathbf{s}^{(n)}} \ln p(\theta, \mathbf{s}^{(n)}|\mathbf{x}^{(n)}, \theta_k)$;
  Maximization Step: Set $\theta_{k+1} \leftarrow \arg\max_{\theta} Q(\theta|\theta_k)$ and $k = k + 1$
until $\theta_{k+1} - \theta_k < \epsilon$;
Return $\hat{\mathbf{s}}^{(n)}$ according to $\arg\max_{\mathbf{s}^{(n)}} p(\mathbf{s}^{(n)}|\mathbf{x}^{(n)}, \theta_k)$

Algorithm 1: Expectation-Maximization for Bayesian Clustering

The algorithm converges when the difference between θ_k and θ_{k+1} is below a pre-defined threshold. A upper bound of the number of iterations can also be set to control the time limit of the algorithm. For each iteration, it holds that

$$p(\theta_{k+1}|\mathbf{x}^{(n)}) \geqslant p(\theta_k|\mathbf{x}^{(n)}). \tag{4.1}$$

Therefore the EM algorithm leads to monotonically increasing probabilities approaching a local maximum of the target function.

4.2 Stochastic Greedy Search

Greedy Search for Bayesian Clustering

For Bayesian clustering tasks with large number of underlying clusters, EM algorithm can be computationally expensive and have poor performance in terms of the speed of convergence. In Article II, we present a stochastic search algorithm to learn the sparse Markov model. This greedy algorithm is introduced by Marttinen et al. (2006) for identifying evolutionary groups and conserved parts of the protein sequences. Given a Markov chain of order m defined on $\mathcal{X}$ with J symbols, under the definitions in 3.1, we have the following algorithm to estimate the SMC structure:

1. Initialize $S_t, t = 0$ with $|\mathcal{X}|^m$ singleton clusters and store for all pairs of states $u, v \in \mathcal{X}^m$ the similarity between posterior mean estimates of their transition probability vectors

$$s_{u,v} = \sum_{j=1}^{J} \left(\frac{n_{u|j} + \alpha q_j}{\sum_{j=1}^{J}(n_{u|j} + \alpha q_j)} - \frac{n_{v|j} + \alpha q_j}{\sum_{j=1}^{J}(n_{v|j} + \alpha q_j)} \right)^2, \tag{4.2}$$

2. Given the current value of $p(\mathbf{x}|S_t)$, apply the following operators iteratively until no more change can be done in S_t:

   (a) Move each state $u \in \mathcal{X}^m$ to the class c in S_t in a random order, if the moving leads to $p(\mathbf{x}|S_{t+1}) > p(\mathbf{x}|S_t)$.

(b) For each pair of classes $c_1, c_2 = 1, \ldots, k$, calculate $p(\mathbf{x}|S^*)$ for the S^* which merges classes c_1, c_2 in S_t. Set $S_{t+1} = S^*$ if $p(\mathbf{x}|S^*) > p(\mathbf{x}|S_t)$, otherwise set $S_{t+1} = S_t$.

(c) For each class $c = 1, \ldots, k$, use complete linkage algorithm e.g. (Mardia et al., 1979) with the similarity defined in (4.2) to split the class into two non-empty subsets of states, and set $S_{t+1} = S^*$ if $p(\mathbf{x}|S^*) > p(\mathbf{x}|S_t)$ for the resulting S^*, otherwise set $S_{t+1} = S_t$.

This greedy algorithm converges to a local maximum when no moves can increase $p(\mathbf{x}|S_t)$. Starting from different initial states may be used to find better solutions.

Greedy Search for Semi-Supervised and Supervised Classification

In the classification scenario, the search space of the algorithm is given by the training data. In Article I, for the supervised simultaneous classifier, we have the following greedy searching algorithm to find the maximum a posteriori estimates of the predictive posterior distribution.

Data: input data: $\mathbf{x}^{(n)}$, training data: $\mathbf{z}^{(m)}$, training labels: $\mathbf{t}^{(m)}$
Result: labels $\mathbf{s}^{(n)}$
Initialize each component s_i in $\mathbf{s}^{(n)}$;
repeat
  for $i = 1, \ldots, n$ **do**
    With the remaining assignment fixed, assign new value for s_i according to

$$s_i \leftarrow \arg\max_{s_i \in \mathcal{T}} p(\mathbf{s}^{(n)}|\mathbf{x}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) \tag{4.3}$$

  end
until *no change occurs in* $\mathbf{s}^{(n)}$;
$\hat{\mathbf{s}}^{(n)} = \mathbf{s}^{(n)}$;

Algorithm 2: Greedy search for supervised simultaneous classifier

This algorithm can be straightforwardly generalized to the semi-supervised classifier by extending the search space of each label to

include even classes lacking training data.

