# Bayesian Nonparametric Multilevel Modelling and Applications

by

Vu Nguyen
BSc. (Honours)

Submitted in fulfilment of the requirements for the degree of
Doctor of Philosophy

Deakin University
May, 2015

# Contents

# List of Figures

# List of Tables

# List of Algorithms

# Acknowledgments

It is the time to finish my exciting and memorable PhD research at centre for Pattern Recognition and Data Analytics, Deakin University. The thesis would not have been possible without the encouragement, support and guidance of many people.

In my deep appreciation, I would like to express my utmost gratefulness to my principal supervisor, Prof. Dinh Phung, for his tireless support, enthusiastic guidance and inspirational encouragement during my time at PRaDA. His expertise in Bayesian nonparametric and graphical models have been proven to be beneficial for me, as well as this thesis. I am also grateful to my co-supervisor, Prof. Svetha Venkatesh for her invaluable ideas, useful discussion and feedback which have been great importance for this thesis.

I was fortunate to benefit from the guidance of two other collaborators Dr. Hung Bui and Dr. XuanLong Nguyen. As long with Prof Dinh Phung, their sharp view is beneficial and influential for me to follow the research path of multilevel modelling.

I would count myself incredibly fortunate to work and enjoy my leisure time with Prada peers, all interesting in their own way, including but not limited to Tu, Thuong, Bo, Cheng, Binh, Viet, Truyen, Thin, Sunil, and Wei Lou. I have enjoyed lunch-time discussion, morning coffee, soccer weekly on Friday and pool in spare time. We have had many interesting discussions regarding work and other useful matters and I look forward to continuing collaboration with them.

Throughout these PhD years and for every single year before that, my parents Ut and Van have provided the love, encouraged my curiosity from the start and have put my best interest ahead of everything that has made this all possible.

# DEAKIN UNIVERSITY
# CANDIDATE DECLARATION

I certify the following about the thesis entitled *Bayesian Nonparametric Multilevel Modelling and Applications* submitted for the degree of doctor of philosophy.

a. I am the creator of all or part of the whole work(s) (including content and layout) and that where reference is made to the work of others, due acknowledgment is given.

b. The work(s) are not in any way a violation or infringement of any copyright, trademark, patent, or other rights whatsoever of any person.

c. That if the work(s) have been commissioned, sponsored or supported by any organisation, I have fulfilled all of the obligations required by such contract or agreement.

I also certify that any material in the thesis which has been accepted for a degree or diploma by any university or institution is identified in the text.

*'I certify that I am the student named below and that the information provided in the form is correct'*

**Full Name: .....TIEN VU NGUYEN.....(a.k.a. VU NGUYEN).....**

**Signed: .....** Signature Redacted by Library **.....**

**Date: .....28 April 2015.....**


# DEAKIN UNIVERSITY
# ACCESS TO THESIS - A

I am the author of the thesis entitled

Bayesian Nonparametric Multilevel Modelling and Applications

submitted for the degree of DOCTOR OF PHILOSOPHY

This thesis may be made available for consultation, loan and limited copying in accordance with the Copyright Act 1968.

*'I certify that I am the student named below and that the information provided in the form is correct'*

**Full Name: .....TIEN VU NGUYEN.....**
(Please Print)

**Signed: .....** Signature Redacted by Library **.....**

**Date: .....17 September 2015.....**

# Abstract

Bayesian nonparametric (BNP) methods have increasingly gained their popularity in machine learning and data science as a useful framework thanks to their model flexibility. A widely-used application of Bayesian nonparametrics is clustering data via inducing nonparametric discrete distributions on the parameter space. As such, BNP can identify the suitable number of clusters, addressing the fundamental problem of model selection in parametric models, which is difficult in practice. This thesis considers a specific research problem of BNP where data are organised at more than one level – a setting known as *multilevel*. The proliferation of multilevel data requires new techniques for analysing and modelling, which has posed unique and important research challenges, both in theory and computation. This thesis introduces a set of methods to address these challenges in multilevel modelling using the theoretical framework of Bayesian nonparametrics. The contributions of the thesis can be summarised as follows.

Considering the multilevel structure in video data, we first propose a novel use of Bayesian nonparametric methods for abnormality detection in video surveillance. We utilise the Infinite Hidden Markov model for stream data segmentation to perform abnormality detection in each segment separately. In addition, we introduce the interactive system allowing a user to inspect and browse suspicious events to overcome the semantic gap caused of the detected event by the system and the true anomalous events. The novelty of this approach lies in the fact that it automatically infers the number of hidden patterns from data and thus discover the abnormal events appropriately.

Next, we address, for the first time, the important task of multilevel clustering where the number of clusters at multilevel are not known in advance. We introduce

the Bayesian nonparametric framework of Multilevel Clustering with Group-Level Context ($MC^2$) to jointly cluster data at multilevel while utilising the context observation if available. Using the Dirichlet Process as the building block, our model constructs a product base-measure with a nested structure to accommodate content and context observations at multiple levels. We provide a Polya-urn view of the model and an efficient collapsed Gibbs inference procedure. $MC^2$ outperforms other baselines methods in various experiments including image clustering and text modelling.

We further address classification task for grouped data where a group-level label is our main interest of classification. The input data are observations at the individual level that can be in high dimension and noise. Therefore, we extract the low-dimensional feature embedded inside the data for each group using probabilistic models for multilevel data. After obtaining probabilistic feature from multilevel models, we propose the Topic Model Kernel to perform document classification. The proposed kernel is not only outperforming baseline kernels on the probabilistic feature, but also achieves comparable performances on generic features (non-probabilistic feature).

In seeking to advance regression theory with multilevel data, we address the multilevel regression for modelling and predicting a continuous outcome, where data observation are organised in multilevel structure. We introduce the Bayesian Nonparametric Multilevel Regression (BNMR) to identify the unknown number of hidden regression patterns inside the multilevel data. Our BNMR employs the group-level context information to induce the group clusters that allow predicting for unseen groups. We derive model presentation and collapsed Gibbs sampler for posterior inference. We perform extensive experiments on econometric panel data and healthcare longitudinal data to demonstrate the effectiveness of the proposed model.

Finally, to tackle the issue of scalability, we formalise the novel method for multilevel clustering, emphasising on the scalability and speed, namely Nested K-means. We introduce the concept of Chinese Restaurant Franchise-Bus upon which our result is derived using the principle of the recent small variance asymptotic analysis. This results in an algorithm that can nestedly cluster data points within group and groups themselves. Furthermore, the number of local clusters within each group and the number of global clusters are also automatically induced due to the inherent property

of Bayesian nonparametric models.

# Relevant Publications

Part of this thesis and some related work has been published in referred conference and journal papers or documented elsewhere. The list of these publications is provided below:

- Chapter 3
  - **Nguyen, T. V.**, Phung, D., Rana, S., Pham, D. S., & Venkatesh, S. (2012). Multi-modal abnormality detection in video with unknown data segmentation. In *International Conference on Pattern Recognition* (ICPR), pp. 1322-1325, Japan.
  - **Nguyen, T. V.**, Phung, D., Gupta, S., & Venkatesh, S. (2013). Interactive browsing system for anomaly video surveillance. In *International Conference on Intelligent Sensors, Sensor Networks and Information Processing* (ISSNIP), pp. 384-389, Australia.
  - **Nguyen, V.**, Phung, D., Pham, D. S., & Venkatesh, S. (2015). Bayesian Nonparametric Approaches to Abnormality Detection in Video Surveillance. *Annals of Data Science*, pp. 1-21.
- Chapter 4
  - **Nguyen, T. V.**, Phung, D., Nguyen, X., Venkatesh, S., & Bui, H. (2014). Bayesian Nonparametric Multilevel Clustering with Group-Level Contexts. In *International Conference on Machine Learning* (ICML), pp. 288-296, China.
- Chapter 5

- **Nguyen, T. V.**, Phung, D., & Venkatesh, S. (2013). Topic model kernel: an empirical study towards probabilistically reduced features for classification. In *International Conference on Neural information processing* (ICONIP), pp. 124-131, Korea.
- **Nguyen, V.**, Phung, D., & Venkatesh, S. Topic model kernel classification with probabilistically reduced features. accepted October, *Journal of Data Science*, 2014.

- Chapter 6

  - **Nguyen, V.**, Phung, D., Venkatesh, S., & Bui, H. H. (2015). A Bayesian Nonparametric Approach to Multilevel Regression. In *Advances in Knowledge Discovery and Data Mining*, pp. 330-342. Springer International Publishing.

- Chapter 7

  - **Nguyen, V.**, Phung, D., Nguyen, X., Venkatesh, S., & Bui, H. (2014). The Nested K-means. (to be submitted).

Besides the main publications listed above, the application of the chapters 2 has resulted in the following collaborative work:

- Chapter 2

  - Luo, W., Phung, D., **Nguyen, V.**, Tran, T., & Venkatesh, S.. Speed up health research through topic modeling of coded clinical data. 2nd *International Workshop on Pattern Recognition for Healthcare Analytics*, ICPR, Sweden, 2014.

# Abbreviations

| Abbreviations | Meaning |
|---|---|
| AP | Affinity Propagation |
| BNFA | Bayesian Nonparametric Factor Analysis |
| BNMR | Bayesian Nonparametric Multilevel Regression |
| CDDP | Conditional Dependent Dirichlet Process |
| CRF | Chinese Restaurant Franchise |
| CRP | Chinese Restaurant Process |
| DP | Dirichlet Process |
| DPM | Dirichlet Process Mixture |
| EM | Expectation Maximization |
| GMM | Gaussian Mixture Model |
| HDP | Hierarchical Dirichlet Process |
| HMM | Hidden Markov Model |
| IHMM | Infinite Hidden Markov Model |
| iid | independently and identically distributed |
| LME | Linear Mixed Effects |
| MC2 | Multilevel Clustering with Context |
| MCMC | Markov Chain Monte Carlos |
| NDP | Nested Dirichlet Process |
| NNMF | Nonnegative Matrix Factorization |
| PCA | Principal Component Analysis |
| RBF | Radial Basic Function |
| SVM | Support Vector Machine |
| TMK | Topic Model Kernel |

# Chapter 1

# Introduction

Dealing with high-dimensional data is one of the most important problems in machine learning, data mining and statistical analysis (Lafferty and Wasserman, 2006). A fundamental approach to addressing this problem is to develop suitable theories and methods to discover low-dimensional space embedded inside the data. These low-dimensional representations play an important role in basic machine learning tasks, such as classification, novelty detection, and summarization. An important body of theory addressing this goal is the latent variables models and probabilistic topic modelling for hierarchical data which have been proposed and quickly attracted research attention due to its expressiveness and flexibility in model extension (Blei et al., 2003; Hofmann, 2001; Bishop, 2006).

However, a key limitation of these models lies in its parametric setting in which one needs to specify the latent dimension or number of topics in advance – a task which is typically difficult to do in practice and theoretically known to be difficult to address elegantly. Furthermore, given the prevalence of big data, these models fall short at handling the growing complexity from ever-growing data in real-world problems. To our hope for a new data modelling paradigm, Bayesian nonparametric (BNP) methods have recently emerged in machine learning and data mining as an extremely useful modelling framework due to their model flexibility, capable of fitting a wide range of data types. A widely-used application of Bayesian nonparametric is clustering data via inducing discrete distributions on the parameter space that Dirichlet process (Ferguson, 1973) and Beta processes (Hjort, 1990) are the most

popular approaches.

This thesis considers a specific research problem of BNP for multilevel modelling where data are organised at more than one level. In many situations, data naturally presents themselves in groups, such as students are organised into classes, words grouped into documents, books are organised in chapters which, in turn, break down into paragraphs and sentences, or patients are phenotyped into different cohorts. Observations in the same group are generally not independent; they tend to be more similar than observations from different groups. We refer these grouped data as multilevel data which are organised in a nested or hierarchical structure. Multilevel data structures also arise in longitudinal studies where individual's responses over time are correlated with each other. Standard single level models are not robust for multilevel data as it assumes the observations across groups are independence. The proliferation of multilevel data requires new techniques for analysing and modelling, which has posed new and interesting research challenges, both in theory and computation. Therefore, multilevel modelling (Hox, 2010; Luke, 2004; Goldstein, 2011; Leyland and Goldstein, 2001a) emerges as one of the central research topics in data science for analysing grouped data. In multilevel modelling, we term *individuals* as students while *groups* are as classes. The multilevel approach offers several advantages including sharing statistical strength between individuals within and between groups. Therefore, multilevel model is often outperformed classical single level analysis for modelling hierarchically structured data.

In this thesis, we aim to address the challenges of multilevel modelling under the statistical framework of Bayesian nonparametrics. The challenges of multilevel modelling, which we are addressing, include two fundamental questions. The first question is how to effectively handle multilevel data for browsing, segmentation and extracting hidden information. The second is how to address the principle multilevel modelling tasks including multilevel clustering, scalable multilevel clustering and multilevel regression. Addressing realistic problems in multilevel modelling requires a need to advance Bayesian nonparametric modelling, both in theory and computation, to accommodate multilevel data for various analysis tasks in a principled way. Towards this end, we base our multilevel work on a probabilistic framework, in particular, the use of Bayesian nonparametric modelling. BNP models allow the complexity to grow as more data are observed that can identify the suitable number of clusters. Thus, BNP is appropriate to model the multilevel data where we do not

know the number of the underlying hidden structure inside the grouped data.

## 1.1 Aims and Approaches

This thesis presents an investigation into the problem of Bayesian nonparametric multilevel modelling. Our key objectives are including to handle multilevel structure in data effectively and efficiently for segmenting, browsing and extracting high-level representation. Specific subgoals include:

- We address multilevel data structure in the problem of video surveillance for abnormality detection task.
- We assume that we have label information for group-level in multilevel setting. Then, our target is classifying the group-level given the multilevel data which are in high dimension and noisy.

Our second key objective is to advance the knowledge base and theory of multilevel modelling. In particular, we address the following goals:

- The task of multilevel clustering that can handle group-level context information if available. We aim to perform the data clustering at multilevel including individuals and groups jointly.
- The multilevel regression task for grouped data where group-level context is available. The observations at individual level, in multilevel setting, are our main of interest for regression. In addition, we want to perform prediction for unseen groups which are not seen during training.
- The scalable multilevel clustering task which identifies the cluster labels for observations and groups in multilevel setting. For scalability, we emphasise on the ability of the multilevel clustering system to handle a growing amount of data.

To address the above objectives, our approach roots in the theory of probabilistic graphical and Bayesian nonparametrics. For probabilistic inference tasks, our solution mainly relies on the theory of Markov Chain Monte Carlo. To summarise, we address the first objective as follows:

- Segmenting video data and simultaneously discover low-dimensional latent patterns are key tasks in abnormality detection in video surveillance. We propose to use the Infinite Hidden Markov model (Beal et al., 2002) for video segmentation and the Bayesian Nonparametric Factor Analysis (Paisley and Carin, 2009) for factors decomposition. We develop an interface to assist users interactively in browsing and filtering suspicious events. We present our proposed approach in Chapter 3.
- We firstly extract the high-level representation of the data using the probabilistic models for multilevel data (e.g., Latent Dirichlet Allocation (Blei et al., 2003), Hierarchical Dirichlet Processes (Teh et al., 2006) and MC$^2$ (Nguyen et al., 2014)). Then, we use Jensen-Shannon divergence (Endres and Schindelin, 2003) for computing the distance between the extracted distributions which is Multinomially distributed. We present the approach in Chapter 5.

We address the second objective as follows:

- Multilevel clustering with a unknown number of clusters at multilevel is a hard and complicated task. We borrow the idea of nonparametric clustering by Dirichlet Process Mixture (DPM) (Antoniak, 1974) and Nested Dirichlet Process (NDP) (Rodriguez et al., 2008) for multilevel clustering (cf. Chapter 4).
- Multilevel regression with a unknown number of clusters on the parameter space is difficult problem. Our approach leverages on the basic structure of the Linear Mixed Effect model together with the nested DP to derive a new Bayesian nonparametric solution for multilevel regression (cf. Chapter 6).
- We utilise the theory of Nested Dirichlet Process (Rodriguez et al., 2008) for multilevel clustering. Then, we advance the small variance asymptotic technique for scalable multilevel clustering (cf. Chapter 7).

## 1.2 Contribution and Significance

The significance of our thesis lies in the theoretical development of Bayesian nonparametric methods for multilevel modelling in different situations including: modelling multilevel structure of video surveillance data for abnormality detection, multilevel clustering with group-level context, classification features extracted from multilevel models, multilevel regression, and scalable multilevel clustering. The key contributions of this thesis are detailed below:

- We present our first contribution to understanding Bayesian nonparametric models for abnormality detection task of the video surveillance domain where video frames and frame blocks are organised into multilevel structure. We propose to use the Infinite Hidden Markov model to segment video stream into an infinite number of coherent sections for multi-model abnormality detection at video frame level. Then, we propose an interactive system for a user to browse and filter abnormal events in which the factors are learned by Bayesian Nonparametric Factor Analysis. Using spatial filtering, the proposed framework can find abnormality at multilevel, video frames level and frame blocks level (e.g., which video frame is abnormal, then in this video frame which specific frame block is abnormal).

- We examine deeper theoretical contribution into multilevel clustering problem to jointly cluster data at multilevel. We introduce the Multilevel Clustering with Group-Level Context (MC2) which perform multilevel clustering and utilise group-level context if available. Using the Dirichlet Process as the building block, our model constructs a product base-measure with a nested structure to accommodate content and context observations at multiple levels. We provide a Polya-urn view of the model and an efficient collapsed Gibbs inference procedure. A detailed analysis of marginalization property is also provided for further understanding. MC2 outperforms other baselines methods in various experiments including image clustering and text modelling.

- We further propose Topic Model Kernel (TMK) for data classification when the group-level latent features are extracted by topic models from multilevel data. We exploit the fact that the extracted mixing proportion for each group is followed the Multinomial distribution. Thus, Jensen-Shannon divergence

is a suitable metric for these probabilistic features. The experimental results demonstrate the advantage of the TMK over baseline kernels for probabilistic features. Furthermore, we have demonstrated that the Topic Model Kernel can be widely applicable for other data types (non-probabilistic features) that achieves competitive performance comparing to baseline kernels.

- We introduce a new Bayesian Nonparametric Multilevel Regression (BNMR) for modelling and predicting continuous outcome, where observations are organised into multilevel structure. Our BNMR also employs the group-level context information to induce the group clusters that allow predicting observations in unseen groups. We derive model presentation and collapsed Gibbs sampler for posterior inference. We perform extensive experiments on econometric panel data and healthcare longitudinal data to demonstrate the effectiveness of the proposed model.

- We formalise novel method for multilevel clustering, emphasizing on the scalability and speed, namely Nested K-means. We introduce the concept of Chinese Restaurant Franchise-Bus upon which our result is derived using the principle of the recent small variance asymptotic analysis. This results in an algorithm that can nestedly cluster data points within group and groups themselves. Furthermore, the number of local clusters within each group and the number of global clusters are also automatically induced due to the inherent property of Bayesian nonparametric.

## 1.3 Thesis Overview

The rest of this thesis is structured as follows. In Chapter 2, we provide a related background to our work. Starting with an overview on probabilistic graphical model and exponential family, we review parameter estimation techniques including optimization approaches (e.g., maximum likelihood, Laplace approximation) and sampling approaches (e.g., Gibbs sampler, collapsed Gibbs sampler). We then go deeper into parametric model (e.g., Gaussian Mixture Model, Latent Dirichlet Allocation and Hidden Markov Model) and describe Bayesian nonparametric theory centralized on the Dirichlet Process (e.g., Dirichlet Process Mixture, Hierarchical

Dirichlet Process, Infinite Hidden Markov Model and Nested Dirichlet Process). Finally, we describe the research topic of multilevel modelling for analysing multilevel data which is our main of interest in this thesis.

In Chapter 3, we begin with a review on abnormal detection task in video surveillance. We next present the Infinite Hidden Markov model (Beal et al., 2002) for video stream segmentation where the number of clusters is not known in advance. Then, we develop the multi-model abnormality detection framework on the segmented video stream. To overcome the semantic gap between the true abnormal events and the detected abnormal events, we propose an interface allowing users to interactively browse and filter events at multilevel data structure (video frames and frame blocks). The motion factors are learned using Bayesian Nonparametric Factor Analysis (Paisley and Carin, 2009). Finally, we present experiments on real-world video surveillance datasets.

Chapter 4 presents our novel contribution on a Bayesian nonparametric framework, namely Multilevel Clustering with Group-Level Context (MC2). MC2 allows modelling and clustering data at multilevel. In addition, our model can handle context observation if available to improve modelling and clustering performance. We first present the related methods in multilevel clustering, then the preliminary background on Bayesian nonparametric. Next, we go into details of model properties and derive collapsed Gibbs inference. After developing a Bayesian nonparametric multilevel clustering model, we present its applications on text modelling (using the context as time, author and title respectively) and image clustering. Through these applications, we clearly demonstrate the benefits of jointly modelling and clustering with group-level context information for improving performance. Before concluding, we describe the analysis on the case of missing data where the group-level context is partially available.

Chapter 5 presents our work on extracting latent representation and performs classification for groups where data are organised in multilevel with individuals and groups. Because the observations at individual level are in high-dimensional and noisy, we use probabilistic frameworks for multilevel data such as LDA (Blei et al., 2003), HDP (Teh et al., 2006) and our recent proposed MC2 (Nguyen et al., 2014) to extract the low-dimensional feature embedded inside the data for each document. Then, we propose the Topic Model Kernel to perform document classification with

Support Vector Machine (Cortes and Vapnik, 1995).

In Chapter 6, we present a novel Bayesian nonparametric approach for multilevel regression namely Bayesian Nonparametric Multilevel Regression (BNMR) to model the relationship between the explanatory and outcome variables, organised at multilevel structure. We perform experiments on econometric panel data and healthcare longitudinal data to demonstrate the advantages of the BNMR on predicting observations in unseen groups.

Next, Chapter 7 presents our theoretical contribution in deriving the novel methods for scalability of multilevel clustering where we aim to cluster words and documents at multilevel. We first propose the Chinese Franchise Restaurant Bus (CFRB) metaphor, then developing the sampling algorithm for it. Next, we derive the small variance asymptotic from the CFRB. We conduct large scale experiment for image clustering task.

Finally, Chapter 8 provides a summary of the work in this thesis and discusses some ideas and directions for possible future work.

# Chapter 2

# Background

In this chapter, we present a literature review and the background for the thesis. The methodology used in this thesis relies on the theory of probabilistic graphical model, especially Bayesian nonparametric statistics. We begin with background material for probabilistic graphical model in Section 2.1. We then review the parametric approaches for data modelling in Section 2.2. Next, we present an essential background on Bayesian nonparametrics. At the cornerstone of this theory is the Dirichlet process which shall be covered in Section 2.3. Finally, Section 2.4 provides a review on multilevel modelling, centring at the key research agenda that this thesis aims at addressing.

## 2.1 Graphical Model and Exponential Family

Dealing with uncertainty is a foundational problem in artificial intelligence and machine learning research (Koller and Friedman, 2009). Probabilistic graphical model (PGM) provides a mathematical language to present and do probabilistic reasoning under uncertainty. Probabilistic graphical models use graph theory to encode conditional independence structures over a set of several variables, hence provides a compact factorized form for the joint distribution over the set of these random variables. Two families of graphical models commonly used are Bayesian networks and Markov networks. Both families encompass the properties of factorization and

independence, but they differ in the set of independences they can encode and the factorization of the distribution that they induce (Koller and Friedman, 2009; Jordan, 2004; Bishop, 2006). Several real-world problems can be represented through Bayesian networks or Markov networks. These methods have been used in a wide range of application domains, including: web search, medical diagnosis, image understanding, speech recognition, natural language processing, and robot navigation, to name a few. The interested readers may refer to these books for details (Koller and Friedman, 2009; Darwiche, 2009; Jordan, 2004; Edwards, 2000; Pearl, 1988; Murphy, 2012).

As the main theoretical theme of the thesis is about Bayesian nonparametric, we present this section on the interplay between probabilistic notions such as conditional independence and d-separation and standard materials on parameter estimation for Bayesian model and conjugate analysis of exponential family.

### 2.1.1 Probability distributions on graphs

We briefly describe graphical formalism. A graph $G = (V, E)$ is formed by a set $V = 1, 2, ..., m$ of vertices or nodes, together with a set $E \subset V \times V$ of edges. Each edge consists of a pair of vertices, e.g., $(s, t) \in E$ that may either be undirected or directed. In the case that the edge is undirected, there is no distinction (or unordered) between edge $(s, t)$ and edge $(t, s)$. If the edge is directed, we write $(s \rightarrow t)$ to indicate the direction. Interested readers may refer to (Bondy and Murty, 1976) for further background on graphs and their properties.

To define a probabilistic graphical model, we encode with each vertex $s \in V$ a random variable $X_s$ taking values in some space $\mathcal{X}_s$ which can be continuous or discrete. Two main realisation of graphical models are of Markov random field (MRF) and Bayesian network (cf. Fig. 2.1.1). MRF is similar to a Bayesian network in its representation of dependencies; the differences being that Bayesian networks are directed and acyclic, whereas Markov networks are undirected and may be cyclic.


(a) Markov random field (b) Bayesian network

Figure 2.1.1: Examples of probabilistic graphical models. Nodes represent variables. By convention, shaded nodes represent observed variables and unshaded nodes represent unobserved variables. Edges represent the probabilistic relationship between variables.

### 2.1.2 Independence and conditional independence

The concept of conditional independence is at the heart of probabilistic graphical models. It means that knowing more about the state of the first variable does not have any impact on our knowledge of the state of the second variable. One can use the independence and conditional independence property to interpret the mutual relevance or irrelevance among variables. This property is further illustrated when we discuss d-separation in the next section.

The conditional probability of $A$ given $B$ is represented by $p(A|B)$ which can be represented via joint probabilities $p\left(A \mid B\right) = \frac{p(A,B)}{p(B)}$. In general, we write $p(A|B)$ to represent a belief in $A$ under the assumption that $B$ is known. The variables $A$ and $B$ are said to be independent if $p(A) = p(A|B)$ (or alternatively if $p(A, B) = p(A)p(B)$ because of the property for conditional probability ).

- Example 1: We suppose Apple and Banana, each tosses separate coins. Let A represent the variable "Apple's outcome", and B represent the variable "Banana's outcome". Both A and B have two possible values (Head and Tail). It would be obvious to assume that A and B are independent. Evidence about B will not change our belief in A.

- Example 2: Now suppose both Apple and Banana toss the same coin. Again let $A$ represent the variable "Apple's outcome", and $B$ represent the variable "Banana's outcome". We also assume that there is a possibility that the coin is biased towards tails but we do not know this for certain. In this case $A$ and $B$ are not independent. For example, observing that $B$ is Tail causes us to increase our belief in $A$ being Tail (in other words $p(A|B) > p(A)$ in the case when $A$=Tail and $B$=Tail).

In Example 2, if we further assume the binary variable $C$ represents the condition "the coin is biased towards Tail", then $A$ and $B$ are both dependent on a separate variable $C$. Although $A$ and $B$ are not independent, it turns out that once we know for certain the value of $C$ then any evidence about $B$ cannot change our belief about $A$, or $P(A|C) = P(A|B, C)$. In such case we say that $A$ and $B$ are conditionally independent given $C$. In many real life situations variables which are believed to be independent are actually only independent conditional on some other variables.

In other words, $A$ and $B$ are conditionally independent given $C$ if and only if, given knowledge that $C$ occurs, knowledge of whether $A$ occurs provides *no information* on the likelihood of $A$ occurring, and knowledge of whether $B$ occurs provides no information on the likelihood of $A$ occurring. This is commonly written: $A \perp B \mid C$ and can be read as "$A$ is independent of $B$, given $C$".

### 2.1.3 D-separation

Bayesian networks encode the dependencies and independencies between variables. The important result which can be gain in Bayesian network is the conditional independencies between variables other than those just involving the parent of a node (with the causal Markov assumption). For this purpose, Pearl (1988) propose the concept of d-separation to evaluate the conditional independence in a Bayesian network. D-separation later on plays an important role in deriving the posterior inference for Bayesian nonparametric methods in this thesis.

In this section, we present the notion of d-separation (Pearl, 1988). The "d" in d-separation and d-connection stands for dependence. Using the idea of indepen-


Figure 2.1.2: D-separation, serial connection. Left: Active trait from $A \overset{C}{\rightarrow} B$ when $C$ is unobserved. Right: Inactive trait from $A \overset{C}{\nrightarrow} B$ when $C$ is observed.

dence and conditional independence in previous section, two variables $A$ and $B$ are conditional independence on $C$ if knowledge about $A$ gives you no extra information about $B$ once you have knowledge of $C$. Thus if two variables are d-separated relative to a variable (or set of variables) $C$ in a directed graph (or Bayesian network), then they are independent conditioning on $C$ in all probability distributions such a graph can represent.

Given a graph, a path is *active* if it carries information, or dependence. Two variables $A$ and $B$ might be connected in a graph, where all, some, or none of the paths is active. $A$ and $B$ are *d-connected*, however, if there is existing *any* active path between them. On the contrary, $A$ and $B$ are *d-separated* if all the paths that connect them are *inactive*, or if no path between them is active.

Now we need to define what makes a path active or inactive. A path is active when every vertex on the path is active. Paths, and vertices on these paths, are active or inactive relative to a set of other vertices $C$. To make this concrete, consider all possible paths between a pair of variables $A$ and $B$ that go through a third variable $C$ as shown in Figs. 2.1.2,2.1.3, and 2.1.4.

For the serial connection case as shown in Fig. 2.1.2, we denote that $A$ as fuel price (high or low), $C$ as the inflation state (high or low), and $B$ as the living cost (high or low). When $C$ is unobserved, variable $A$ will have impact on variable $B$. If we know about fuel price ($A$), but not for inflation state ($C$), this affects our belief about the cause of living cost $B$. When $C$ is observed, $A$ will have no impact on $B$. If we observe high inflation ($C$), we will increase our belief that living cost ($B$) is high. In this case, it does not matter the information from fuel price ($A$).


Figure 2.1.3: D-separation, diverging connection. Left: Active trait from $A \overset{C}{\rightarrow} B$ when $C$ is unobserved. Right: Inactive trait from $A \overset{C}{\nrightarrow} B$ when $C$ is observed.


Figure 2.1.4: D-separation, converging connection. Left: Active trait from $A \overset{C}{\rightarrow} B$ when $C$ is observed. Right: Inactive trait from $A \overset{C}{\nrightarrow} B$ when $C$ is unobserved.

For the diverging connection case as shown in Fig. 2.1.3, we denote that $A$ is maths level (high or low) of the student, $C$ is student's intelligence ability (high or low), and $B$ is physic level (high or low). The casual inference in this case is similar to the first case. When $C$ is unobserved, variable $A$ will have impact on variable $B$. If we know that he is good at maths ($A$), this affects our belief about that he is good at physic ($B$) as well although we do not know his intelligence level. When $C$ is observed, $A$ will have no impact on $B$. If we observe his intelligence level ($C$), we will increase our belief about his physic as maths. Therefore, the information from maths level ($A$) and physic level ($B$) are no longer affect each other.

For the converging connection case as shown in Fig. 2.1.4, $A$ and $B$ have common effect in $C$. Let $A$ represent for fire (yes or no), $C$ be alarm (yes or no), and $B$ be thief (yes or no). The casual inference in this case is different to the first two cases. When $C$ is unobserved, there is no causal connection between $A$ and $B$. If we observe there is no fire ($A$), but not observing alarm ($C$), we would have no idea about whether or not the thief is coming. When $C$ is observed, the path between $A$ and $B$ is active. We assume that we hear the alarm ($C$) is on, but we know that there is no thief ($B$), our belief in being fire will increase. Telling you that there

is no fire that tells you nothing about whether there is thief. However, telling you that the alarm is on, after I have told you that the fire is off, tells me that the thief is likely coming.

In converging connection case in Fig. 2.1.4, we need to consider the descendants nodes of $C$ to check if they are activated by conditioning on $C$, then does conditioning on any of its descendants.

### 2.1.4 Parameter estimation

There are two main inference problem that we wish to solve: computing the marginal probability and computing the conditional probabilities upon observing evidence. A relevant auxiliary problem is to compute the maximum a posterior (MAP) estimation. The inference for probabilistic graphical model depends not only on the complexity of the model, such as hierarchical, tree structure, recursive, but also the support of the random variables which can be continuous or discrete (MacKay, 2003; Jordan and Weiss, 2002; Wainwright and Jordan, 2008). If the model is tractable and the support of node variables are discrete we can calculate the posterior exactly by *Variable Elimination* algorithm (Koller and Friedman, 2009). The running time of these exact algorithms are exponential in the size of the largest cluster, assuming all hidden nodes are discrete; this size is called the *induced width of the graph*, and the exact computation for it is NP-hard (Koller and Friedman, 2009). When the model is more complicated and exact solutions do not exist, one can turn to approximate inference methods.

An approximate inference algorithms fall in two categories - *optimization approaches* and *sampling approaches*. Widely used optimization-based approaches are Maximum Likelihood Estimation (MLE), Expectation Maximization (EM) (Dempster et al., 1977) and Variational Bayes (VB) (Bishop, 2006; Wainwright and Jordan, 2008). Sampling approaches are mainly based on Markov Chain Monte Carlo (MCMC) (Andrieu et al., 2003; Gilks et al., 1996). Whilst MCMC methods seek to generate independent samples from the posterior, EM and VB optimize a simplified parametric distribution to be closed in Kullback-Leibler divergence (Kullback and Leibler, 1951) to the posterior. Although the choice of approximate posterior introduces

bias, optimization approaches (MLE, EM, and VB) are empirically shown to be faster than MCMC. While VB is an attractive option when applying Bayesian models to large datasets, MCMC is guaranteed to give arbitrarily precise estimates with sufficient computation.

We present in the subsequences sections the two main approaches for Bayesian inference, including optimization methods and sampling methods. We note that this thesis would rely more on sampling methods than optimization methods.

#### 2.1.4.1 Optimization approaches

In this section, we describe four widely used optimization approaches for statistical inference. We start with maximum likelihood estimation which selects the set of model parameters that maximizes the likelihood function. Next, we describe Laplace approximation to estimate a posterior distribution using Gaussian form. Then, we present the Expectation Maximization and Variational Bayes techniques.

**Maximum Likelihood.** We revisit the maximum likelihood estimation (MLE) (Scholz, 1985; Bilmes et al., 1998). We have a density function $p(x \mid \Theta)$ that is associated by the set of parameters $\Theta$ (e.g., $\Theta$ could be the mean and covariance for Gaussian distribution). We also have a collection of $N$ observations, supposedly drawn from this distribution, e.g., $X = \{x_1, x_2, ...x_N\}$. We assume that these observations are independent and identically distributed (i.i.d.) with distribution $p(x \mid \Theta)$. Therefore, the resulting likelihood for these samples is:

$$p(X \mid \Theta) = \prod_{i=1}^{N} p(x_i \mid \Theta) = \mathcal{L}(\Theta, X)$$

where $\mathcal{L}(\Theta, X)$ is called likelihood function of the data $X$ given the parameter $\Theta$. In the maximum likelihood problem, our aim is to find the $\hat{\Theta}$ that maximizes $\mathcal{L}$:

$$\hat{\Theta} = \underset{\Theta}{\operatorname{argmax}}\, \mathcal{L}(\Theta, X).$$

This optimization problem can be easy or hard depending on the form of $p(x \mid \Theta)$. For detailed derivation of maximum likelihood estimation, we refer to Section 2.4.2.1 where we derive MLE for linear regression.

We summarize the pros and cons of MLE (Natrella, 2010). The advantage is that MLE provides a consistent approach to parameter estimation problem. As the sample size increased, the average value of the estimated parameters will be theoretically exactly equal to the population value and the estimator has the smallest variance. There are two main disadvantages of MLE. The first drawback is that the likelihood formula need to be specifically worked out for a given distribution and estimation problem. The numerical estimation is usually non-trivial, except for a few cases where the maximum likelihood formulas are simple. The second drawback of MLE is that it can be heavily biased for small samples and sensitive to the choice of starting values.

**Laplace Approximation.** Laplace approximation (Laplace, 1986) is a popular method for approximating an integration due to its close relationship with the Gaussian integrals. Specifically, the posterior mode is estimated for each parameter, assumed to be unimodal and Gaussian. As a Gaussian distribution, the posterior mean is the same as the posterior mode, and the variance is estimated. Laplace approximation shares many limitations of MLE, including asymptotic estimation with respect to sample size.

The idea of Laplace approximation is evaluating the integration $A = \int_x f(x)dx$ for some positive functions $f(x) > 0$, but this integral does not have a closed form solution. We can alternatively try to approximate $f(x)$ with a log-quadratic function from the multivariate Gaussian density. Then we integrate the result using what we already know about the Gaussian integrals.

We proceed by approximate $\ln f(x)$ with a quadratic form using Taylor expansion up to second order for $\ln f(x)$ around $x_0$:

$$\ln f\left(x\right) \approx \ln f\left(x_0\right) + \left(x - x_0\right)^{\intercal} \left.\frac{\partial \ln f(x)}{\partial x}\right|_{x_0} + \frac{1}{2}\left(x - x_0\right)^{\intercal} \left.\frac{\partial^2 \ln f\left(x\right)}{\partial x \partial x^{\top}}\right|_{x_0} \left(x - x_0\right).$$

Since $f$ has a global maximum at $x_0$, the first derivative disappears and we yield:

$$\ln f(x) \approx \ln f(x_0) + \frac{1}{2}(x - x_0)^{\mathsf{T}} \operatorname{Hess}(x_0)(x - x_0)$$

where $\operatorname{Hess}(x)$ is the Hessian matrix of $\ln f(x)$. Now switching back to $f(x)$ from equation above:

$$f(x) \approx f(x_0) \exp\left\{\frac{1}{2}(x - x_0)^{\mathsf{T}} \operatorname{Hess}(x_0)(x - x_0)\right\}.$$

We can now integrate this form when the Hessian matrix is negative definite which is often the case if $x_0$ is the mode. Taking integration over $x$ on the density function of multivariate Gaussian yields:

$$\int_x \exp\left\{-\frac{1}{2}(x - \mu)^{\mathsf{T}} \Sigma^{-1}(x - \mu)\right\} = (2\pi)^{d/2} |\Sigma|^{1/2}.$$

Let substitute $\Sigma^{-1}$ with $\operatorname{Hess}(x_0)$ and put things together, we obtain the final Laplace approximation form

$$\begin{aligned} A = \int_x f(x)\, dx &\approx \int_x f(x_0) \exp\left\{\frac{1}{2}(x - x_0)^{\mathsf{T}} \operatorname{Hess}(x_0)(x - x_0)\right\} \\ &= f(x_0)(2\pi)^{d/2} |\operatorname{Hess}(x_0)|^{-1/2}. \end{aligned}$$

**Expectation Maximization.** Typically the statistical models involve latent variables in addition to unknown parameters and known data observations. For example, a mixture model can be described by assuming that each observed data point has a corresponding latent variable, specifying the mixture component that each data point belongs to.

The EM algorithm (Dempster et al., 1977; Wu, 1983) is an iterative method to compute the maximum likelihood in the presence of missing or hidden data. We aim to estimate the model parameters for which the observed data obtain the most likelihood. The EM estimation comprises of two stages: The Expectation step (or E-step) and the Maximization step (or M-step). In the E-step, the missing data are estimated based on the observed data and the current estimation of the model parameters. This stage involves the use of the conditional expectation of the profitability of the hidden variables given the observed variables. In the M-step,

the likelihood function is maximized under the assumption that the missing data are observed. The algorithm is guaranteed to converge because the likelihood is increased at each iteration.

We start deriving the EM algorithm by the log likelihood function of the parameter $\theta$ given the data $\mathbf{X}$ that need to be maximized as:

$$L(\theta) = \ln p(\mathbf{X} \mid \theta).$$

At iteration $t$ and $t+1$, we have the log likelihood function $L(\theta_t)$ and $L(\theta_{t+1})$ respectively. Because the aim is to maximize $L(\theta_{t+1})$, we seek an updated $\theta_{t+1}$ such that $L(\theta_{t+1}) > L(\theta_t)$. Then we will maximize the following objective function:

$$L(\theta_{t+1}) - L(\theta_t) > 0. \tag{2.1.1}$$

Given the observed data $\mathbf{X}$ and hidden variables $\boldsymbol{z}$, we will further express the Eq. 2.1.1 below:

$$\begin{aligned}\ln p(\mathbf{X} \mid \theta_{t+1}) - \ln p(\mathbf{X} \mid \theta_t) &= \ln \left\{ \sum_{\mathbf{z}} p(\mathbf{X} \mid \mathbf{z}, \theta_{t+1})\, p(\mathbf{z} \mid \theta_{t+1}) \frac{p(\mathbf{z} \mid \mathbf{X}, \theta_t)}{p(\mathbf{z} \mid \mathbf{X}, \theta_t)} \right\} - \ln p(\mathbf{X} \mid \theta_t) \\ &= \ln \left\{ \sum_{\mathbf{z}} p(\mathbf{z} \mid \mathbf{X}, \theta_t) \frac{p(\mathbf{X} \mid \mathbf{z}, \theta_{t+1})\, p(\mathbf{z} \mid \theta_{t+1})}{p(\mathbf{z} \mid \mathbf{X}, \theta_t)} \right\} - \ln p(\mathbf{X} \mid \theta_t).\end{aligned}$$

Using Jensen inequality (Jensen, 1906) that $f\left(\sum_{i=1}^{n} \lambda_i x_i\right) \leq \sum_{i=1}^{n} \lambda_i f(x_i)$ with $f(x)$ is a concave function of $\ln(x)$ in our case, we obtain:

$$\begin{aligned}L(\theta_{t+1}) - L(\theta_t) &\geq \sum_{\mathbf{z}} p(\mathbf{z} \mid \mathbf{X}, \theta_t) \ln \left\{ \frac{p(\mathbf{X} \mid \mathbf{z}, \theta_{t+1})\, p(\mathbf{z} \mid \theta_{t+1})}{p(\mathbf{z} \mid \mathbf{X}, \theta_t)\, p(\mathbf{X} \mid \theta_t)} \right\} \\ &= \Delta(\theta_{t+1} \mid \theta_t) \geq 0.\end{aligned} \tag{2.1.2}$$

At the iteration $t+1$, the EM updates the parameter $\theta_{t+1}$ so that the likelihood in the next step is maximized w.r.t the current iteration $t$. In other words, we maximize the objective function in Eq. 2.1.1 that is equivalent to $\Delta(\theta_{t+1} \mid \theta_t)$ in Eq. 2.1.2. Therefore, the Eq. 2.1.2 is the lower bound of the objective function in EM algorithm. Thus, the algorithm is guaranteed to be converged monotonically to

a (local) minimum. Formally,

$$\begin{aligned}\theta_{t+1} &= \underset{\theta}{\operatorname{argmax}} \left\{\Delta\left(\theta \mid \theta_t\right)\right\} \\ &= \underset{\theta}{\operatorname{argmax}} \left\{\sum_{\mathbf{z}} p\left(\mathbf{z} \mid \mathbf{X}, \theta_t\right) \ln \left\{\frac{p\left(\mathbf{X} \mid \mathbf{z}, \theta\right) p\left(\mathbf{z} \mid \theta\right)}{p\left(\mathbf{z} \mid \mathbf{X}, \theta_t\right) p\left(\mathbf{X} \mid \theta_t\right)}\right\}\right\}.\end{aligned}$$

We drop terms which are constant w.r.t $\theta_{t+1}$ and yields the final form $\theta_{t+1} = \underset{\theta}{\operatorname{argmax}} \left\{E_{p(\mathbf{z}|\mathbf{X},\theta_t)}[\ln \mathrm{p}\left(\mathbf{X}, \mathbf{z} \mid \theta\right)]\right\}$. To sum up, the EM algorithm includes two steps:

1. E-step: Estimate the conditional expectation $E_{p(\mathbf{z}|\mathbf{X},\theta_t)}[\ln \mathrm{p}\left(\mathbf{X}, \mathbf{z} \mid \theta\right)]$.
2. M-step: Maximize the conditional expectation

$$\theta_{t+1} = \underset{\theta}{\operatorname{argmax}} \left\{E_{p(\mathbf{z}|\mathbf{X},\theta_t)}[\ln \mathrm{p}\left(\mathbf{X}, \mathbf{z} \mid \theta\right)]\right\}.$$

There are two main advantages of the EM algorithm (Bilmes et al., 1998). The first happens when the data has missing values. The second occurs when optimizing the likelihood function is analytically intractable but when the likelihood function can be simplified by assuming the existence of additional but hidden (or missing) parameters. We later derive the EM algorithm for Gaussian Mixture Model in Section 2.2.1.1.

**Variational Bayes.** Variational Bayes (VB) (Wainwright and Jordan, 2008) is the deterministic optimization algorithm that approximates marginal posterior distribution with an approximating distribution. VB usually converges slower than Laplace Approximation and faster than MCMC. Variational Bayesian methods are primarily used for two purposes: to provide an analytical approximation to the posterior probability of the unobserved variables for performing statistical inference over these variables, to derive a lower bound for the marginal likelihood of the observed data.

Variational Bayes can be seen as an extension of the Expectation Maximization algorithm described in the previous Section 2.1.4.1. Similar to EM, VB finds a set of optimal parameter values, and it has the same alternating structure as that of the EM.

We provide two ways of deriving the lower bound for VB. The first way is based on simple calculus and the second is using Jensen inequality (Jensen, 1906). We begin the first derivation using the log marginal distribution over observed variable $x$ and we denote $\theta$ for model parameters.

$$\begin{aligned}
\ln p(x) &= \ln \frac{p(x,\theta)}{p(\theta \mid x)} \\
&= \int q(\theta) \left[ \ln \frac{q(\theta)}{p(\theta \mid x)} + \ln \frac{p(x,\theta)}{q(\theta)} \right] d\theta \\
&= \underbrace{\int q(\theta) \ln \frac{q(\theta)}{p(\theta \mid x)} d\theta}_{KL(q||p)} + \underbrace{\int q(\theta) \ln \frac{p(x,\theta)}{q(\theta)} d\theta}_{F(q,x)} \\
&\geq \underbrace{\int q(\theta) \ln \frac{p(x,\theta)}{q(\theta)} d\theta}_{F(q,x)}
\end{aligned} \tag{2.1.3}$$

where $F(q,x)$ is the lower bound for $\ln p(x)$ as the term $KL(q||p) \geq 0$. Our aim is to maximize the lower bound $F(q,x)$. As the quantity of $\ln p(x)$ in Eq. 2.1.3 is unknown but fixed, maximizing the lower bound $F(q,x)$ is equivalent to minimising $KL(q||p)$. In other words, we estimate the variational distribution $q(\theta)$ such that minimising $KL(q||p)$. We note that the lower bound in Eq. 2.1.3 can be obtained by using Jensen inequality (Jensen, 1906) as follows:

$$\begin{aligned}
\ln p(x) &= \ln \int_\theta q(\theta) \frac{p(x,\theta)}{q(\theta)} d\theta \\
&\geq \underbrace{\int q(\theta) \ln \frac{p(x,\theta)}{q(\theta)} d\theta}_{F(q,x)}.
\end{aligned}$$

When the Bayesian model involves with several parameters, a common way of restricting the class of approximate posteriors $q(\theta)$ is to use mean field assumption. Particularly, we factorize the approximate posteriors into independent partitions $q(\theta) = \prod_i q_i(\theta_i)$ where $q_i(\theta_i)$ is the approximate posterior for the $i^{th}$ subset of parameter. We denote $q_i \equiv q_i(\theta_i)$ and further extend the lower bound $F(q,x)$ by

splitting the index set $i$ into two sets $j$ and $\backslash j \equiv \{\forall i \mid i \neq j\}$:

$$\begin{aligned} F(q,x) &= \int q_j \prod_{\backslash j} q_i \left(\ln p(x,\theta) - \ln q_j\right) d\theta - \int q_j \prod_{\backslash j} q_i \sum_i \ln q_i d\theta \\ &= \int q_j \left[\prod_{\backslash j} q_i \ln p(x,\theta)\, d\theta_{\backslash j} - \ln q_j\right] d\theta_j - \int q_j \prod_{\backslash j} q_i \ln \prod_i q_i d\theta_{\backslash j} d\theta_j. \end{aligned}$$

We observe that $\prod_{\backslash j} q_i \ln p(x,\theta)\, d\theta_{\backslash j} = \mathop{\mathbb{E}}_{q_{\backslash j}} [\ln p(x,\theta)]$ and the above equation becomes:

$$\begin{aligned} F(q,x) &= \int q_j \ln \frac{\exp\left[\mathop{\mathbb{E}}_{q_{\backslash j}} [\ln p(x,\theta)]\right]}{q_j} d\theta_j + \text{const} \\ &= -KL\left(q_j || \exp\left[\mathop{\mathbb{E}}_{q_{\backslash j}} [\ln p(x,\theta)]\right]\right) + \text{const}. \end{aligned}$$

Therefore, the approximate posterior $q(\theta_j)$ that maximizes $F(q,x)$ is given by:

$$q_j^* = \mathop{\text{argmax}}_{q_j} F(q,x) \propto \exp\left[\mathop{\mathbb{E}}_{q_{\backslash j}} [\ln p(x,\theta)]\right].$$

The derivation of variational inference for Dirichlet Process Mixture can refer to (Blei and Jordan, 2006).

#### 2.1.4.2 Monte Carlo approaches

The idea of Markov Chain Monte Carlo (MCMC) sampling was firstly introduced by (Metropolis et al., 1953) as a method for the efficient simulation of the energy levels of atoms in a crystalline structure. By using random samples to simulate probabilistic models, Monte Carlo methods (Andrieu et al., 2003; Gilks et al., 1996) provide complementary solutions to the learning tasks, especially for estimating posterior distribution in Bayesian inference. In contrast with optimization approaches, they are guaranteed to give precise estimation with sufficient computation. In practice, however, it is known to be very slow for many problems.

A sequence $X_1, X_2, ...X_n$ of random elements of some set is a *Markov chain* (Norris, 1998) if the conditional distribution of $X_{n+1}$ given $X_1, ..., X_n$ depends only on $X_n$ . The set in which the $X_i$ takes values is called the state space of the Markov chain. A Markov chain has *stationary transition probabilities* if the conditional distribution of $X_{n+1}$ given $X_n$ does not depend on $n$. This is the key property of Markov chain of interest in MCMC, in which Markov chain is defined as a process, whose stationary distribution is our posterior of interest.

We below present four MCMC methods including importance sampling, Metropolis Hasting, Gibbs sampling, blocked Gibbs sampling, and collapsed Gibbs sampling. We note that Gibbs and collapsed Gibbs versions are intensively used in this thesis.

**Importance Sampling.** The simplest approach in Markov Chain Monte Carlo class is the importance sampling (Srinivasan, 2002) for estimating properties of a particular distribution. Consider a collection $\{x_i\}_{i=1}^N$ generated from a given probability distribution $p(x)$. Then the expectation of $f(x)$ under $p(x)$ can be approximated by the average of $f(x)$: $\mathbb{E}[f(x)] = \int f(x)p(x)dx$.

We assume that it is possible to evaluate $p(x)$ given $x$, but sampling from $p(x)$ is difficult. Therefore, importance sampling method introduces a sampling distribution $g(x)$ from which we draw a sample $x_i$ instead. The principle of importance sampling is presented:

$$\begin{aligned}\mathbb{E}[f(x)] &= \int f(x)\frac{p(x)}{g(x)}g(x)d(x) \\ &\backsimeq \frac{1}{N}\sum_{i=1}^{N}\frac{p(x_i)}{g(x_i)}f(x_i) \\ &= \frac{1}{N}\sum_{i=1}^{N}w(x_i)f(x_i)\end{aligned}$$

where we sample $x_i$ from $g(x_i)$, then compute the important sampling weight $w(x_i) = \frac{p(x_i)}{g(x_i)}$. Finally, the desired expectation is estimated as $\mathbb{E}[f(x)] = \frac{1}{N}\sum_{i=1}^{N} w(x_i)f(x_i)$. In other words, sampling $x_i$ from $p(x)$ is equivalent to sampling $x_i \times w(x_i)$ from $g(x)$.

**Metropolis Hasting.** Metropolis–Hastings algorithm is a Markov chain Monte Carlo (MCMC) method for obtaining a sequence of random samples from a probability distribution for which direct sampling is difficult. This sequence can be used to approximate the distribution (i.e., to generate a histogram), or to compute an integral (such as an expected value). Metropolis–Hastings and other MCMC algorithms are generally used for sampling from multi-dimensional distributions, especially when the number of dimensions is high.

The derivation of the algorithm starts with the condition of balance: $p(x)\,p(x \to x') = p(x')\,p(x' \to x)$ which can be rewritten as $\frac{p(x \to x')}{p(x' \to x)} = \frac{p(x')}{p(x)}$. We decompose the transition probability $p(x \to x')$ into proposal distribution $g(x \to x')$ and acceptance distribution $A(x \to x')$. We perform similar decomposition for $p(x' \to x)$ and yield:

$$\frac{A(x \to x')}{A(x' \to x)} = \frac{p(x')\,g(x' \to x)}{p(x)\,g(x \to x')}.$$

A common choice in Metropolis hasting for acceptance distribution is as $A(x \to x') = \min\left(1, \frac{p(x')g(x' \to x)}{p(x)g(x \to x')}\right)$.

**Gibbs Sampling.** Gibbs sampling (Turchin, 1971; Geman and Geman, 1984) is one member of Metropolis-Hasting algorithm in which the acceptance rate for each move is always as 1. The general idea is to approximately find the stationary distribution of the target distribution (e.g., posterior distribution in the Bayesian Mixture Model in Section 2.2.1.3). Gibbs sampling is applicable when the joint distribution is not known explicitly or is difficult to sample from directly, but the conditional distribution of each variable is known and easy to sample from. A Gibbs sampler generates a draw from the distribution of each parameter or variable in turn, conditional on the current values of the other parameters. Particularly, to sample variables $A, B$ and $C$ from the joint distribution $p(A, B, C)$ where there is no closed form solution for $p(A, B, C)$, Gibbs sampler will use the conditional distribution to infer sequentially $p(A \mid B, C), p(B \mid A, C)$ and $p(C \mid A, B)$ following Algorithm 2.1.

There are numerous variations of Gibbs sampling, such as blocked Gibbs sampling, collapsed Gibbs sampling (Liu, 1994) that we present below.

**Algorithm 2.1** Gibbs sampler routine.

1. Initialize randomly $A^0, B^0, C^0$
2. For $t = 1, ...T$
   (a) $A^{t+1} \sim p\left(A \mid B^t, C^t\right)$
   (b) $B^{t+1} \sim p\left(B \mid A^{t+1}, C^t\right)$
   (c) $C^{t+1} \sim p\left(C \mid A^{t+1}, B^{t+1}\right)$
3. Return $A, B, C$

**Blocked Gibbs sampler.** A more efficient version of Gibbs sampling, especially in the case of deterministic constraints, is the blocked Gibbs sampler which groups two or more variables together and samples from their joint distribution conditioned on all other variables, rather than sampling from each one individually as in previous section of Gibbs sampler.

This procedure is valid even if blocks overlap; variables in multiple blocks will simply be sampled more often. Typically you will not have to specify the blocking for Gibbs sampling as default blocking is automatically based on the deterministic factors and constraints in your graphical model. The routine for Blocked Gibbs sampler is summarized in Algorithm 2.2 where we conditionally sample the variables $A$ and $B$ at once as $p\left(A, B \mid C\right)$ and $p\left(C \mid A, B\right)$.

**Algorithm 2.2** Blocked Gibbs sampler routine.

1. Initialize randomly $A^0, B^0, C^0$
2. For $t = 1, ...T$
   (a) $A^{t+1}, B^{t+1} \sim p\left(A, B \mid C^t\right)$
   (b) $C^{t+1} \sim p\left(C \mid A^{t+1}, B^{t+1}\right)$
3. Return $A, B, C$

**Collapsed Gibbs sampler.** A collapsed Gibbs sampler integrates out one or more variables when sampling for some other variables. For example, we assume that a model consists of three variables $A$, $B$, and $C$. The original version of Gibbs

sampler would sample from $p(A|B,C)$, then $p(B|A,C)$, and then $p(C|A,B)$. A collapsed Gibbs sampler might replace the sampling step for $A$ with a sample taken from the marginal distribution $p(A|C) = \int_B p\left(A \mid B,C\right) p\left(B\right) dB$, with variable B integrated out in this case. The distribution over a variable $A$ that arises when collapsing a parent variable $B$ is called a compound distribution; sampling from this distribution is generally tractable when $B$ is the conjugate prior for $A$, particularly when $A$ and $B$ are members of the exponential family (Liu, 1994). In fact, we can often collapse $B$ out entirely in cases where we do not actually care about its value, then we get Algorithm 2.3.

**Algorithm 2.3** Collapsed Gibbs sampler routine.

1. Initialize randomly $A^0, C^0$
2. For $t = 1, ...T$
   (a) $A^{t+1} \sim \int_B p\left(A \mid B, C^t\right) p\left(B\right) dB$
   (b) $C^{t+1} \sim \int_B p\left(C \mid A^{t+1}, B\right) p\left(B\right) dB$
3. Return $A, C$

### 2.1.5 Exponential family and conjugacy analysis

Throughout this thesis, we widely utilise collapsed Gibbs sampler (Liu, 1994) to perform posterior inference (to estimate posterior distribution of latent parameters) of Bayesian nonparametric models. By using collapsed Gibbs sampler, we analytically integrate out one or more latent variables due to conjugacy prior of exponential family. Therefore, in this section, we review the exponential family and its conjugate prior structure.

#### 2.1.5.1 Exponential family

In statistics and probability, the exponential family is an important class of probability distributions sharing a specific form (Andersen, 1970; Pitman, 1936). The exponential families (Andersen, 1970) include many of the most common distributions, including the Normal, Exponential, Gamma, Chi-squared, Beta, Dirichlet,

Bernoulli, Categorical, Poisson, Wishart, Inverse Wishart and many others. A number of common distributions are exponential families only when certain parameters are considered fixed and known, e.g., Multinomial (with fixed number of trials). Exponential families are also important in Bayesian statistics in which a prior distribution is multiplied by a likelihood function and then normalized to produce a posterior distribution. If the likelihood belongs to the exponential family, there exists a conjugate prior which is often also followed the exponential family.

For the observation $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N\}$ and the parameter $\boldsymbol{\theta}$, the probability density function of the exponential family is defined as:

$$f_X\left(\boldsymbol{x} \mid \boldsymbol{\theta}\right) = h\left(\boldsymbol{x}\right) \exp\left[\eta(\boldsymbol{\theta}).T(\boldsymbol{x}) - A\left(\boldsymbol{\theta}\right)\right]$$

where $T(\boldsymbol{x})$ is a vector of sufficient statistics (or feature vector), $A\left(\boldsymbol{\theta}\right)$ is a log partition function.

In Bayesian setting, there is a hyperparameter $\lambda$ for $\boldsymbol{\theta}$ which defines the prior density $p\left(\boldsymbol{\theta} \mid \lambda\right)$. A group of prior densities $p\left(\boldsymbol{\theta} \mid \lambda\right)$ is called as *conjugate prior* to $p\left(\boldsymbol{x} \mid \boldsymbol{\theta}\right)$ if the posterior distribution possesses the same form as the prior distribution.

#### 2.1.5.2 Conjugate priors in Bayesian statistics

As discussed in Section 2.1.4.2, collapsed Gibbs sampler integrates out one or more variables when sampling for some other variables. To utilise collapsed Gibbs sampler, we make use of variables followed exponential family that will be analytically integrated out. We below go into details how to use conjugate priors for conveniently computing posterior distribution, marginal likelihood and predictive likelihood.

In Bayesian probability theory, if the posterior distributions $p(\boldsymbol{\theta}|\boldsymbol{x})$ are in the same family as the prior probability distribution $p(\boldsymbol{\theta})$, the prior and posterior are then called conjugate distributions, and the prior is called a conjugate prior for the likelihood function (Schlaifer and Raiffa, 1961). A conjugate prior is an algebraic convenience, giving a closed-form expression for the posterior; otherwise a difficult numerical integration may be needed. Further, conjugate priors may give intuition, by more transparently showing how a likelihood function updates a prior distribution

(Murphy, 2007; Diaconis et al., 1979; Sudderth, 2006).

We consider the Bayesian inference task that the data observation is denoted as $\mathbf{x}_{1:N}$. The parameter which directly generating the observation $\mathbf{x}_{1:N}$ is $\boldsymbol{\theta}$. We place the hyperparameter $\alpha_0$ serving as a prior distribution for the parameter $\boldsymbol{\theta}$.

Let $p(\boldsymbol{x} \mid \mathbf{w}) = h(\boldsymbol{x}) \exp\{\langle \mathbf{w}, t(\boldsymbol{x}) \rangle - A(\mathbf{w})\}$ be an exponential family density and $p(\mathbf{w} \mid \boldsymbol{\alpha}, \alpha_0) = g(\mathbf{w}) \exp(\langle \boldsymbol{\alpha}, \mathbf{w} \rangle - \alpha_0 A(\mathbf{w}) - B(\boldsymbol{\alpha}, \alpha_0))$ is its conjugate prior. We have the following propositions for posterior distribution, marginal likelihood and predictive likelihood.

**Proposition 2.1.** *The posterior distribution of the data, given the parameter $\boldsymbol{\theta}$ and hyperparameters $\alpha_0$ can be written as:*

$$p(\boldsymbol{\theta} \mid \boldsymbol{x}_{1:n}, \boldsymbol{\alpha}, \alpha_0) = p\left(\mathbf{w} \mid \boldsymbol{\alpha}^{[n]}, \alpha_0^{[n]}\right) \tag{2.1.4}$$

*We denote the updated hyperparameter $\boldsymbol{\alpha}^{[n]} = \boldsymbol{\alpha} + \sum_{i=1}^{n} t(\boldsymbol{x}_i)$ and $\alpha_0^{[n]} = \alpha_0 + n$.*

For computing marginal likelihood of the data, we have the below proposition.

**Proposition 2.2.** *The marginal likelihood of the data point $\boldsymbol{x}_{1:N}$ is expressed as:*

$$p(\boldsymbol{x}_{1:N} \mid \boldsymbol{\alpha}, \alpha_0) = \exp\left\{ B\left(\boldsymbol{\alpha}^{[n]}, \alpha_0^{[n]}\right) - B(\boldsymbol{\alpha}, \alpha_0) + \sum_{i=1}^{n} \log h(\boldsymbol{x}_i) \right\}.$$

The predictive likelihood can be presented in a closed form as:

**Proposition 2.3.** *The predictive likelihood of new data point $\mathbf{x}_{new}$ given the previous data points $\boldsymbol{x}_{1:N}$:*

$$p(\boldsymbol{x}_{new} \mid \boldsymbol{x}_{1:N}, \boldsymbol{\alpha}, \alpha_0) = \exp\left\{ B(\boldsymbol{\alpha}^{new}, \alpha_0^{new}) - B\left(\boldsymbol{\alpha}^{[n]}, \alpha_0^{[n]}\right) \right\} \tag{2.1.5}$$

*where $\boldsymbol{\alpha}^{\text{new}} = \boldsymbol{\alpha}^{[n]} + t(\boldsymbol{x}_{\text{new}})$ and $\alpha_0^{\text{new}} = \alpha_0^{[n]} + 1$.*

When the number of observation $N$ is large relative to $\alpha_0$, the posterior distribution of Eq. 2.1.4 is mainly determined by the observed sufficient statistics.

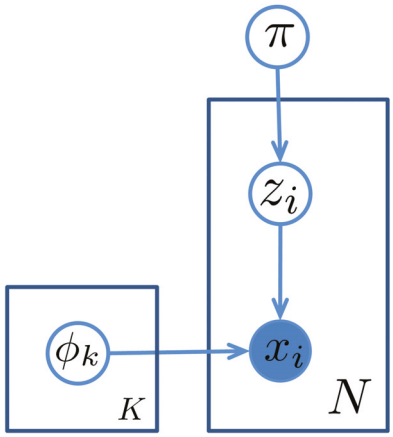



Figure 2.2.1: Finite Mixture Model

## 2.2 Parametric Approaches for Data Modelling

Our dissertation is about Bayesian nonparametric modelling. Before going to details of Bayesian nonparametric, we revisit a family of parametric approaches from which our Bayesian nonparametric methods would be developed. The difference between parametric model and nonparametric model is that the former has a fixed number of parameters, while the latter grows the number of parameters with the amount of training data (Murphy, 2012). In this section, we present the probabilistic mixture model (Everitt et al., 1981) for flat data clustering, Hidden Markov model (Rabiner and Juang, 1986) for time-series modelling, and Latent Dirichlet Allocation (Blei et al., 2003) for document modelling.

### 2.2.1 Mixture Model

We start this section by describing the mixture model, a simple but useful framework for data modelling.

Modelling non-homogenous structure of the data by a single mode (e.g., approximate the whole data points by a single Gaussian distribution) will not be sufficiently enough to represent the data. Hence, mixture model (Everitt et al., 1981) is proposed as a probabilistic model for representing the presence of subpopulations within an overall population, aimed at providing a richer class of density models than the

single one (Bishop, 2006). Given a collection of observations, statistical inferences are then necessary to estimate the mixture model and to assign each data point to a suitable subpopulation.

The formal definition of mixture model can be any convex combination such that: $\sum_{k=1}^{K} \pi_k f\left(x \mid \phi_k\right)$ with $\sum_{k}^{K} \pi_k = 1$. Where parameters $(\pi_1, \pi_2, ..., \pi_K)$ called the weights or mixture proportion, $\phi_k$ is a parameter for each mixture component $k$, and $f\left(. \mid \phi_k\right)$ is a probability density function given the component's parameter $\phi_k$. Then, the data point is represented under a mixture model as:

$$p\left(x \mid \phi_k\right) = \sum_{k=1}^{K} \pi_k \times f\left(x \mid \phi_k\right)$$

where $K$ is number of mixture components, $N$ is the total number of observations, $\{z_i\}_{i=1}^{N}$is the latent variable, assigning observation $i$ to component $k$, $\{x_i\}_{i=1}^{N}$is the data observation, and $f\left(x \mid \phi_k\right)$ is probability distribution of an observation, parametrized on $\phi_k$.

#### 2.2.1.1 Expectation Maximization for Gaussian Mixture Model

Using Expectation Maximization algorithm presented in Section 2.1.4.1, we derive parameter estimation for Gaussian Mixture Model. Our data observations include $D = \{x_n\}_{n=1}^{N}$ and the latent variable $\{z_n\}_{n=1}^{N}$ are unobserved. As we have summarized in Section 2.1.4.1 that there are two steps need to be estimated:

1. E-step: We estimate the conditional expectation $\underset{p(\mathbf{z}|\mathbf{X},\theta_t)}{\mathbb{E}}\left[\ln \mathrm{p}\left(\mathbf{X}, \mathbf{z} \mid \theta_{\mathrm{t+1}}\right)\right]$. The expectation step (E-step) consists of calculating the expected value of the complete data likelihood function where

$$p\left(z_i = k \mid \boldsymbol{X}, \theta_t\right) = \frac{p\left(x_i \mid z_i = k, \theta_t\right) p\left(z_i = k \mid \theta_t\right)}{p\left(x_i \mid \theta_t\right)}.$$

2. M-step: We compute $\theta_{t+1} = \underset{\theta_{t+1}}{\operatorname{argmax}} \left\{ \underset{p(\mathbf{z}|\mathbf{X},\theta_t)}{\mathbb{E}}\left[\ln \mathrm{p}\left(\mathbf{X}, \mathbf{z} \mid \theta_{\mathrm{t+1}}\right)\right] \right\}$.

The expected complete log likelihood $\mathcal{L}(\Phi \mid D)$ looks like:

$$\begin{aligned}\mathcal{L}(\Phi \mid D) &= \log \prod_n p(x_n, z_n \mid \Phi) \\ &= \sum_n \log \sum_k \left[ p(x_n \mid z_n, \phi_k)\, p(z_n \mid \pi) \right]. \end{aligned} \tag{2.2.1}$$

We then maximize the log likelihood $\mathcal{L}(\Phi \mid D)$ in Eq. 2.2.1 by taking derivative w.r.t $\mu_k$, $\Sigma_k$, and $\pi_k$ setting results to zero, and solve these equations.

- Optimizing $\mu_k$. Getting the partial derivative w.r.t. $\mu_k$, we have:

$$\begin{aligned}\frac{\delta \mathcal{L}(\Phi \mid D)}{\delta \mu_k} &= \sum_n \frac{\pi_k}{\sum_k \mathcal{N}(x_n \mid \mu_k, \Sigma_k) * \pi_k} \frac{\delta \mathcal{N}(x_n \mid \mu_k, \Sigma_k)}{\delta \mu_k} \\ &= \sum_n \frac{\pi_k * \mathcal{N}(x_n \mid \mu_k, \Sigma_k)}{\sum_k \mathcal{N}(x_n \mid \mu_k, \Sigma_k) * \pi_k} \frac{\delta \log \mathcal{N}(x_n \mid \mu_k, \Sigma_k)}{\delta \mu_k} \\ &= \sum_n r_n^k \Sigma_k^{-1} (x_n - \mu_k). \end{aligned} \tag{2.2.2}$$

  Setting the result in Eq. 2.2.2 to zero and solve the equation:

$$\mu_k = \frac{\sum_n r_n^k x_n}{\sum_n r_n^k}.$$

- Optimizing $\Sigma_k$. Getting the partial derivative w.r.t $\Sigma_k$:

$$\begin{aligned}\frac{\delta \mathcal{L}(\Phi \mid D)}{\delta \Sigma_k} &= \sum_n \frac{\pi_k}{\sum_k \mathcal{N}(x_n \mid \mu_k, \Sigma_k) * \pi_k} \frac{\delta \mathcal{N}(x_n \mid \mu_k, \Sigma_k)}{\delta \Sigma_k} \\ &= \sum_n \frac{\pi_k * \mathcal{N}(x_n \mid \mu_k, \Sigma_k)}{\sum_k \mathcal{N}(x_n \mid \mu_k, \Sigma_k) * \pi_k} \frac{\delta \log \mathcal{N}(x_n \mid \mu_k, \Sigma_k)}{\delta \Sigma_k} \\ &= \frac{\delta}{\delta \Sigma_k} \left\{ \log |\Sigma_k^{-1}|^{\frac{1}{2}} - \frac{1}{2} (x_n - \mu_k)^T \Sigma_k^{-1} (x_n - \mu_k) \right\} \\ \frac{\delta \mathcal{L}(\Phi \mid D)}{\delta \Sigma_k} &= \sum_n r_n^k \left\{ \frac{|\Sigma_k|}{2} - \left( \frac{1}{2} (x_n - \mu_k)(x_n - \mu_k)^T \right) \right\}. \end{aligned} \tag{2.2.3}$$

  Setting the result in Eq. 2.2.3 to zero and solve the equation:

$$\Sigma_k = \frac{\sum_n r_n^k (x_n - \mu_k)(x_n - \mu_k)^T}{\sum_n r_n^k}.$$

- Optimizing $\pi_k$. We apply Lagrange multiplier for $\mathcal{L}(\Phi \mid D)$ with the

constraint of $\sum_k \pi_k = 1$. Then take derivative w.r.t $\pi_k$

$$\frac{\delta}{\delta \pi_k} \left\{ \sum \log \sum_k \mathcal{N}\left(x_n \mid \mu_k, \Sigma_k\right) * \pi_k + \lambda \left(1 - \sum_k \pi_k\right) \right\} = 0$$

$$\sum_n \frac{\mathcal{N}\left(x_n \mid \mu_k, \Sigma_k\right)}{\sum_k \mathcal{N}\left(x_n \mid \mu_k, \Sigma_k\right) * \pi_k} - \lambda = 0. \quad (2.2.4)$$

Let multiply $\sum_k \pi_k$ both sides of the Eq. 2.2.4 yields:

$$\begin{aligned} \sum_k \pi_k \sum_n \frac{\mathcal{N}\left(x_n \mid \mu_k, \Sigma_k\right)}{\sum_k \mathcal{N}\left(x_n \mid \mu_k, \Sigma_k\right) * \pi_k} - \sum_k \pi_k \lambda &= 0 \\ \lambda &= N. \end{aligned}$$

Again, multiplying $\pi_k$ both sides of the Eq. 2.2.4, we obtain $\pi_k$:

$$\sum_n \frac{\mathcal{N}\left(x_n \mid \mu_k, \Sigma_k\right) * \pi_k}{\sum_k \mathcal{N}\left(x_n \mid \mu_k, \Sigma_k\right) * \pi_k} - \lambda * \pi_k = 0$$

$$\pi_k = \frac{1}{N} \sum_n r_n^k.$$

We summarize the EM algorithm for Gaussian Mixture Model in Algorithm 2.4.

#### 2.2.1.2 Illustrating GMM using EM algorithm

We create three Gaussian distributions ($K = 3$) in two-dimensional space defined by the following parameters $\mu_1 = [4; 4]$, $\mu_2 = [8; 8]$, $\mu_3 = [12; 12]$ with the diagonal covariance matrix $\sigma = [1; 1]$. We then generate $N = 1000$ data points following 2 dimensional Gaussian distribution (cf. Fig. 2.2.2a) which organised into three groups. Next, the EM algorithm is used to learn GMM with the number of clusters is specified as $K = 3$. The clustering result is plotted in Fig. 2.2.2b for visualization (each cluster of data is represented in different colors).

**Algorithm 2.4** EM algorithm for Gaussian Mixture Model

Input: data observation $\boldsymbol{X} = \{x_n\}_{n=1}^{N}$, number of cluster $K$

1. Initialize $\boldsymbol{z} = [z_1, ..., z_N], z_n \in \{1, ...K\}, \forall n = 1...N$
2. Initialize parameters

   **for** $k = 1, 2, ...K$ **do**

   $\pi_k = \frac{1}{N}\sum_n \mathbb{I}(z_n, k) \quad \mu_k = \frac{\sum_n \mathbb{I}(z_n,k)x_n}{\sum_n \mathbb{I}(z_n,k)} \quad \Sigma_k = \frac{\sum_n \mathbb{I}(z_n,k)(x_n-\mu_k)(x_n-\mu_k)^T}{\sum_n \mathbb{I}(z_n,k)}$

   **end for**
3. Loop until convergence

   (a) Expectation Step

   **for** $n = 1, 2...N$ **do**

   $r_n^k = \frac{\pi_k * \mathcal{N}(x_n|\mu_k,\Sigma_k)}{\sum_k \mathcal{N}(x_n|\mu_k,\Sigma_k)*\pi_k} \ \forall k = 1, 2, ...K$

   **end for**

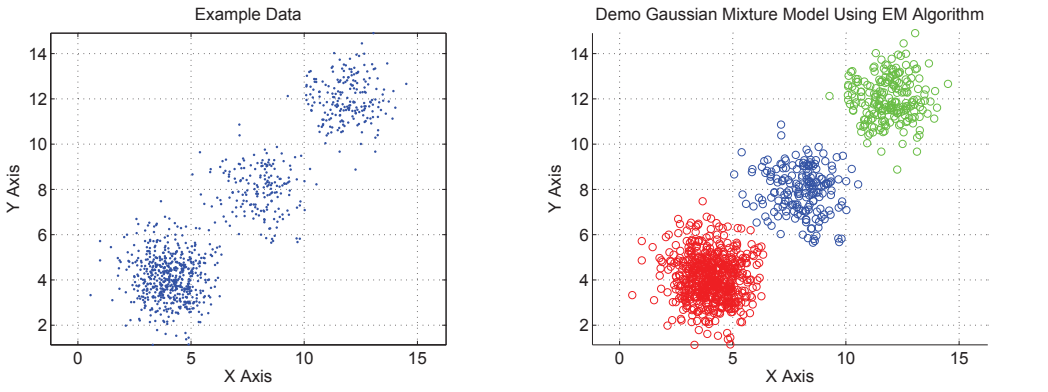
   (b) Maximization Step

   **for** $k = 1, 2, ...K$ **do**

   $\pi_k = \frac{1}{N}\sum_n r_n^k \quad \mu_k = \frac{\sum_n r_n^k x_n}{\sum_n r_n^k} \quad \Sigma_k = \frac{\sum_n r_n^k (x_n-\mu_k)(x_n-\mu_k)^T}{\sum_n r_n^k}$

   **end for**

Output $\pi = [\pi_1...\pi_K], \mu = [\mu_1...\mu_K], \Sigma = [\Sigma_1, ...\Sigma_K], \boldsymbol{z}$




(a) Example data (b) Clustering by GMM using EM algorithm

Figure 2.2.2: Demo Gaussian Mixture Model using EM algorithm.

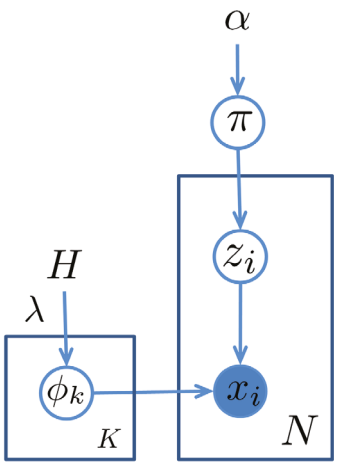



Figure 2.2.3: Bayesian Mixture Model

#### 2.2.1.3 Bayesian Mixture Model

In the previous section, we use Expectation Maximization to learn GMM parameters under point estimation setting (a.k.a. Frequentist view). We further explore the Bayesian view (Berger, 1985; Bayes and Price, 1763) of the Mixture Model as Bayesian Mixture Model. In Bayesian Mixture Model, the mixing proportion is assumed to be drawn from a Dirichlet distribution with parameter $\alpha$: $\pi \sim \text{Dir}(\alpha)$, and the topic is generated from a prior distribution $\phi_k \overset{\text{iid}}{\sim} H(\lambda), \forall k \in \{1, 2...K\}$. The remaining variables of $z_i$ and $x_i$ are sampled similar to Mixture Model in previous section as $z_i \overset{\text{iid}}{\sim} \text{Mult}(\pi)$ and $x_i \sim F(\phi_{z_i}), \forall i \in \{1, 2...N\}$.

When function $F$ is Gaussian distribution $x_i \sim \mathcal{N}(\phi_{z_i})$ and $\phi_k \sim \mathcal{N}(\mu_0, \Sigma_0)$, we obtain Bayesian Gaussian Mixture Model. Specifically, the graphical representation of Bayesian Mixture Model is displayed in Fig. 2.2.3.

Since exact inference for Bayesian Mixture Model is intractable, one can utilise Markov Chain Monte Carlo methods (Andrieu et al., 2003) for posterior inference. Using conjugacy property in Section 2.1.5.2, we would analytically integrate out the variable $\pi$ and $\phi_k$. Then, we need to sample the remaining latent variable $\boldsymbol{z}$. For collapsed Gibbs sampler, the conditional distribution on $z_i$ given the remaining variables is written as:

$$p(z_i = k \mid \boldsymbol{z}_{-i}, \boldsymbol{x}, \alpha, H) \propto p(z_i = k \mid z_{-i}, \alpha) \times p(x_i \mid z_i = k, \{x_j \mid z_j = k, j \neq i\}, H)$$

where the first term can be seen as the proportion likelihood on number of data point in cluster $k$: $p\left(z_i = k \mid z_{-i}, \alpha\right) = \frac{n_k}{N}$ and the second term is the conjugate predictive likelihood under component $k$ as shown in Eq. 2.1.5.

#### 2.2.1.4 Applications of Mixture Models

There are two main themes of using probabilistic mixture model as Gaussian Mixture Models (GMM) and Multinomial Mixture Model (MMM). Gaussian Mixture Models are widely used in computer vision to model natural images for the purposes of automatic clustering, retrieval, and classification (Barnard et al., 2003; Jeon et al., 2003). GMM is also a popular approach for background subtraction task (Grimson et al., 1998) in computer vision and multimedia. In image segmentation, the likelihood of the pixel to the object or background are often estimated using Gaussian Mixture Model (Rother et al., 2004; Nguyen et al., 2011, 2012a; Kohli et al., 2009) that they use a GMM to model a single object or background. Then, the likelihood of a pixel to an object is equivalent to a predictive likelihood of a data point to a GMM.

GMM is popular for image applications as color histogram usually has a suitable shape for GMM while MMM is widely used for text and document domains because the term frequency in documents is multinomially distributed (Hofmann, 1999). Rigouste et al. (2007) investigate the use of Multinomial Mixture Model for text clustering in which a bag-of-words approach to vector document representation is employed. Similarly, Masada et al. (2007) propose a method for image clustering using Multinomial Mixture Models. In addition, MMM can be utilised for class-conditional distributions in document classification task (Novovičová and Malík, 2003).

### 2.2.2 Hidden Markov Model

A Hidden Markov model (HMMs) (Rabiner, 1989) can be considered as a generalization of mixture model where the hidden variables, which control the mixture component to be selected for each observation, are related through a Markov process

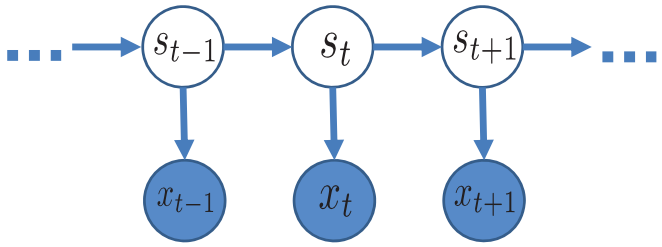



Figure 2.2.4: Hidden Markov Model representation. The observation is denoted as $x_t$ while hidden state is $s_t$.

rather than independent of each other. In this section, we describe the HMM which is a parametric counterpart of the Infinite Hidden Markov model that we later use for video segmentation and abnormal detection in Chapter 3.

In Hidden Markov model, we make two main assumptions about the data including Markov assumption and discrete state space assumption. For the first assumption, the conditional probability distribution of the hidden variable $s(t)$ at time $t$, given the values of the hidden variable $s$ at all times, depends only on the value of the hidden variable $s_{t-1}$. In other words, the values at time $t-2$ and before have no influence on $s(t)$ given $s(t-1)$. This is called the *Markov property*, a.k.a. Markov assumption mentioned earlier in our review of Markov chain, used in HMM.