Data: input data: $\mathbf{x}^{(n)}$, training data: $\mathbf{z}^{(m)}$, training labels: $\mathbf{t}^{(m)}$
Result: labels $\mathbf{s}^{(n)}$
Initialize each component s_i in $\mathbf{s}^{(n)}$;
repeat
  for $i = 1, \ldots, n$ **do**
    With the remaining assignment fixed, assign new value for s_i according to

$$s_i \leftarrow \arg \max_{s_i \in \{\mathcal{T}, \mathcal{C}\}} p(\mathbf{s}^{(n)} | \mathbf{x}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) \qquad (4.4)$$

  end
until *no change occurs in* $\mathbf{s}^{(n)}$;
$\hat{\mathbf{s}}^{(n)} = \mathbf{s}^{(n)}$;

Algorithm 3: Greedy search for semi-supervised simultaneous classifier

By modifying the proof by Gyllenberg et al. (1997), we can show that these greedy searching algorithms increase the value of target function monotonically and will converge to local maxima over the space of the classification structure in a finite number of steps.

4.3 Deterministic Recursive Learning

In some learning tasks, the search space for the algorithm can be large. For example, for learning the SMC, as the order of the Markov chain increases, the parameter space grows exponentially and even the stochastic greedy optimization method may not converge fast enough. In Article IV, we developed a recursive algorithm for optimizing the partition for an SMC model by considering Delaunay triangulation of the parameter space of a higher order Markov Chain.

Data: Input sequence: $\{X_t\}_{t=1}^n$
Result: Sparse Markov model $(S, \mathcal{P})$
calculate the transition counts from $\{X_t\}_{t=1}^n$ for $MC(m)$;
estimate each transition probability distribution $\mathbf{p}$ of $MC(m)$ using the posterior mean based on the transition counts;
obtain Delaunay triangulation G of $\mathcal{X}^m$ by using values of free parameters in $\mathbf{p}$ as coordinates;
calculate the log Bayes factor $\log BF_{uv} = \log P(\mathbf{x}|M(\{u, v\})) - \log P(\mathbf{x}|M(\{u\}, \{v\}))$ for each edge u, v in G;
find the edge (u^*, v^*) with max log Bayes factor value w.;
set $\mathcal{U} = u^*$,$\mathcal{V} = v^*$ and $\mathcal{W} = w$, ;
while $\mathcal{W} > 0$ **do**
    merge $\mathcal{V}$ to $\mathcal{U}$ by the following steps: ;
    a) add the sufficient statistics counts of $\mathcal{V}$ to $\mathcal{U}$;
    b) for each node r in G who has a connection with $\mathcal{V}$, if edge $(\mathcal{U}, r)$ does not exist, redirect the edge $(\mathcal{V}, r)$ to $(\mathcal{U}, r)$;
    c) delete $\mathcal{V}$ from G;
    update the Bayes factors for all the edges (include the edges added by merging) connected to $\mathcal{U}$.;
    find the an edge $(u^{*'}, v^{*'})$ with max log Bayes factor value w';
    set $\mathcal{U} = u^{*'}$,$\mathcal{V} = v^{*'}$ and $\mathcal{W} = w'$;
end

Algorithm 4: Deterministic recursive learning for sparse Markov models

This methods yields a consistent estimate of the SMC structure in a local neighbourhood when the sequence length tends to infinity and considerably faster than the stochastic optimization used in Article II.

4.4 MCMC Algorithms

The EM algorithm and stochastic greedy search algorithms provide solutions to the maximum a *posteriori* (MAP) estimation problems which lead to numerical point estimates of the parameters associated with Bayesian learning. However, in some tasks, like the super-

vised and semi supervised *marginalized classification*, some alternative methods are required to evaluate the whole posterior distribution instead of only point estimates. Markov chain Monte Carlo algorithms represent a class of stochastic methods for sampling from the posterior distribution. By simulating from a reversible Markov Chain, the stationary distribution of the samples converges to the target posterior distribution.

Metropolis Hastings

Metropolis-Hastings(MH) algorithm uses a random acceptance/rejection rule to generate a Markov chain to approximate the posterior distribution. The acceptance ratio of a proposal S^* generated with probability $q(S^*|S)$, given the current value S, can be calculated with

$$r = \frac{m(S^*)q(S|S^*)}{m(S)q(S^*|S)} \tag{4.5}$$

The proposal value S^* is accepted with probability $\min(r, 1)$, otherwise rejected.

One important issue for the Metropolis-Hastings algorithms is to choose good search operators and proposal distribution to ensure a decent average acceptance ratio(Robert and Casella, 2013). In Bayesian clustering, Corander et al. (2004) used a search operator with four move types:

1. With probability 0.5, combine two randomly chosen classes c_i, c_j.
2. With probability 0.5, split a randomly chosen class c_i into two new classes, whose sizes are uniformly distributed between 1 and $|c_i| - 1$(the cardinality minus one), and whose elements are randomly chosen from c_i
3. Move an arbitrary sampling unit from a randomly chosen class c_i with cardinality $|c_i| > 1$, into another randomly chosen class c_j .
4. Choose one sampling unit randomly from each of two randomly chosen classes c_i and c_j , and exchange them between the classes.

This strategy is similar to that used in Dawson and Belkhir (2001). The proposal probabilities for the four move types have the following expressions:

1. $\binom{k}{2}^{-1}/2$, where k is the current number of clusters

2. $[|c_i|/2]^{-1}\binom{|c_i|}{|c_j|}^{-1}$ for $|c_j| < |s_i|/2$, and $[|c_i|/2]^{-1}\binom{|c_i|}{|c_j|}^{-1}/2$ for $|c_j| = |c_i|/2$, where c_j is one of the two new classes from the split of c_i, with minimal cardinality $|c_j|$.

3. $\tau(S)^{-1}(k-1)^{-1}, |c_i|^{-1}$, where $\tau(S)$ is the number of classes with cardinality larger than one, and c_i is the chosen class.

4. $\binom{k}{2}^{-1}|c_i|^{-1}|c_j|^{-1}$.

Gibbs Sampler

The Gibbs sampler offers an alternative way of implementing an MCMC algorithm(Geman and Geman, 1984). In Article I, it is adopted in the supervised and semi-supervised **marginalized classifier**. In Gibbs sampler, we simulate each single component $s_i, i = 1, \ldots, n$ of $\mathbf{s}^{(n)} = (s_1, \ldots, s_n)$ iteratively from its full conditional distribution.

Data: input data: $\mathbf{x}^{(n)}$, training data: $\mathbf{z}^{(m)}$, training labels: $\mathbf{t}^{(m)}$
Result: labels $\mathbf{s}^{(n)}$

Initialize each component s_i in $\mathbf{s}^{(n)}$ as $\mathbf{s}^{(n)}(0)$;
for *each iteration iter* $= 0, 1, \ldots$ **do**
  $\mathbf{s}^{(n)}(temp) \leftarrow \mathbf{s}^{(n)}(iter)$;
  for *i = 1,. . . , n* **do**
    draw a new value for $s_i(temp)$ from $\mathcal{T}$ according the posterior full conditional distribution:

$$p(s_i|\mathbf{x}^{(n)}, s_{-i}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) \propto p(\mathbf{s}^{(n)}|\mathbf{x}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})$$

  end
  $\mathbf{s}^{(n)}(u+1) = \mathbf{s}^{(n)}(temp)$
end

Algorithm 5: Gibbs Sampler for supervised marginalized classifier

Similar to the greedy searching algorithms for simultaneous classifier, this algorithm can also be generalized to the semi-supervised classifier by extending the search space of each label to include even classes lacking training data.

Data: input data: $\mathbf{x}^{(n)}$, training data: $\mathbf{z}^{(m)}$, training labels: $\mathbf{t}^{(m)}$
Result: labels $\mathbf{s}^{(n)}$

Initialize each component s_i in $\mathbf{s}^{(n)}$ as $\mathbf{s}^{(n)}(0)$;
for *each iteration* $iter = 0, 1, \ldots$ **do**
  $\mathbf{s}^{(n)}(temp) \leftarrow \mathbf{s}^{(n)}(iter)$;
  for $i = 1, \ldots, n$ **do**
    draw a new value for $s_i(temp)$ from $\{\mathcal{T}, \mathcal{C}\}$ according the posterior full conditional distribution:

$$p(s_i|\mathbf{x}^{(n)}, s_{-i}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)}) \propto p(\mathbf{s}^{(n)}|\mathbf{x}^{(n)}, \mathbf{z}^{(m)}, \mathbf{t}^{(m)})$$

  end
  $\mathbf{s}^{(n)}(u+1) = \mathbf{s}^{(n)}(temp)$
end

Algorithm 6: Gibbs Sampler for semi-supervised marginalized classifier

In fact, the Gibbs sampler is a special case of Metropolis-Hastings algorithms. Generally, it is easier to use since the new state generated from the proposal distribution is always accepted, which has made the algorithm hugely popular. However, its convergence may still be prohibitively slow for large target spaces.