Secondly, we have the discrete state space assumption that the state space of the hidden variables is discrete, while the observations themselves can either be discrete or continuous (e.g., observations from a Gaussian distribution).

#### 2.2.2.1 Model representation and parameter estimation

A HMM is characterized by a set of $T$ states, by an initial probability distribution for the first state $\pi$, by a transition probability matrix connecting states $A$, and by a state-dependent probability distribution on the outputs $B$. The model parameter is $\Theta = (\pi, A, B)$ including *initial probability transition probability* and *emission probability* (a.k.a. output probability). The initial probability $\pi$ specifies the initial probability for the first state of the sequence. The transition probability $A$ indicates the likelihood of the hidden state at time $t$ is chosen given the hidden state at time $t-1$. The emission probability $B$ governs the distribution of the observed variable

at a particular time given the state of the hidden variable at that time.

We assume there are $K$ latent states. The initial probability $\boldsymbol{\pi} = [\pi_1 \pi_2 ... \pi_K]$ indicates the likelihood for starting the sequence with state $k$ is defined as: $\pi_k = p(s_1 = k), \forall k = 1....K$, and $\sum_{k=1}^{K} \pi_k = 1$. The state transition matrix is $A = \{a_{ij}\}$ where $a_{ij} = p(s_{t+1} = j \mid s_t = i), \forall i, j = 1...K$. We note that $\sum_{j=1}^{K} a_{ij} = 1$. The observation probability matrix denotes as $B = \{b_k(v)\}$ where $b_k(v) = p(x_t = v \mid s_t = k), \forall k = 1...K$ and $v = 1...V$ if we assume the observed value $x_t$ is discrete, taking value from $v = 1...V$.

Given an output (or observation) sequence $X = \{x_t\}_{t=1}^{T}$ and the hidden sequence $S = \{s_t\}_{t=1}^{T}$, the parameter learning task in HMMs is to find the best set of state transition probability $A$, emission probability $B$ and initial probability $\pi$. The task is usually to derive the maximum likelihood estimation for the parameters of the HMM given the set of observation sequences. There is no algorithm for solving this problem exactly, but a local maximum likelihood can be derived efficiently using the Baum–Welch algorithm (Rabiner and Juang, 1986). The Baum–Welch algorithm is a special case of the Expectation-Maximization algorithm (Dempster et al., 1977).

HMM joint probability distribution is written as:

$$p(X, S \mid \Theta) = p(s_1 \mid \pi) \prod_{i=2}^{T} p(s_i \mid s_{i-1}, A) \prod_{i=2}^{T} p(x_i \mid s_i, B).$$

Three basic problems associated with HMM outlined in (Rabiner, 1989) are:

- Computing the likelihood of the sequence of observations: $p(X \mid \Theta)$. We can utilise the forward-backward algorithm for this task.
- Finding the most likely underlying explanation of the sequence of observation: $\hat{S} = \underset{S}{\text{argmax}} p(S \mid X, \Theta)$. The Viterbi-algorithm is used for finding the best explanation of the sequence of observation.
- Estimating the parameter $\hat{\Theta}$ that maximizes the likelihood: $\hat{\Theta} = \underset{\Theta}{\text{argmax}} p(X \mid \Theta)$. For this setting, the Expectation Maximization (Dempster et al., 1977) and its special case as Baum–Welch algorithm (Rabiner and Juang, 1986) is applied.

For details on these problems, we refer the readers to (Rabiner, 1989). In this thesis, we are interested in the Infinite Hidden Markov model, a nonparametric counterpart of HMM, for video segmentation described in Section 2.3.5 and further in Chapter 3. Therefore, we briefly discuss below the algorithm for compute the first problem above.

**Forward and Backward recursion algorithms.** The forward recursion algorithm (a.k.a. alpha recursion) enables us to compute the desired likelihood given a sequence of observations. We would like to obtain a recursion between $\alpha(s_t)$ and $\alpha(s_{t+1})$. The idea is to condition on a state then use the conditional independence to decompose the probabilities. The forward/backward variables can be efficiently computed recursively via dynamic programming based on the conditional dependency from Markov property. The recursion for the forward variable (or alpha algorithm), for example, can be calculated as:

$$\begin{aligned}
\alpha(s_n) &= p(x_1, ..., x_n \mid s_n)\, p(s_n) \\
&= p(x_n \mid s_n)\, p(x_1, ..., x_{n-1} \mid s_n)\, p(s_n) \\
&= p(x_n \mid s_n) \sum_{s_{n-1}} p(x_1, ..., x_{n-1}, s_{n-1}, s_n) \\
&= p(x_n \mid s_n) \sum_{s_{n-1}} p(x_1, ..., x_{n-1}, s_{n-1})\, p(s_n \mid s_{n-1}) \\
&= B(x_n \mid s_n) \sum_{s_{n-1}} \alpha(s_{n-1})\, A(s_n \mid s_{n-1})
\end{aligned}$$

where $B$ is a emission probability and $A$ is a transition probability estimated from training data. We will recursively compute:

$$\alpha(s_t) = B(x_t \mid s_t) \sum_{s_{t-1}} \alpha(s_{t-1})\, A(s_t \mid s_{t-1}).$$

Similarly, we obtain a backward recursion (or beta algorithm) as follows $\beta(s_t) = B(x_t \mid s_t) \sum_{s_{t+1}} \beta(s_{t+1})\, A(s_t \mid s_{t+1})$.

#### 2.2.2.2 Applications of HMM

Hidden Markov models are widely used in almost all current speech recognition systems (Rabiner, 1989), in computational molecular biology (Bishop and Thompson, 1986), and in pattern recognition and computer vision (Starner and Pentland, 1997; Bui et al., 2004; Duong et al., 2005). Hidden Markov models are also popular in video analysis and video segmentation. We describe the applications of HMM for video segmentation that we further extend to Infinite Hidden Markov model for video surveillance abnormal detection in Chapter 3. As discussed in (Phung, 2005), there are two main themes of using Hidden Markov model for video segmentation: window-based and Viterbi-based methods.

In the window-based approach (Huang et al., 2000; Xie et al., 2002; Kijak et al., 2003), assuming a finite set of content categories exist, a HMM-based classification process is performed before the segmentation stage. A sliding window is then scanned through the entire video, in which at each step, the content within the window is input to a pool of HMMs to compute the likelihood it belongs to one of the content categories. The likelihood values are further used to determine the segmentation points, most widely by using dynamic programming to determine an optimal likelihood path for the entire video.

In the Viterbi-based approach (Iurgel et al., 2001; Boykin and Merlino, 2000; Merlino et al., 1997), instead of training a series of HMMs for a pre-defined set of video content classes, this approach uses a single HMM to model the entire video, usually with some specific domain knowledge such as typical information of the video. After training, a video is segmented based on a Viterbi decoding on the video.

### 2.2.3 Latent Dirichlet Allocation

Latent Dirichlet allocation (LDA) (Blei et al., 2003) is a generative model that allows groups of observations to be explained by the collection of hidden topics. For example, if observations are words collected into documents, it posits that each document is a mixture of a small number of topics and that each word's creation is attributable to one of the document's topics. In chapter 5, we would use LDA as one

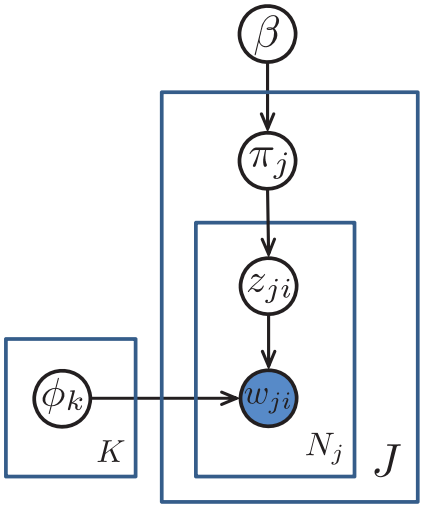



Figure 2.2.5: Latent Dirichlet Allocation. There are $J$ documents, each document has $N_j$ (observed) words $w_{ji}$.

of the extraction tools for classification task to exploit the multinomial distribution from mixing proportion.

#### 2.2.3.1 Model representation and inference

Each document is assumed to be characterized by a particular set of topics. This is akin to the standard bag of words model assumption and makes the individual words exchangeable. This is similar to probabilistic latent semantic analysis (pLSA) (Hofmann, 1999), except that in LDA the topic distribution is assumed to have a Dirichlet prior. In practice, this results in more reasonable mixtures of topics. It has been noted, however, that the pLSA model is equivalent to the LDA model under a uniform Dirichlet prior distribution (Girolami and Kabán, 2003).

LDA is a parametric model that requires a pre-defined number of topics. The graphical representation of LDA is displayed in Fig. 2.2.5. For the full model specification, we refer the interested readers to the original paper (Blei et al., 2003) that we sketchy summarize the model in this section. In LDA, there are $K$ topics $\beta_k$, $k \in \{1, ..., K\}$ ($K$ is initialized and fixed), which are discrete distributions over words. The mixture proportion outputs $\pi_j$ in Fig. 2.2.5 is a random mixture over hidden topics that reflect the topic assignment distribution over each document. We

describe the generative process of LDA as follows. We generate a collection of topics $\phi_k$ from Dirichlet distribution and a mixture proportion for each document $\pi_j$.

$$\phi_k \sim \text{Dir}(\eta) \;\; k \in \{1, ..., K\} \qquad\qquad \pi_j \sim \text{Dir}(\alpha) \;\; j \in \{1, ..., J\}$$

Then for each word $i$ in the document $j$, we draw a topic index $z_{ji} \in \{1, ..., K\}$ from the mixture weight $\pi_j$ for document $j$. Next, we generate the observation $w_{ji}$ from the selected topic $\phi_{z_{ji}}$.

$$z_{ji} \overset{\text{iid}}{\sim} \text{Categorical}(\pi_j) \;\; j \in \{1, ..., J\}, i \in \{1, ..., N_j\}$$
$$w_{ji} \overset{\text{iid}}{\sim} \text{Multinomial}\left(\phi_{z_{ji}}\right) \;\; j \in \{1, ..., J\}, i \in \{1, ..., N_j\}$$

If we sum over the topic assignments $\boldsymbol{z}$, then we get $p(w_{ji} = v \mid \pi_j, \phi) = \sum_{k=1}^{K} \pi_{jk} \phi_{kv}$.

According to the conjugate property of Multinomial-Dirichlet (Schlaifer and Raiffa, 1961), the posterior estimation of $\pi_j$ is also followed Multinomial Dirichlet distribution. In this setting, the extracted features $\pi_j$ are assumed to be drawn from multiple group-specific distributions and this *allows documents within a class share the same set of weights* - this nature will be beneficial in classification. We would exploit this property for classification task in Chapter 5.

Parameter estimation in such a Bayesian network of LDA is a problem of Bayesian inference. The original paper (Blei et al., 2003) used a variational Bayes approximation of the posterior distribution while the alternative inference techniques such as Gibbs sampling (Griffiths and Steyvers, 2004) and expectation propagation (Minka and Lafferty, 2002) can also be used. We would briefly describe the collapsed Gibbs inference for LDA (Griffiths and Steyvers, 2004) where the mixing proportion $\pi_j$ and topic $\phi_k$ are analytically integrated out due to conjugacy property (Diaconis et al., 1979). The remaining latent parameter $\boldsymbol{z}$ need to be sampled as

$$\begin{aligned} p(z_{ji} = k \mid \boldsymbol{z}_{-ji}, \mathbf{w}_{-ji}, \alpha) &\propto p(z_{ji} = k \mid \boldsymbol{z}_{-ji}, \alpha)\, p(w_{ji} = v \mid \mathbf{w}_{-ji}, z_{-ji}) \\ &= \frac{n^{k}_{j,-i}}{n^{*}_{j,-i} + K\alpha} \times \frac{C^{v}_{k,-(ji)} + \eta}{C^{*}_{k,-(ji)} + V\eta} \end{aligned}$$

where we denote that $n^{k}_{j,-i} = \sum_{\forall i' \neq i} \mathbb{I}(z_{ji'}, k)$, $n^{*}_{j,-i} = \sum_{k=1}^{K} n^{k}_{j,-i}$, $V$ is a dictionary size, $C^{v}_{k,-(ji)}$ is the count of the set $\{w_{j'i'} \mid z_{j'i'} = k, w_{j'i'} = v, \forall (j'i') \neq (ji)\}$, and

$C^{*}_{k,-(ji)} = \sum_{v=1}^{V} C^{v}_{k,-(ji)}$.

#### 2.2.3.2 Applications of LDA

LDA has been used widely for uncovering the underlying semantic structure of a document collection based on a hierarchical Bayesian analysis (Blei et al., 2003; Deerwester et al., 1990; Griffiths and Steyvers, 2004). By discovering patterns of words and connecting documents that exhibit similar patterns, LDA has emerged as a powerful new technique for finding useful structure in a collection of documents. For large scale data analysis with topic model, online learning for LDA (Hoffman et al., 2010) is developed following the idea of online stochastic optimization that can handle millions of articles. Spatial Latent Dirichlet Allocation (SLDA) model (Wang and Grimson, 2007) encodes the spatial structure among visual words. SLDA clusters visual words (e.g. an eye patch and a nose patch), which often occur in the same images and are close in space, into one topic (e.g., face).

Topic models can connect words with similar meanings and distinguish between uses of words with multiple meanings. Author Topic Model (Rosen-Zvi et al., 2004) is proposed to investigate the interests of authors from a large collection of documents. Topic Over Time (Wang and McCallum, 2006) is introduced to capture not only the low-dimensional structure of data, but also how the structure changes over time. Another way to analyse time series data, Blei and Lafferty (2006) propose the Dynamic Topic Models using state space models. A correlated topic model of science (Blei and Lafferty, 2007) is introduced to overcome the limitation of LDA that cannot learn the correlation of the topics. Another variant of topic model is the contextual focused topic model (Chen et al., 2012) which aims to infer a sparse set of topics for each document whilst leveraging context information about author and document venue.

## 2.3 Bayesian Nonparametric Data Modelling

Given the number of clusters, we can identify the subpopulations from the data using probabilistic mixture model in Section 2.2.1. However, the question is how

many clusters should we use in our mixture model (Fraley and Raftery, 1998)? This question regularly exercise scientists by fitting the data with different numbers of clusters, then selecting one using model comparison metrics (Claeskens and Hjort, 2008).

Bayesian nonparametric (BNP) models (Orbanz and Teh, 2010) provide better solution to this problem. Instead of comparing models with different number of clusters, BNP approach allows the model complexity growing with the data. Therefore, Bayesian nonparametric approach overcomes the problem of model selection.

The term 'nonparametric' does not mean that the models are free of parameters (Murphy, 2012). Instead, these models can have a larger set of parameters (compare to the finite models). However, these parameters get adjusted automatically with the data. While parametric models assume a fixed set of parameters, Bayesian nonparametric model defines a prior distribution over an infinite dimensional parameter space. The parameter space represents the set of all possible solutions for a given learning problem – for example, the set of smooth functions in nonlinear regression, or of all probability densities in a density estimation problem.

We describe the foundation background on Bayesian nonparametric approach for data modelling, as a major theme of this thesis. We start with Dirichlet Process (Ferguson, 1973) and Dirichlet Process Mixture (Antoniak, 1974). Then, we derive the small variance asymptotic version of Dirichlet Process Mixture, a.k.a. DPmeans (Kulis and Jordan, 2012). Next, we present the Hierarchical Dirichlet Processes (Teh et al., 2006) for modelling grouped data and HDPmeans for scalable modelling grouped data. We further review the Infinite Hidden Markov model (Teh et al., 2006) for modelling time series data and the Nested Dirichlet Process (Rodriguez et al., 2008) for document clustering. All of these Bayesian nonparametric frameworks build up a theoretical foundation from which this thesis would develop for multilevel modelling throughout Chapters 3,4,5,6, and 7.

### 2.3.1 Dirichlet Process and Dirichlet Process Mixture

In this section, we present Dirichlet Process (Ferguson, 1973) and nonparametric mixture modelling using Dirichlet Process prior, a.k.a. Dirichlet Process Mixture

(Antoniak, 1974).

#### 2.3.1.1 Dirichlet Proceses

We provide a brief account of the Dirichlet Process and its variants.

**Definition 2.4.** (Dirichlet Process) A Dirichlet process $\mathrm{DP}(\gamma, H)$ is a distribution of a random probability measure $G$ over the measurable space $(\Theta, \mathcal{B})$ where $H$ is a *base* probability measure and $\gamma > 0$ is the *concentration* parameter. It is defined such that, for any finite measurable partition $(A_k : k = 1, \ldots, K)$ of $\Theta$, the resultant finite-dimensional random vector $(G(A_1), \ldots, G(A_k))$ is distributed according to a Dirichlet distribution with parameters $(H(A_1), \ldots, H(A_k))$. Formally, we have

$$(G(A_1), \ldots, G(A_k)) \sim \mathrm{Dir}(H(A_1), \ldots, H(A_k))$$

Dirichlet process and its existence was established by Ferguson (1973) who has also showed that draws from a DP are discrete with probability one.

**Stick-breaking.** Before going into details of Stick-breaking construction for Dirichlet Process, we describe the stick-breaking metaphor in Fig. 2.3.1. Let assume we have a stick's length of 1. We want to divide the stick into countably infinite pieces $\pi_k$ such that $\sum_{k=1}^{\infty} \pi_k = 1$. Let $\beta_k \overset{\text{iid}}{\sim} \mathrm{Beta}(1, \alpha)$ for $k = 1, 2, 3 \ldots \infty$ is the proportion of the $k$-th partition of the stick. Firstly, we cut a stick by a length of $\pi_1 = \beta_1 \times 1$, the remainder of the stick is $1 - \beta_1$. We continue splitting the remaining stick repeatedly. At the time $k$-th, the remaining stick's length is $(1 - \beta_1) \ldots \times (1 - \beta_{k-1})$ which is further cut by a piece of $\pi_k = \beta_k \times (1 - \beta_1) \ldots \times (1 - \beta_{k-1})$. We continue the stick-breaking process until the remaining length of the stick as $k \to \infty$ approaches zero. Formally, each stick length at iteration $k$ can be expressed as: $\pi_k = \beta_k \prod_{i=1}^{k-1} (1 - \beta_i)$ with $\beta_k \overset{\text{iid}}{\sim} \mathrm{Beta}(1, \alpha)$ for $k = 1, 2, 3 \ldots \infty$. We can shortly write: $\pi_{1:\infty} = \mathrm{Stick}(\alpha)$.

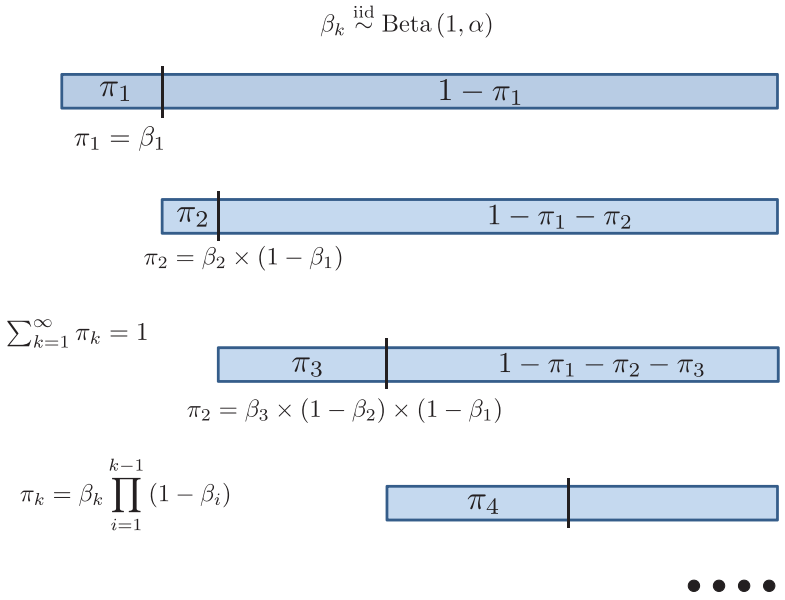



Figure 2.3.1: Stick-breaking illustration $\boldsymbol{\pi} = [\pi_1\pi_2, ...\pi_K] \sim \text{Stick}(\alpha)$

Sethuraman (1994) provides an alternative construction which makes the discreteness property of a Dirichlet process explicitly via a stick-breaking construction

$$G = \sum_{k=1}^{\infty} \beta_k \delta_{\phi_k} \tag{2.3.1}$$

where $\phi_k \overset{\text{iid}}{\sim} H, k = 1, \ldots, \infty$ and $\boldsymbol{\beta} = (\beta_k)_{k=1}^{\infty}$ are the weights constructed through a 'stick-breaking' process $\beta_k = v_k \prod_{s<k}(1 - v_s)$ with $v_k \overset{\text{iid}}{\sim} \text{Beta}(1, \gamma), k = 1, \ldots, \infty$. It can be shown that $\sum_{k=1}^{\infty} \beta_k = 1$ with probability one, and as a convention (Pitman, 2002), we hereafter write $\boldsymbol{\beta} \sim \text{GEM}(\gamma)$.

**Polya-urn scheme.** Another useful interpretation for the Dirichlet process is given by the Polya-urn scheme (Blackwell and MacQueen, 1973). It shows that drawing from the Dirichlet process are not only discrete, but also exhibit a clustering property. More concretely, let $\theta_1, \theta_2, \ldots, \theta_{N+1}$ be iid draws from $G$, Blackwell and MacQueen (1973) showed that $G$ can be integrated out to give the following marginal conditional distribution form

$$\theta_{N+1} \mid \theta_1, \ldots, \theta_N, \gamma, H \sim \sum_{i=1}^{n} \frac{1}{N + \gamma} \delta_{\theta_i} + \frac{\gamma}{N + \gamma} H. \tag{2.3.2}$$

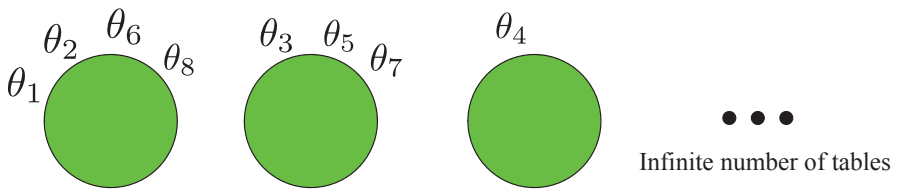



Figure 2.3.2: Chinese Restaurant Process visualization. Customers are denoted as $\theta_1, \theta_2, ...$ gathering in the countably infinite number of table.

If we further group identical values in the set $\{\theta_1, \ldots, \theta_N\}$ together and let $K$ be the number of such distinct values, each represented by $\phi_k$ with $n_k$ be its count, then Eq. 2.3.2 is equivalent to:

$$\theta_{n+1} \mid \theta_1, \ldots, \theta_n, \gamma, H \sim \sum_{k=1}^{K} \frac{n_k}{N+\gamma} \delta_{\phi_k} + \frac{\gamma}{N+\gamma} H. \tag{2.3.3}$$

This expression is clearly showing the clustering property induced by $G$: a future draw $\theta$ is likely to return to an existing atom $\phi_k$ and it does so with a probability proportional to the popularity $n_k$ of the respective atom. However, it may also pick on a new cluster with a probability proportional to the concentration parameter $\gamma$.

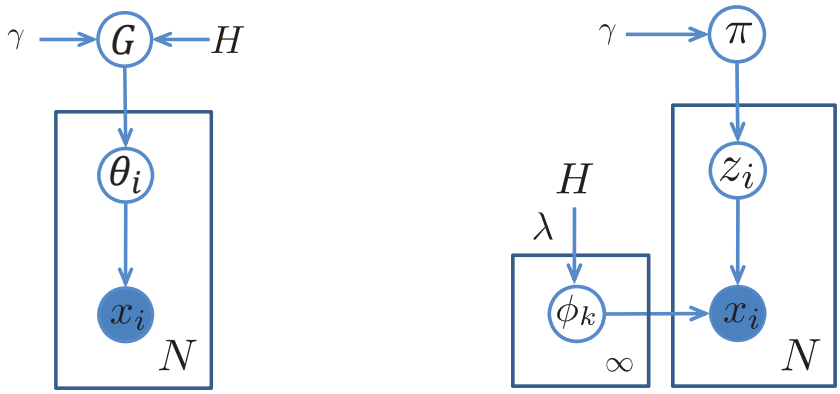
**Chinese Restaurant Process.** Chinese Restaurant Process is a distribution over partitions of integers. Let imagine that there are $N$ customers $(\theta_1, \theta_2..., \theta_N)$ coming to the restaurant which has the infinite number of table (cf. Fig. 2.3.2). The tables are chosen by the customers following the random process. The first customer sits at the first table. The $i$-th customer sits at an occupied or new table with different probabilities. He takes the occupied table with the probability of $\frac{n_k}{N+\gamma-1}$ where $n_k$ is the number of people sitting at that table $k$-th. He will select the unoccupied (or new) table with the probability of $\frac{\gamma}{N+\gamma-1}$ where $\gamma$ is the concentration parameter that indicates the customer's tendency to pick a new table.

Some properties of the Chinese Restaurant Process are identified. It is exchangeable that we can permute the order of customers without changing the process's probability. The rich get richer such that the customers tend to sit at the most crowded table. A customer belongs to one table only. The bigger concentration parameter $\alpha$ is, the more number of table we have.

The expression in Eq. 2.3.3 can also be viewed as the Chinese Restaurant Process. As such, the probability of a customer to seat at an occupied table $k$ is proportional to the number of customers already in the table. The probability of a customer to choose a new table is proportion to the concentration parameter $\gamma$:

$$\theta_{n+1} \mid \theta_1, \ldots, \theta_n, \gamma, H = \begin{cases} \frac{n_k}{N+\gamma-1}\delta_{\phi_k} & \text{used table } k \\ \frac{\gamma}{N+\gamma-1}H & \text{new table.} \end{cases}$$

#### 2.3.1.2 Dirichlet Process Mixture




(a) Stochastic view (b) Stick-breaking view

Figure 2.3.3: Graphical model representation for Dirichlet Process Mixture.

Due to its discreteness, the Dirichlet Process (Ferguson, 1973) is often not applied directly to modelling the data (e.g., it is unable to model continuous data) instead it can be effectively used as a nonparametric prior on the mixture components $\theta$, which in turn serves as the parameters within another likelihood function $F$ to generate data - a model which is known as Dirichlet Process Mixture model (DPM) (Antoniak, 1974; Escobar and West, 1995). To be precise, under a DPM formalism an observation $x_n$ is generated from a two-step process: $x_n \sim F(x_n \mid \theta_n)$ where $\theta_n \sim G$. Using the stick-breaking representation in Eq. 2.3.1, it is not hard to see that DPM yields an *infinite* mixture model representation:

$$p(x \mid \gamma, H) = \sum_{k=1}^{\infty} \beta_k \times f(x \mid \phi_k) \tag{2.3.4}$$

where $f$ denotes the density function for $F$. Dirichlet Process Mixture models have been embraced with a great success and enthusiasm recently (Gelfand et al., 2005; Neal, 2000). The crucial advantage is its ability to naturally address the problem of model selection - a major obstacle encountered in several parametric mixture modelling, such as the Gaussian Mixture Models where the number of mixtures cannot be specified apriori in a principal way.

We present the stochastic process of Dirichlet Process Mixture. Firstly, we draw a global atom from Dirichlet Process $G \sim \mathrm{DP}(\gamma H)$, then each local atom (for each data point) is iid sampled as $\theta_i \overset{\text{iid}}{\sim} G$. Finally, the data observation is generated from a corresponding local atom $x_i \sim F(\theta_i)$.

To characterize the stochastic process, we provide the stick-breaking representation for posterior inference. Given the concentration parameter $\gamma$, the mixing proportion is drawn from $\pi_\infty \sim \mathrm{GEM}(\gamma)$. A collection of topics is also sampled from a base measure $\phi_k \overset{\text{iid}}{\sim} H(\lambda)$. The global atom $G$ from stochastic process above can be represented as $G = \sum_{k=1}^{\infty} \pi_k \delta_{\phi_k}$. Each local indicator (or topic assignment) is then generated as $z_i \overset{\text{iid}}{\sim} \pi$. Finally, we draw the data observation $x_i \sim F(\phi_{z_i})$.

#### 2.3.1.3 Posterior inference for Dirichlet Process Mixture

Our Dirichlet Process Mixture is built as a Bayesian model. Therefore, we need to perform posterior inference to estimate the posterior distributions for all hidden variables. Using conjugacy property (Diaconis et al., 1979), we would integrate out the variable $\pi$ and $\phi_k$. We need to sample the latent variable $z$ and concentration parameter $\gamma$. We below utilise a Bayes rule: $p(A \mid B) = \frac{p(A)p(B|A)}{p(B)} = \frac{p(A)p(B|A)}{\sum_A p(A)p(B|A)}$.

**Sampling $z$:** Consider the conditional independence (Koller and Friedman, 2009) of the graphical representation in Fig. 2.3.4, we ignore the variables that do not influence on the conditional likelihood of latent assignment $z_i = k$ in Fig. 2.3.3 given other variables, is written as:

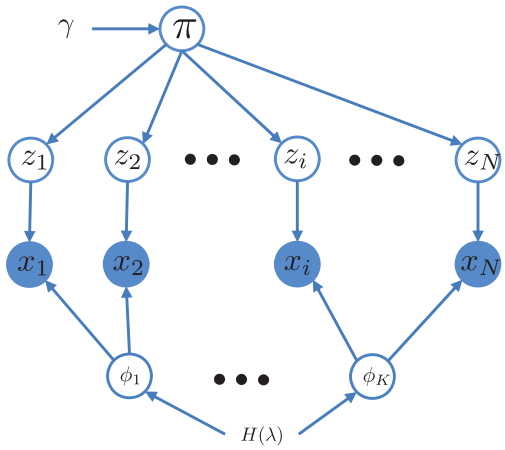



Figure 2.3.4: Visualising the variables in Dirichlet Process Mixture.

$$p\left(z_i = k \mid .\right) = \frac{p\left(x_i \mid z_i = k, \mathbf{x}_{-i}, \boldsymbol{z}_{-i}, H\right) \times p\left(z_i = k \mid \mathbf{z}_{-i}, \gamma\right)}{p\left(x_i \mid \mathbf{x}_{-i}, \boldsymbol{z}_{-i}, \gamma, H\right)} \tag{2.3.5}$$

where we split $\mathbf{x}$ into $x_i$ (at data point $i$-th) and $x_{-i}$ (at other data points), similarly $\mathbf{z}$ is split into $z_i$ and $\mathbf{z}_{-i}$. In Eq. 2.3.5, the denominator of $p\left(\mathbf{x}, \boldsymbol{z}_{-i}, \gamma, H\right)$ serves a role as normalization term which is constant and fixed (as we observe all of these variables $\boldsymbol{x}, \boldsymbol{z}_{-i}, \gamma, H$ ) in sampling $z_i$. We can ignore the denominator and compute the approximate likelihood in Eq. 2.3.5 to yield:

$$p\left(z_i = k \mid .\right) \propto \underbrace{p\left(x_i \mid z_i = k, \mathbf{x}_{-i}, H\right)}_{\text{predictive likelihood}} \times \underbrace{p\left(z_i = k \mid \mathbf{z}_{-i}, \gamma\right)}_{\text{CRP}}. \tag{2.3.6}$$

The first term of $p\left(x_i \mid z_i = k, \mathbf{x}_{-i}, H\right)$ in Eq. 2.3.6 is the predictive likelihood of observation $x_i$ to component $\phi_k$ after integrating $\phi_k$. This can be evaluated analytically due to conjugacy of $F$ and $H$. Specifically, let $f\left(\cdot \mid \phi\right)$ and $h\left(\cdot\right)$ be the density function for $F\left(\phi\right)$ and $H$, the conjugacy between $F$ and $H$ allows us to integrate out the mixture component parameter $\phi_k$, leaving us the conditional density of $x_i$

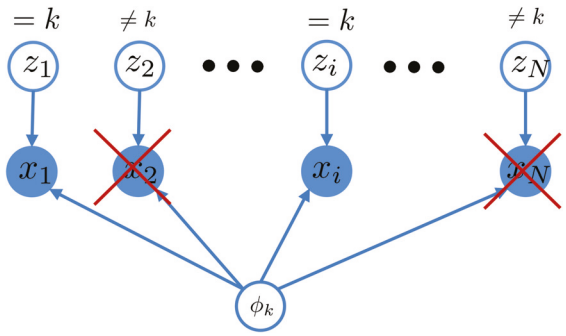



Figure 2.3.5: Visualising the conditional independent when observing $z_i = k$.

under the mixture component $k$ given all the data observations exclude $x_i$:

$$p\left(x_i \mid z_i = k, \left\{x_{i'} \mid z_{i'} = k, i' \neq i\right\}, H\right) = \frac{\int_{\phi_k} f\left(x_i \mid \phi_k\right) \prod_{\forall i'} f\left(x_{i'} \mid \phi_k\right) h\left(\phi_k\right) d\phi_k}{\int_{\phi_k} \prod_{\forall i'} f\left(x_{i'} \mid \phi_k\right) h\left(\phi_k\right) d\phi_k}$$
$$= f_k^{-x_i}\left(x_i\right).$$

The second term in Eq. 2.3.6 is followed Chinese Restaurant Process after we integrate out $\pi$:

$$p\left(z_i = k \mid \mathbf{z}_{-i}, \gamma\right) = \int_{\pi} p\left(z_i = k \mid \pi\right) p\left(\pi \mid \gamma\right) d\pi$$
$$= \begin{cases} \frac{n_k}{N+\gamma-1} & \text{used k} \\ \frac{\gamma}{N+\gamma-1} & \text{new k} \end{cases}$$

where $n_k$ is the count of number of data points belonged to component $k$-th and $N$ is the total number of data points. To summarize, the sampling procedure for $z_i$ in DPM is as:

$$p\left(z_i = k \mid \boldsymbol{x}, \boldsymbol{z}_{-i}, \gamma, H\right) \propto \begin{cases} \frac{n_k}{\gamma+n-1} \times f_k^{-x_i}\left(x_i\right) & \text{for } 1 \leq k \leq K \\ \frac{\gamma}{\gamma+n-1} \times f_{k_{\text{new}}}^{-x_i}\left(x_i\right) & \text{for } k = K+1. \end{cases} \tag{2.3.7}$$

**Sampling concentration parameter $\gamma$.** The hyperparameter $\gamma$ can be seen as prior pseudo-counts, i.e. the number of the data points in a cluster that we apriori specify before observing any data. In practice, these hyperparameters are further endowed with distributions (e.g., Gamma distribution) and integrated out. This approach is a typical scheme in Bayesian hierarchical modelling (Gelman et al., 2003) to

ensure the robustness of the model. Therefore, sampling hyperparameter makes the model robust in identifying the unknown number of clusters. By robust, we mean the results are resilient the changes of the hyperparameters. The posterior distribution of the hyperparameter can be computed as a function of the prior hyperparameters and the observed data. Sampling $\gamma$ is similar to (Escobar and West, 1995). We place a Gamma prior over $\gamma$, assuming $\gamma \sim \text{Gamma}\,(\gamma_1, \gamma_2)$. Given $N$ is the number of data point, we define the auxiliary variable $t$ as $p\,(t \mid \gamma, K) \propto \text{Beta}\,(\gamma_1 + 1, N)$. Then, the posterior distribution for sampling $\gamma$ is as:

$$p\,(\gamma \mid t, K) \sim \pi_t \text{Gamma}\,(\gamma_1 + K, \gamma_2 - \log(t)) + (1 - \pi_t)\text{Gamma}\,(\gamma_1 + K - 1, \gamma_2 - \log(t))$$

where $\pi_k$ are computed as $\frac{\pi_t}{1-\pi_t} = \frac{\gamma_1+K-1}{N(\gamma_2-\log t)}$ given the auxiliary variable $t$.

#### 2.3.1.4 Illustrating DPM for nonparametric clustering

To illustrate the nonparametric clustering task using Dirichlet Process Mixture, when the number of clusters is not known in advance, we use simulated data. We generate $K = 5$ clusters using five 2D Gaussian distributions with different means and variances (cf. Top Left Fig. 2.3.6). Then, we initialize our collapsed Gibbs sampler for DPM with $K = 2$ clusters. We plot the data clustering behavior w.r.t. iterations in Fig. 2.3.6. After 39 iterations, DPM identifies the correct number of clusters ($K = 5$) as shown in Right Bottom Fig. 2.3.6.

#### 2.3.1.5 Applications of DPM

Due to the nonparametric prior, we build DPM as a single mixture model in which the number of mixture components is unknown and growing with the data. This means that DPM does not require the number of clusters apriori and it allows us to adapt the number of active clusters as we feed more data to our model over time. Thus, Dirichlet Process Mixture (Antoniak, 1974) is suitable for nonparametric clustering where the true number of clusters is not known in advance.

DPM has been widely used in a large number or applications. Wood et al. (2006) have used DPM to perform spike sorting and identify the number of different neurons that were monitored by a single electrode. Sudderth et al. (2008) have used this

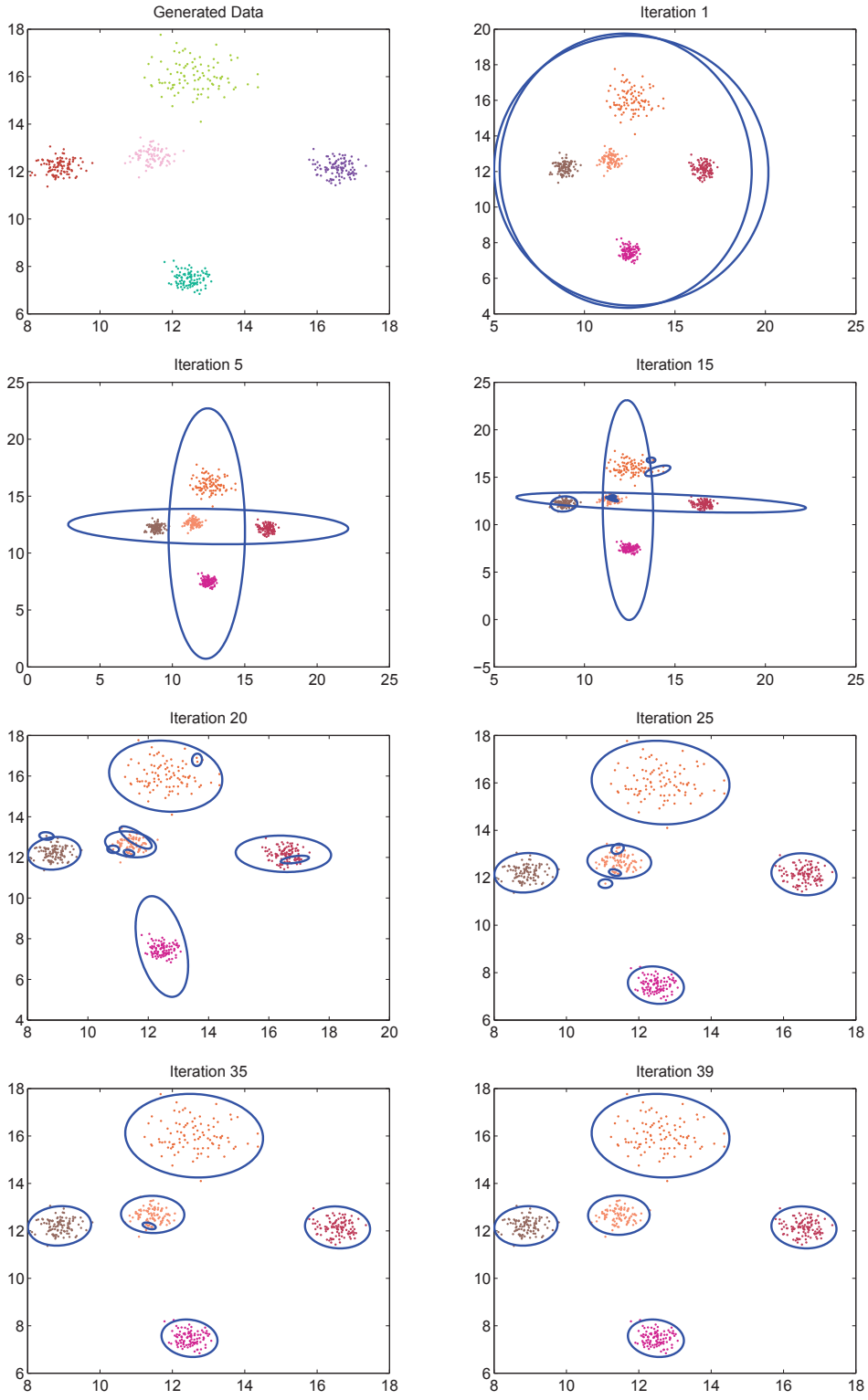



Figure 2.3.6: Dirichlet Process Mixture demo using 2-dimensional data. Top Left: The simulated data. Top Right: Initialization (Iteration 1). Middle and Bottom Rows: Gibbs outputs from subsequent iterations.

model to perform visual scene analysis and identify the number of objects, parts and features that a particular image contains. Blei and Jordan (2006) utilise variational inference for DPM on image clustering task where each image is reduced to a 192-dimensional real-valued vector given by an 8×8 grid of average red, green, and blue values. They fit a DP mixture model in which the mixture components are Gaussian with mean μ and covariance matrix $\sigma^2 I$. In this work, Vlachos et al. (2008) apply Dirichlet Process Mixture Models to a learning task in natural language processing (NLP) of lexical-semantic verb clustering. They assess the performance on a dataset based on verb classes using the recently introduced Vmeasure metric. Another application of DPM is in density estimation (Escobar and West, 1995; Rasmussen, 1999; Lo, 1984) where we are interested in modelling the density from which a given set of observations is drawn.

### 2.3.2 DPmeans

The classical Gaussian Mixture Model (GMM) is related to K-means via small variance asymptotic: as the covariance of the Gaussians tends to zero, the negative log-likelihood of the mixture of Gaussians model approaches the K-means objective function, and the EM algorithm approaches the K-means algorithm. Kulis and Jordan (2012) use this observation to obtain a novel K-means-like algorithm from a Gibbs sampler for the Dirichlet Process Mixture (DPM), a.k.a. DPmeans. In this section, we provide detailed derivation of the key results (Kulis and Jordan, 2012) aiming to help non-specialist to follow, which is not fully described in the original paper. Motivated from DPmeans, we later on propose the Nested Kmeans for scalable multilevel clustering in Chapter 7.

#### 2.3.2.1 DPmeans derivation from DPM

We consider a full Gibbs inference scheme for DPM where only the stick-breaking weights $\{\pi_1, \pi_2, \ldots\}$ are integrated out. Let $K$ be the current active number of atoms at each iteration; the state space includes $\{z_1, \ldots, z_n, \phi_1, \phi_2, \ldots, \phi_K\}$ and we shall iteratively sample individual variable conditioning on the remainder set:

- Sample $z_i$: given $\alpha$ is a concentration parameter, $N$ is the total number of data points, $n_k$ is the number of data points in cluster $k$, and $\sum_{k=1}^{K} n_k = N$

$$p\left(z_i = k \mid \alpha, \boldsymbol{x}, \boldsymbol{z}_{-i}, \phi_{1:K}\right) = \begin{cases} \frac{n_k}{\alpha+N-1} f\left(x_i \mid \phi_{z_i}\right) & \text{for } 1 \leq k \leq K \\ \frac{\alpha}{\alpha+N-1} \int_{\phi} p\left(x_i \mid \phi\right) dG\left(\phi\right) & \text{for } k = K+1. \end{cases} \tag{2.3.8}$$

- Sample $\phi_k = \{\mu_k, \sigma\mathbf{I}\}$ : as we fixe the covariance of $\sigma\mathbf{I}$, the prior distribution $G(\phi)$ is only parameterized for $\mu$. Let $\{x_i \mid z_i = k, \forall i\}$ be the set of data points associated with component $k$, then:

$$\begin{aligned} p\left(\mu_k \mid \boldsymbol{x}, \boldsymbol{z}, G\right) &\propto p\left(\{x_i \mid z_i = k, \forall i\} \mid \mu_k, \sigma\mathbf{I}\right) p\left(\mu_k \mid G\right) \\ &= \prod_{\forall i, z_i = k} p\left(x_i \mid \mu_k, \sigma\mathbf{I}\right) p\left(\mu_k \mid G\right). \end{aligned} \tag{2.3.9}$$

Our line of analysis is to start with the full Gibbs sampling for DPM with Gaussian likelihood, then examine the limit form of the sampling Eqs. 2.3.8 and 2.3.9 when the variance goes to zero.

**Hard assignment cluster indicator $z_i$ .** We begin with deriving hard assignment for cluster indicator $z_i$ by starting with standard Gibbs probabilities, then providing asymptotic analysis on the Gibbs probabilities.

**Gibbs probabilities for cluster indicator.** The Gibbs sampling equation for $z_i$ is presented in Eq. 2.3.8. Let $f$ and $G$ be (multivariate) Gaussian distributions, we consider two cases in sampling $z_i$:

$$p\left(z_i = k \mid .\right) = \begin{cases} \frac{n_k}{\alpha+N-1} \left(2\pi\sigma\right)^{-d/2} \exp\left\{-\frac{1}{2\sigma} \left\|x_i - \mu_k\right\|^2\right\} & \text{for } 1 \leq k \leq K \\ \frac{\alpha}{\alpha+N-1} \int_{\phi} f\left(x_i \mid \phi\right) dG(\phi) & \text{for } k = K+1. \end{cases} \tag{2.3.10}$$

We below present the proof that the case of $z_i = K+1$ in Eq. 2.3.10 can be expressed as:

$$p\left(z_i = K+1 \mid .\right) = \frac{\alpha}{\alpha+N-1} \left(2\pi(\sigma+\rho)\right)^{-d/2} \exp\left(-\frac{1}{2(\sigma+\rho)} \parallel x_i \parallel^2\right).$$

*Proof.* The integration term can be characterized as:

$$\begin{aligned}&\int_{\phi} f\left(x_i \mid \phi\right) dG(\phi)\\ &= \int \underbrace{\exp\left\{-\frac{1}{2\sigma}(x_i-\mu)^T(x_i-\mu)-\frac{1}{2\rho}\mu^T\mu\right\}}_{[1]}\underbrace{\frac{1}{(2\pi)^d(\sigma\rho)^{d/2}}}_{[2]}d\mu.\end{aligned}$$

Since $\mu^T x_i$ is scalar, we can write $\mu^T x_i = x_i^T\mu$, then the term [1] becomes:

$$[1] \;=\; \exp\left\{-\frac{1}{2}\left(\mu^T A\mu - B^T\mu + C\right)\right\} \tag{2.3.11}$$

where $A = (\frac{1}{\sigma}+\frac{1}{\rho})I,\ B = \frac{2x}{\sigma},\ C = \frac{x^Tx}{\sigma}$. Moreover, using the integral of multivariate Gaussian density function, we have

$$\int \exp\left\{-\frac{1}{2}(y-\lambda)^T\Sigma^{-1}(y-\lambda)\right\}dy = (2\pi)^{d/2}\left|\sum\right|^{1/2}. \tag{2.3.12}$$

Let express the term inside exponential formula:

$$(y-\lambda)^T\Sigma^{-1}(y-\lambda) \;=\; y^T\Sigma^{-1}y - 2\lambda^T\Sigma^{-1}y + \lambda^T\Sigma^{-1}\lambda. \tag{2.3.13}$$

Let denote $D = \sum^{-1}$ and $E^T = 2\lambda^T\sum^{-1} = 2\lambda^T D$, we have $\lambda^T = \frac{E^TD^{-1}}{2}$ or equivalently $\lambda = \frac{(D^{-1})^TE}{2}$, and $\lambda^T\sum^{-1}\lambda = \frac{E^T(D^{-1})^TE}{4}$. The Eq. 2.3.12 can be rewritten as follows

$$\int \exp\left\{-\frac{1}{2}\left(\mu^T D\mu - E^T\mu + \frac{E^T(D^{-1})^TE}{4}\right)\right\}d\mu \;=\; \frac{(2\pi)^{d/2}}{|D|^{1/2}}. \tag{2.3.14}$$

Let combine Esq. 2.3.11 and 2.3.14 and denote $U = \frac{B^T(A^{-1})^TB}{4}$, we obtain

$$\begin{aligned}[1] &= \exp\left\{-\frac{1}{2}\left(C-U\right)\right\}\int \exp\left(-\frac{1}{2}\left[\mu^TA\mu - B^T\mu + U\right]\right)d\mu\\ &= (2\pi)^{d/2}\left|A^{-1}\right|^{1/2}\exp\left\{-\frac{1}{2}\left(C-\frac{B^T(A^{-1})^TB}{4}\right)\right\}.\end{aligned} \tag{2.3.15}$$

We replace Eq. 2.3.15 using $A = (\frac{1}{\sigma} + \frac{1}{\rho})I, \quad B = \frac{2x}{\sigma}, \quad C = \frac{x^T x}{\sigma}$,

$$[1] = \left(2\pi\frac{\sigma\rho}{\sigma+\rho}\right)^{d/2} \exp\left\{-\frac{1}{2(\sigma+\rho)} \parallel x_i \parallel^2\right\}.$$

Therefore, we have the final form in sampling $z_i = K+1$ in Eq. 2.3.8 as:

$$p\left(z_i = K+1 \mid .\right) = \frac{\alpha}{\alpha+N-1}\left(2\pi(\sigma+\rho)\right)^{-d/2} \exp\left(-\frac{1}{2(\sigma+\rho)} \parallel x_i \parallel^2\right).$$

□

**Asymptotic derivation for cluster indicator.** We have the following probabilities to be used during Gibbs sampling:

$$\gamma\left(z_i = k\right) = \begin{cases} \frac{1}{Z} \times p\left(z_i = k \mid \boldsymbol{x}, \boldsymbol{z}_{-i}, \phi_{1:K}\right) & \text{for } 1 \le k \le K \\ \frac{1}{Z} \times p\left(z_i = K+1 \mid \alpha, G(\phi)\right) & \text{for } k = K+1 \end{cases} \tag{2.3.16}$$

where $Z = p\left(z_i = K+1 \mid \alpha, G(\phi)\right) + \sum_{k=1}^{K} p\left(z_i = k \mid \boldsymbol{x}, \boldsymbol{z}_{-i}, \phi_{1:K}\right)$. We below proof that, in the limit as $\sigma \to 0$, the Eq. 2.3.16 becomes the hard assignments as:

$$\lim_{\sigma\to 0} \gamma\left(z_i = k\right) = \begin{cases} \left\|x_i - \mu_k\right\|^2 & \text{for } 1 \le k \le K \\ \lambda & \text{for } k = K+1. \end{cases}$$

*Proof.* Let $\exp\left(-\frac{\lambda}{2\sigma}\right) = \alpha(1+\rho/\sigma)^{-d/2}$ for some $\lambda > 0$, we consider two cases of the cluster assignment. The probabilities of $z_i$ assigned to cluster $k$ $(1 \le k \le K)$ will be as (after canceling the common factors)

$$\gamma(z_i = k) = \frac{\exp\left\{-\frac{1}{2\sigma}\left\|x_i - \mu_k\right\|^2\right\}}{\exp\left\{-\frac{\lambda}{2\sigma} - \frac{\|x_i\|^2}{2(\sigma+\rho)}\right\} + \sum_{c=1}^{K} \exp\left\{-\frac{1}{2\sigma}\left\|x_i - \mu_c\right\|^2\right\}}.$$

Divide the above fraction for the numerator, the fraction becomes

$$\gamma(z_i = k) = \frac{1}{A + \sum_{c=1}^{K} B}$$

where $A = \exp\left\{-\frac{1}{2\sigma}\left(\lambda - \|x_i - \mu_k\|^2 + \frac{\sigma}{(\sigma+\rho)} \parallel x_i \parallel^2\right)\right\}$ and

$B = \exp\left\{-\frac{1}{2\sigma}\left(\|x_i - \mu_c\|^2 - \|x_i - \mu_k\|^2\right)\right\}$. Let take the limit $\sigma \to 0$ the denominator, the term $A + \sum_{c=1}^{K} B$ will be:

$$\lim_{\sigma\to 0}\left(A + \sum_{c=1}^{K} B\right) = \begin{cases} 1 & \text{if } \|x_i - \mu_k\|^2 \leq \lambda \text{ and } \|x_i - \mu_k\|^2 \leq \|x_i - \mu_c\|^2 \\ \infty & \text{if } \|x_i - \mu_k\|^2 > \lambda \text{ or } \|x_i - \mu_k\|^2 > \|x_i - \mu_c\|^2. \end{cases}$$

If $\|x_i - \mu_k\|^2$ is the smallest value of $\left\{\|x_i - \mu_1\|^2, \ldots, \|x_i - \mu_K\|^2, \lambda\right\}$, then we have $\lim_{\sigma\to 0}\gamma(z_i = k) = \frac{1}{0+\ldots+1+\ldots+0} = 1$. If $\|x_i - \mu_k\|^2$ is not the smallest value of $\left\{\|x_i - \mu_1\|^2, \ldots, \|x_i - \mu_K\|^2, \lambda\right\}$, then $\lim_{\sigma\to 0}\gamma(z_i = k) = \frac{1}{0+\ldots+\infty} = 0$.

The probabilities of $z_i$ assigned to new cluster $k = K + 1$ will be as (after canceling the common factors)

$$\gamma(z_i = K+1) = \frac{\exp\left\{-\frac{\lambda}{2\sigma} - \frac{\|x_i\|^2}{2(\sigma+\rho)}\right\}}{\exp\left\{-\frac{\lambda}{2\sigma} - \frac{\|x_i\|^2}{2(\sigma+\rho)}\right\} + \sum_{c=1}^{K}\exp\left\{-\frac{1}{2\sigma}\|x_i - \mu_c\|^2\right\}}.$$

Divide the above fraction for the numerator, we have

$$\gamma(z_i = K+1) = \frac{1}{1 + \sum_{c=1}^{K}\exp\left\{-\frac{1}{2\sigma}\left(\|x_i - \mu_c\|^2 - \lambda - \frac{\sigma}{(\sigma+\rho)} \parallel x_i \parallel^2\right)\right\}}.$$

Denote $A = \|x_i - \mu_k\|^2 - \lambda - \frac{\sigma}{(\sigma+\rho)} \parallel x_i \parallel^2$, we take the limit when $\sigma \to 0$ as

$$\lim_{\sigma\to 0}\exp\left\{-\frac{1}{2\sigma}A\right\} = \begin{cases} 0 & \text{if } \|x_i - \mu_k\|^2 > \lambda \\ 1 & \text{if } \|x_i - \mu_k\|^2 = \lambda \\ \infty & \text{if } \|x_i - \mu_k\|^2 < \lambda. \end{cases}$$

If $\lambda$ is the smallest value of $\left\{\|x_i - \mu_1\|^2, \ldots, \|x_i - \mu_K\|^2, \lambda\right\}$, then

$$\lim_{\sigma\to 0}\gamma(z_i = k) = \frac{1}{1 + \cdots + 0} = 1.$$

If $\lambda$ is not the smallest value of $\left\{\|x_i - \mu_1\|^2, \ldots, \|x_i - \mu_K\|^2, \lambda\right\}$, then

$$\lim_{\sigma \to 0} \gamma(z_i = k) = \frac{1}{1 + \infty + \cdots + 0} = 0.$$

□

**Hard assignment cluster mean $\mu_k$.** We will derive the computation for mean $\mu_k$ from Eq. 2.3.9 as

$$p\left(\mu_k \mid \boldsymbol{x}, \boldsymbol{z}, G\right) = \mathcal{N}\left(\mu_k \mid \tilde{\mu}_k, \tilde{\sum}_k\right)$$

where $\tilde{\sum}_k = \frac{\tilde{\sigma}_k \rho}{\tilde{\sigma}_k + \rho n_k} I$ with $\frac{1}{\tilde{\sigma}_k} = \frac{n_k}{\sigma} + \frac{1}{\rho}$, and $\tilde{\mu}_k = \left(\frac{\rho n_k}{\sigma + \rho n_k}\right) \overline{x_k}$ with $\overline{x_k} = \frac{\sum_{\forall i, z_i = k} x_i}{n_k}$.

*Proof.* We consider the Eq. 2.3.9: $p\left(\mu_k \mid \boldsymbol{x}, \boldsymbol{z}, G\right) = \prod_{\forall i, z_i = k} p\left(x_i \mid \mu_k, \sigma \mathbf{I}\right) p\left(\mu_k \mid G\right)$ where the prior distribution $p\left(\mu_k \mid G\right) = \mathcal{N}\left(\mu_k \mid 0, \rho \mathbf{I}\right) = \frac{1}{(2\pi)^{d/2} |\rho \mathbf{I}|^{d/2}} \exp\left\{-\frac{1}{2} \mu_k^{\mathsf{T}} (\rho I)^{-1} \mu_k\right\}$ and the data likelihood in cluster $k$ can be written as

$$\prod_{\forall i, z_i = k} p\left(x_i \mid \mu_k, \sigma \mathbf{I}\right) = \frac{1}{(2\pi)^{n_k d/2} |\sigma \mathbf{I}|^{n_k d/2}} \exp\left\{\sum_{\forall i, z_i = k} -\frac{1}{2} \left(x_i - \mu\right)^T (\sigma I)^{-1} (x_i - \mu_k)\right\}.$$

The posterior distribution for the mean $\mu_k$ is computed as a product of prior distribution and likelihood distribution:

$$\begin{aligned} p\left(\mu_k \mid \boldsymbol{x}, \boldsymbol{z}, G\right) &= \prod_{\forall i, z_i = k} \mathcal{N}\left(x_i \mid \mu_k, \sigma \mathbf{I}\right) \times \mathcal{N}\left(\mu_k \mid 0, \rho \mathbf{I}\right) \\ &= C \times \exp\left\{\sum_{\forall i, z_i = k} -\frac{1}{2\sigma} \left(x_i^{\mathsf{T}} x_i + \mu_k^{\mathsf{T}} \mu_k - 2 x_i^{\mathsf{T}} \mu_k\right) - \frac{1}{2\rho} \mu_k^{\mathsf{T}} \mu_k\right\}. \end{aligned}$$

Let ignore the constant, the term inside the exponential becomes:

$$p\left(\mu_k\right) \propto \exp\left\{\frac{-\mu_k^{\mathsf{T}} \mu_k}{2} \left(\frac{n_k}{\sigma} + \frac{1}{\rho}\right) + \left(\frac{\sum_{\forall i, z_i = k} x_i}{\sigma}\right) \mu_k - \frac{1}{2\sigma} \sum_{\forall i, z_i = k} x_i^{\mathsf{T}} x_i\right\} \quad (2.3.17)$$

where $n_k$ is the number of data points in cluster $k$. Since the posterior $p\left(\mu_k \mid \boldsymbol{\mu}_{-k}, \boldsymbol{x}, \boldsymbol{z}, \lambda\right)$

is following Gaussian distribution, we can rewrite this in the form

$$p\left(\mu_k \mid \boldsymbol{\mu}_{-k}, \boldsymbol{x}, \boldsymbol{z}, \lambda\right) = \mathcal{N}\left(\mu_k \mid \tilde{\mu}_k, \tilde{\sum}_k\right) \propto \exp\left\{-\frac{\mu_k^{\mathsf{T}}\mu_k}{2\tilde{\sigma}_k} + \frac{1}{\tilde{\sigma}_k}\tilde{\mu}_k^{\mathsf{T}}\mu_k - \frac{1}{2\tilde{\sigma}_k}\tilde{\mu}_k^{\mathsf{T}}\tilde{\mu}_k\right\}. \tag{2.3.18}$$

Matching coefficients of $\mu_k^{\mathsf{T}}\mu_k$ in Eq. 2.3.17 and Eq. 2.3.18, we compute $\tilde{\sigma}_k$ such that $\frac{1}{\tilde{\sigma}_k} = \frac{n_k}{\sigma} + \frac{1}{\rho}$. Then, we compute $\tilde{\sum}_k = \frac{\tilde{\sigma}_k\rho}{\tilde{\sigma}_k+\rho n_k}I$. Similarly, let match the coefficients of $\mu_k$ in Eq. 2.3.17 and in Eq. 2.3.18, we compute $\tilde{\mu}_k = \left(\frac{\rho n_k}{\sigma+\rho n_k}\right)\overline{x_k}$ where $\overline{x_k} = \frac{\sum_{\forall i, z_i=k} x_i}{n_k}$. Finally, we obtain the posterior distribution for mean $\mu_k$ as $p\left(\mu_k \mid \tilde{\mu}_k, \tilde{\sum}_k\right)$ that is similar to the result presented in Kulis and Jordan (2012). □

#### 2.3.2.2 Applications of DPmeans

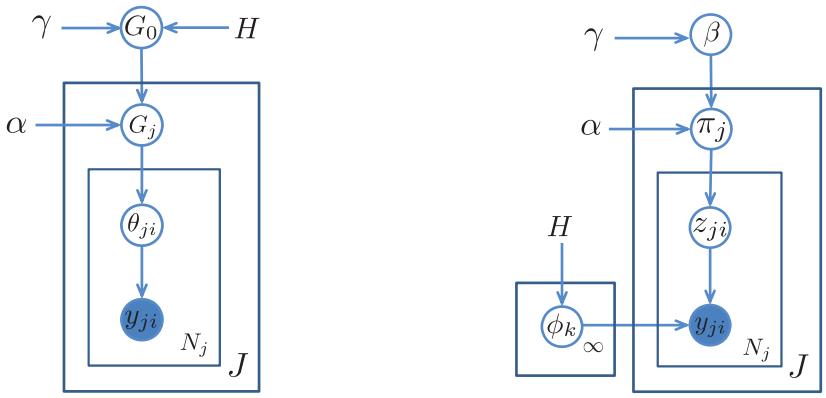
DPmeans (Kulis and Jordan, 2012) has been used for scalable data clustering while the number of cluster is automatically identified. To demonstrate the scalability, Kulis and Jordan (2012) examine DPmeans on 312,320 images patches of Photo Tourism dataset. DPmeans takes 29.4 seconds and converge in 63 iterations which is infeasible for Gibbs sampler. Axial DP-means (Cabeen and Laidlaw, 2014), an extension from DPmeans, presents an efficient approach to hard clustering of spatial and axial data that is effective for segmenting brain white matter. Cabeen and Laidlaw (2014) evaluate the Axial DP-means to diffusion tensor atlas segmentation.

For generic case of distributions (non Gaussian case), small variance derivation for exponential family is proposed in (Jiang et al., 2012). Particularly, they show that in the limit Multinomial Dirichlet distribution likelihood can be approximated by Kullback–Leibler divergence which is suitable for working with discrete-data domains. More recently, the asymptotic work of (infinite) HMM (Roychowdhury et al., 2013) and Dependent Dirichlet Process Mixture (Campbell et al., 2013) offer scalable analysis for sequential data.

Borrowing the idea of small variance asymptotic for Dirichlet Process Mixture, recent work has consider the small variance asymptotic for Pitman Yor Process Mixture

(Fan et al., 2013; Zhou et al., 2015). To achieve the hard clustering, Zhou et al. (2015) treat the Pitman-Yor exchangeable partition probability function (EPPF) as a regularizer to graph cut objectives. Because the resulting objectives cannot be solved by relaxing via eigenvectors, they derive a simple iteration algorithm to locally optimize the objectives. Moreover, Zhou et al. (2015) show that the proposed algorithm can be viewed as performing MAP inference on a Pitman-Yor mixture model.

### 2.3.3 Hierarchical Dirichlet Processes




(a) Stochastic View (b) Stick-Breaking View

Figure 2.3.7: Graphical model representation for HDP.

The Dirichlet Process (Ferguson, 1973) can also be utilised as nonparametric prior for modelling grouped data. Under this setting, each group is modelled as a Dirichlet Process Mixture model and these models are 'linked' together to reflect the dependency among them. The goal is to exploit the mutual statistical strength across groups, and at the same time provide the clustering flexibility at the group level - a formalism which is generally known as dependent Dirichlet Process (MacEachern, 1999). One particular attractive formalism is the Hierarchical Dirichlet Processes (Teh et al., 2006; Teh and Jordan, 2009) which posits the dependency among the group-level DPM by another Dirichlet Process (cf. Fig. 2.3.7).

#### 2.3.3.1 Representation of Hierarchical Dirichlet Process

Let $J$ be the number of groups and $\left\{x_{j1}, \ldots, x_{jN_j}\right\}$ be $N_j$ observations associated with the group $j$ which are assumed to be exchangeable within the group. Under HDP framework, each group $j$ is endowed with a random group-specific mixture distribution $G_j$ which is statistically connected with other mixture distributions via another Dirichlet Process sharing the same base probability measure $G_0$:

$$G_j \mid \alpha, G_0 \overset{\text{iid}}{\sim} \mathrm{DP}\left(\alpha, G_0\right), j = 1, \ldots, J \tag{2.3.19}$$

This generative process further indicates that $G_j$ (s) are exchangeable at the group level and conditionally independent given the base measure $G_0$ , which is also a random probability measure distributed according to another Dirichlet Process

$$G_0 \mid \gamma, H \sim \mathrm{DP}\left(\gamma, H\right). \tag{2.3.20}$$

It is clear from the definition of the Hierarchical Dirichlet Process that $G_j$'s ,$G_0$ and $H$ share the same support $\Theta$. Then the local atoms in group $j$ is draw as $\theta_{ji} \overset{\text{iid}}{\sim} G_j$ and the observation is generated following $x_{ji} \sim F\left(\theta_{ji}\right)$.

We present the stick-breaking representation of HDP for posterior inference which can be summarized following. We draw a global mixing weight $\beta \sim \mathrm{GEM}(\gamma)$, then generate the topics $\phi_k \overset{\text{iid}}{\sim} H(\lambda)$. The global atom $G_0$ in Eq. 2.3.20 can be characterized as $G_0 = \sum_{k=1}^{\infty} \delta_{\phi_k} \times \beta_m$. We next sample the mixing proportion for each document $j$ such that $\pi_j \overset{\text{iid}}{\sim} \mathrm{DP}\left(\alpha\beta\right)$. The local atom in each document is represented as $G_j = \sum_{k=1}^{\infty} \pi_{j,k} \times \delta_{\phi_k}$ . Finally, we draw the latent assignment $z_{ji} \overset{\text{iid}}{\sim} \mathrm{Mult}(\pi_j)$ and observation $x_{ji} \overset{\text{iid}}{\sim} F(\phi_{z_{ji}})$ accordingly.

Teh et al. (2006) present three ways of estimating posterior inference for HDP. The first and second ways of Gibbs sampler is built upon the Chinese Restaurant Franchise metaphor. The remaining one is from direct assignment scheme. We will describe the first and the third approaches of posterior inference for HDP below.

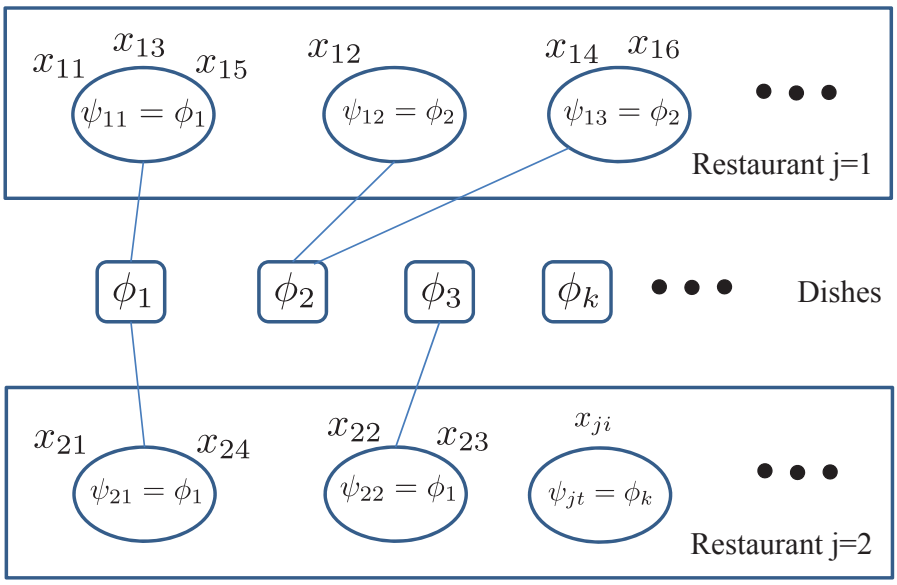



Figure 2.3.8: Chinese Restaurant Franchise metaphor. Customers (words) $x_{ji}$ are organised into restaurants (documents) (e.g., customer $x_{11}$ is a member of restaurant $j = 1$). There are infinite (but countably) number of table in a restaurant. Each table will take a dish (e.g., table $\psi_{11}$ in restaurant $j = 1$ takes dish $\phi_1$).

#### 2.3.3.2 Gibbs sampler via Chinese Restaurant Franchise metaphor

We next to describe the posterior sampling using the Chinese Restaurant Franchise (CRF) metaphor (Teh, 2006) serving as the main machinery for developing Gibbs sampler for HDP.

**Chinese Restaurant Franchise.** We have the following notations: $\theta_{ji}$: customer $i$-th in restaurant $j$-th. $t_{ji}$: table of customer $\theta_{ji}$. $\psi_{jt}$: table $t$-th in restaurant $j$-th. $k_{jt}$: table $t$-th in restaurant $j$-th served dish $k$-th. $\phi_k$: global dish.

We use the notation $n_{jtk}$ to denote the number of customers in restaurant $j$ at table $t$ eating dish $k$. Marginal counts are represented with dots. Thus, $n_{jt*}$ represents the number of customers in restaurant $j$ at table $t$ and $n_{j*k}$ represents the number of customers in restaurant $j$ eating dish $k$. The notation $m_{jk}$ denotes the number of tables in restaurant $j$ serving dish $k$. Thus, $m_{j*}$ represents the number of tables in restaurant $j$, $m_{*k}$ represents the number of tables serving dish $k$, and $m_{**}$ the total number of tables occupied.

**Sampling $t_{ji}$.** We assign customer $i$ in restaurant $j$ to table $t$

$$p\left(t_{ji}=t \mid \mathbf{t}^{-ji}, \mathbf{k}\right) \propto \begin{cases} \frac{n_{jt.}^{-ji}}{n_{j**}^{-ji}+\alpha_0} \times f_{k_{jt}}^{-x_{ji}}\left(x_{ji}\right) & \text{used } t \\ \frac{\alpha_0}{n_{j**}^{-ji}+\alpha_0} \times p\left(x_{ji} \mid \mathbf{t}^{-ji}, t_{ji}=t^{new}, \mathbf{k}\right) & \text{new } t \end{cases} \tag{2.3.21}$$

where the likelihood function for new table is computed as

$$p\left(x_{ji} \mid \mathbf{t}^{-ji}, t_{ji}=t^{new}, \mathbf{k}\right)=\sum_{k=1}^{K} \frac{m_{.k}}{m_{**}+\gamma} f_k^{-x_{ji}}\left(x_{ji}\right)+\frac{\gamma}{m_{**}+\gamma} f_{k^{new}}^{-x_{ji}}\left(x_{ji}\right).$$

If $t=t^{new}$ in Eq. 2.3.21, we further sampling $t^{new} \in \{1, ...k, k^{new}\}$

$$p\left(k_{jt^{new}}=k \mid \mathbf{t}, \mathbf{k}^{-jt^{new}}\right) \propto \begin{cases} m_{*k} \times f_k^{-x_{ji}}\left(x_{ji}\right) & \text{used } k \\ \gamma \times f_{k^{new}}^{-x_{ji}}\left(x_{ji}\right) & \text{new } k. \end{cases} \tag{2.3.22}$$

Thus, sampling $t$ will give three possible outcomes:

$$p\left(t_{ji}=t \mid .\right)=\begin{cases} \frac{n_{jt.}^{-ji}}{n_{j**}^{-ji}+\alpha_0} \times f_{k_{jt}}^{-x_{ji}}\left(x_{ji}\right) & \text{used } t \\ \frac{\alpha_0}{n_{j**}^{-ji}+\alpha_0} \times \frac{m_{.k}^{-ji}}{m_{**}+\gamma} \times f_k^{-x_{ji}}\left(x_{ji}\right) & \text{new } t, \text{used } k \\ \frac{\alpha_0}{n_{j**}^{-ji}+\alpha_0} \times \frac{\gamma}{m_{**}+\gamma} \times f_{k^{new}}\left(x_{ji}\right) & \text{new } t, \text{new } k. \end{cases} \tag{2.3.23}$$

**Sampling $k$.** We assign a table $t$ in restaurant $j$ to a dish $k$. Then, we sample the table assignment $k_{jt}$ in restaurant $j$ to global dish which is sharing across restaurant.

$$p\left(k_{jt}=k \mid \mathbf{t}, \mathbf{k}^{-jt}\right) \propto \begin{cases} m_{.k}^{-jt} \times f_k^{-\mathbf{x}_{jt}}\left(\mathbf{x}_{jt}\right) & k \\ \gamma \times f_{k^{new}}^{-\mathbf{x}_{jt}}\left(\mathbf{x}_{jt}\right) & k^{new}. \end{cases} \tag{2.3.24}$$

We have presented the first approach for sampling HDP using Chinese Restaurant Franchise, we next present the third approach for sampling HDP using collapsed Gibbs sampler via direct sampling scheme.

#### 2.3.3.3 Collapsed Gibbs sampling via direct sampling scheme

For collapsed Gibbs sampling (Liu, 1994), we would integrate out $\pi_j$ and $\phi_k$ due to conjugacy property. Thus, there are two latent variables $z_{ji}$ and $\beta$ that we need to sample (cf. Fig. 2.3.7).

**Sampling $z_{ji}$.** We assign a data point $x_{ji}$ to its component $\phi_k$. The conditional distribution for $z_{ji}$ is influenced by a collection of words associated with topic $k$ across documents:

$$\begin{aligned} p\left(z_{ji}=k \mid \boldsymbol{x}, \mathbf{z}_{-ji}, \alpha, \beta, H\right) =& p\left(z_{ji}=k \mid \boldsymbol{z}_{-j}, \alpha, \beta\right) \\ & \times p\left(x_{ji} \mid z_{ji}=k, \left\{x_{j'i'} \mid z_{j'i'}=k, \forall\left(j'i' \neq ji\right)\right\}, H\right) \\ =& \begin{cases} \left(n_{jk}^{-ji}+\alpha\beta_k\right) \times f_k^{-x_{ji}}\left(x_{ji}\right) & \text{used } k \\ \alpha \times \beta_{\text{new}} \times f_{k^{\text{new}}}^{-x_{ji}}\left(x_{ji}\right) & \text{new } k. \end{cases} \end{aligned}$$

The first term can be recognized as the Chinese Restaurant Franchise, such as the number of data points in group $j$ follows topic $k$. The second term is the predictive likelihood $f_k^{-x_{ji}}\left(x_{ji}\right)$.

**Sampling $\boldsymbol{\beta}$.** We sample the global mixing weight. We have the posterior distribution for $\boldsymbol{\beta}$:

$$p\left(\boldsymbol{\beta} \mid \boldsymbol{z}, \gamma, \alpha\right) \propto p\left(\boldsymbol{z} \mid \boldsymbol{\beta}, \alpha, \gamma\right) p\left(\boldsymbol{\beta} \mid \gamma\right).$$

Integrating out $\boldsymbol{\pi}_j$ using the conjugacy property of Multinomial-Dirichlet and recall that $\boldsymbol{\pi}_j \sim \text{Dir}\left(\alpha\beta_1, \ldots, \alpha\beta_K\right)$ and $\sum_{k=1}^{K} \beta_k = 1$ the first term becomes

$$\begin{aligned} p\left(\boldsymbol{z} \mid .\right) &= \prod_{j=1}^{J}\left[p\left(\boldsymbol{z}_j \mid \boldsymbol{\beta}_{1:K}, \alpha, \gamma\right)\right] = \prod_{j=1}^{J} \int_{\boldsymbol{\pi}_j} p\left(\boldsymbol{z}_j \mid \boldsymbol{\pi}_j\right) p\left(\boldsymbol{\pi}_j \mid \alpha\beta_1, \ldots, \alpha\beta_K\right) d\boldsymbol{\pi}_j \\ &= \prod_{j=1}^{J} \frac{\Gamma\left(\alpha\right)}{\Gamma\left(\alpha+N_j\right)} \prod_{k=1}^{K} \frac{\Gamma\left(\alpha\beta_k+n_{jk}\right)}{\Gamma\left(\alpha\beta_k\right)}. \end{aligned}$$

For the second term, note that $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_K, \beta_{\text{new}}) \sim \text{Dir}\left(\frac{\gamma}{L}, \ldots, \frac{\gamma}{L}, \frac{L-K}{L}\gamma\right)$, let $\gamma_r = \frac{\gamma}{L}$ and $\gamma_{\text{new}} = \frac{L-K}{L}\gamma$ then

$$p\left(\boldsymbol{\beta} \mid \gamma\right) = \frac{\Gamma\left(\overbrace{K\gamma_r + \gamma_{\text{new}}}^{\gamma}\right)}{\left[\Gamma\left(\gamma_r\right)\right]^K \Gamma\left(\gamma_{\text{new}}\right)} \left(\prod_{k=1}^{K} \beta_k^{\gamma_r - 1}\right) \beta_{\text{new}}^{\gamma_{\text{new}} - 1}.$$

Put them together, we get:

$$p\left(\boldsymbol{\beta}, \boldsymbol{z} \mid \gamma, \alpha\right) = \underbrace{\frac{\Gamma\left(\gamma\right)}{\left[\Gamma\left(\gamma_r\right)\right]^K \Gamma\left(\gamma_{\text{new}}\right)} \beta_{\text{new}}^{\gamma_{\text{new}} - 1} \prod_{j=1}^{J} \frac{\Gamma\left(\alpha\right)}{\Gamma\left(\alpha + N_j\right)} \prod_{k=1}^{K} \beta_k^{\gamma_r - 1}}_{\text{(A)}} \frac{\Gamma\left(\alpha\beta_k + n_{jk}\right)}{\Gamma\left(\alpha\beta_k\right)}.$$

Using the results from Teh et al. (2006), let $\boldsymbol{m} = (m_{jk} : \text{ for all } j \text{ and } k)$ and $\text{Stirl}\left(n, k\right)$ is the Stirling number of the second kind, we have:

$$\frac{\Gamma\left(\alpha\beta_k + n_{jk}\right)}{\Gamma\left(\alpha\beta_k\right)} = \sum_{m_{jk}=0}^{n_{jk}} \text{Stirl}\left(n_{ij}, m_{jk}\right) \left(\alpha\beta_k\right)^{m_{jk}}$$

$$p\left(\boldsymbol{\beta}, \boldsymbol{z} \mid \gamma, \alpha\right) = A \times \sum_{m_{jk}=0}^{n_{jk}} \text{Stirl}\left(n_{ij}, m_{jk}\right) \left(\alpha\beta_k\right)^{m_{jk}}.$$

Dropping the summation over $m_{jk}$, it is easy to see that

$$\sum_{\boldsymbol{m}} A \times \text{Stirl}\left(n_{ij}, m_{jk}\right) \left(\alpha\beta_k\right)^{m_{jk}} = p\left(\boldsymbol{\beta}, \boldsymbol{z} \mid \gamma, \alpha\right).$$

This defines a joint distribution over $\boldsymbol{\beta}, \boldsymbol{z}, \boldsymbol{m}$:

$$\begin{aligned} p\left(\boldsymbol{\beta}, \boldsymbol{z}, \boldsymbol{m}\right) &= A \times \text{Stirl}\left(n_{ij}, m_{jk}\right) \left(\alpha\beta_k\right)^{m_{jk}} \\ &= \frac{\Gamma\left(\gamma\right) \times \beta_{\text{new}}^{\gamma_{\text{new}} - 1}}{\Gamma\left(\gamma_r\right)^K \Gamma\left(\gamma_{\text{new}}\right)} \prod_{j=1}^{J} \frac{\Gamma\left(\alpha\right)}{\Gamma\left(\alpha + N_j\right)} \prod_{k=1}^{K} \beta_k^{\gamma_r - 1} \text{Stirl}\left(n_{ij}, m_{jk}\right) \left(\alpha\beta_k\right)^{m_{jk}}. \end{aligned}$$

We sample $\boldsymbol{\beta}$ jointly with the auxiliary variable $\boldsymbol{m}$:

$$p\left(m_{jk} = m \mid \boldsymbol{z}, \boldsymbol{m}_{-jk}, \boldsymbol{\beta}\right) \propto \text{Stirl}\left(n_{ij}, m_{jk}\right) \left(\alpha\beta_k\right)^{m} \quad 0 \leq m \leq n_{jk}$$

$$p\left(\boldsymbol{\beta} \mid \boldsymbol{m}, \boldsymbol{z}, \alpha, \gamma\right) \propto \beta_{\text{new}}^{\gamma_{\text{new}} - 1} \prod_{k=1}^{K} \beta_k^{\sum_j m_{jk} + \gamma_r - 1} \overset{\infty}{=} \beta_{\text{new}}^{\gamma - 1} \prod_{k=1}^{K} \beta_k^{\sum_j m_{jk} - 1} (\text{as } L \to \infty)$$

where we note that $\gamma_{\text{new}} = \left(\frac{L-K}{L}\right)\gamma \to \gamma$ and $\gamma_r = \frac{K}{L} \to 0$ when $L \to \infty$.

**Sampling hyperparameters $\alpha$ and $\gamma$.** To make the model robust in identifying the unknown number of clusters, we resample hyperparameters in each Gibbs iteration. The lower concentration parameter $\alpha$ is described in (Teh et al., 2006). The upper concentration parameter $\gamma$ is followed the techniques of (Escobar and West, 1995).