Chapter 5

Discussion

In this thesis, we have presented predictive framework for Bayesian clustering and classification, as well as several models adopted for analysing specific type of datasets. In Article I, different rules for predictive Bayesian sequential classification are introduced, and the asymptotic properties of the classifiers are explored. In Article II, sparse Markov model is defined and its properties are investigated for analysing sequence data. Article III develops predictive inference for data with complex dependent features with graphical models and introduced stratified graphical models to allow label based independence between the features. The learning algorithm for SMC is further developed in Article IV by introducing a recursive deterministic approach that uses Delaunay triangulation and Bayes factors. Finally, the Bayesian clustering and classification frameworks are enabled to allow features with continuous and discrete values in one dimension by a hierarchical way of modelling.

One of the main topic in the thesis is model based inference in machine learning, especially in the classification scenario.

HMM vs i.i.d.

In Article I, different strategies of Hidden Markov models are introduced into the classification framework to formulate the predictive distribution for time-ordered sequential data. Compare to the i.i.d

model, the HMMs are illustrated to have superior performance in terms of classification accuracy on the synthetic and strong robustness even when the data do not follow closely the assumptions of the generating models.

SMC vs VLMC

In Article II, sparse Markov models are compared against different state-of-the-art variable order Markov models in data compression and different sequence classification applications. Sparse Markov models need not have a hierarchical structure as in context tree based variable order Markov models, which leads to a reduced parameter space and enhanced performance in data compression and sequence classification. However, learning the SMC structure requires more computational resources than those context tree based methods.

SGM vs GM

In Article III, the Naive Bayes model, graphical models and stratified graphical models are compared for predictive classification. Naive predictive Bayes classifier is simple and straightforward to use, however it often oversimplifies the model by assuming full conditional independence among the features. By introducing a graphical model, the dependence structure among the features can be captured to some extent. However, the stratified graphical models extend the graphical models to have a more precise and sparse representation of the dependence structure by introducing context-specific independences among the features. The SGM classifiers are illustrated to improve the rate of success with which the items are classified.

A second topic of this thesis is optimization algorithms for Bayesian clustering. Different algorithms are developed and adopted to handle large datasets and complicated optimization tasks.

Greedy Searching vs MCMC

We considered both greedy search algorithms and Markov chain Monte Carlo Samplers for Bayesian computation. MCMC samplers provide unbiased approximation to the full posterior distribution, but are computationally expensive and may even fail in practice to find the representative areas of the posterior as illustrated for clustering applications(Corander et al., 2006). In some circumstances, non-reversible Markov chains can be applied to achieve a faster exploration of the search space(Corander et al., 2006; Corander and Tang, 2007). In most of the cases covered by this thesis, point estimates of the optimal solution are preferred rather than the full posterior distribution. For this purpose, the greedy searching algorithms converge much faster than the MCMC samplers. However, these algorithms for maximum a *posteriori* (MAP) estimation do not provide estimates of the uncertainty, which can be problematic if the posterior distribution is not sufficiently concentrated around its mode. Nevertheless, it is possible to also use the greedy search algorithms to find the initial state of the MCMC samplers to ensure more rapid convergence of the Markov chain.

Future Work

There is still plenty of room in the research area of this thesis for future improvement. From the model point of view, it would be valuable to improve some of the models to handle massively larger datasets. From the algorithmic point of view, it would be attractive to extends the algorithms to handle much higher-dimensional optimization problems.

References

S. Basu. *Semi-supervised clustering: probabilistic models, algorithms and experiments*. University of Texas at Austin, 2005.

J. M. Bernardo and A. F. Smith. *Bayesian theory*. John Wiley & Sons, 1994.

C. M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.

P. J. Cameron. *Oligomorphic permutation groups*, volume 152. Cambridge University Press, 1990.

P. Cheeseman, M. Self, J. Kelly, and J. Stutz. Bayesian classification. 1996.

J. Corander and J. Tang. Bayesian analysis of population structure based on linked molecular information. *Mathematical biosciences*, 205(1):19–31, 2007.

J. Corander, P. Waldmann, P. Marttinen, and M. J. Sillanpää. Baps 2: enhanced possibilities for the analysis of genetic population structure. *Bioinformatics*, 20(15):2363–2369, 2004.