#### 2.3.3.4 Applications

HDP (Teh et al., 2006) is applied for nonparametric text modelling. Liang et al. (2007); Finkel et al. (2007) use Hierarchical Dirichlet Processes in the field of natural language processing to detect how many grammar symbols exist in a particular set of sentences. HDP can also be applied in health care for modelling of coded clinical data Luo et al. (2014).Another extension of HDP as Dynamic Hierarchical Dirichlet Processes (Ren et al., 2008) is for modelling time series documents. In addition, Zhang et al. (2010) formulates Evolutionary HDP for multiple correlated time-varying corpora by adding time dependencies to the adjacent epochs. In Evo-HDP, each HDP is built for multiple corpora at each time epoch, and the time dependencies are incorporated into adjacent epochs under the Markovian assumption. Specifically, the dependency is formulated by mixing two distinct Dirichlet processes (DPs). One is the DP model for the previous epoch, and the other is an updating DP model. To infer the EvoHDP model, Zhang et al. (2010) propose a cascaded Gibbs sampling scheme for model inference.

HDP can be constructed to place a Dirichlet prior over number of state in Hidden Markov model (Rabiner and Juang, 1986) results in Hierarchical Dirichlet Processes Hidden Markov model (HDP-HMM) (Teh et al., 2006). In Section 2.3.5, we will go into details of the HDP-HMM model for sequential modelling.

### 2.3.4 HDPmeans

In this section, we derive the asymptotic limit for the Hierarchical Dirichlet Process (Teh et al., 2006) to obtain the HDPmeans algorithm. We aim to provide the full details for non-specialist can follow that has not been described thoroughly in (Kulis and Jordan, 2012). We start the HDPmeans algorithm from asymptotic analysis using Gibbs sampling for the Chinese Restaurant Franchise (CRF) metaphor. Then, we derive the objective function of HDPmeans algorithm which is similar to (Kulis and Jordan, 2012).

The HDPmeans algorithm can be extended the asymptotic argument that we employ for the DPmeans. The derivation is analogous to the derivation for the single DP mixture model. We will have a threshold that determines when to introduce a new cluster. Specifically, for HDPmeans, we need to have two parameters $\lambda_l$ as the local threshold parameter and $\lambda_g$ as the global threshold parameter.

The posterior sampling for HDP using CRF is described in Section 2.3.3.2 where we need to sample the table assignment $t_{ji}$ for each customer $i$ in the restaurant $j$ and sample table assignment $k_{jt}$ in restaurant $j$ to global dish $k$.

#### 2.3.4.1 Small variance asymptotic analysis for HDP

**Sampling $t$.** For ease of interpretation, we can assume $f$ follow Gaussian distribution as used in DPMeans (other distributions are easily accommodated with their Bregman divergence). Note that only the term in $f$ will affect the final result that will dominate the count terms (e.g., $n_{jt.}^{-ji}$). Denote $\alpha = \exp\left(-\frac{\lambda_l}{2\sigma}\right)$ and $\beta = \left(\frac{\sigma}{\sigma+\rho}\right)^{-d/2} \exp\left(-\frac{\lambda_g}{2\sigma}\right)$, when $\sigma \to 0$ with a fixed $\rho$, the Eq. 2.3.23 becomes:

$$\lim_{\sigma\to 0} p\left(t_{ji} \mid .\right) = \hat{\gamma}\left(t_{ji}\right) = \begin{cases} ||x_{ji} - \mu_t||^2 & \text{used } t\text{, used } k \\ \lambda_l + ||x_{ji} - \mu_k||^2 & \text{new } t\text{, used } k \\ \lambda_l + \lambda_g & \text{new } t\text{, new } k. \end{cases} \tag{2.3.25}$$

*Proof.* From Eq. 2.3.23, we only consider terms that would not be constants after

we do the asymptotic analysis:

$$A = \frac{n_{jt*}^{-ji}}{n_{j**}^{-ji} + \alpha_0} f_{k_{jt}}^{-x_{ji}}(x_{ji}) \propto n_{jt*}^{-ji} \times f_{k_{jt}}^{-x_{ji}}(x_{ji})$$
$$B = \frac{\alpha_0}{n_{j**}^{-ji} + \alpha_0} \times \frac{m_{.k}^{-ji}}{m_{**} + \gamma} \times f_k^{-x_{ji}}(x_{ji}) \propto \alpha_0 \times m_{.k}^{-ji} \times f_k^{-x_{ji}}(x_{ji})$$
$$C = \frac{\alpha_0}{n_{j**}^{-ji} + \alpha_0} \times \frac{\gamma}{m_{**} + \gamma} \times f_{k^{new}}(x_{ji}) \propto \alpha_0 \times \gamma \times f_{k^{new}}(x_{ji}).$$

Substitute $\alpha_0 = \exp\left(-\frac{\lambda_l}{2\sigma}\right)$ and $\gamma = \left(\frac{\sigma}{\sigma+\rho}\right)^{-d/2} \exp\left(-\frac{\lambda_g}{2\sigma}\right)$ into $B$ and $C$ (using the same strategy for DPmeans in Section 2.3.2), we have the following:

$$A = n_{jt*}^{-ji} (2\pi\sigma)^{-d/2} \exp\left(-\frac{1}{2\sigma}||x_{ji} - \mu_{k_{jt}}||^2\right)$$
$$B = m_{*k}^{-ji} (2\pi\sigma)^{-d/2} \exp\left(-\frac{1}{2\sigma}\left[||x_{ji} - \mu_k||^2 + \lambda_l\right]\right)$$
$$C = (2\pi\sigma)^{-d/2} \exp\left(-\frac{1}{2\sigma}\left[\lambda_l + \lambda_g + \frac{\sigma}{\rho+\sigma}||x_i||^2\right]\right).$$

Similar to DPMeans case presented in Section 2.3.2, in the limit of $\sigma \to 0$ only the smallest of these values $\left\{ \underbrace{||x_{ji} - \mu_{k_{jt}}||^2, ...}_{\text{used } t}, \underbrace{||x_{ji} - \mu_k||^2 + \lambda_l, ...}_{\text{new } t,\text{used } m}, \underbrace{\lambda_l + \lambda_g}_{\text{new } t,\text{new } m} \right\}$ will receive a non-zero value (the remaining terms are zero). □

**Sampling $k$.** Similarly, we have the same derivation for assigning table $t$ in restaurant $j$ to dish $k$ in the limit as

$$\lim_{\sigma\to 0} p(k_{jt} \mid .) = \hat{\gamma}(k_{jt}) \propto \begin{cases} ||\mathbf{x}_{jt} - \mu_k||^2 & \text{used } k \\ \lambda_g & \text{new } k. \end{cases} \tag{2.3.26}$$

The proof is similar to the above case of sampling table indicator $t_{ji}$ by substituting $\gamma = \left(\frac{\sigma}{\sigma+\rho}\right)^{-d/2} \exp\left(-\frac{\lambda_g}{2\sigma}\right)$ into Eq. 2.3.24.

#### 2.3.4.2 Algorithm for HDP hard clustering

We present the high-level algorithm (see Algorithm 2.5) to learn the hard clustering for the asymptotic HDP that is similar to the Algorithm 2 in (Kulis and Jordan, 2012). The high-level algorithm is working as follows: for each customer $x_{ji}$, we compute the distance to every global dish $\mu_k$ as in Eq. 2.3.25, then we assign this customer into the appropriate table or create a new table. Next, we will compute the distance between the local table $t$ in restaurant $j$ to a global dish $k$ using Eq. 2.3.26 and assign it to a used or a new dish.

**Algorithm 2.5** High-level algorithm for Asymptotic HDP

**Input**: $\{x_{ji}\}$: input data; $\lambda_l$: local dish penalty; $\lambda_g$: global dish penalty

1: Initialization
2: Repeat steps 3-5 until convergence
3: **for** each restaurant $j = 1, ..., J$ **do**
4: **for** each customer $i = 1, ..., N_j$ **do**
5: Assigning customer-table indicator $t_{ji}$ and update $K$ (Eq. 2.3.25)
6: **end for**
7: **end for**
8: **for** each restaurant $j = 1, ...J$ **do**
9: Assigning local dish $k_{jt}$ to global dish (Eq. 2.3.26)
10: **end for**
11: Updating global dish $\psi_1, \psi_2, ..., \psi_M$

**Output**: Global dishes $\psi_{1...M}$, and number of cluster $k_j$ for all restaurant.

#### 2.3.4.3 Objective function for HDPmeans

In this section, we derive the objective function for HDPmeans. We compute the joint likelihood of

$$p\left(z, x \mid \alpha, \gamma, \lambda\right) = p\left(z \mid \alpha, \gamma\right) + p\left(x \mid z, \lambda\right). \tag{2.3.27}$$

- The first term is calculated by integrating out stick breaking $\beta$ and mixture

proportion $\pi_j$.

$$p(z \mid \gamma, \alpha) = \int_\beta p(\beta \mid \gamma) \times \prod_{j=1}^{J} \int_{\pi_j} p(z_j \mid \pi_j) \times p(\pi_j \mid \alpha\beta_1, ..., \alpha\beta_K) \, d\pi_j d\beta$$
$$= \int_\beta p(\beta \mid \gamma) \times \prod_{j=1}^{J} \frac{\Gamma(\alpha)}{\Gamma(\alpha + N_j)} \prod_{k=1}^{K} \frac{\Gamma(\alpha\beta_k + n_{jk})}{\Gamma(\alpha\beta_k)} d\beta. \quad (2.3.28)$$

The probability of $p(\beta \mid \gamma)$ can be expressed by the EPPF (Pitman, 1995; Broderick et al., 2013b) $p(\beta \mid \gamma) = \gamma^{K-1} \frac{\Gamma(\gamma+1)}{\Gamma(\gamma+m_{**})} \prod_{k=1}^{K} \Gamma(m_{*k})$ where $m_{jk}$ is the number of table taking dish $k$-th in restaurant $j$-th. We express the fraction $\frac{\Gamma(\alpha\beta_k+n_{jk})}{\Gamma(\alpha\beta_k)} = \text{Stirl}(n_{jk}, m_{jk}) \times (\alpha\beta_k)^{m_{jk}}$. Put them together, we have the Eq. 2.3.28 becomes the joint likelihood of $z$ and $m$:

$$p(z, m) = \int_\beta \gamma^{K-1} \frac{\Gamma(\gamma + 1)}{\Gamma(\gamma + m_{**})} \prod_{k=1}^{K} \Gamma(m_{*k})$$
$$\times \prod_{j=1}^{J} \frac{\Gamma(\alpha)}{\Gamma(\alpha + N_j)} \prod_{k=1}^{K} \text{Stirl}(n_{jk}, m_{jk}) (\alpha\beta_k)^{m_{jk}} \, d\beta.$$

Take integration of Dirichlet distribution and get log of the likelihood, we obtain:

$$\log p(z, m) = \log \frac{\Gamma(\gamma + 1)}{\Gamma(\gamma + m_{**})} + \sum_{k=1}^{K} \log \Gamma(m_{*k}) + \log \left[ \frac{\prod_{k=1}^{K} \Gamma(m_{*k} + 1)}{\Gamma(m_{**} + 1)} \right] \quad (2.3.29)$$
$$+(K-1) \log \gamma + \sum_{j=1}^{J} \left[ \log \frac{\Gamma(\alpha)}{\Gamma(\alpha + N_j)} + \sum_{k=1}^{K} \text{Stirl}(n_{jk}, m_{jk}) + m_{j*} \log \alpha \right].$$

- The second term is computed as the likelihood of observation $x$:

$$p(x \mid z, \lambda) = \prod_{j=1}^{J} \prod_{i=1}^{N_j} p\left(x_{ji} \mid \psi_{z_{ji}}\right) \times p(\psi \mid \lambda). \quad (2.3.30)$$

We replace $p\left(x_{ji} \mid \psi_{z_{ji}}\right) = \mathcal{N}\left(x_{ji} \mid \mu_{z_{ji}}, \sigma^2 I_d\right)$ and $p(\psi \mid \lambda) = \prod_{k=1}^{K} \mathcal{N}(\mu_k \mid 0, \rho^2 I_d)$ as in (Kulis and Jordan, 2012) into Eq. 2.3.30. In order to retain the impact of hyperparameters $\alpha$ and $\gamma$ in the limit, we can define some constants $\lambda_l$,$\lambda_g > 0$ such

that $\alpha = \exp\left(-\frac{\lambda_l}{2\sigma^2}\right)$, and $\gamma = \exp\left(-\frac{\lambda_g}{2\sigma^2}\right)$ . The Eq. 2.3.27 becomes:

$$\begin{aligned}\log p(z,x) = \exp\left[-\frac{1}{2\sigma^2}\left\{\sum_{j=1}^{J}\sum_{i=1}^{N_j}||x_{ji}-\mu_{z_{ji}}||^2 + \lambda_l\sum_{j=1}^{J}(m_{j*}) + \lambda_g(K-1)\right\}\right] \\ + \sum_{j=1}^{J}\left[\log\frac{\Gamma(\alpha)}{\Gamma(\alpha+N_j)} + \sum_{k=1}^{K}\mathrm{Stirl}(n_{jk},m_{jk})\right] + \log\frac{\Gamma(\gamma+1)}{\Gamma(\gamma+m_{**})} \\ + \sum_{k=1}^{K}\log\Gamma(m_{*k}) + \log\left[\frac{\prod_{k=1}^{K}\Gamma(m_{*k}+1)}{\Gamma(m_{**}+1)}\right].\end{aligned}$$

As we let $\sigma^2 \to 0$, we consider the terms which are remaining in the limit

$$\lim_{\sigma\to 0} -2\sigma^2 p(z,x) = \sum_{j=1}^{J}\sum_{i=1}^{N_j}||x_{ji}-\mu_{z_{ji}}||^2 + \lambda_l\sum_{j=1}^{J}(m_{j*}) + \lambda_g(K-1).$$

We result in the objective function for maximizing the joint probability in HDP-means

$$\min_{z_{ji}}\sum_{j=1}^{J}\sum_{i=1}^{N_j}||x_{ji}-\mu_{z_{ji}}||^2 + \lambda_l\sum_{j=1}^{J}(m_{j*}) + \lambda_g(K-1) \tag{2.3.31}$$

such that $K \geq z_{ji} \geq 1$ and $\sum_{j=1}^{J}\sum_{i=1}^{N_j}\delta(z_{ji},k) \geq 1$ where $\delta$ is a Dirac delta function.

This constraint can be intuitively understood as (1) the value of $z_{ji}$ should be less than the total number of topic $K$ and (2) for each topic (or dish), there is at least one customer. The objective function in Eq. 2.3.31 has a kmeans-like objective form with penalized terms as used in (Kulis and Jordan, 2012) which has been proved to monotonically decrease to a local convergence. We have two penalty parameters: $\lambda_l$ for controlling the number of tables used in a restaurant and $\lambda_g$ for controlling the number of global dishes.

### 2.3.5 Infinite Hidden Markov model

In this section, we present the Infinite Hidden Markov model (IHMM) (Teh et al., 2006), namely a Hierarchical Dirichlet Process Hidden Markov model (HDP-HMM)

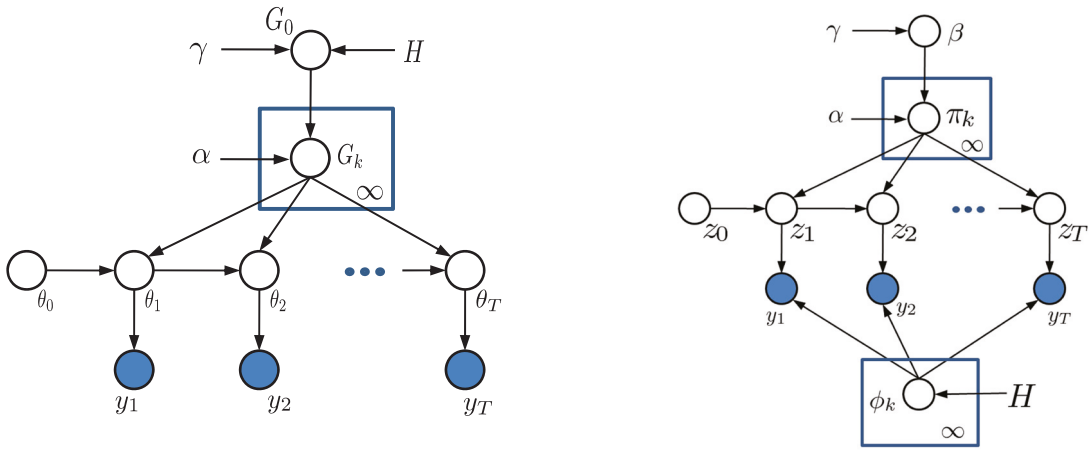



Figure 2.3.9: The infinite Hidden Markov model representation. Left: Stochastic process of iHMM. Right: Stick-breaking representation of the data.

which provides an alternative method to place a Dirichlet prior over the number of state. In Chapter 3, we will utilise IHMM to segment the video surveillance for abnormal detection. Therefore, the (unknown) number of states in HMM is identified in the same way as HDP.

#### 2.3.5.1 Model representation for HDP-HMM

Using HDP (Teh et al., 2006) as a nonparametric prior for building block, the stochastic process of HDP-HMM is described as follows. The global atom $G_0$ is drawn from Dirichlet Process $G_0 \sim \mathrm{DP}(\gamma, H \times S)$. Then, for each topic $k$, we generate a topic-specific atom $G_k$ by another Dirichlet Process from the global atom, as in HDP (Teh et al., 2006), $G_k \overset{\text{iid}}{\sim} \mathrm{DP}(\alpha, G_0)$ $k = 1, 2, ...\infty$. Next, for each state $t = 1, 2, ...T$, we randomly sample local atoms $\theta_t \overset{\text{iid}}{\sim} G_k$ and the data observation $y_t \sim F(\theta_t)$ accordingly.

The stick-breaking of HDP-HMM is illustrated in Fig. 2.3.9 in which the parameters have the following distributions:

$$\begin{aligned} \boldsymbol{\beta} &\sim \mathrm{GEM}(\gamma) \\ \phi_k &\sim H \quad k = 1, 2, ...\infty \end{aligned}$$

$$\begin{aligned} \boldsymbol{\pi}_k &\sim \mathrm{DP}(\alpha, \boldsymbol{\beta}) \\ z_t &\sim \pi_{z_{t-1}} \quad t = 1, 2, ..., T \\ y_t &\sim F(\phi_{z_t}). \end{aligned}$$

#### 2.3.5.2 Model inference

In IHMM, the use of Markov model ensures that the temporal dynamics nature of the data is taken into consideration and the number of coherent topics will be automatically identified due to the property of Bayesian nonparametric. Our data observations correspond the observed variables $\{y_t\}_{t=1}^{T}$, and $\{z_t\}_{t=1}^{T}$ plays the role the latent state variables as in a standard HMM. $H$ is the base measure from which parameters $\{\phi_k\}_{k=1}^{\infty}$ will be sampled from. For example, if we model $y_t$ as a univariate Gaussian and thus each $\phi_k$ is a tuple of $\{\mu_k, \sigma_k^2\}$ where both $\mu_k$ and $\sigma_k^2$ are unknown and treated as random variables. We use $H$ as a conjugate prior, and thus $H$ in our case is a Gaussian-invGamma distribution. A graphical model representation is shown in Fig. 2.3.9.

We use collapsed Gibbs inference for iHMM as described in (Van Gael et al., 2008) in which the latent state $z_t$ and the stick-breaking weight $\beta_k$ are sequentially sampled by explicitly integrating out parameters $\{\phi_k\}$ for the emission probability and $\{\pi_k\}$ for the transition probability. For example, given $z_{t-1} = i, z_{t+1} = j$ from the previous iteration, the conditional Gibbs distribution to sample $z_t$ has the form below.

- Sampling $z_t$. Consider the conditional probability of $z_t$

$$p\left(z_t = k \mid z_{-t}, \boldsymbol{y}, \boldsymbol{\beta}, H\right) \propto \underbrace{p\left(y_t \mid z_t = k, \boldsymbol{z}_{-t}, \boldsymbol{y}_{-t}, H\right)}_{\text{observation likelihood}} \times \underbrace{p\left(z_t = k \mid \boldsymbol{z}_{-t}, \alpha, \boldsymbol{\beta}\right)}_{\text{CRP of transition}}.$$

The first term is the likelihood of the observation $y_t$ given the component $\phi_k$. In other words, this likelihood can be expressed as $\int_{\phi_k} p\left(y_t \mid z_t = k, \phi_k\right) p\left(\phi_k \mid \boldsymbol{y}_{-t}, \boldsymbol{z}_{-t}, H\right) d\phi_k$ which is easily analysed using the conjugate property, described in Section 2.1.5.2. The second probability is simply the Chinese Restaurant Process of transition. Denote $n_{ij}$ as the number of transitions from state $i$ to state $j$, $n_{*j}$ as the number of all transitions to state $j$, and $n_{i*}$ is the number of all transitions departing from state $i$, the CRP likelihood under Markov property can be analysed as:

$$p\left(z_t = k \mid \boldsymbol{z}_{-t}, \alpha, \boldsymbol{\beta}\right) \propto \underbrace{p\left(z_t = k \mid z_{t-1}, \alpha, \boldsymbol{\beta}\right)}_{\text{from previous state t-1 to state t}} \times \underbrace{p\left(z_t = k \mid z_{t+1}, \alpha, \boldsymbol{\beta}\right)}_{\text{from state t to next state t+1}}.$$

We then have four cases to compute this probability:

$$p\left(z_t = k \mid \boldsymbol{z}_{-t}, \alpha, \boldsymbol{\beta}\right) \propto \begin{cases} \left(n_{z_{t-1},k} + \alpha\beta_k\right) \frac{n_{k,z_{t+1}} + \alpha\beta_{z_{t+1}}}{n_{k*}+\alpha} & k \leq K, k \neq z_{t-1} \\ \left(n_{z_{t-1},k} + \alpha\beta_k\right) \frac{n_{k,z_{t+1}} + 1 + \alpha\beta_{z_{t+1}}}{n_{k*}+1+\alpha} & z_{t-1} = k = z_{t+1} \\ \left(n_{z_{t-1},k} + \alpha\beta_k\right) \frac{n_{k,z_{t+1}} + \alpha\beta_{z_{t+1}}}{n_{k*}+1+\alpha} & z_{t-1} = k \neq z_{t+1} \\ \alpha\beta_{\text{new}}\beta_{z_{t+1}} & k = K+1. \end{cases}$$

- Sampling stick-breaking $\boldsymbol{\beta}$, and hyperparameters $\alpha$,$\gamma$ are exactly the same as for HDP describing in Teh et al. (2006).

For robustness we also let the concentration hyper-parameters $\alpha$ and $\gamma$ follow Gamma distributions and they will also be re-sampled at each Gibbs iteration.

#### 2.3.5.3 Applications of Infinite Hidden Markov model

A number of extensions to the IHMM have been proposed recently, including application of the HDP-HMM to speech (Goldwater et al., 2006), the problem of segmenting an audio stream into a sequence of words. Speech is surprisingly continuous with few obvious breaks between words and the problem of word segmentation that of identifying coherent segments of words and their boundaries in continuous speech is nontrivial. Goldwater et al. (2006) propose a statistical approach to word segmentation based upon the HDP-HMM where the latent states of the HMM correspond to words.

Another extension of IHMM includes the tempered HDP-HMM that exhibits a congurable bias for self-transitions in the hidden states (Fox et al., 2008). Tempered HDP-HMM is a hierarchical model that uses an IHMM to identify system sub-regimes that are modeled by Kalman filters (Fox et al., 2007), and a model that shares a library of hidden states across a collection of IHMMs that model separate processes (Ni et al., 2007).

IHMM has been demonstrated as the state of the art method for speaker diarization (Fox et al., 2008). The problem of speaker diarization is of segmenting the audio

recording into time intervals associated with individual speakers (Wooters and Huijbregts, 2008). Posterior inference in the IHMM (or HDP-HMM) yields estimates of the spectral content of each speaker's voice, an estimation of the number of speakers participating in the meeting, and a diarization of the audio stream.

The block diagonal infinite Hidden Markov model (Stepleton et al., 2009) presents a generalization of this framework that introduces nearly block-diagonal structure in the transitions between the hidden states. In identifying such structure, the model classifies, or partitions, data sequence according to these sub-behaviors in an unsupervised way. Stepleton et al. (2009) present applications of this model to video gesture classification task, and a musical theme labeling task.

We later extend IHMM to segment video for abnormality detection (Nguyen et al., 2012b, 2013a, 2015a) in Chapter 3.

### 2.3.6 Nested Dirichlet Processes

Another way of using DP for modelling multilevel data is to construct random measure in a nested structure in which the DP base measure is itself another DP. This formalism is the Nested Dirichlet Process (NDP) (Rodriguez et al., 2008). On one hand, HDP (Teh et al., 2006) concentrates on exploiting the statistical strength across group via sharing atoms $\phi_k$(s), but it does not partition groups into clusters. On the other hand, NDP (Rodriguez et al., 2008) focuses on inferring the clusters of observations and groups of partitions. The original NDP(Rodriguez et al., 2008) does not force sharing property among atoms, but it can be accommodated by introducing a DP prior for the NDP base measure as in Chapter 4 (Nguyen et al., 2014; Phung et al., 2012). In this section, we will shortly present the original version of NDP (Rodriguez et al., 2008).

The Nested Dirichlet Process (Rodriguez et al., 2008) can specifically present as $G_j \overset{\text{iid}}{\sim} \text{DP}\left(\alpha \times \text{DP}\left(\gamma H\right)\right)$. Modelling $G_j$(s) hierarchically as in HDP and nestedly as in NDP yields different effects. HDP focuses on exploiting statistical strength across groups via sharing atoms $\phi_k$ (s), but it does not partition groups into clusters. This statement is made precisely by noting that $P\left(G_j = G_{j'}\right) = 0$ in HDP. Whereas, NDP emphasises on inducing clusters on both observations and distributions, hence

it partitions groups into clusters. To be precise, the prior probability of two groups being clustered together is $P\left(G_j = G_{j'}\right) = \frac{1}{a+1}$. Finally we note that this original definition of NDP (Rodriguez et al., 2008) does not force the atoms to be shared across clusters of groups, but this can be achieved by simply introducing a DP prior for the NDP base measure, a modification that we use in Chapter 4.

Particularly, the stochastic process of the NDP can be presented as $\{G_1, G_2, ...G_J\} \sim \text{nDP}(\alpha, \gamma, H)$ that is equivalent to $G_1^*, G_2^*, ..., G_k^* \overset{\text{iid}}{\sim} \text{DP}\left(\gamma H\right)$ and $G_j \overset{\text{iid}}{\sim} \text{DP}\left(\alpha \text{DP}\left(\gamma H\right)\right)$. Then, the local atom $\theta_{ji} \overset{\text{iid}}{\sim} G_j$ and the observation is drawn as $x_{ji} \sim F\left(\theta_{ji}\right)$. For posterior inference, we provide the stick-breaking representation (Rodriguez et al., 2008).

$$
\begin{aligned}
&\phi_{kl} \overset{\text{iid}}{\sim} H && w_k \overset{\text{iid}}{\sim} \text{GEM}\left(\gamma\right) && G_k^* = \sum_{l=1}^{\infty} w_{kl} \delta_{\phi_{kl}} \\
&\pi \sim \text{GEM}\left(\alpha\right) && z_j \sim \pi && \\
&l_{ji} \sim w_{z_j} && x_{ji} \sim F\left(\phi_{z_j l_{ji}}\right) && G_j^* \sim \sum_{k=1}^{\infty} \pi_k \delta_{G_k^*}
\end{aligned}
$$

Two documents $j$ and $j$' are considered to be in the same cluster if and only if they take on the same atomic measure component $G_k^*$ such as $G_j, G_{j'} = G_k^*$. Details of inference and properties for the Nested Dirichlet Process model can refer to (Rodriguez et al., 2008). The extended version of NDP for sharing atoms is referred to Chapter 4 in this thesis.

## 2.4 Multilevel Data Modelling

Very often, data naturally present themselves in groups, to form a multilevel or hierarchical data structure. The classic example is schools and pupils: we have a collection of schools, then within each school we have a collection of pupils. We would then say that pupils are nested within schools. Other examples can be individuals nested within countries (survey data), words are nested within documents. We term *individuals* as students while *groups* are as classes. A simple visualization of multilevel data including groups and individuals is displayed in Fig. 2.4.1. Multilevel data structures also arise in longitudinal studies where an individual's responses over time are correlated with each other. For instances, the collection of

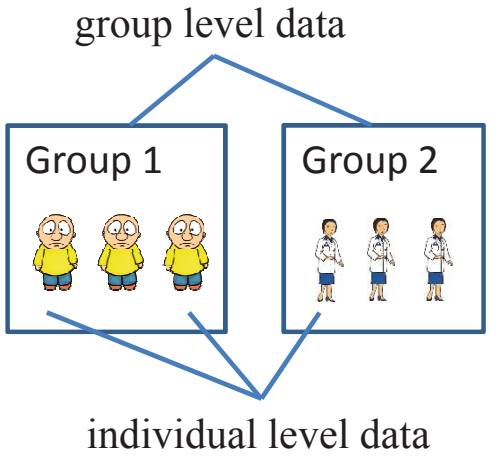



Figure 2.4.1: Multilevel data visualization. There are two groups of data. Each group has three individuals.

GDP per year for states in the USA, each state is a group while GDP per year in a state is individual. Dealing with grouped data, a popular setting known as multilevel analysis (Snijders, 2011; Hox, 2010; Goldstein, 2011) has a board applications from multilevel regression (Gelman et al., 2003; Muthén and Asparouhov, 2009) to multilevel document modelling and clustering (Nguyen et al., 2014).

With the presented background in previous sections from graphical model and exponential family in Section 2.1, parametric approaches for data modelling in Section 2.2, Bayesian nonparametric approaches in Section 2.3. In this section, we present our thesis's research topic of multilevel data modelling that are particularly appropriate for research designs where data for participants are organised at more than one level (i.e., nested data) (Tabachnick et al., 2001). The units of analysis are usually individuals (at a lower level) who are nested within a group of units (at a higher level). While the lowest level of data in multilevel models is usually an individual, repeated measurements of individuals may also be examined (Tabachnick et al., 2001; Goldstein, 2011; Leyland and Goldstein, 2001a). As such, multilevel models provide an alternative type of analysis for univariate or multivariate analysis of repeated measures.

### 2.4.1 Multilevel Models

Multilevel models (a.k.a. hierarchical models, nested models, mixed models, random-effects models) are statistical models of parameters that vary at more than one level (Hox, 2010; Raudenbush and Bryk, 2002). Multilevel models are called multilevel or hierarchical, for two different reasons. The first reason is from the multilevel structure of the data (such as students are grouped within schools). The second reason is from the model itself, which has its own hierarchy, with the parameters of the within-group regression at the bottom, controlled by the hyperparameters of the upper-level model (Gelman and Hill, 2006).

Multilevel models are also known as random-effects or mixed-effects models (Gelman and Hill, 2006). The regression coefficients that are being modelled are called random effects, in the sense that they are considered random outcomes of a process identified with the model that is predicting them. In contrast, fixed effects correspond either to parameters that do not vary (for example, fitting the same regression line for each of the schools) or to parameters that vary but are not modelled themselves (for example, fitting a least squares regression model with various predictors, including indicators for the schools). A mixed-effects model includes both fixed and random effects. Fixed effects have levels that are of primary interest and would be used again if the experiment were repeated. Random effects have levels that are not of primary interest, but rather are thought of as a random selection from a much larger set of levels.

We note that hierarchical Bayesian nonparametric models, including of Hierarchical Dirichlet Processes (Teh et al., 2006) described in previous Section 2.3.3, Nested Dirichlet Processes (Rodriguez et al., 2008) described in Section 2.3.6, are in class of multilevel models. These models are considered as multilevel models due to being able to handle multilevel data and their hierarchy construction of parameters. Particularly, HDP allows sharing statistical strength between individuals within groups while NDP can do multilevel clustering, discussed in Section 2.4.3.

Multilevel models can be used on data with many levels although 2-level models are the most common. In this dissertation, we focus on the 2-level data structure. The dependent variable (e.g., outcome in regression task and label in clustering task) are often examined at the lowest level of analysis (Raudenbush and Bryk, 2002). It

is particularly the case in multilevel regression tasks (Nguyen et al., 2015c) where outcomes and observations are examined at the lowest level. However, for multilevel clustering task (Nguyen et al., 2014) in Chapter 4, the dependent variables, e.g., cluster labels, are examined at both levels of individuals and groups.

### 2.4.2 Multilevel Regression

Regression is a large research field. In this section, we narrow down our focus on a background of linear regression, then we sketchy describe the Gaussian Process as a nonlinear regression setting. We review the multilevel regression task using Linear Mixed Effect model from which we develop our Bayesian Nonparametric Multilevel Regression in Chapter 6.

#### 2.4.2.1 Single-level regression

Regression is a statistical process for estimating the relationships among variables. Particularly, regression models the relationship between a scalar *outcome* variable (or *dependent* variable) $y$ and one or more *explanatory* variables (or *independent* variables) denoted $\boldsymbol{x}$. Regression analysis helps one understand how the typical value of the outcome variable changes when any one of the explanatory variables is varied while the other explanatory variables are held fixed.

Regression analysis is widely used for predicting and forecasting. Many techniques for carrying out regression analysis have been developed. A familiar method such as linear regression is parametric that the regression function is defined in terms of a finite number of unknown parameters that are estimated from the data. Non-parametric regression refers to techniques that allow the regression function to lie in a specified set of functions, which may be infinite-dimensional, such as Gaussian Processes (Rasmussen, 2006).

The case of one explanatory variable is called simple regression. For more than one explanatory variable, the process is called multiple regression (Aiken et al., 1991; Pedhazur and Kerlinger, 1982). This term should be distinguished from multivariate

regression (Chamberlain, 1982), where multiple correlated dependent variables are predicted, rather than a single scalar variable.

**Linear Regression.** In linear regression, data are modelled using linear predictor functions, and unknown model parameters are estimated from the data. Given a data collection $\left\{y_i \in \mathcal{R}, \boldsymbol{x}_i \in \mathcal{R}^d\right\}_{i=1}^N$ of $N$ units, linear regression model assumes the relationship between the outcome variable $y_i$ and the $d$-dimension vector of observation $\boldsymbol{x}_i$ is linear. Linear regression model takes the form:

$$\begin{aligned} y_i &= \beta_0 + x_{i1}\beta_1 + x_{i2}\beta_2 + ... + x_{id}\beta_d \\ &= \boldsymbol{x}_i^T\boldsymbol{\beta} + \epsilon_i \end{aligned} \tag{2.4.1}$$

where $\epsilon_i$ is a residual or error term, $\boldsymbol{\beta}$ is a regression coefficient, including *intercept* and *slope* parameters, and we insert the constant 1 to the first element of $\boldsymbol{x}_i$ that will multiply by intercept parameter $\beta_0$. Using ordinal least squared fitting, the solution for $\boldsymbol{\beta}$ (Bishop, 2006) is: $\hat{\boldsymbol{\beta}} = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{Y}$ where $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^N$ and $\boldsymbol{Y} = \{y_i\}_{i=1}^N$.

We present the two approaches for estimating parameter $\boldsymbol{\beta}$ including ordinal least squared (OLR) and maximum likelihood estimation (MLE). We observe that the two solutions of estimating regression coefficient $\boldsymbol{\beta}$ using OLS and MLE are identical.

**Ordinal Least Square (OLS).** The cost function associated with linear regression is:

$$C = \sum_{i=1}^{N}\left(y_i - \boldsymbol{x}_i^T\boldsymbol{\beta}\right)^2. \tag{2.4.2}$$

The cost function is the sum of squared distance between true outcome $y_i$ and the expected outcome of the data $i$-th in this line $(\boldsymbol{x}_i^T\boldsymbol{\beta} + \epsilon_i)$. We use square errors because we can further compute derivatives of $C$. minimising the cost function in Eq. 2.4.2 is called Ordinal Least Square approximation. We can take the partial derivative of $C$ w.r.t. $\boldsymbol{\beta}$ equate to zero and solve it:

$$\frac{\delta C}{\delta \boldsymbol{\beta}} = \sum_{i=1}^{N} 2\left(y_i - \boldsymbol{x}_i^T\boldsymbol{\beta}\right)(-\boldsymbol{x}_i) = 0.$$

Finally, we get the solution $\boldsymbol{\beta} = \frac{\sum_{i=1}^{N}(\boldsymbol{x}_i - \bar{\boldsymbol{x}})(y_i - \bar{y})}{\sum(\boldsymbol{x}_i - \bar{\boldsymbol{x}})^2}$ where $\bar{\boldsymbol{x}} = \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{x}_i$ and $\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$.

**Maximum Likelihood Estimation (MLE).** MLE aims to estimate for hidden parameter $\boldsymbol{\beta}$ of the given probability distribution $f$ from an i.i.d sample observation (e.g. our data input $\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N \in R^d$ ) so that the likelihood of the data from the given distribution (with the hidden parameter) $f(\boldsymbol{x} \mid \boldsymbol{\beta})$ is maximized. The likelihood function is defined as:

$$\mathcal{L}(\boldsymbol{x}_1, ..., \boldsymbol{x}_N; \boldsymbol{\beta}) = \prod_{i=1}^{N} f(\boldsymbol{x}_i; \boldsymbol{\beta}).$$

We assume the residual error $\epsilon_i$ in Eq. 2.4.1 is followed Normal distribution with mean $\mu = 0$ and variance $\sigma^2$ such that $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$. The likelihood of a single observation $\boldsymbol{x}$ given the parameter $\boldsymbol{\beta}$ is following:

$$p\left(y \mid \boldsymbol{x}^T\boldsymbol{\beta}, \sigma^2\right) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{1}{2\sigma^2}||y - \boldsymbol{x}^T\boldsymbol{\beta}||^2\right\}.$$

The likelihood from a collection of observations $\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N$ is written as:

$$\mathcal{L}(\boldsymbol{x}_1, ..., \boldsymbol{x}_N; \boldsymbol{\beta}) = \frac{1}{(2\pi)^{N/2}\sigma^N} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{N}||y_i - \boldsymbol{x}_i^T\boldsymbol{\beta}||^2\right\}.$$

For ease of computation, we get log likelihood function

$$\log\mathcal{L} = \frac{N}{2}\log(2\pi) + N\log\sigma - \frac{1}{2\sigma^2}\sum_{i=1}^{N}||y_i - \boldsymbol{x}_i^T\boldsymbol{\beta}||^2.$$

We maximize the above log likelihood function by taking partial derivat(ive w.r.t $\boldsymbol{\beta}, \sigma$ and equate to zero respectively.

$$\begin{aligned} \frac{\delta\log\mathcal{L}}{\delta\boldsymbol{\beta}} &= 0 & \boldsymbol{\beta} &= \frac{\sum_{i=1}^{N}(\boldsymbol{x}_i - \bar{\boldsymbol{x}})(y_i - \bar{y})}{\sum(\boldsymbol{x}_i - \bar{\boldsymbol{x}})^2} \\ \frac{\delta\log\mathcal{L}}{\delta\sigma} &= 0 & \sigma &= \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{N} \end{aligned}$$

where $\bar{\boldsymbol{x}} = \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{x}_i$, $\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$, and $\hat{y}_i = -\boldsymbol{x}_i^T\boldsymbol{\beta}$.

We have presented two methods of Ordinal Least Square (OLS) and Maximum Likelihood Estimation (MLE) for estimating regression model. The solutions from both methods are identical in estimating $\boldsymbol{\beta}$.

**Bayesian Linear Regression.** We below present Bayesian Linear Regression which is useful to integrate linear regression into Bayesian nonparametric models in Chapter 6. Bayesian linear regression (Bishop, 2006) is an approach to linear regression in which the statistical analysis is undertaken within the context of Bayesian inference with a prior distribution for parameter $\boldsymbol{\beta}$. In this setting, the regression errors (or residual) is assumed to follow normal distribution $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$. Given a data point $\boldsymbol{x} \in \mathcal{R}^d$ and its respond variable $y$, the likelihood of Bayesian linear regression model with parameter $\boldsymbol{\beta}$ is defined as:

$$p\left(y \mid \boldsymbol{x}, \boldsymbol{\beta}, \sigma^2\right) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left\{-\frac{1}{2}||y - \boldsymbol{x}^T\boldsymbol{\beta}||^2\right\}.$$

Posterior probability distributions of the model's parameter under conjugate prior distribution $\boldsymbol{\beta} \sim \mathcal{N}(0, \Sigma_0)$ is estimated following:

$$p\left(\boldsymbol{\beta} \mid \boldsymbol{x}_{1:N}, y_{1:N}, \Sigma_0, \sigma\right) \propto \mathcal{N}\left(\mu_n, \Sigma_n\right) \tag{2.4.3}$$

where the posterior mean $\mu_n = \Sigma_n\left\{\boldsymbol{X}\sigma^{-1/2}\boldsymbol{Y}\right\}$, and posterior covariance $\Sigma_n = \left(\Sigma_0^{-1} + \boldsymbol{X}\sigma^{-1/2}\boldsymbol{X}^T\right)^{-1}$. We provide detailed derivation of this posterior computation for $\boldsymbol{\beta}$ is Section 6.2.4. The likelihood for predicting new explanatory $\boldsymbol{x}_{\text{new}}$ with new response $y_{\text{new}}$ is computed:

$$\begin{aligned} p\left(y_{\text{new}} \mid \boldsymbol{x}_{\text{new}}, \mu_n, \Sigma_n\right) &= \int_{\boldsymbol{\beta}} p\left(y_{\text{new}} \mid \boldsymbol{x}_{\text{new}}, \boldsymbol{\beta}, \sigma^2\right) p\left(\boldsymbol{\beta} \mid .\right) d\boldsymbol{\beta} \\ &= \mathcal{N}\left(\boldsymbol{x}_{\text{new}}^T\mu_n, \sigma_n^2(\boldsymbol{x}_{\text{new}})\right) \end{aligned} \tag{2.4.4}$$

where $\sigma_n^2(\boldsymbol{x}_{\text{new}}) = \sigma^2 + \boldsymbol{x}_{\text{new}}^T\Sigma_n\boldsymbol{x}_{\text{new}}$.

**Gaussian Processes for Regression.** Gaussian Processes (GP) (Rasmussen, 2006) extends multivariate Gaussian distribution to infinite dimensionality. For-

mally, Gaussian process generates data located throughout some domain such that any finite subset of the range follows a multivariate Gaussian distribution. Given $N$ observations $\boldsymbol{Y} = \{y_1, y_2, ...y_N\}$ can always be imagined as a single point sampled from some multivariate Gaussian distributions.

The mean of GP is assumed to be zero everywhere. What relates one observation to another in such cases is just the covariance function, $k(x, x')$. From the assumption of Gaussian Process, we have $\boldsymbol{y} \sim \mathcal{N}(0, K)$ where the covariance matrix is defined following:

$$K = \begin{matrix} k(x_1, x_1) & k(x_2, x_2) & \cdots & k(x_1, x_N) \\ k(x_2, x_1) & k(x_2, x_2) & \cdots & k(x_2, x_N) \\ \vdots & \vdots & \ddots & \vdots \\ k(x_N, x_1) & k(x_N, x_2) & \cdots & k(x_N, x_N). \end{matrix}$$

A popular choice for the covariance function $K$ is the squared exponential function: $k(x, x') = \sigma_f^2 \exp\left[\frac{-(x-x')^2}{2l^2}\right]$ where $\sigma_f^2$ defines the maximum allowable covariance. It $x \approx x'$, then $k(x, x')$ approaches this maximum of $\sigma_f^2$, indicating that $f(x)$ is perfectly correlated with $f(x')$. If $x$ is far from $x'$, we have instead $k(x, x') \approx 0$. The length parameter $l$ will control this separation when $x$ is not closed to $x'$.

For prediction on the new data point $y_*$, we can update the covariance matrix with $K_* = \begin{matrix} k(x_*, x_1) & k(x_*, x_2) & \cdots & k(x_*, x_N) \end{matrix}$ and $K_{**} = k(x_*, x_*)$. Hence, we can write

$$\begin{bmatrix} \boldsymbol{y} \\ y_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K & K_*^T \\ K_* & K_{**} \end{bmatrix}\right).$$

The conditional probability of $p(y_* \mid \boldsymbol{y})$ is followed Gaussian distribution as $p(y_* \mid \boldsymbol{y}) \sim \mathcal{N}\left(K_* K^{-1} \boldsymbol{y}, K_{**} - K_* K^{-1} K_*^T\right)$ (Rasmussen, 2006; Ebden, 2008).

We summarize the advantage and disadvantage of Gaussian Processes (Rasmussen, 2006). The main advantage of Gaussian Processes for data regression is that they can be optimized exactly, like other kernel methods, given the values of their hyper-parameters (such as the weight decay and the spread of a Gaussian kernel), and this often allows a fine and precise trade-off between fitting the data and smoothing. The disadvantage of Gaussian Processes includes that it uses the whole samples/features information to perform the prediction. Thus, Gaussian Processes loses efficiency in high-dimensional spaces – namely when the number of features exceeds a few

dozens. It might give poor performance and it loses computational efficiency.

#### 2.4.2.2 Linear Mixed Effects model for multilevel regression

Within the scope of the chapter, we focus on the linear part of multilevel regression models where the data presented in groups. Observations in the same group are generally not independent, they tend to be more similar than observations from different groups. Standard single level regression models are not robust due to violation of the independence assumption. That is why we need special multilevel treatment. A multilevel model, which is widely used in multilevel analysis (Hox, 2010; Leyland and Goldstein, 2001b; Diez-Roux, 2000), is Linear Mixed Effect (McLean et al., 1991).

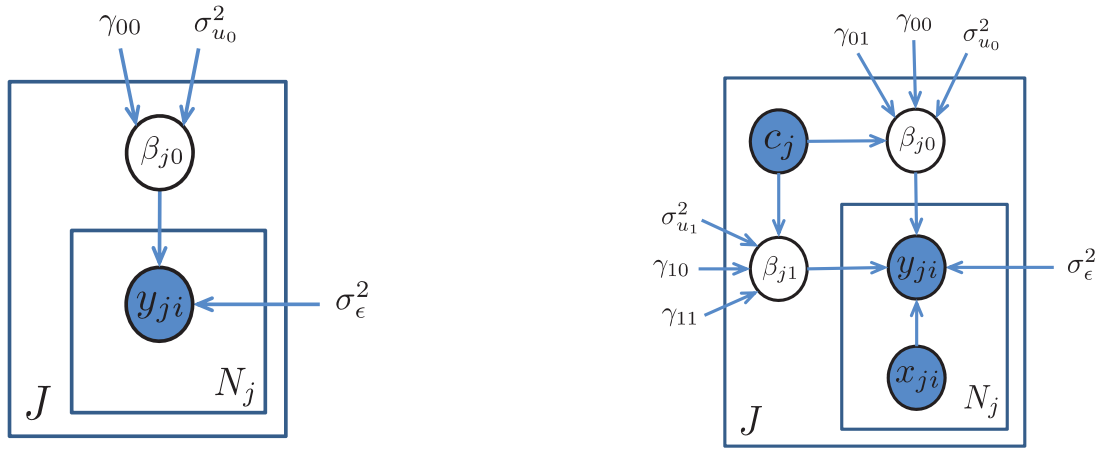
We consider a pair of outcome and observation in hierarchical structure ($y_{ji} \in R, \boldsymbol{x}_{ji} \in R^d$) where $y_{ji}$ is an outcome (or response) and $\boldsymbol{x}_{ji}$ is an observation for trial $i$ in group $j$. The multilevel models are the appropriate choice that can be used to estimate the intraclass correlation and regression in the multilevel data. Specifically, we consider Linear Mixed Effects models (McLean et al., 1991) which are extensions of linear regression models for data that are organized in groups. We begin with the basic intercept-only model.

**Intercept only model.** The intercept-only model (null model, baseline model) uses only the intercept to explain the data. In this model, the outcome variable $y_{ji}$ in group $j$ is estimated as:

$$y_{ji} = \beta_{j0} + \epsilon_{ji} \tag{2.4.5}$$

where $\epsilon_{ji} \sim \mathcal{N}(0, \sigma_\epsilon^2)$. The variance of error for each individual is denoted as $\sigma_\epsilon^2$ . To make all individuals in the same group share the same parameter, the regression coefficient $\beta_{j0}$ is assumed as: $\beta_{j0} = \gamma_{00} + u_{j0}$ where $u_{ji} \sim \mathcal{N}(0, \sigma_u^2)$ and $\sigma_u^2$ is the variance of error in group level. Therefore, the single equation for the intercept-only model becomes:

$$y_{ji} = \gamma_{00} + u_{j0} + \epsilon_{ji}$$




(a) Intercept-only model.

(b) Linear Mixed Effects model.

Figure 2.4.2: Graphical representation. Left: Intercept Only. Right: Linear Mixed Effects model.

In practice, the intercept-only model above is often used as a baseline comparison for evaluation multilevel regression models.

**Linear Mixed Effects model.** The LME model (McLean et al., 1991) describes the relationship between a response variable and independent variables in multilevel structure, with coefficients that can vary with respect to one or more grouping variables. A mixed-effects model consists of two parts, fixed effects and random effects. Fixed-effects terms are usually the conventional linear regression part, and the random effects are associated with individual experimental units drawn randomly from population. The random effects have prior distributions whereas fixed effects do not. Linear Mixed Effects model can represent the covariance structure related to the grouping of data by associating the common random effects to observations in the same group. The standard form of a linear mixed-effects model is as follows:

$$y_{ji} = \boldsymbol{\beta}_{j0} + \boldsymbol{x}_{ji}^T \boldsymbol{\beta}_{j1} + \epsilon_{ji} \qquad \epsilon_{ji} \sim \mathcal{N}\left(0, \sigma_\epsilon^2\right)$$

where the regression coefficients (for group $j$) $\boldsymbol{\beta}_{j0}$ and $\boldsymbol{\beta}_{j1}$ are computed as $\boldsymbol{\beta}_{j0} = \gamma_{00} + \gamma_{01} c_j + u_{j0}$ where $u_{j0} \sim \mathcal{N}\left(0, \sigma_{u0}^2\right)$, and $\boldsymbol{\beta}_{j1} = \gamma_{10} + \gamma_{11} c_j + u_{j1}$ where $u_{j1} \sim \mathcal{N}\left(0, \sigma_{u1}^2\right)$. Therefore, the final form to predict the individual outcome

variable $y_{ji}$ using individual explanatory variables $x_{ji}$ and group explanatory variable $c_j$ is followed:

$$y_{ji} = \underbrace{\gamma_{00} + \gamma_{01}c_j + \gamma_{10}x_{ji} + \gamma_{11}c_j x_{ji}}_{\text{fixed effects}} + \underbrace{u_{j0} + u_{j1}x_{ji} + \epsilon_{ji}}_{\text{random effects}}.$$

Fixed effects have levels that are of primary interest and would be used again if the experiment were repeated. Random effects have levels that are not of primary interest, but rather are thought of as a random selection from a much larger set of levels. We present the graphical representation of LME model in Fig. 2.4.2b. The common parameter estimation methods for linear mixed effect include Iterative Generalized Least Squares (Goldstein, 1986) and Expectation Maximization algorithm (Raudenbush, 1992).

### 2.4.3 Multilevel Clustering

We review related works in multilevel clustering task for multilevel data where individuals are organised into groups. Using multilevel clustering, one aims to discover the cluster labels at multilevel of individuals and groups. For instance, given a nested structure of students and classes, multilevel clustering task will assign students to a suitable student's cluster across classes (e.g., based on similarity in student characteristic), multilevel clustering also groups classes together (e.g., based on the overall student's academic performance in each class). In chapter 4, we would introduce the novel model, namely MC$^2$ for multilevel clustering.

We note that multilevel clustering is completely different from hierarchical clustering (Johnson, 1967). In data mining, hierarchical clustering (Johnson, 1967; Navarro et al., 1997) is a method of cluster analysis which seeks to build a hierarchy of clusters. Strategies for hierarchical clustering generally fall into two types: Agglomerative and Divisive. Agglomerative (Zhang et al., 2013; Beeferman and Berger, 2000) is a 'bottom up' algorithm: each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy. Divisive (Gowda and Ravi, 1995; Chavent et al., 2006) is a 'top down' approach: all observations start in one cluster, and splits are performed recursively as one moves down the hierarchy.

#### 2.4.3.1 Single-level clustering

Before describing multilevel clustering, we start with single-level data clustering. Clustering data based on a measure of similarity is a critical step in scientific data analysis and in engineering systems (Frey and Dueck, 2007). A common approach is to use data to learn a set of centres such that the sum of squared errors between data points and their nearest centres is small (MacQueen et al., 1967).

The popular k-means (MacQueen et al., 1967) clustering technique begins with an initial set of randomly selected centre and iteratively refines this set so as to decrease the sum of squared errors. K-means clustering is sensitive to the initial selection of exemplars, so it is usually starting with different initializations in an attempt to find a good solution. Finally, this algorithm aims at minimising an objective function, in this case a squared error function. The objective function of K-means is as:

$$J = \sum_{k=1}^{K} \sum_{i=1}^{N} \mathbb{I}\left(z_i, k\right) ||x_i - c_k||^2$$

where $z_i$ is the label assignment of data point $i$ to a cluster $k$, and $||x_i - c_k||^2$ is a chosen distance measure between a data point $x_i$ and the cluster centre $c_k$.

Another popular clustering technique is Affinity Propagation (AP) (Frey and Dueck, 2007) which based on the concept of 'message passing' between data points. Unlike clustering algorithms such as k-means or k-medoids, AP does not require the number of clusters to be determined or estimated before running the algorithm.

Using Bayesian nonparametric approaches, such as Dirichlet Process Mixture in Section 2.3.1.2, for nonparametric clustering is also a promising way. Due to the characterization of Dirichlet Process prior, the Bayesian nonparametric methods, using DP, can discover the suitable number of clusters by itself. Thus, it overcomes the problem of model selection for clustering task.

#### 2.4.3.2 Approaches for multilevel clustering

We consider the clustering task for multilevel data structure where observations are organised into groups. Our aim of multilevel clustering is clustering both observations and groups. For example, given a collection of pupils nested within schools, we want to cluster pupils into a number of pupil's clusters and we are also interested in clustering schools into school's clusters.

We consider the naive approach for multilevel clustering that is to employ a two-stage process where clustering at individual level and clustering at group level are treated as two separated tasks. First, topic models (e.g., LDA or HDP) are applied to extract the topics and their mixture proportion for each document in which the words (individual level) are grouped into topics. Then, this is used as feature input to another clustering algorithms. Some examples of this approach include the use of LDA and K-means for image clustering (Elango and Jayaraman, 2005) and HDP and Affinity Propagation (Frey and Dueck, 2007) for clustering human activities (Nguyen et al., 2013b; Phung et al., 2014).

A more principled approach is to conduct multilevel clustering jointly. By jointly, we treat this multilevel clustering task, which include clustering observations and clustering groups, as a dependent task, not separate as used in previous approaches (Elango and Jayaraman, 2005; Nguyen et al., 2013b). To have matters concrete, we give an example of multilevel clustering in words and documents. The two tasks - word clustering (assign words into topics) and document clustering (assign documents into document clusters) are not totally independent. On one hand, a good word topic assignment can generate effective latent representations in the documents, which are the input of document clustering step and deeply effect such document clustering performance. On the other hand, document cluster labels obtained from document clustering can serve as the supervised information to guide word clustering process. Multilevel clustering of words and documents follows a chicken-and-egg relationship. A better document clustering results produce the better word clustering and better words in turn contributes to better documents. Thus performing them jointly in a unified model can help them mutually benefit each other (Xie and Xing, 2013; Nguyen et al., 2014) that is multilevel clustering.

We further review related tasks in topic modelling that can perform multilevel clus-

tering. The first Bayesian nonparametric model proposed for this task is the nested Dirichlet Process (nDP) (Rodriguez et al., 2008) where documents in a cluster share the same distribution over topic atoms. Although the original nDP does not force the topic atoms to be shared across document clusters, this can be achieved by simply introducing a DP prior for the nDP base measure (Nguyen et al., 2014). The same observation was also made by (Wulsin et al., 2012) who introduce the MLC-HDP, a 3-level extension to the nDP. This model thus can cluster words, documents and document-corpora with shared topic atoms throughout the group hierarchy. Xie and Xing (2013) recently introduced the Multi-Grain Clustering Topic Model which allows mixing between global topics and document-cluster topics. However, this is a parametric model which requires fixing the number of topics in advance.

We note that all of these existing models do not attempt to utilise group-level context information. Therefore, in Chapter 4, we introduce MC2, the Bayesian nonparametric framework for multilevel clustering which utilising group-level context information.

## 2.5 Closing Remarks

In this chapter, we have discussed the necessary background over which the material of following chapters would be built. We firstly described the background in the probabilistic graphical model which provides the basic foundation for the subsequent sections. Next, we present the parametric approaches in data modelling, as a closely related counterpart for the Bayesian nonparametric modelling. Then, we go into detail of the Bayesian nonparametric modelling using Dirichlet Process as the prior distribution. Further, we describe multilevel modelling for handling multilevel data. Bayesian nonparametric modelling and multilevel analysis play as a key research interest in this thesis.

In the next chapter, we present our first contribution to the abnormality detection task in video surveillance where the data are organised at multilevel.

# Chapter 3

# Abnormal Detection with Multilevel Structure in Video Surveillance

In this chapter we address a special type of data: video. A video is a sequence of images which is rich in its content and structures. Video organisation is often naturally sequential and hierarchical. We particularly address the problem of video surveillance in this chapter. As widely acknowledged in the computer vision community and security management, discovering suspicions and irregularities of events in a video sequence are the key issue for abnormal detection in video surveillance. The important steps in identifying such events include stream data segmentation and hidden patterns discovery. However, the crucial challenge in stream data segmentation and hidden patterns discovery are the number of coherent segments in surveillance stream and the number of traffic patterns is unknown and hard to specify.

The theory of Bayesian Nonparametric (BNP) holds a promise to address these challenges. As such, BNP can automatically identify the suitable number of cluster from the data. Therefore, in this chapter we revisit the abnormality detection problem through the lens of BNP and develop a novel usage of BNP methods for this problem. In particular, we employ the Infinite Hidden Markov Model (Beal et al., 2002) and Bayesian Nonparametric Factor Analysis (Paisley and Carin, 2009).

The first advantage of our methods includes identifying the unknown number of coherent sections of the video stream would result in better detection performance.

Each coherent section of motion (e.g., traffic movements at night time and day time) would contain different types of abnormality. Unlike traditional abnormality detection methods which typically build upon a unified model across data stream. The second benefit of our system is to provide an interface allowing users to interactively examine rare events in an intuitive manner. Because the abnormal events detected by algorithms and what is considered anomalous by users may be inconsistent.

To this end, in this chapter, we make two major contributions to abnormal detection in video surveillance: (1) the novel adaptation of the Infinite Hidden Markov Model for stream data segmentation, and (2) the development of a novel interactive system with Bayesian Nonparametric Factor Analysis allowing users to inspect and browse suspiciously abnormal events.

We organise the rest of this chapter as follows. We present our overview on abnormality detection in video surveillance and the need of segmenting the data and interaction in Section 3.1. In Section 3.3, we describes our contribution on Bayesian nonparametric data stream segmentation for abnormal detection. Section 3.4 presents our introduced browsing system for abnormal detection. Finally, we present a summary of the chapter with some concluding remarks in Section 3.6.

## 3.1 Abnormality Detection for Video Surveillance

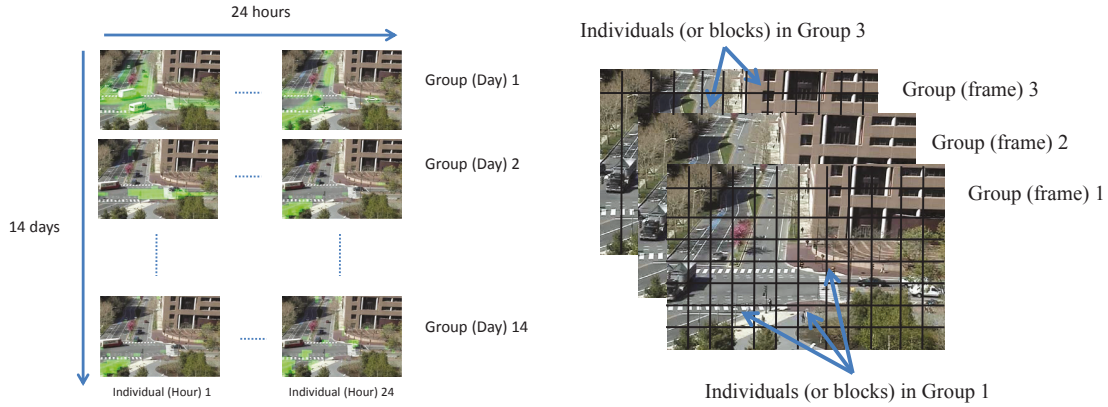
Ideally, abnormality detection algorithms should report only events that require intervention - however, this is impossible to achieve with the current state-of-art, and a large semantic gap exists between what is perceived as abnormal and what are computationally realisable outlier events. An alternate framework in which the algorithm reports a fraction ($< 1\%$) of rarest events to a human operator for potential intervention (Budhaditya et al., 2009) has been successful commercially (*ic-etana.com*). By retaining humans in the loop, whilst drastically reducing the footage that needs scrutiny, the framework provides a practical recourse to machine-assisted video surveillance. A typical medium sized city council has to handle hundreds of cameras simultaneously, and it is imperative that the computational cost be low. This is achieved via an efficient algorithm based on PCA analysis of the video feature data. Motion based features are computed within a fixed duration video clip

(typically 10-30 secs). PCA analysis is performed on the training data set to obtain the residual subspace, and the threshold corresponding to a desired false alarm rate. During testing, if the projection of the test vector in the residual subspace exceeds the computed threshold, the event is reported to the operator. Since the algorithm is based on PCA, it is important that the training data be coherent, so as to have most of the energy concentrated within a low-dimensional principal subspace. In this case, most normal events remain within the principal space upon projection, and the residual subspace retains the fidelity for detecting subtle but rare events. However, for typical outdoor surveillance, the feature vectors generally exhibit different modes - depending on the time of day, climatic variations etc. If we try to fit all these incoherent modes into a single model, we reduce the sensitivity of detection. If we construct one principle subspace for a 24 hours period, we are likely to miss events at night, because nights have very different motion profiles to that of the daytime.

Thus, it is of paramount importance that video data be separated into coherent sections on which subsequent statistical analysis, for tasks such as anomaly detection, can be performed. One solution to provide this data segmentation into coherent modes is to use Markov models such as the Hidden Markov Model. However, these models require apriori specification of the number of modes. To circumvent this problem, we model the activity levels as a mixture of Gaussian states for the Infinite Hidden Markov Model (iHMM) (Beal et al., 2002) segmentation. We show application of the model to such stream data and present the collapsed Gibbs inference to achieve automatic data segmentation. To demonstrate the model, we perform experiments with 336 hours of footage obtained from a live surveillance scene. We show how the use of model selection as a preliminary process improves typical downstream processes such as anomaly detection.

## 3.2 Multilevel Structure of Video Surveillance Data

In video surveillance, the data is usually captured and processed into multilevel structure of groups and individuals. Individuals are nested within groups. In this chapter, we consider two ways of multilevel video surveillance. The first construction of video surveillance (e.g., 14 days) that the group level is one day footage (14 groups)




(a) Multilevel setting 1.

(b) Multilevel setting 2.

Figure 3.2.1: Multilevel structure visualization of video surveillance data. a) Multilevel setting 1: Group level is data collected within a day. Individual level is data within an hour. b) Multilevel setting 2: Group is a video frame. Individual is a block within a video frame.

and the individual unit is each hour footage (e.g., 24 hours or units a day). An example of the first multilevel setting is displayed in Fig. 3.2.1a. The second view is that the video frame is a group level, the number of video frame is the number of group. Then, each video frame is split into multiple frame blocks (individual units). It is necessary to divide the frame into blocks, instead of manipulating on pixel levels, for computational efficiency. We plot a visualization for the second multilevel setting in Fig. 3.2.1b. The first construction is further used in Section 3.3 and the second multilevel construction is in Section 3.4.

The novelty of our contribution is in tacking a novel problem in large scale multilevel stream data - model fitting to find coherent data sections of individual units (hours) across groups (days), on which suitable models can be subsequently constructed. The significance of our solution is that the use of iHMM allows incremental use, and thus lends itself to large scale data analysis. In addition, we introduce the browsing framework to overcome the semantic gap between the returned events by the algorithms and the true events. The browsing system assists user in analysing and filtering suspicious events at multilevel (individual unit of frame blocks to group level of video frames).

## 3.3 iHMM Stream Data Segmentation

For data segmentation using standard HMM, one needs to specify the number of states in advance and use the EM algorithm to estimate the parameters. The iHMM (Beal et al., 2002) overcomes this restriction, allowing the number of states to grow unboundedly according to the data. In other words, the number of states will be automatically inferred from the data. It was later shown in (Teh et al., 2006) that this model can be interpreted using the hierarchical Dirichlet process formalism in which the number of groups is dynamically changed according to the state assignments. This interpretation is significant as it provides a deeper understanding and formal framework to work with the iHMM. Interested readers are referred to (Beal et al., 2002; Teh et al., 2006) for details.

### 3.3.1 Multi-model abnormality detection framework

We below describe the multilevel data construction of the video surveillance.

#### 3.3.1.1 Multilevel data construction

We use video footage spanning multiple days for model selection and abnormality detection. A surveillance video in 14 days is divided into a sequence of fixed 20 sec clips. Optic flow vectors are computed (Horn and Schunck, 1981). For each clip, we first aggregate the total count of optic flow vectors at each pixel location over all the frames, and then spatially bin them into a $10 \times 10$ uniform grid. After vectorization, we obtain a 100 dimensional feature vector for each clip as in (Budhaditya et al., 2009).

Specifically, the multilevel structure of the data is presented that each day is a group-level and each hour (for each day) is a unit-level. Totally, we have 14 groups each of which comprises of 24 individuals. For the model selection phase, we unroll the multilevel structure of video surveillance data into flat structure constructed by activities within each hour across days. We use the total activity level in an hour, computed by summing the feature vectors over an hourly window and then

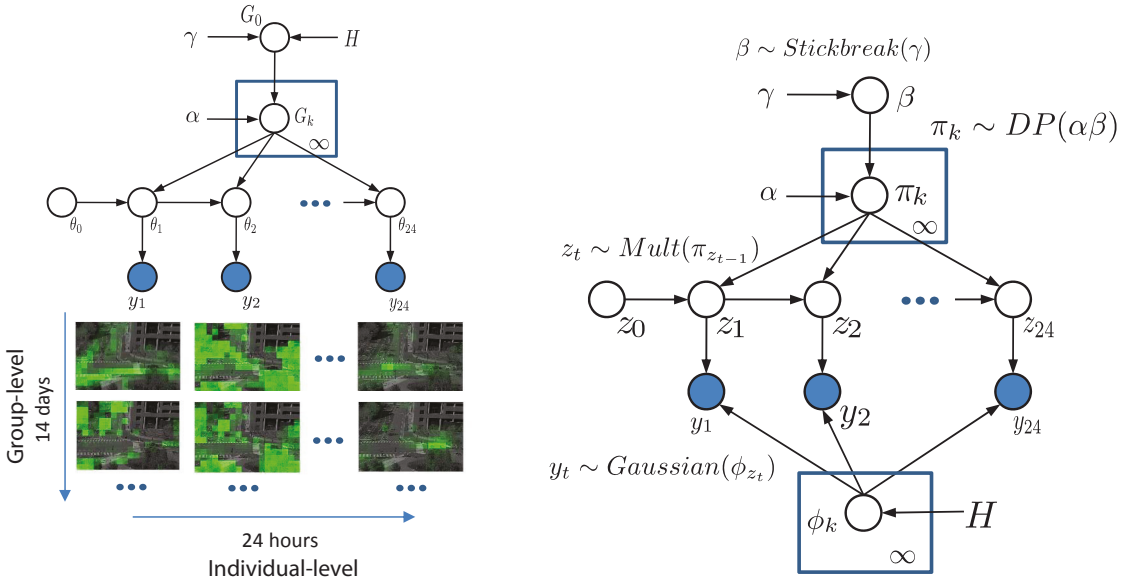



Figure 3.3.1: The infinite Hidden Markov model representation. Left: Stochastic process of iHMM. Each observation $y_t$ indicates for a traffic movement for an hour $t$ (individual-level), including 24 hours, collected from 14 days (group-level). Right: Stick-breaking representation of the data.

summing across the length of the resultant vector generating a scalar value for the total activity. The activity level is then modelled by a mixture of Gaussian states for the infinite-HMM (Beal et al., 2002; Teh et al., 2006) segmentation. Once we obtain the segmentation of hours based on the activity levels, we run separate anomaly detectors for each model. In the following sections, we present the framework for iHMM followed by a brief description of the core anomaly detection algorithm of (Budhaditya et al., 2009).

**iHMM for data segmentation.** Under the Hierarchical Dirichlet Process specification (Teh et al., 2006), the building block property can be adopted to represent the infinite Hidden Markov model (iHMM) (Beal et al., 2002). Teh et al. (2006) describe the infinite Hidden Markov model, namely a Hierarchical Dirichlet Process Hidden Markov model (HDP-HMM) which provides an alternative method to place a Dirichlet prior over the number of state. Therefore, the (unknown) number of states in HMM is identified in the same way as HDP.

Using HDP (Teh et al., 2006) as a nonparametric prior for building block, the stochastic process of HDP-HMM is described as:

$$
\begin{aligned}
G_0 &\sim \mathrm{DP}(\gamma, H \times S) & \theta_t &\overset{\text{iid}}{\sim} G_k \\
G_k &\overset{\text{iid}}{\sim} \mathrm{DP}(\alpha, G_0) \;\; k = 1, 2, ...\infty & y_t &\sim F(\theta_{t-1}) \;\; t = 1, 2, ..., T.
\end{aligned}
$$

There are $T$ timestamps (e.g., number of hours in a day the data is collected). The stick-breaking of HDP-HMM is illustrated in Fig. 3.3.1 in which the parameters have the following distributions:

$$
\begin{aligned}
\boldsymbol{\beta} &\sim \mathrm{GEM}(\gamma) & \boldsymbol{\pi}_k &\sim \mathrm{DP}(\alpha, \boldsymbol{\beta}) \\
\phi_k &\sim H \;\; k = 1, 2, ...\infty & z_t &\sim \pi_{z_{t-1}} \;\; t = 1, 2, ..., T \\
& & y_t &\sim F(\phi_{z_t}).
\end{aligned}
$$

**Inference for HDP-HMM.** In this work, we use the iHMM at the first stage to segment the data into coherent sections before building the abnormality detection models. The use of Markov model ensures that the temporal dynamics nature of the data is taken into consideration. The number of coherent sections is unknown and will be estimated from the data. Our first goal is perform a rough data segmentation at hourly intervals; thus there are 24 data points for each day using the average motion at each hour as the input. These inputs correspond the observed variables $\{y_t\}$, and $\{z_t\}$ plays the role the latent state variables as in a standard HMM. $H$ is the base measure from which parameters $\{\phi_k\}$ will be sampled from. In our case, we model $y_t$ as a univariate Gaussian and thus each $\phi_k$ is a tuple of $\{\mu_k, \sigma_k^2\}$ where both $\mu_k$ and $\sigma_k^2$ are unknown and treated as random variables. We use $H$ as a conjugate prior, and thus $H$ in our case is a Gaussian-invGamma distribution. A graphical model representation is shown in Fig. 3.3.1.

We use collapsed Gibbs inference (Liu, 1994) for iHMM as described in (Van Gael et al., 2008) in which the latent state $z_t$ and the stick-breaking weight $\beta_k$ are sequentially sampled by explicitly integrating out parameters $\{\phi_k\}$ for the emission probability and $\{\pi_k\}$ for the transition probability. For example, given $z_{t-1} = i, z_{t+1} = j$ from the previous iteration, the conditional Gibbs distribution to sample $z_t$ has the form:

- Sampling $z_t$. Consider the conditional probability of $z_t$

$$p\left(z_t = k \mid z_{-t}, \boldsymbol{y}, \boldsymbol{\beta}, H\right) \propto \underbrace{p\left(y_t \mid z_t = k, \boldsymbol{z}_{-t}, \boldsymbol{y}_{-t}, H\right)}_{\text{observation likelihood}} \times \underbrace{p\left(z_t = k \mid \boldsymbol{z}_{-t}, \alpha, \boldsymbol{\beta}\right)}_{\text{CRP of transition}}.$$

The first term is the likelihood of the observation $y_t$ given the component $\phi_{z_t}$ that can be expressed as $\int_{\phi_k} p\left(y_t \mid z_t = k, \phi_k\right) p\left(\phi_k \mid \boldsymbol{y}_{-t}, \boldsymbol{z}_{-t}, H\right) d\phi_k$ which is easily analysed using the conjugate property. The second probability is simply the Chinese Restaurant Process of transition. Denote $n_{ij}$ as the number of transitions from state $i$ to state $j$, $n_{*j}$ as the number of all transitions to state $j$. Similarly, $n_{i*}$ is the number of all transitions departing from state $i$. The CRP likelihood under Markov property can be analysed as:

$$p\left(z_t = k \mid \boldsymbol{z}_{-t}, \alpha, \boldsymbol{\beta}\right) \propto \underbrace{p\left(z_t = k \mid z_{t-1}, \alpha, \boldsymbol{\beta}\right)}_{\text{from previous state t-1 to state t}} \times \underbrace{p\left(z_t = k \mid z_{t+1}, \alpha, \boldsymbol{\beta}\right)}_{\text{from state t to next state t+1}}.$$

We then have four cases to compute this probability:

$$p\left(z_t = k \mid \boldsymbol{z}_{-t}, \alpha, \boldsymbol{\beta}\right) \propto \begin{cases} \left(n_{z_{t-1},k} + \alpha\beta_k\right) \frac{n_{k,z_{t+1}} + \alpha\beta_{z_{t+1}}}{n_{k*} + \alpha} & k \leq K, k \neq z_{t-1} \\ \left(n_{z_{t-1},k} + \alpha\beta_k\right) \frac{n_{k,z_{t+1}} + 1 + \alpha\beta_{z_{t+1}}}{n_{k*} + 1 + \alpha} & z_{t-1} = k = z_{t+1} \\ \left(n_{z_{t-1},k} + \alpha\beta_k\right) \frac{n_{k,z_{t+1}} + \alpha\beta_{z_{t+1}}}{n_{k*} + 1 + \alpha} & z_{t-1} = k \neq z_{t+1} \\ \alpha\beta_{\text{new}}\beta_{z_{t+1}} & k = K + 1. \end{cases}$$

- Sampling stick-breaking $\boldsymbol{\beta}$, and hyperparameters $\alpha$,$\gamma$ are exactly the same as for HDP describing in (Teh et al., 2006).

For robustness we also let the concentration hyper-parameters $\alpha$ and $\gamma$ to follow Gamma distributions and they will also be re-sampled at each Gibbs iteration.

**Abnormality detection algorithm.** Let assume that $X \in \mathbb{R}^{d \times n}$ is the data matrix with $n$ centralized feature vectors of $d$ dimensions and $C$ is the covariance matrix with its SVD factorization:

$$C = U\Sigma U^T.$$

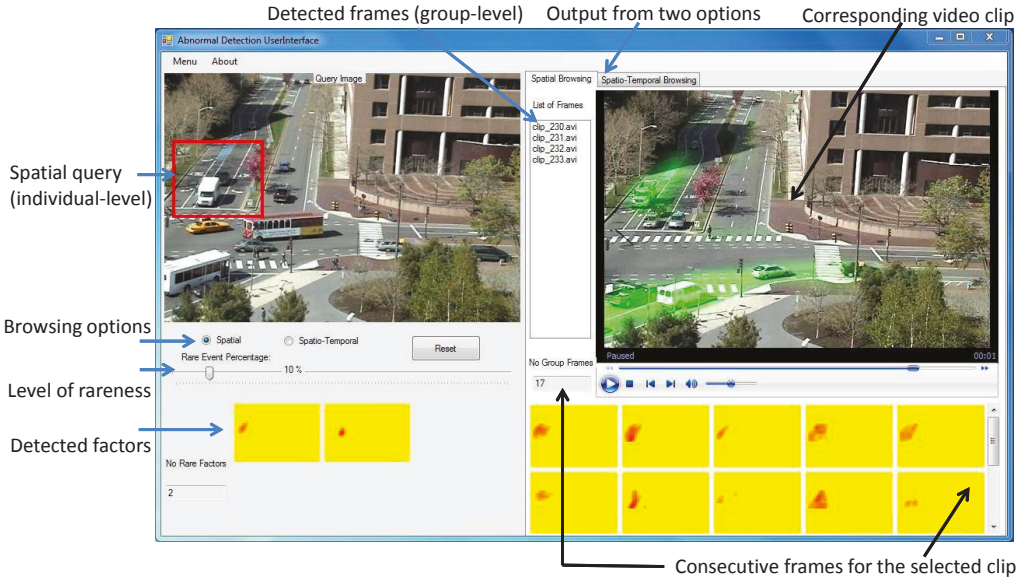



Figure 3.4.1: User interface for browsing abnormal events in multilevel setting. User provides a spatial query (at individual-level) to search for abnormal events, then the system returns a list of detected video clips (at group-level). User can vary the degree of rareness (from 0 to 100%) which will result in more or less suspicious events.

We divide the eigenvectors from $U$ in two groups:

$$C = [U_1\, U_2] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} U^T$$

such that $\frac{\mathrm{tr}(\Sigma_1)}{\mathrm{tr}(\Sigma_1)+\mathrm{tr}(\Sigma_2)} = 0.9$, i.e., selecting the most significant eigenvectors such that it covers the 90% of the total energy. $U_1$ is called the principal subspace and $U_2$ is called the residual subspace. The abnormality detection algorithm works by projecting the test vectors to the residual subspace $U_2$ and comparing it to the detection threshold ($\lambda$), also called the Q-statistic, and is a function of the non-principle eigenvalues in residual subspace.

## 3.4 Interactive System for Browsing Anomalous Events at Multilevel

Security and surveillance systems focus on rare and anomalous events detection. Typically, these events are detected by estimating the statistics from the "normal" data - anything that deviates is termed as *rare*. The problem, however, is that in surveillance data, there is a semantic gap between *statistically* rare events produced by the detection algorithms and what the user would consider as *semantically* rare.

In this section, we raise the question: Is there an alternative to examining these anomalies, at least retrospectively? Consider security officers being given location/time of an incident - they now wish to find the matching footages. We propose a novel interface that permits the operators to specify such queries, and retrieve potential footages of *rare events* that match. This geometric query can be either *spatial* (rare events in region of interest) or *spatial-temporal* (rare events at location A, then B).

Our solution is firstly to find the hidden patterns in the scene. Since the number of latent factors is unknown in advance, we employ recent advances in Bayesian nonparametric factor analysis. The generative process models non-negative count data with a Poisson distribution (Gupta et al., 2012). The presence or absence of a factor is modelled through a binary matrix. Its nonparametric distribution follows the Indian Buffet Process (Griffiths and Ghahramani, 2006), and is modelled through a draw from Beta process, which allows infinitely many factors. The extracted factors correspond to patterns of movement in the scene. The rareness of each extracted factor is determined by how much it is used across the whole data set. The factors are then ordered in decreasing rarity, and the user is allowed to choose a proportion of rare factors for consideration. Three top candidates' rare factors from MIT dataset are visualized in the right column of Fig. 3.5.5 while three other common patterns are on the left hand side. Frames that contain these factors are considered as potential candidates.

The solution to a given geometric query is candidate frames that satisfy the specified spatial or spatial-temporal constraints. We demonstrate this browsing paradigm, with spatial and spatial-temporal queries in video surveillance. The user interface

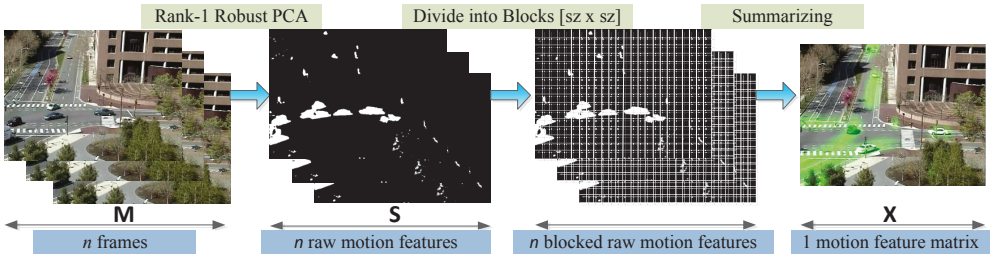



Figure 3.4.2: Foreground extraction and feature computation using rank-1 robust PCA.

of our system is displayed in Fig. 3.4.1.

The significance of this paradigm is that it allows an operator to browse rare events, either spatially or spatial-temporally, at different "scales" of rarity. The use of non-parametric factor analysis models allows the framework to gracefully adapt to the data, without the need for *a priori* intervention. The framework can also easily be extended to accommodate multiple cameras. To our knowledge, there is no such existing system in the literature. Our main contributions in this interface include:

- The anomaly detection frame work based on part-based matrix decomposition that utilises our recently introduced rank-1 robust background subtraction for motion video from static camera and nonparametric pattern analysis.
- The new browsing scheme allowing users not only to control the rareness degree but also to query spatial or spatial-temporal searching to overcome the difficulty due to the semantic gap.

### 3.4.1 Proposed browsing framework

A schematic illustration of the proposed system is shown in Fig. 3.4.2. The first step is to perform background subtraction followed by the feature extraction step detailed in Section 3.4.2. Once the features are extracted, latent factors are learned as detailed in Section 3.4.3. We use non-parametric factor analysis to recover the decomposition of factors (motion patterns) and constituent factor weights. For each latent factor, a rareness score is derived based on their overall contribution

to the scene, and sorted in a decreasing order of rareness level. Since we follow a part-based decomposition approach for scene understanding, each latent factor is a *sparse* image having the same dimension of the original video frame. Therefore, a query for rare events at a spatial location can directly 'interact' with latent factors. The user is then able to select a proportion of rare factors for consideration. Based on the rareness degree of each latent factors, the interface returns to the user the corresponding footages. We shall now describe these steps in detail.