J. Corander, M. Gyllenberg, and T. Koski. Bayesian model learning based on a parallel mcmc strategy. *Statistics and computing*, 16(4): 355–362, 2006.

K. J. Dawson and K. Belkhir. A bayesian approach to the identification of panmictic populations and the assignment of individuals. *Genetical research*, 78(01):59–77, 2001.

B. de Finetti. Theory of probability. 1974.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.

R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern classification*. John Wiley & Sons, 2012.

N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine learning*, 29(2-3):131–163, 1997.

S. Geisser. *Predictive inference*, volume 55. CRC Press, 1993.

S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6):721–741, 1984.

M. Gyllenberg, T. Koski, and M. Verlaan. Classification of binary vectors by stochastic complexity. *Journal of Multivariate Analysis*, 63(1):47–72, 1997.

T. Hastie, R. Tibshirani, J. Friedman, T. Hastie, J. Friedman, and R. Tibshirani. *The elements of statistical learning*, volume 2. Springer, 2009.

M. Hilbert and P. López. The world's technological capacity to store, communicate, and compute information. 332(6025):60–65, 2011.

X. Huang, Y. Ariki, and M. Jack. *Hidden Markov Models for Speech Recognition*. Addison-Wesley, New York, 1990.

Q. Huo and C.-H. Lee. On-line adaptive learning of the continuous density hidden Markov model based on approximate recursive Bayes estimate. *Speech and Audio Processing, IEEE Transactions on*, 5 (2):161 –172, mar 1997. ISSN 1063-6676. doi: 10.1109/89.554778.

Q. Huo and C.-H. Lee. A Bayesian predictive classification approach to robust speech recognition. *Speech and Audio Processing, IEEE Transactions on*, 8(2):200 –204, mar 2000. ISSN 1063-6676. doi: 10.1109/89.824706.

Q. Huo, H. Jiang, and C.-H. Lee. A Bayesian predictive classification approach to robust speech recognition. In *Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on*, volume 2, pages 1547 –1550, apr 1997. doi: 10.1109/ICASSP.1997.596246.

V. Jääskinen, V. Parkkinen, L. Cheng, and J. Corander. Bayesian clustering of dna sequences using markov chains and a stochastic partition model. *Statistical applications in genetics and molecular biology*, 13(1):105–121, 2014.

H. Jiang, K. Hirose, and Q. Huo. Improving Viterbi Bayesian predictive classification via sequential Bayesian learning in robust speech recognition. In *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, volume 1, pages 77 –80, may 1998. doi: 10.1109/ICASSP.1998.674371.

T. Koski. *Hidden Markov models for bioinformatics*, volume 2. Springer, 2001.

M. G. Madden. On the classification performance of tan and general bayesian networks. *Knowledge-Based Systems*, 22(7):489–495, 2009.

K. Mardia, J. Kent, and J. Bibby. *Multivariate analysis*. Probability and mathematical statistics. Academic Press, 1979.

P. Marttinen, J. Corander, P. Törönen, and L. Holm. Bayesian search of functionally divergent protein subgroups and their function specific residues. *Bioinformatics*, 22(20):2466–2474, 2006.

I. Morlini. A latent variables approach for clustering mixed binary and continuous variables within a gaussian mixture model. *Advances in Data Analysis and Classification*, 6(1):5–28, 2012.

A. Nadas. Optimal solution of a training problem in speech recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 33(1):326–329, 1985.

H. Nyman, J. Pensar, T. Koski, J. Corander, et al. Stratified graphical models-context-specific independence in graphical models. *Bayesian Analysis*, 9(4):883–908, 2014.

B. D. Ripley. *Statistical inference for spatial processes*. Cambridge university press, 1991.

B. D. Ripley. *Pattern recognition and neural networks*. Cambridge university press, 1996.

J. Rissanen et al. A universal data compression system. *IEEE Transactions on information theory*, 29(5):656–664, 1983.

C. Robert and G. Casella. *Monte Carlo statistical methods*. Springer Science & Business Media, 2013.

S. Watanabe, Y. Minami, A. Nakamura, and N. Ueda. Variational bayesian estimation and clustering for speech recognition. *Speech and Audio Processing, IEEE Transactions on*, 12(4):365–381, 2004.

J. Yamato, J. Ohya, and K. Ishii. Recognizing human action in time-sequential images using hidden Markov model. In *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR '92., 1992 IEEE Computer Society Conference on*, pages 379–385, 1992. doi: 10.1109/CVPR.1992.223161.