### 3.4.2 Foreground extraction and data representation

Since our framework focuses on scene understanding and therefore, features are extracted directly from the foreground information. To do so, we require a robust foreground extraction algorithm which can operate incrementally and in real-time. To this end, we utilise a recently proposed robust PCA approach (Pham et al., 2011) which is a special case of the robust PCA theory (Candes et al., 2011; Eriksson and van den Hengel, 2010) developed specifically for static surveillance camera. Given a short window time size of $n$ and $\boldsymbol{M} = [M_1, M_2, \ldots, M_n]$ being the data matrix consisting of $n$ consecutive frames, the goal of robust PCA theory is to decompose

$$\boldsymbol{M} = \boldsymbol{L} + \boldsymbol{S},$$

where $L$ is a low-rank matrix and $S$ is a sparse matrix. A standard algorithm to perform robust PCA is principal component pursuit (PCP) (Candes et al., 2011) which involves SVD decomposition at each optimization iteration step. However, it can be very costly to compute. Static cameras, on the other hand, pose a strong rank-1 characteristic wherein the background remains unchanged within a short duration. Given this assumption, an algorithm for rank-1 robust PCA can be efficiently developed which is shown to be a robust version of the temporal median filter (Pham et al., 2011). This makes the foreground extraction, contained in $\boldsymbol{S}$, becomes extremely efficient[1] since it can avoid the costly SVD computation in the original formulation of (Candes et al., 2011). Moreover, it can be operated incrementally in real-time.

[1]In practice, it is noted to be 10-20 times faster than a standard optical flow implementation.

Next, using the sparse matrix $\boldsymbol{S}$, a fixed $sz \times sz$ block is super-imposed and the foreground counts in each cell is accumulated to form a feature vector $X$ summarizing the data matrix $\boldsymbol{M}$ over a short window time of size $n$. An illustration of this step is shown in Fig. 3.4.2.

### 3.4.3 Learning of latent factors

Recall that a foreground feature $X_t$ is collected for each short window $t$. Let $\boldsymbol{X} = [X_1 X_2 \dots X_T]$ be the feature matrix over such $T$ collections. In other words, $\boldsymbol{X}$ contains a collection of grouped data where $X_t$ is a group (or video frame) in our multilevel structure. Our next goal is to learn latent factors from $\boldsymbol{X}$, each of which represents a 'part' or basis unit that constitutes our scene. Using a part-based decomposition approach, a straightforward approach is to use Nonnegative Matrix Factorization (NNMF) of (Lee and Seung, 2001) which factorizes $\boldsymbol{X}$ into

$$\boldsymbol{X} \approx \boldsymbol{W}\boldsymbol{H}, \tag{3.4.1}$$

where $\boldsymbol{W}$ and $\boldsymbol{H}$ are nonnegative matrices. The columns of $\boldsymbol{W}$ contains $K$ latent factors and $\boldsymbol{H}$ contains the corresponding coefficients of each factor contribution to the original data in $\boldsymbol{X}$. Due to the nonnegativity of $\boldsymbol{H}$, a part-based or additive decomposition is achieved and each columns of $\boldsymbol{X}$ is represented by $X_j = \sum_{k=1}^{K} W_k H_{kj}$. However, a limitation of NNMF for our framework is that it requires the number of latent factors $K$ in advance. This can severely limit the applicability of the proposed framework since such knowledge on $K$ is very difficult to obtain.

To address this issue, we employ recent advances in Bayesian nonparametric factor analysis for this task which can automatically infer the number of latent factors from the data (Paisley and Carin, 2009; Teh et al., 2007). In particular, we use a recent work (Gupta et al., 2012) that models count data using Poisson distribution. For the sake of completeness, we shall briefly describe it here. A nonparametric Bayesian factor analysis can be written as follows:

$$\boldsymbol{X} = \boldsymbol{W}\left(\boldsymbol{Z} \odot \boldsymbol{F}\right) + \boldsymbol{E}, \tag{3.4.2}$$

wherein $\odot$ denotes as the Hadamard product, $\boldsymbol{Z}$ is a newly added binary matrix

whose nonparametric prior distribution follows an Indian Buffet Process (IBP) (Griffiths and Ghahramani, 2006). Its binary values indicates the presence or absence of a factor (i.e. a column of matrix $\boldsymbol{W}$) and the matrix $\boldsymbol{F}$ contains the coefficients when working with matrix $\boldsymbol{Z}$. Formally, $Z_{kn} = 1$ implies that the $k$-th factor is used while reconstructing the $n$-th data vector, i.e. $n$-th column of the matrix $\boldsymbol{X}$. In this nonparametric model, $\boldsymbol{Z}$ is modelled through a draw from Beta process which allows infinitely many factors. Given the data, the number of active factors[2] are automatically discovered using the inference procedure.

The distributions on the parameters $\boldsymbol{W}, \boldsymbol{F}$ of the above nonparametric model is as

$$W_{mk} \sim \text{Gamma}\left(a_w, b_w\right), \ \boldsymbol{F}_i \sim \Pi_{k=1}^{K} \text{Gamma}\left(a_F, b_F\right), \tag{3.4.3}$$

where $a_w$, $b_w$, $a_F$ and $b_F$ are the shape and scale parameters. Similarly, given the parameters, the data is modelled using a Poisson distribution in the following manner

$$X_i \mid \boldsymbol{X}, Z_i, F_i \sim \text{Poisson}\left(\boldsymbol{X}\left(Z_i \odot F_i\right) + \lambda\right), \tag{3.4.4}$$

where $\lambda$ is a parameter which expresses modelling error $\boldsymbol{E}$ such that $E_{mn} \sim \text{Poisson}\left(\lambda\right)$.

We use Gibbs sampling to infer $\boldsymbol{W}$ and $\boldsymbol{F}$. We introduce the auxiliary variables to make the inference become tractable. For example, the Gibbs update equation for $i$-th row of $\boldsymbol{W}$, denoted by $\boldsymbol{W}_{(i)}$, is given as:

$$\begin{aligned} p(\boldsymbol{W}_{(i)}|\boldsymbol{Z}, \boldsymbol{F}, \boldsymbol{X}, \lambda, \mathbf{s}) \propto \Pi_{k=1}^{K} \left(W_{ik}\right)^{a+\sum_{j=1}^{T} s_j^{ik}-1} \\ \times \exp\left\{-\left(b + \sum_{j=1}^{T} H_{kj}\right) W_{ik}\right\}, \end{aligned} \tag{3.4.5}$$

where the auxiliary variables $\mathbf{s} = \left\{s_j^{ik}\right\}_{k=1}^{K+1}$ can be sampled from a Multinomial distribution for each $j \in \{1, \ldots, T\}$ satisfying $\sum_{k=1}^{K+1} s_j^{ik} = 1$:

$$p\left(s_j^{i1}, \ldots s_j^{iK}, s_j^{i(K+1)} \mid \cdot\right) \propto \frac{X_{ij}!}{\prod_{k=1}^{K+1} s_j^{ik}!} \Pi_{k=1}^{K} \left(W_{ik} H_{kj}\right)^{s_j^{ik}} \lambda^{s_j^{i(K+1)}}. \tag{3.4.6}$$

The matrix $\boldsymbol{F}$ and $\boldsymbol{Z}$ can also be sampled in a similar manner proposed in (Teh et al., 2007).

[2]e.g., $k$-th factor is an active factor, if $k$-th row of the matrix $Z$ has at least one non-zero entry.

### 3.4.4 Browsing functionalities

Using the latent factors $\boldsymbol{W}$ across video frames (or group level in multilevel structure) learning in the previous steps, we propose the following functionalities for our system.

#### 3.4.4.1 Discovering rare factors and footages

For each factor $W_k$ within $K$ factors discovered in the previous step, we define a score to measure its rareness based on its overall contribution to the scene. Since $X_j = \sum_k W_k H_{kj}$, it is clear that $H_{kj}$ is the contribution of factor $W_k$ to reconstruct $X_j$. Hence, we have the term of $\sum_j H_{kj}$ is the overall contribution of factor $k$ to $\boldsymbol{X}$. We define the rareness score of a factor as a function reciprocal to this quantity:

$$\text{r-score}\left(W_k\right) = -\log\left(\sum_j H_{kj}\right). \tag{3.4.7}$$

In our system, we rank the scores for those factors learned in Section 3.4.3 using Eq. 3.4.7 and allows the user to interactively choose the percentage $\alpha$ of rare factors to be displayed and interacted with (cf. Fig. 3.5.3 and Fig. 3.4.1.A). The list of footages, a footage is a group level in multilevel setting, associated with this factor is also returned to the user (cf. Fig. 3.4.1.G). Denote $S\left(W_k\right)$ as the corresponding index set, then:

$$S\left(W_k\right) = \left\{j \mid H_{kj} > \epsilon, j = 1, \ldots, T\right\}, \tag{3.4.8}$$

where $\epsilon$ is a small threshold, mainly used for the stability of the algorithm. Further, let $K_\alpha$ be the collection of all rare factors, then the index set of all detected footages is:

$$\mathcal{F}_\alpha = \bigcup_{W \in K_\alpha} S\left(W\right). \tag{3.4.9}$$

#### 3.4.4.2 Spatial searching

Given a spatial region of interest $R$ being input to the system, spatial filtering on rare events can now be efficiently carried out by analysing the intersecting region between the spatial region $R$ and the set of rare factors $W$. We note that the spatial region $R$ contains one or multiple blocks (individual level) in multilevel setting. First we extend $R$ to $R'$ to have the full size of the video frame by zero padding and mask it with each rare factor $W$ which will be selected if the resultant matrix is non-zero. Let $\mathrm{SP}_\alpha(R)$ be the set of output indices returned, then formally we have:

$$\mathrm{SP}_\alpha(R) = \bigcup_{W \in \mathrm{SPF}(K_\alpha, R)} S(W) \quad \text{where}$$
$$\mathrm{SPF}(K_\alpha, R) = \{W \mid W \in K_\alpha, ||W \odot R'||_0 > 0\}.$$

Here, $\alpha$ is a percentage of rareness degree in as described in Section 3.4.4.1 and $\odot$ is element-wise multiplication, $||A||_0$ is the $l_0$-norm which counts the number of non-zero elements in the matrix $A$. The demonstration of this browsing capacity is shown in Fig. 3.5.3 which reveals that the security officer can scrutinize the red rectangle region in the left window to inspect any unusual things happened in the right panel such as an event that one person is crossing the street.

#### 3.4.4.3 Spatial-temporal searching

More significantly, the spatial-temporal criteria searching is included in our model in Fig. 3.5.3. The semantic can be understood as "show me the events here (red rectangle) followed by the events there (blue rectangle)" that is set temporally as within $\Delta t$ seconds. Once again our filters extracted the frames data into the potential candidates for rare frames (group level). Initially, an user indicates a queue region of interest at individual level. For this purpose, we illustrate them into two regions, say red and blue rectangle. Spatial scanning in previous section will be applied into both rectangles. Those output patterns are considered as the necessary input for this process. In accordance with the mathematical formula in Eq. 3.4.10, the typical

illustration of this searching category can be found in Fig. 3.5.3.

$$\mathrm{ST}_{\alpha}\left(R_1, R_2\right) = \left\{(i,j) \mid i \in \mathrm{SP}_{\alpha}\left(R_1\right), j \in \mathrm{SP}_{\alpha}\left(R_2\right), |i-j| < \Delta t\right\}. \tag{3.4.10}$$

## 3.5 Experiment

In this experiment, we first demonstrate quantitatively the abnormality detection performance, then present the user interface system.

### 3.5.1 Quantitative Experiment

We use a 14 day long video footage from an existing surveillance camera mounted at a street junction overlooking a busy thoroughfare. For each hour, we have 14 separate observations from each of the 14 days - this is used as the input matrix for the iHMM inference. The total number of Gibbs iterations performed for the inference is 1500, with 500 burnings. An example of the discovered segmentation is shown in Fig 3.5.1. We discover two segments including 8.00am - 8.59pm, and 9.00pm - 7.59am. The total running time is 10.58 sec on the X5690 based server.

We next show why such data segmentation improves downstream processes like anomaly detection. We divide the data into two parts. The first 7 days are used for training, i.e. computing residual subspace and the detection threshold set. The detection threshold ($\lambda$) is set at 0.1%. The remaining 7 days of video are used for testing, i.e. projecting each feature vector onto the residual space and declaring an anomaly if the projected energy in the residual space exceeds $\lambda$. We run two anomaly detectors: 1) The uni-model, that runs on the whole data, and 2) The multi-model, catering to multiple modes for the segmented hours as obtained by iHMM, with separate anomaly detectors for each mode.

The energy distribution of the test vectors in the residual subspace for the two settings are shown in Fig. 3.5.2a. The energy distribution for the multi-model decays more sharply, and thus an application of detection threshold will not 'leak' normal events as anomalous ones. Fig. 3.5.2b shows the energy signal for a chain of

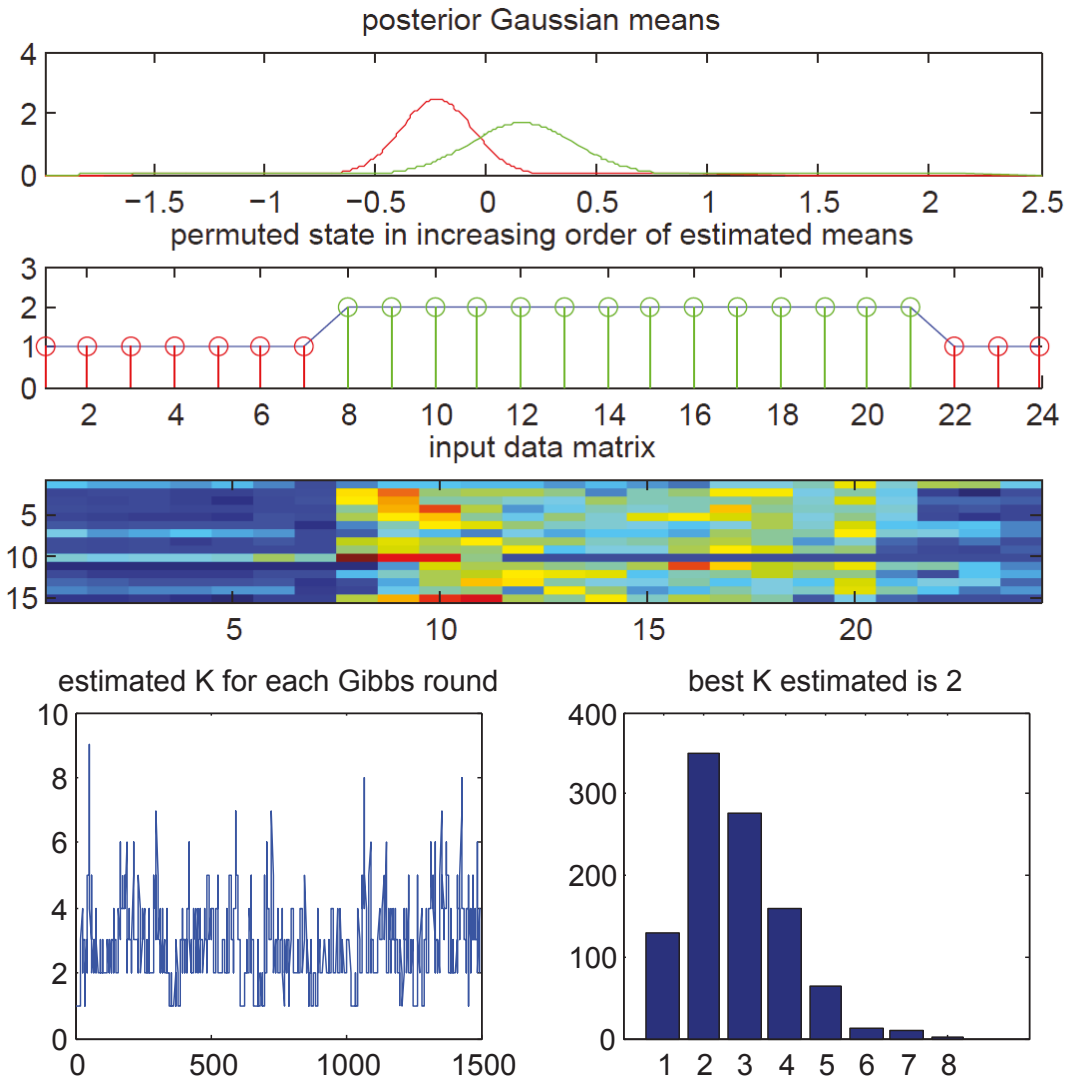
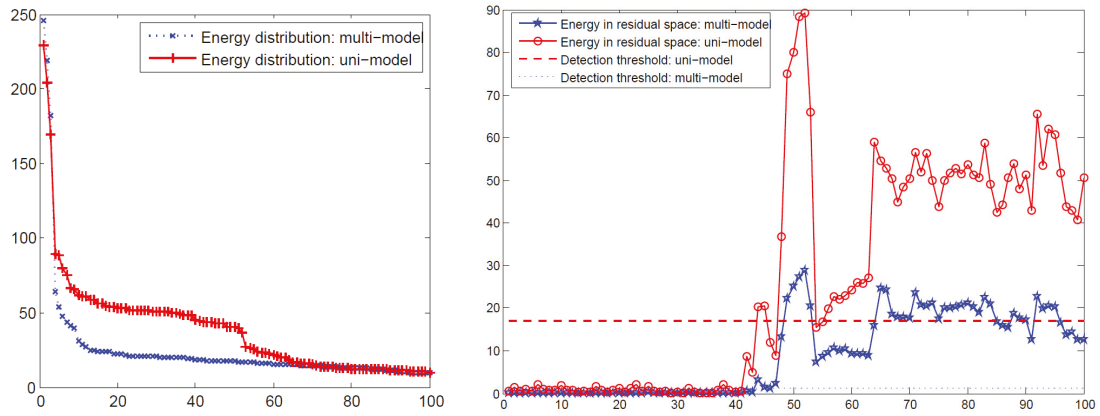



Figure 3.5.1: Example of of iHMM segmentation for 1-day data.

anomalous events - a street fight followed by police intervention. It shows that whilst the overall projection energy is higher for the uni-model, the detection threshold is also much higher, resulting in missed events (between frames 40-45 of Fig. 3.5.2a, for example). For the multi-model, the detection threshold is low, and the energy for this entire period remains above the detection threshold.

This effect is illustrated quantitatively in Table 3.1 which shows the number of events detected by both set-ups. The multi-model is more effective than the uni-model – detecting more loitering events (all of which occur at night, and thus are missed by the uni-model) and the full sequence of events in the street fight period. Incidentally, both models declared one (different) event as anomalous, which we consider a false




(a) Energy distribution in the residual subspace.

(b) Residual signals

Figure 3.5.2: Comparative signal in residual space

| Event type | # detected (uni-model) | # detected (multi-model) |
|---|---|---|
| Street fight | 57 | 63 |
| Loitering | 1 | 7 |
| Truck-unusual stopping | 4 | 4 |
| Big truck blocking camera | 2 | 2 |
| No apparent reason | 1 | 1 |

Table 3.1: Description of anomalous events.

positive. For both models, the training and testing of the total 14 days of video were achieved in less than 0.5 sec.

### 3.5.2 User Interface Demonstration

Next, we demonstrate the proposed system using the MIT dataset (Wang et al., 2008). In this public dataset, the traffic scene are recorded by the static camera, especially the traffic flows such as truck, car, pedestrian, bicycle, and other noisy motions such as leaves flickering due to wind etc. These objects generate various motion patterns in the intersection area of the traffic scene. The image dimension of the traffic scene is 480×720 pixel per frame (cf. Fig. 3.5.5). As mentioned earlier in Section 3.4.2, static cameras own the rank-1 property which is the necessary condition for our background subtraction task.

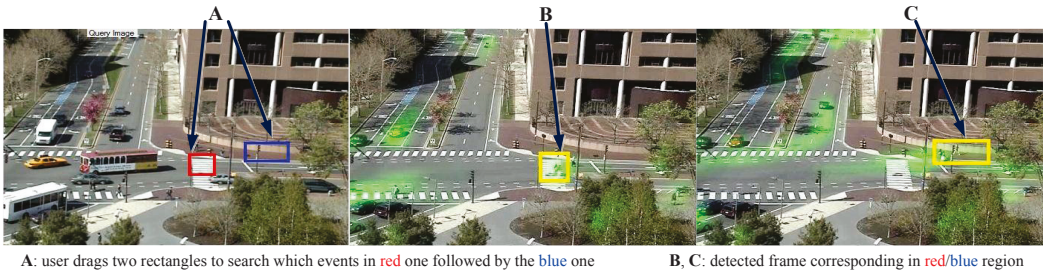



Figure 3.5.3: Example of spatial-temporal browsing. User draw two rectangles: red and blue to find the abnormal incident that turn up at blue area followed the one in red section. The system caught the pedestrian which is compatible with his motion direction in this zones.

For the motion feature extraction stage, we choose the block size of 20x20 and a sequential footage of $n = 200$. In order to deal with matrix factorization problems when we do not know the number of latent factors beforehand, one possible solution is to do model selection by varying the number of latent factor $K$. The visualization of the model selection step is depicted in Fig. 3.4.1, in which we restrain the parameter scope from 20 to 56. Using our nonparametric model, however, the parameter $K$ is automatically identified as 40. From 40 learned patterns (cf. Fig. 3.5.4), we sort all in an increasing order of rareness amount that is explained in Section 3.4.4.1. For example, three candidates for common factors and three rare factors are shown in Fig. 3.5.5.

We establish the browsing paradigm by assisting users to restrict their searching region by spatial and spatial-temporal criteria. One typical example is presented in Fig. 3.5.3. A user draws two regions: red and blue rectangles to investigate which patterns will followed by others in those windows. Initially, the system will automatically detects suitable candidate patterns in those regions with regard to the proportion of rareness level that user are querying. Through the candidate factors, we will reverse to all the consecutive frames (each frame is a group comprising of individuals as blocks) and clips associated with the selected factors. Then, the most appropriate event will be discovered following Eq. 3.4.10. In Fig. 3.5.3, people who cross the zebra-crossing (red rectangle) and turn right (blue rectangle) are caught by our system.

One false positive is also recorded. Because of the big traffic flow in the period of $n = 200$ serial frames in the selected rectangle, the system will treat it as an abnormal episode. When a user draws a spatial interrogation in this area, the

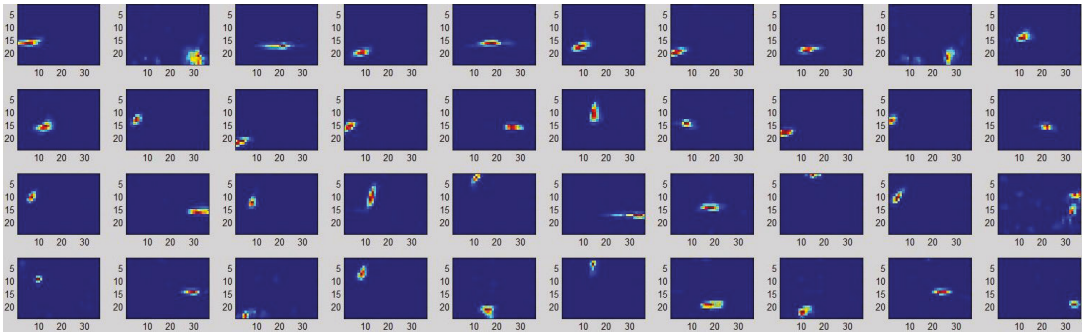


Figure 3.5.4: Factors learned from MIT dataset, shown from top-left to right-bottom in the increasing order of their 'rareness'.

machine will give back this flow as a possible candidate for abnormality. However, the user can control, fortunately, the rareness level and alter it following the true abnormality semantically in the scene. Concerning with the input rareness rate, multiple patterns and clips are discovered so that the user can decide which one is a real affair. Thus, our proposed framework surmounts successfully the semantic gap between the statistical perspective and human perception.

Our focus is on browsing interactively the abnormal activities locally in a scene for multilevel surveillance data. There is no such existing interactive system available for comparison. Moreover, the difficult thing in evaluating our experimental results for interactivity is that there is no suitable ground truth which can satisfy all of user spatial and temporal queries. Because a user can examine in different locations: top left, right bottom, or middle region, and with different window sizes and time interval. For that reason, the quantitative evaluation of our abnormality detection approach can be referred to Section 3.5.1.

This system was programmed in C# and Matlab. The experiment was running on a PC Intel Core i7 3.4 GHz, with 8GB RAM. A query system took approximately less than 0.2 second, as the motion feature extraction step was preprocessed. As mentioned, the rare patterns are understood as human perception, so we select roughly $p = 10\%$ for the number of rare events that the user can slide the bar to alter the number of rare events following their interests.

Figure 3.5.5: Illustration of our learned factors overlaid with the data from MIT dataset. The left column presents three common patterns. Three rare factors are displayed in the right column.

## 3.6 Closing Remark

Identifying the meaningfully anomalous events in video surveillance are essential to security management. In this chapter, we address the problem of abnormality detection in video surveillance data using Bayesian nonparametric methods. The video surveillance data are constructed in multilevel setting. In the first multilevel setting, video recorded in each day is a group while video recorded in each hour within a day is individual. In the second setting, video frame is considered as a group and video frame block is as individual. We propose a framework for nonparametric data segmentation and multi-modal abnormality detection. By building multiple abnormality detection models on different coherent sections of the stream data, our proposed framework is more robust for abnormality detection in a large scale video data. Especially, when the video cameras are monitored across many days and exhibit strong variations in the data. Our experiments on a collection of video data over 14 days has demonstrated the superior performance of the proposed multi-modal anomaly detector compared to uni-model detectors.

In addition, we have address the problem of interactive monitoring in video surveillance, allowing users to examine the rare events. The anomalous and rare events

are detected in an unsupervised manner and can be filtered out interactively. We establish the browsing paradigm with spatial and temporal-spatial approaches to overcome the limitation of pure computational processing.

In the next chapter, we present a deeper theoretical contribution to Bayesian non-parametric that we consider the multilevel clustering problem when the number of clusters in the grouped data are not known.

# Chapter 4

# Bayesian Nonparametric Multilevel Clustering with Group-Level Contexts

In the previous chapter, we have presented an exposition into abnormality detection in video surveillance using Bayesian nonparametric methods. In this chapter, we delve deeper into Bayesian nonparametric modelling and present a new nonparametric framework for multilevel clustering namely *Multilevel Clustering with Context* ($MC^2$) which utilises group-level context information to simultaneously discover low-dimensional structures of the group contents and partitions groups into clusters. Particularly, we jointly cluster both the content data and their groups when there is group-level context information. By *context,* we mean a secondary data source attached to the group of primary *content* data. An example is the problem of multielvel clustering documents and words (nested in documents), where each document is a group of words associated with group-level context information such as time-stamps, list of authors, etc.

Using the Dirichlet process as the building block, our model constructs a product base-measure with a nested structure to accommodate content and context observations at multiple levels. The proposed model possesses properties that link the nested Dirichlet processes (nDP) and the Dirichlet process mixture models (DPM) in an interesting way: integrating out all contents results in the DPM over con-

texts, whereas integrating out group-specific contexts results in the nDP mixture over content variables. We provide a Polya-urn view of the model and an efficient collapsed Gibbs inference procedure. Extensive experiments on real-world datasets demonstrate the advantage of utilising context information via our model in both text and image domains.

The advantages of our proposed solution are: (1) the model automatically discovers the (unspecified) number of groups clusters and the number of topics while fully utilising the context information; (2) content topic modelling is informed by group-level context information, leading to more predictive content topics; (3) the model is robust to partially missing context information. In our experiments, we demonstrate that our proposed model achieves better document clustering performances and more predictive word topics in real-world datasets in both text and image domains.

This remainder of this chapter is organised as follows. We present our overview on multilevel clustering when the data are organised into groups in Section 4.1. In Section 4.2, we describes the related background to our work. Section 4.3 presents our proposed model. We perform extensive experiments in Section 4.4.

## 4.1 Multilevel Clustering

In many situations, content data naturally present themselves in groups, e.g., students are grouped into classes, classes grouped into schools, words grouped into documents, etc. Furthermore, each content group can be associated with additional context information (teachers of the class, authors of the document, time and location stamps). Dealing with grouped data, a setting known as *multilevel analysis* (Hox, 2010; Diez-Roux, 2000), has diverse application domains ranging from document modelling (Blei et al., 2003) to public health (Leyland and Goldstein, 2001b).

This chapter considers specifically the multilevel clustering problem in multilevel analysis: to jointly cluster both the content data and their groups when there is group-level context information. By *context,* we mean a secondary data source attached to the group of primary *content* data. An example is the problem of clustering documents, where each document is a group of words associated with

group-level context information such as time-stamps, list of authors, etc. Another example is image clustering where visual image features (e.g., SIFT) are the content and image tags are the context.

To cluster groups together, it is often necessary to perform dimensionality reduction of the content data by forming content topics, effectively performing clustering of the content as well. For example, in document clustering, using bag-of-words directly as features is often problematic due to the large vocabulary size and the sparsity of the in-document word occurrences. Thus, a typical approach is to first apply dimensionality reduction techniques such as LDA (Blei et al., 2003) or HDP (Teh et al., 2006) to find word topics (i.e., distributions on words), then perform document clustering using the word topics and the document-level context information as features. In such a cascaded approach, the dimensionality reduction step (e.g., topic modelling) is not able to utilise the context information. This limitation suggests that a better alternative is to perform context-aware document clustering and topic modelling jointly. With a joint model, one can expect to obtain improved document clusters as well as context-guided content topics that are more predictive of the data.

Recent work has attempted to jointly capture word topics and document clusters. Parametric approaches (Xie and Xing, 2013) are extensions of the LDA (Blei et al., 2003) and require specifying the number of topics and clusters in advance. Bayesian nonparametric approaches including the nested Dirichlet process (nDP) (Rodriguez et al., 2008) and the multi-level clustering hierarchical Dirichlet Process (MLC-HDP) (Wulsin et al., 2012) can automatically adjust the number of clusters. We note that none of these methods can utilise context data.

We propose in this chapter the *Multilevel Clustering with Context* (MC$^2$), a Bayesian nonparametric model to jointly cluster both content and groups while fully utilising group-level context. Using the Dirichlet process as the building block, our model constructs a product base-measure with a nested structure to accommodate both content and context observations. The MC$^2$ model possesses properties that link the nested Dirichlet process (nDP) and the Dirichlet process mixture model (DPM) in an interesting way: integrating out all contents results in the DPM over contexts, whereas integrating out group-level context results in the nDP mixture over content variables. For inference, we provide an efficient collapsed Gibbs sampling procedure for the model.

## 4.2 Related Background

There have been extensive works on clustering documents in the literature. Due to limited scope of the chapter, we only describe works closely related to probabilistic topic models. We note that standard topic models such as LDA (Blei et al., 2003) or its nonparametric Bayesian counterpart, HDP (Teh et al., 2006) exploits the group structure for word clustering. However these models do not cluster documents.

An approach to document clustering is to employ a two-stage process. First, topic models (e.g., LDA or HDP) are applied to extract the topics and their mixture proportion for each document. Then, this is used as feature input to another clustering algorithm. Some examples of this approach include the use of LDA+Kmeans for image clustering (Xuan et al., 2011; Elango and Jayaraman, 2005) and HDP+Affinity Propagation for clustering human activities (Nguyen et al., 2013b).

A more elegant approach is to simultaneously cluster documents and discover topics. The first Bayesian nonparametric model proposed for this task is the nested Dirichlet Process (nDP) (Rodriguez et al., 2008) where documents in a cluster share the same distribution over topic atoms. Although the original nDP does not force the topic atoms to be shared across document clusters, this can be achieved by simply introducing a DP prior for the nDP base measure. The same observation was also made by (Wulsin et al., 2012) who introduced the MLC-HDP, a 3-level extension to the nDP. This model thus can cluster words, documents and document-corpora with shared topic atoms throughout the group hierarchy. Xie and Xing (2013) recently introduced the Multi-Grain Clustering Topic Model which allows mixing between global topics and document-cluster topics. However, this is a parametric model which requires fixing the number of topics in advance. More crucially, all of these existing models do not attempt to utilise group-level context information.

### Modelling with Dirichlet Process

We have provided a general background in Bayesian nonparametrics in Chapter 2, to provide the context we recall a brief account on the Dirichlet process and its variants here.

*Dirichlet process* (Ferguson, 1973) is a basic building block in Bayesian nonparametric. Let $(\Theta, \mathcal{B}, H)$ be a probability measure space, and $\gamma$ is a positive number, a Dirichlet process $\mathrm{DP}(\gamma, H)$ is a distribution over discrete random probability measure $G$ on $(\Theta, \mathcal{B})$. Sethuraman (1994) provides an alternative constructive definition which makes the discreteness property of a draw from a Dirichlet process explicit via the stick-breaking representation: $G = \sum_{k=1}^{\infty} \beta_k \delta_{\phi_k}$ where $\phi_k \overset{\text{iid}}{\sim} H, k = 1, \ldots, \infty$ and $\boldsymbol{\beta} = (\beta_k)_{k=1}^{\infty}$ are the weights constructed through a 'stick-breaking' process $\beta_k = v_k \prod_{s<k} (1 - v_s)$ with $v_k \overset{\text{iid}}{\sim} \mathrm{Beta}(1, \gamma)$. It can be shown that $\sum_{k=1}^{\infty} \beta_k = 1$ with probability one, and as a convention (Pitman, 2002), we hereafter write $\boldsymbol{\beta} \sim \mathrm{GEM}(\gamma)$.

Due to its discrete nature, Dirichlet process has been widely used in Bayesian mixture models as the prior distribution on the mixing measures, each is associated with an atom $\phi_k$ in the stick-breaking representation of $G$ above. A likelihood kernel $F(\cdot)$ is used to generate data $x_i \mid \phi_k \overset{\text{iid}}{\sim} F(\cdot \mid \phi_k)$, resulting in a model known as the *Dirichlet process mixture model* (DPM), pioneered by the work of (Antoniak, 1974) and subsequently developed by many others. In section 4.3 we provide a precise definition for DPM.

While DPM models exchangeable data within a *single* group, the Dirichlet process can also be constructed hierarchically to provide prior distributions over *multiple* exchangeable groups. Under this setting, each group is modelled as a DPM and these models are 'linked' together to reflect the dependency among them – a formalism which is generally known as dependent Dirichlet processes (MacEachern, 1999). One particular attractive approach is the *hierarchical Dirichlet processes* (Teh et al., 2006) which posits the dependency among the group-level DPM by another Dirichlet process, i.e., $G_j \mid \alpha, G_0 \sim \mathrm{DP}(\alpha, G_0)$ and $G_0 \mid \gamma, H \sim \mathrm{DP}(\gamma, H)$ where $G_j$ is the prior for the $j$-th group, linked together via a discrete measure $G_0$ whose distribution is another DP.

Yet another way of using DP to model multiple groups is to construct random measure in a nested structure in which the DP base measure is itself another DP. This formalism is the *nested Dirichlet Process* (Rodriguez et al., 2008), specifically $G_j \overset{\text{iid}}{\sim} U$ where $U \sim \mathrm{DP}(\alpha \times \mathrm{DP}(\gamma H))$. modelling $G_j$ (s) hierarchically as in HDP and nestedly as in nDP yields different effects. HDP focuses on exploiting statistical strength across groups via sharing atoms $\phi_k$ (s), but it does not partition groups

into clusters. This statement is made precisely by noting that $P(G_j = G_{j'}) = 0$ in HDP. Whereas, nDP emphasises on inducing clusters on both observations and distributions, hence it partitions groups into clusters. To be precise, the prior probability of two groups being clustered together is $P(G_j = G_{j'}) = \frac{1}{a+1}$. Finally we note that this original definition of nDP in (Rodriguez et al., 2008) does not force the atoms to be shared across clusters of groups, but this can be achieved by simply introducing a DP prior for the nDP base measure, a modification that we use in this chapter. This is made clearly in our definition for nDP mixture in section 4.3.

## 4.3 The Proposed Framework

In this section, we describe the proposed model description and stick-breaking representation. Then, we present the posterior inference and marginalization property.

### 4.3.1 Model description and stick-breaking

Consider data presented in a two-level group structure as follows. Denote by $J$ the number of groups; each group $j$ contains $N_j$ exchangeable data points, represented by $\mathbf{w}_j = \{w_{j1}, w_{j2}, \ldots, w_{jN_j}\}$. For each group $j$, the group-specific context data is denoted by $x_j$. Assuming that the groups are exchangeable, the overall data is $\{(x_j, \mathbf{w}_j)\}_{j=1}^{J}$. The collection $\{\mathbf{w}_1, \ldots, \mathbf{w}_J\}$ represents observations of the group contents, and $\{x_1, \ldots, x_J\}$ represents observations of the group-level contexts.

We now describe the generative process of MC² that generates a two-level clustering of this data. We use a group-level DP mixture to generate an infinite cluster model for groups. Each group cluster $k$ is associated with an atom having the form of a pair $(\phi_k, Q_k^*)$ where $\phi_k$ is a parameter that generates the group-level contexts within the cluster and $Q_k^*$ is a measure that generates the group contents within the same cluster.

To generate atomic pairs of context parameter and measure-valued content parameter, we introduce a product base-measure of the form $H \times \mathrm{DP}(vQ_0)$ for the group-level DP mixture. Drawing from a DP mixture with this base measure, each

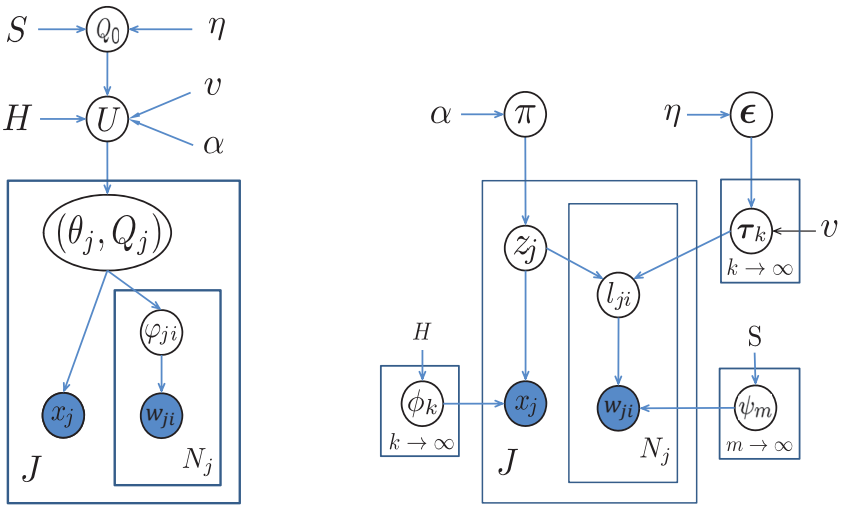



Figure 4.3.1: Graphical model representation for the proposed model. Right Fig. illustrates a stick breaking representation.

realisation is a pair $(\theta_j, Q_j)$; $\theta_j$ is then used to generate the context $x_j$ and $Q_j$ is used to repeatedly produce the set of content observations $w_{ji}$ within the group $j$. Specifically,

$$\begin{aligned} U \sim \mathrm{DP}\left(\alpha(H \times \mathrm{DP}(\upsilon Q_0))\right) \text{ where } Q_0 \sim \mathrm{DP}\left(\eta S\right) \\ (\theta_j, Q_j) \overset{\text{iid}}{\sim} U \text{ for each group } j \\ x_j \sim F(.|\theta_j), \quad \varphi_{ji} \overset{\text{iid}}{\sim} Q_j, \quad w_{ji} \sim Y\left(.|\varphi_{ji}\right). \end{aligned} \tag{4.3.1}$$

In the above, $H$ and $S$ are respectively base measures for context and content parameters $\theta_j$ and $\varphi_{ji}$. The context and content observations are then generated via the likelihood kernels $F\left(\cdot \mid \theta_j\right)$ and $Y\left(\cdot \mid \varphi_{ji}\right)$. To simplify inference, $H$ and $S$ are assumed to be conjugate to $F$ and $Y$ respectively. The generative process is illustrated in Fig. 4.3.1.

### Stick-breaking representation

We now derive the stick-breaking construction for MC$^2$ where all the random discrete measures are specified by a distribution over integers and a countable set of atoms. The random measure $U$ in Eq. (4.3.1) has the stick-breaking form:

$$U = \sum_{k=1}^{\infty} \pi_k \delta_{(\phi_k, Q_k^*)} \tag{4.3.2}$$

where $\pi \sim \text{GEM}(\alpha)$ and $(\phi_k, Q_k^*) \overset{\text{iid}}{\sim} H \times \text{DP}(vQ_0)$. Equivalently, this means $\phi_k$ is drawn i.i.d. from $H$ and $Q_k^*$ drawn i.i.d. from $\text{DP}(vQ_0)$. Since $Q_0 \sim \text{DP}(\eta S)$, $Q_0$ and $Q_k^*$ have the standard HDP (Teh et al., 2006) stick-breaking forms: $Q_0 = \sum_{m=1}^{\infty} \epsilon_m \delta_{\psi_m}$where $\boldsymbol{\epsilon} \sim \text{GEM}(\eta)$, $\psi_m \overset{\text{iid}}{\sim} S$; $Q_k^* = \sum_{m=1}^{\infty} \tau_{k,m} \delta_{\psi_m}$ where $\boldsymbol{\tau}_k = (\tau_{k1}, \tau_{k2}, \ldots) \sim \text{DP}(v, \boldsymbol{\epsilon})$.

For each group $j$ we sample the parameter pair $(\theta_j, Q_j) \overset{\text{iid}}{\sim} U$; equivalently, this means drawing $z_j \overset{\text{iid}}{\sim} \pi$ and letting $\theta_j = \phi_{z_j}$ and $Q_j = Q_{z_j}^*$. For the $i$-th content data within the group $j$, the content parameter $\varphi_{ji}$ is drawn $\overset{\text{iid}}{\sim} Q_j = Q_{z_j}^*$; equivalently, this means drawing $l_{ji} \overset{\text{iid}}{\sim} \tau_{z_j}$ and letting $\varphi_{ji} = \psi_{l_{ji}}$. Fig. 4.3.1 presents the graphical model of this stick-breaking representation.

### 4.3.2 Inference and Polya Urn View

We provide detailed derivations for model inference with the graphical model displayed in Fig. 4.3.1. The variables $\phi_k$, $\psi_m$, $\pi$, $\tau_k$ are integrated out due to conjugacy property. We need to sample these latent variables $z$, $l$, $\epsilon$ and hyper parameters $\alpha$, $v$, $\eta$. For convenience of notation, we denote $z_{-j}$ is a set of latent context variable $z$ in all documents excluding document $j$, $\boldsymbol{l}_{j*}$ is all of hidden variables $l_{ji}$ in document $j$, and $\boldsymbol{l}_{-j*}$ is all of $l$ in other documents rather than document $j$-th.

**Sampling $\boldsymbol{z}$.** Sampling context index $z_j$ needs to take into account the influence of the corresponding context topics:

$$\begin{aligned} p(z_j = k \mid \mathbf{z}_{-j}, \boldsymbol{l}, \mathbf{x}, \alpha, H) \propto & \underbrace{p(z_j = k \mid \mathbf{z}_{-j}, \alpha)}_{\text{CRP for context topic}} \underbrace{p(x_j \mid z_j = k, \mathbf{z}_{-j}, \mathbf{x}_{-j}, H)}_{\text{context predictive likelihood}} \\ & \times \underbrace{p(\boldsymbol{l}_{j*} \mid z_j = k, \boldsymbol{l}_{-j*}, \mathbf{z}_{-j}, \epsilon, v)}_{\text{content latent marginal likelihood}}. \end{aligned} \tag{4.3.3}$$

The first term can easily be recognized as a form of Chinese Restaurant Process (CRP):

$$p\left(z_j = k \mid \mathbf{z}_{-j}, \alpha\right) = \begin{cases} \frac{n_{-j}^k}{n_{-j}^* + \alpha} & \text{if } k\text{is previously used} \\ \frac{\alpha}{n_{-j}^* + \alpha} & \text{if } k\text{is new} \end{cases}$$

where $n_{-j}^k$ is the number of data $z_j = k$ excluding $z_j$, and $n_{-j}^*$ is the count of all $\mathbf{z}$, except $z_j$.

The second expression is the predictive likelihood from the context observations under the context component $\phi_k$. Specifically, let $f\left(\cdot \mid \phi\right)$ and $h\left(\cdot\right)$ be respectively the density function for $F\left(\phi\right)$ and $H$, the conjugacy between $F$ and $H$ allows us to integrate out the mixture component parameter $\phi_k$ , leaving us the conditional density of $x_j$ under the mixture component $k$ given all the context data items exclude $x_j$:

$$\begin{aligned} p\left(x_j \mid z_j = k, \mathbf{z}_{-j}, \mathbf{x}_{-j}, H\right) =& \frac{\int_{\phi_k} f\left(x_j \mid \phi_k\right) \prod\limits_{j' \neq j, z_{j'} = k} f\left(x_{j'} \mid \phi_k\right) h\left(\phi_k\right) d\phi_k}{\int_{\phi_k} \prod\limits_{j' \neq j, z_{j'} = k} f\left(x_{j'} \mid \phi_k\right) h\left(\phi_k\right) d\phi_k} \\ =& f_k^{-x_j}\left(x_j\right). \end{aligned}$$

Finally, the last term is the contribution from the multiple latent variables of corresponding topics to that context. Since $l_{ji} \mid z_j = k \overset{\text{iid}}{\sim} \text{Mult}\left(\boldsymbol{\tau}_k\right)$ where $\boldsymbol{\tau}_k \sim \text{Dir}\left(v\epsilon_1, \ldots, v\epsilon_M, \epsilon_{\text{new}}\right)$, we shall attempt to integrate out $\boldsymbol{\tau}_k$ . Using the Multinomial-Dirichlet conjugacy property we proceed to compute the last term in Eq. 4.3.3 as follows:

$$p\left(\boldsymbol{l}_{j*} \mid z_j = k, \mathbf{z}_{-j}, \boldsymbol{l}_{-j*}, \epsilon, v\right) = \int_{\boldsymbol{\tau}_k} p\left(\boldsymbol{l}_{j*} \mid \boldsymbol{\tau}_k\right) \times p\left(\boldsymbol{\tau}_k \mid \left\{l_{j'*} \mid z_{j'} = k, j' \neq j\right\}, \epsilon, v\right) d\boldsymbol{\tau}_k \tag{4.3.4}$$

Recognizing the term $p\left(\boldsymbol{\tau}_k \mid \left\{\boldsymbol{l}_{j'*} \mid z_{j'} = k, j' \neq j\right\}, \boldsymbol{\epsilon}, v\right)$ is a posterior density, it is Dirichlet-distributed with the updated parameters

$$p\left(\boldsymbol{\tau}_k \mid \left\{\boldsymbol{l}_{j'*} \mid z_{j'} = k, j' \neq j\right\}\right) = \text{Dir}\left(v\epsilon_1 + c_{k,1}^{-j}, \ldots, v\epsilon_M + c_{k,M}^{-j}, v\epsilon_{\text{new}}\right) \tag{4.3.5}$$

where $c_{k,m}^{-j} = \sum_{j' \neq j} \sum_{i=1}^{N_{j'}} \mathbb{I}\left(l_{j'i} = m, z_{j'} = k\right)$ is the count of topic $m$ being assigned

to context $k$ excluding document $j$. Using this result, $p\left(\boldsymbol{l}_{j*} \mid \boldsymbol{\tau}_k\right)$ is a predictive likelihood for $\boldsymbol{l}_{j*}$ under the posterior Dirichlet parameters $\boldsymbol{\tau}_k$ in Eq. 4.3.5 and therefore can be evaluated to be:

$$
\begin{aligned}
p\left(\boldsymbol{l}_{j*} \mid z_j = k, ...\right) &= \int_{\tau_k} p\left(\boldsymbol{l}_{j*} \mid \boldsymbol{\tau}_k\right) \times \mathrm{Dir}\left(v\epsilon_1 + c_{k,1}^{-j}, \ldots, v\epsilon_M + c_{k,M}^{-j}, v\epsilon_{\mathrm{new}}\right) d\boldsymbol{\tau}_k \\
&= \int_{\tau_k} \prod_{m=1}^{M} \tau_{k,m}^{c_{k,m}^{j}} \times \frac{\Gamma\left(\sum_{m=1}^{M}\left(v\epsilon_m + c_{k,m}^{-j}\right)\right)}{\prod_{m=1}^{M}\Gamma\left(v\epsilon_m + c_{k,m}^{-j}\right)} \times \prod_{m=1}^{M} \tau_{k,m}^{v\epsilon_m + c_{k,m}^{-j} - 1} d\boldsymbol{\tau}_k \\
&= \frac{\Gamma\left(\sum_{m=1}^{M}\left(v\epsilon_m + c_{k,m}^{-j}\right)\right)}{\prod_{m=1}^{M}\Gamma\left(v\epsilon_m + c_{k,m}^{-j}\right)} \times \int_{\tau_k} \prod_{m=1}^{M} \tau_{k,m}^{v\epsilon_m + c_{k,m}^{-j} + c_{k,m}^{j} - 1} d\boldsymbol{\tau}_k \\
&= \frac{\Gamma\left(\sum_{m=1}^{M}\left(v\epsilon_m + c_{k,m}^{-j}\right)\right)}{\prod_{m=1}^{M}\Gamma\left(v\epsilon_m + c_{k,m}^{-j}\right)} \times \frac{\prod_{m=1}^{M}\Gamma\left(v\epsilon_m + c_{k,m}^{-j} + c_{k,m}^{j}\right)}{\Gamma\left(\sum_{m=1}^{M}\left(v\epsilon_m + c_{k,m}^{-j} + c_{k,m}^{j}\right)\right)} \\
&= \frac{\Gamma\left(\sum_{m=1}^{M}\left(v\epsilon_m + c_{k,m}^{-j}\right)\right)}{\Gamma\left(\sum_{m=1}^{M}\left(v\epsilon_m + c_{k,m}^{-j}\right) + N_j\right)} \times \prod_{m=1}^{M} \frac{\Gamma\left(v\epsilon_m + c_{k,m}^{-j} + c_{k,m}^{j}\right)}{\Gamma\left(v\epsilon_m + c_{k,m}^{-j}\right)} \\
&= \begin{cases} A = \frac{\Gamma\left(\sum_m \left[v\epsilon_m + c_{k,m}^{-j}\right]\right)}{\Gamma\left(\sum_m \left[v\epsilon_m + c_{k,m}\right]\right)} \prod_m \frac{\Gamma\left(v\epsilon_m + c_{k,m}\right)}{\Gamma\left(v\epsilon_m + c_{k,m}^{-j}\right)} & \text{if } k \text{ previously used} \\ B = \frac{\Gamma\left(\sum_m v\epsilon_m\right)}{\Gamma\left(\sum_m v\epsilon_m + N_j\right)} \prod_m \frac{\Gamma\left(v\epsilon_m + c_{k,m}^{j}\right)}{\Gamma(v\epsilon_m)} & \text{if } k = k_{\mathrm{new}} \end{cases}
\end{aligned}
$$

note that $\epsilon = (\epsilon_1, \epsilon_2, ...\epsilon_M, \epsilon_{\mathrm{new}})$, here $\epsilon_{1:M} = (\epsilon_1, \epsilon_2, ...\epsilon_M)$, when sampling $z_j$ we only use $M$ active components from the previous iteration. In summary, the conditional distribution to sample $z_j$ is given as:

$$
p\left(z_j = k \mid \mathbf{z}_{-j}, \boldsymbol{l}, \mathbf{x}, \alpha, H\right) \propto \begin{cases} n_{-j}^{k} \times f_k^{-x_j}\left(x_j\right) \times A & \text{if } k \text{ previousely used} \\ \alpha \times f_{k_{\mathrm{new}}}^{-x_{ji}}\left(x_{ji}\right) \times B & \text{if } k = k_{\mathrm{new}}. \end{cases}
$$

Implementation note: to evaluate A and B, we make use of the marginal likelihood resulted from a Multinomial-Dirichlet conjugacy.

**Sampling $l$.** Let $w_{-ji}$ be the same set as $w$ excluding $w_{ji}$, i.e $w_{-ji} = \{w_{uv} : u \neq j \cap v \neq i\}$, then we can write

$$p\left(l_{ji} = m \mid l_{-ji}, z_j = k, v, \epsilon, \mathbf{w}, \rho, S\right) \propto \underbrace{p\left(w_{ji} \mid w_{-ji}, l_{ji} = m, \rho\right)}_{\text{content predictive likelihood}} \times \underbrace{p\left(l_{ji} = m \mid \boldsymbol{l}_{-ji}, z_j = k, \epsilon_m, v\right)}_{\text{CRF for content topic}}. \tag{4.3.6}$$

The first argument is computed as log likelihood predictive of the content with the component $\psi_m$

$$\begin{aligned} p\left(w_{ji} \mid w_{-ji}, l_{ji} = m, \rho\right) =& \frac{\int_{\lambda_m} s\left(w_{ji} \mid \lambda_m\right) \left[\prod_{u \in w_{-ji}(m)} y(u \mid \lambda_m)\right] s(\lambda_m) d\lambda_m}{\int_{\lambda_m} \left[\prod_{u \in w_{-ji}(m)} y\left(u \mid \lambda_m\right)\right] s\left(\lambda_m\right) d\lambda_m} \\ \triangleq& y_m^{-w_{ji}}\left(w_{ji}\right). \end{aligned} \tag{4.3.7}$$

And the second term is inspired by Chinese Restaurant Franchise (CRF) as:

$$p\left(l_{ji} = m \mid \boldsymbol{l}_{-ji}, \epsilon_m, v\right) = \begin{cases} c_{k,m} + v\epsilon_m & \text{if } m\text{is used previously} \\ v\epsilon_{\text{new}} & \text{if } m = m_{\text{new}} \end{cases} \tag{4.3.8}$$

where $c_{k,m}$ is the number of data point $|\{l_{ji} | l_{ji} = m, z_j = k, 1 \leq j \leq J, 1 \leq i \leq N_j\}|$. The final form to sample $l_{ji}$ is given as:

$$p\left(l_{ji} = m \mid \boldsymbol{l}_{-ji}, z_j = k, w, v, \epsilon\right) \propto \begin{cases} \left(c_{k,m} + v\epsilon_m\right) \times y_m^{-w_{ji}}\left(w_{ji}\right) & \text{if } m\text{is used previously} \\ v\epsilon_{\text{new}} \times y_m^{-w_{ji}}(w_{ji}) & \text{if } m = m_{\text{new}}. \end{cases}$$

**Sampling $\epsilon$.**

Note that sampling $\epsilon$ require both $z$ and $l$

$$p\left(\epsilon \mid \boldsymbol{l}, \mathbf{z}, v, \eta\right) \propto p\left(\boldsymbol{l} \mid \epsilon, v, z, \eta\right) \times p\left(\epsilon \mid \eta\right). \tag{4.3.9}$$

Isolating the content variables $l_{ji}^k$ generated by the same context $z_j = k$ into one group

$l_j^k = \{l_{ji} : 1 \leq i \leq N_j, z_j = k\}$ the first term of 4.3.9 can be expressed following:

$$\begin{aligned} p\left(l \mid \epsilon, v, z, \eta\right) =& \prod_{k=1}^{K} \int_{\tau_k} p\left(l_{**}^k \mid \tau_k\right) p\left(\tau_k \mid \epsilon\right) d\tau_k \\ =& \prod_{k=1}^{K} \frac{\Gamma(v)}{\Gamma\left(v + n_{k*}\right)} \prod_{m=1}^{M} \frac{\Gamma(v\epsilon_m + n_{km})}{\Gamma(v\epsilon_m)} \end{aligned}$$

where $n_{km} = |\{w_{ji} \mid z_j = k, l_{ji} = m, 1 \leq j \leq J, 1 \leq i \leq N_j, \}|$ and

$n_{k*} = |\{w_{ji} \mid z_j = k, i = 1, ...N_j\}|$. Let $\eta_r = \frac{\eta}{R}$, $\eta_{\text{new}} = \frac{R-M}{R}\eta$ and recall that $\epsilon \sim \text{Dir}\left(\eta_r, \ldots, \eta_r, \eta_{\text{new}}\right)$, the last term of Eq. 4.3.9 is a Dirichlet density:

$$\begin{aligned} p\left(\boldsymbol{\epsilon} \mid \eta\right) =& \text{Dir}\left(\underbrace{\eta_1, \eta_2, ...\eta_M}_{M}, \eta_{\text{new}}\right) \\ =& \frac{\Gamma(M \times \eta_r + \eta_{\text{new}})}{[\Gamma(\eta_r)]^M \eta_{\text{new}}} \prod_{m=1}^{M} \epsilon_m^{\eta_r - 1} \epsilon_{\text{new}}^{\eta_{\text{new}} - 1}. \end{aligned}$$

Using the result:

$$\frac{\Gamma(v\epsilon_m + n_{km})}{\Gamma(v\epsilon_m)} = \sum_{o_{km}=0}^{n_{km}} \text{Stirl}\left(o_{km}, n_{km}\right) (v\epsilon_m)^{o_{km}}.$$

Thus, Eq. 4.3.9 becomes:

$$\begin{aligned} p\left(\boldsymbol{\epsilon} \mid \boldsymbol{l}, \mathbf{z}, v, \eta\right) =& \epsilon_{\text{new}}^{\eta_{\text{new}} - 1} \prod_{k=1}^{K} \frac{\Gamma(v)}{\Gamma\left(v + n_{k*}\right)} \prod_{m=1}^{M} \epsilon_m^{\eta_m - 1} \sum_{o_{km}=0}^{n_{km}} \text{Stirl}\left(o_{km}, n_{km}\right) (v\epsilon_m)^{o_{km}} \\ =& \epsilon_{\text{new}}^{\eta_{\text{new}} - 1} \sum_{o_{km}=0}^{n_{km}} \prod_{k=1}^{K} \frac{\Gamma(v)}{\Gamma\left(v + n_{k*}\right)} \prod_{m=1}^{M} \epsilon_m^{\eta_m - 1} \text{Stirl}\left(o_{km}, n_{km}\right) (v\epsilon_m)^{o_{km}} \\ p\left(\boldsymbol{\epsilon}, \boldsymbol{o} \mid \boldsymbol{l}, \mathbf{z}, v, \eta\right) =& \epsilon_{\text{new}}^{\eta_{\text{new}} - 1} \prod_{k=1}^{K} \frac{\Gamma(v)}{\Gamma\left(v + n_{k*}\right)} \prod_{m=1}^{M} \epsilon_m^{\eta_m - 1} \text{Stirl}\left(o_{km}, n_{km}\right) (v\epsilon_m)^{o_{km}}. \end{aligned}$$

The probability of the auxiliary variable $o_{km}$ is computed as:

$$p(o_{km}) = \sum_{o_{km}=0}^{n_{km}} \text{Stirl}\left(o_{km}, n_{km}\right) (v\epsilon_m)^{o_{km}}.$$

Now let $o = (o_{km} : \forall k, m)$ we derive the following joint distribution:

$$p\left(\boldsymbol{\epsilon} \mid o, \boldsymbol{l}, \mathbf{z}, v, \eta\right) = \epsilon_{\text{new}}^{\eta_{\text{new}}-1} \prod_{m=1}^{M} \epsilon_m^{\sum_K o_{km}+\eta_m-1}.$$

As $R \to \infty$, we have $p\left(\boldsymbol{\epsilon} \mid \boldsymbol{o}, \boldsymbol{l}, \mathbf{z}, v, \eta\right) \stackrel{\infty}{=} \epsilon_{\text{new}}^{\eta-1} \prod_{m=1}^{M} \epsilon_m^{\sum_K o_{km}-1}$. Finally, we sample $\epsilon$ jointly with the auxiliary variable $o_{km}$ by:

$$p\left(o_{km} = h \mid \cdot\right) \propto \text{Stirl}\left(h, n_{km}\right)\left(v\epsilon_m\right)^h,\ h = 0, 1, \ldots, n_{km}$$
$$p(\epsilon) \propto \epsilon_{\text{new}}^{\eta-1} \prod_{m=1}^{M} \epsilon_m^{\sum_K o_{km}-1}.$$

**Sampling hyperparameters**

In the proposed model, there are three hyper-parameters which need to be sampled: $\alpha, v$ and $\eta$.

**Sampling $\eta$ .** Using the technique from Escobar and West (Escobar and West, 1995), we have

$$p\left(M \mid \eta, u\right) = \text{Stirl}\left(M, u\right)\eta^M \frac{\Gamma\left(\eta\right)}{\Gamma\left(\eta + u\right)}$$

where $u = \sum_m u_m$ with $u_m = \sum_K o_{km}$ is in the previous sampling $\epsilon$ and $M$ is the number of active content atoms. Let $\eta \sim \text{Gamma}\left(\eta_1, \eta_2\right)$. Recall that:

$$\frac{\Gamma\left(\eta\right)}{\Gamma\left(\eta + u\right)} = \int_0^1 t^\eta \left(1 - t\right)^{u-1} \left(1 + \frac{u}{\eta}\right) dt$$

that we have just introduced an auxiliary variable $t$

$$p\left(t \mid \eta\right) \propto t^\eta \left(1 - t\right)^{u-1} = \text{Beta}\left(\eta + 1, u\right).$$

Therefore,

$$
\begin{aligned}
p(\eta \mid t) \propto & \eta^{\eta_1 - 1 + M} \exp\{-\eta\eta_2\} \times t^{\eta} (1-t)^{u-1} \left(1 + \frac{u}{\eta}\right) \\
& = \eta^{\eta_1 - 1 + M} \times \exp\{-\eta(\eta_2 - \log t)\} \times (1-t)^{u-1} \\
& + \eta^{\eta_1 - 1 + M - 1} \exp\{-\eta(\eta_2 - \log t)\} \times (1-t)^{u-1} u \\
& \propto \eta^{\eta_1 - 1 + M} \exp\{-\eta(\eta_2 - \log t)\} + u\eta^{\eta_1 - 1 + M - 1} \exp\{-\eta(\eta_2 - \log t)\} \\
& = \pi_t \text{Gamma}(\eta_1 + M, \eta_2 - \log t) \\
& + (1 - \pi_t) \text{Gamma}(\eta_1 + M - 1, \eta_2 - \log t)
\end{aligned} \tag{4.3.10}
$$

where $\pi_t$ satisfies this following equation to make the above expression a proper mixture density:

$$
\frac{\pi_t}{1 - \pi_t} = \frac{\eta_1 + M - 1}{u(\eta_2 - \log t)}. \tag{4.3.11}
$$

To re-sample $\eta$, we first sample $t \sim \text{Beta}(\eta + 1, u)$, compute $\pi_t$ as in Eq. 4.3.11, and then use $\pi_t$ to select the correct Gamma distribution to sample $\eta$ as in Eq. 4.3.10.

**Sampling $\alpha$.** Again sampling $\alpha$ is similar to (Escobar and West, 1995). Assuming $\alpha \sim \text{Gamma}(\alpha_1, \alpha_2)$ with the auxiliary variable $t$:

$$
\begin{aligned}
p(t \mid \alpha, K) \propto & t^{\alpha_1} (1-t)^{J-1} \\
& p(t \mid \alpha, K) \text{Beta}(\alpha_1 + 1, J)
\end{aligned}
$$

where $J$ is the number of document. Then, the final form for sampling $\alpha$ is as

$$
p(\alpha \mid t, K) \sim \pi_t \text{Gamma}(\alpha_1 + K, \alpha_2 - \log(t)) + (1 - \pi_t) \text{Gamma}(\alpha_1 + K - 1, \alpha_2 - \log(t))
$$

where $a_1, a_2$ are prior parameter for sampling $\alpha$ following Gamma distribution and $\frac{\pi_t}{1-\pi_t} = \frac{\alpha_1 + K - 1}{J(\alpha_2 - \log t)}$.

**Sampling $v$.** Sampling $v$ is similar to sampling concentration parameter in HDP (Teh et al., 2006). Denote $o_{k*} = \sum_m o_{km}$, where $o_{km}$ is defined previously during the sampling step for $\boldsymbol{\epsilon}$, $n_{k*} = \sum_m n_{km}$, where $n_{km}$ is the count of $|\{l_{ji} \mid z_{ji} = k, l_{ji} = m\}|$.

Using similar technique in (Teh et al., 2006), we write:

$$p\left(o_{1*}, o_{2*}.., o_{K*} \mid v, n_{1*}, ...n_{K*}\right) = \prod_{k=1}^{K} \text{Stirl}(n_{k*}, o_{k*}) \alpha_0^{o_{k*}} \frac{\Gamma(v)}{\Gamma\left(v + n_{k*}\right)}$$

where the last term can be expressed as

$$\frac{\Gamma(v)}{\Gamma\left(v + n_{k*}\right)} = \frac{1}{\Gamma(n_{k*})} \int_0^1 b_k^v \left(1 - b_k\right)^{n_{k**}-1} \left(1 + \frac{n_{k*}}{v}\right) db_k.$$

Assuming $v \sim \text{Gamma}\left(v_1, v_2\right)$, define the auxiliary variables $b = \left(b_k \mid k = 1, \ldots, K\right), b_k \in [0, 1]$ and $t = \left(t_k \mid k = 1, \ldots, K\right), t_k \in \{0, 1\}$ we have

$$q\left(v, b, t\right) \propto v^{v_1 - 1 + \sum_k M_k} \exp\left\{-v v_1\right\} \prod_{k=1}^{K} b_k^v \left(1 - b_k\right)^{M_k - 1} \left(\frac{M_k}{v}\right)^{t_k}.$$

We will sample the auxiliary variables $b_k$, $t_k$ in accordance with $v$ that are defined below:

$$\begin{aligned}
q(b_k \mid v) =& \text{Beta}\left(v + 1, o_{k*}\right) \\
q\left(t_k \mid .\right) =& \text{Bernoulli}\left(\frac{o_{k*}/v}{1 + o_{k*}/v}\right) \\
q(v \mid .) =& \text{Gamma}\left(v_1 + \sum_k \left(o_{k*} - t_k\right), v_2 - \sum_k \log b_k\right).
\end{aligned}$$

**Polya Urn View**

Our model exhibits a Polya-urn view using the analogy of a fleet of buses, driving customers to restaurants. Each bus represents a group and customers on the bus are data points within the group. For each bus $j$, $z_j$ acts as the index to the restaurant for its destination. Thus, buses form clusters at their destination restaurants according to a CRP: a new bus drives to an existing restaurant with the probability proportional to the number of other buses that have arrived at that restaurant, and with probability proportional to $\alpha$, it goes to a completely new restaurant.

Once all the buses have delivered customers to the restaurants, *all customers at the restaurants start to behave in the same manner as in a Chinese restaurant franchise*

*(CRF) process*: customers are assigned tables according to a restaurant-specific CRP; tables are assigned with dishes $\psi_m$ (representing the content topic atoms) according to a global franchise CRP. In addition to the usual CRF, at restaurant $k$, a single dessert $\phi_k$ (which represents the context-generating atom, drawing $\overset{\text{iid}}{\sim}$ from $H$) will be served to all the customers at that restaurant. Thus, every customer on the same bus $j$ will be served the same dessert $\phi_{z_j}$. We observe three sub-CRPs, corresponding to the three DP(s) in our model: the CRP at the dish level is due to the DP $(\eta S)$, the CRP forming tables inside each restaurant is due to the DP$(vQ_0)$, and the CRP aggregating buses to restaurants is due to the DP $(\alpha(H \times \text{DP}(vQ_0)))$.

**Inference Complexity Analysis**

The majority of computations in the proposed model arises from the following. *Each* Gibbs iteration is dominated by $O(J \times N \times M)$ where: (1) sampling all $z_j$ (s) having complexity $O(J \times K)$ where $J$ is the number of groups, $K$ the active number of context topics; (2) sampling all $l_{ji}$ (s) having complexity $O(J \times N \times M)$ where $N$ is the average number of data points in one group, $M$ is the active number of content atoms; (3) sampling $\epsilon$ having complexity of $O(K \times M)$. Finally, this complexity assumes the unsigned Stirling number of the first kind has been pre-computed, hence excludes its computation time.

### 4.3.3 Marginalization property

We study marginalization property for our model when either the content topics $\varphi_{ji}$ (s) or context topics $\theta_j$ (s) are marginalized out. Our main result is established in Theorem 4.5 where we show an interesting link to nested DP and DPM via our model.

Let $H$ be a measure over some measurable space $(\Theta, \Sigma)$. Let $\mathbb{P}$ be the set of all measures over $(\Theta, \Sigma)$, suitably endowed with some $\sigma$-algebra. Let $G \sim \text{DP}(\alpha H)$ be a draw from a Dirichlet process.

**Lemma 4.1.** *Let $S_1 \dots S_n$ be $n$ measurable sets in $\Sigma$. We form a measurable partition of $\Theta$, a collection of disjoint measurable sets, that generate $S_1, \dots, S_n$ as follows.*

*If $S$ is a set, let $S^1 = S$ and $S^{-1} = \Theta \backslash S$. Then $S^* = \{\bigcap_{i=1}^n S_i^{c_i} | c_i \in \{1, -1\}\}$ is a partition of $\Theta$ into a finite collection of* disjoint *measurable sets with the property that any $S_i$ can be written as a union of some sets in $S^*$. Let the element of $S^*$ be $A_1 \dots A_{n^*}$ (note $n^* \leq 2^n$). Then the expectation*

$$\mathop{\mathbb{E}}_{G} [G(S_1), \dots, G(S_n)] = \int \prod_{i=1}^{n} G(S_i) \, DP(dG \mid \alpha H) \tag{4.3.12}$$

*depends only on $\alpha$ and $H(A_i)$. In other words, the above expectation can be written as a function $E_n(\alpha, H(A_1), \dots H(A_{n^*}))$.*

It is easy to see that since $S_i$ can always be expressed as the sum of some disjoints $A_i$, $G(S_i)$ can respectively be written as the sum of some $G(A_i)$. Furthermore, by definition of a Dirichlet process, the vector $(G(A_1), \dots, G(A_{n^*}))$ distributed according to a finite Dirichlet distribution $(\alpha H(A_1), \dots, \alpha H(A_{n^*}))$, therefore the expectation $\mathop{\mathbb{E}}_{G} [G(S_i)]$ depends only on $\alpha$ and $H(A_i)$ (s).

**Definition 4.2.** (DPM) A DPM is a probability measure over $\Theta^n \ni (\theta_1, \dots, \theta_n)$ with the usual product sigma algebra $\Sigma^n$ such that for every collection of measurable sets $\{(S_1, \dots, S_n) : S_i \in \Sigma, i = 1, \dots, n\}$:

$$\mathrm{DPM}(\theta_1 \in S_1, \dots, \theta_n \in S_n | \alpha, H) = \int \prod_{i=1}^{n} G(S_i) \, \mathrm{DP}(dG \mid \alpha H).$$

We now state a result regarding marginalization of draws from a DP mixture with a joint base measure. Consider two measurable spaces $(\Theta_1, \Sigma_1)$ and $(\Theta_2, \Sigma_2)$ and let $(\Theta, \Sigma)$ be their product space where $\Theta = \Theta_1 \times \Theta_2$ and $\Sigma = \Sigma_1 \times \Sigma_2$. Let $H^*$ be a measure over the product space $\Theta = \Theta_1 \times \Theta_2$ and let $H_1$ be the marginal of $H^*$ over $\Theta_1$ in the sense that for any measurable set $A \in \Sigma_1$, $H_1(A) = H^*(A \times \Theta_2)$. Then drawing $(\theta_i^{(1)}, \theta_i^{(2)})$ from a DP mixture with base measure $\alpha H$ and marginalizing out $(\theta_i^{(2)})$ is the same as drawing $(\theta_i^{(1)})$ from a DP mixture with base measure $H_1$. Formally

**Proposition 4.3.** *Let $H^*$ be a measure over the product space $\Theta = \Theta_1 \times \Theta_2$. Let $H_1$ be the marginal of $H^*$ over $\Theta_1$ in the sense that for any measurable set $A \in \Sigma_1$,*

$H_1(A) = H^*(A \times \Theta_2)$. *Then:*

$$\begin{aligned} &DPM\left(\theta_1^{(1)} \in S_1, \ldots, \theta_n^{(1)} \in S_n \mid \alpha, H_1\right) \\ &= DPM\left(\left(\theta_1^{(1)}, \theta_1^{(2)}\right) \in S_1 \times \Theta_2, \ldots, \left(\theta_n^{(1)}, \theta_n^{(2)}\right) \in S_n \times \Theta_2 \mid \alpha, H^*\right) \end{aligned}$$

*for every collection of measurable sets* $\{(S_1, \ldots, S_n) : S_i \in \Sigma_1, i = 1, \ldots, n\}$.

*Proof.* Since $\{(S_1, \ldots, S_n) : S_i \in \Sigma_1, i = 1, \ldots, n\}$ are rectangles, expanding the RHS using Definition 4.2 gives:

$$RHS = \int G(S_1 \times \Theta_2) \ldots G(S_n \times \Theta_2)\, d\mathrm{DP}(dG|\alpha, H^*)$$

Let $T_i = S_i \times \Theta_2$, the above expression is the expectation of $\prod_i G(T_i)$ when $G \sim DP(\alpha H^*)$. Forming collection of the disjoint measurable sets $T^* = (B_1 \ldots B_{n^*})$ that generates $T_i$, then note that $B_i = A_i \times \Theta_2$, and $S^* = (A_1 \ldots A_{n^*})$ generates $S_i$. By definition of $H_1$, $H_1(A_i) = H^*(A_i \times \Theta_2) = H^*(B_i)$. Using the Lemma 4.1 above, $RHS = E_n(\alpha, H^*(B_1) \ldots H^*(B_{n^*}))$, while $LHS = E_n(\alpha, H_1(A_1) \ldots H_1(A_{n^*}))$ and they are indeed the same. □

We note that $H^*$ can be any arbitrary measure on $\Theta$ and, in general, we do not require $H^*$ to factorize as product measure.

Next we give a formal definition for the nDP mixture: $\varphi_{ji} \overset{\text{iid}}{\sim} Q_j$, $Q_j \overset{\text{iid}}{\sim} U$, $U \sim \mathrm{DP}(\alpha \mathrm{DP}(vQ_0))$, $Q_0 \sim DP(\eta S)$.

**Definition 4.4.** (nested DP Mixture) An nDPM is a probability measure over $\Theta^{\sum_{j=1}^{J} N_j} \ni (\varphi_{11}, \ldots, \varphi_{1N_1}, \ldots, \varphi_{JN_J})$ equipped with the usual product sigma algebra $\Sigma^{N_1} \times \ldots \times \Sigma^{N_J}$ such that for every collection of measurable sets

$\{(S_{ji}) : S_{ji} \in \Sigma, j = 1, \ldots, J, i = 1 \ldots, N_j\}$:

$$\begin{aligned} \mathrm{nDPM}(\varphi_{ji} \in S_{ji}, \forall i, j|\alpha, v, \eta, S) = \int \int \left\{ \prod_{j=1}^{J} \int \prod_{i=1}^{N_j} Q_j(S_{ji})\, U(dQ_j) \right\} \\ \times \mathrm{DP}(dU \mid \alpha \mathrm{DP}(vQ_0))\, \mathrm{DP}(dQ_0 \mid \eta, S). \end{aligned}$$

We now have the sufficient formalism to state the marginalization result for our model.

**Theorem 4.5.** *Given $\alpha, H$ and $\alpha, v, \eta, S$, let $\boldsymbol{\theta} = (\theta_j : \forall j)$ and $\boldsymbol{\varphi} = (\varphi_{ji} : \forall j, i)$ be generated as in Eq. 4.3.1. Then, marginalizing out $\boldsymbol{\varphi}$ results in $DPM(\boldsymbol{\theta} \mid \alpha, H)$, whereas marginalizing out $\boldsymbol{\theta}$ results in $nDPM(\boldsymbol{\varphi}|\alpha, v, \eta, S)$.*

*Proof.* First we make observation that if we can show Proposition 4.3 still holds when $H_1$ is random with $H_2$ is fixed and vice versa, then the proof required is an immediate corollary of Proposition 4.3 by letting $H^* = H_1 \times H_2$ where we first let $H_1 = H$, $H_2 = \mathrm{DP}(vQ_0)$ to obtain the proof for the first result, and then swap the order $H_1 = \mathrm{DP}(vQ_0), H_2 = H$ to get the second result.

To see that Proposition 4.3 still holds when $H_2$ is a random measure and $H_1$ is fixed, we let the product base measure $H^* = H_1 \times H_2$ and further let $\mu$ be a prior probability measure for $H_2$, i.e, $H_2 \sim \mu(\cdot)$. Consider the marginalization over $H_2$:

$$\begin{aligned}
&\int_{H_2} \mathrm{DPM}\left(\left(\theta_1^{(1)}, \theta_1^{(2)}\right) \in S_1 \times \Theta_2, \ldots, \left(\theta_n^{(1)}, \theta_n^{(2)}\right) \in S_n \times \Theta_2 \mid \alpha, H^*\right) \mu(H_2) \\
&\quad = \int_{\Sigma_2} \underbrace{\mathrm{DPM}\left(\theta_1^{(1)} \in S_1, \ldots, \theta_n^{(1)} \in S_n \mid \alpha, H_1\right)}_{\text{constant w.r.t } H_2} \mu(H_2) \\
&\quad = \mathrm{DPM}\left(\theta_1^{(1)} \in S_1, \ldots, \theta_n^{(1)} \in S_n \mid \alpha, H_1\right) \int_{\Sigma_2} \mu(H_2) \\
&\quad = \mathrm{DPM}\left(\theta_1^{(1)} \in S_1, \ldots, \theta_n^{(1)} \in S_n \mid \alpha, H_1\right).
\end{aligned}$$

When $H_1$ is random and $H_2$ is fixed. Let $\lambda(\cdot)$ be a prior probability measure for $H_1$, ie., $H_1 \sim \lambda(\cdot)$. It is clear that Proposition 4.3 holds for each draw $H_1$ from $\lambda(\cdot)$. This complete our proof. □

## 4.4 Experiments

We first evaluate the model via simulation studies, then demonstrate its applications on text and image modelling using three real-world datasets. Throughout this section, unless explicitly stated, discrete data is modelled by Multinomial with

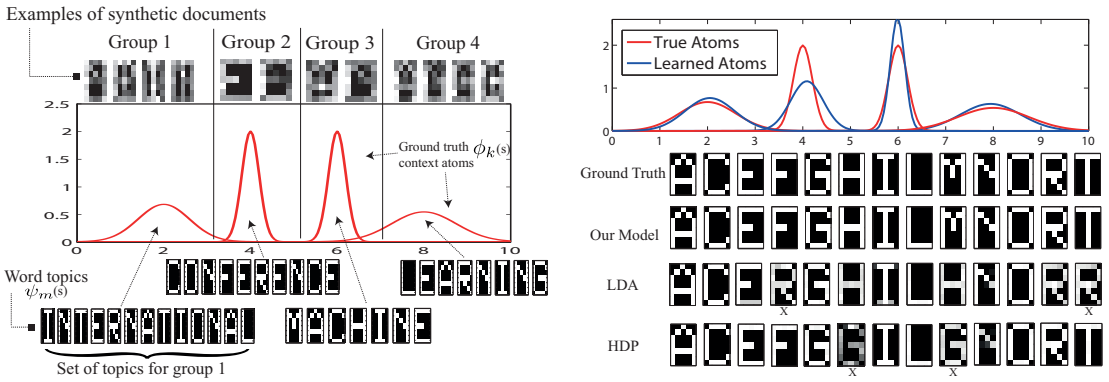



Figure 4.4.1: Results from simulation study. Left: illustration of data generation with ground truth for context atoms are 4 univariate Gaussians centreed at $2, 4, 6$ and 8 respectively (different variances). Right: Our model recovers the correct 4 group clusters, their context distributions and the set of shared topics. LDA and HDP are unable to recover the true content topics without using contexts.

Dirichlet prior, while continuous data is modelled by Gaussian (unknown mean and unknown variance) with Gaussian-Gamma prior.

### 4.4.1 Numerical simulation

The main goal is to investigate the posterior consistency of the model, i.e., its ability to recover the true group clusters, context distribution and content topics. To synthesize the data, we use $M = 13$ topics which are the 13 unique letters in the ICML string "INTERNATIONAL CONFERENCE MACHINE LEARNING". Similar to (Griffiths and Steyvers, 2004), each topic $\psi_m$ is a distribution over 35 words (pixels) and visualized as a $7 \times 5$ binary image. We generate $K = 4$ clusters of 100 documents each. For each cluster, we choose a set of topics corresponding to letters in the each of 4 words in the ICML string. The topic mixing distribution $\tau_k$ is an uniform distribution over the chosen topic letters. Each cluster is also assigned a context-generating univariate Gaussian distribution. These generating parameters are shown in Fig. 4.4.1 (left). Altogether we have $J = 400$ documents; for each document we sample $N_j = 50$ words and a context variable $x_j$ drawing from the cluster-specific Gaussian.

We model the word $w_{ji}$ with Multinomial and Gaussian for context $x_j$. After 100 Gibbs iterations, the number of context and content topics ($K = 4, M = 13$) are

recovered correctly: the learned context atoms $\phi_k$ and topic $\psi_m$ are almost identical to the ground truth (Fig. 4.4.1, right) and the model successfully identifies the 4 clusters of documents with topics corresponding to the 4 words in the ICML string.

To demonstrate the importance of context observation, we then run LDA and HDP with only the word observations (ignoring context) where the number of topic of LDA is set to 13. As can be seen from Fig. 4.4.1 (right), LDA and HDP have problems in recovering the true topics in this simulation setup.

**Roles of context and content data**

The relative contribution of context and content data raises further question on the inference of the clusters defined by our model. We attempt to provide some empirical insights by offering some intuition and through numerical simulation.

In the graphical model representation (cf. Fig. 4.3.1), there are qualitatively more latent variables associated with the content than the context. Therefore, it is tempting to think that increasing the number of data points within each group will cause the content data's likelihood to quickly overwhelm that of the context and make the likelihoods unbalanced.

Regarding the inference of the cluster index $z_j$, to obtain the marginal likelihood (the third term in Eq. 4.3.3 used to sample $z_j$), one has to integrate out the words' topic labels $l_{ji}$. In doing so, it can be shown that the *sufficient* statistics coming from the content data toward the inference of the topic frequencies and the clustering labels will *just be the empirical word frequency from each document*. As each document becomes sufficiently long, the empirical word frequency quickly concentrates around its mean by the central limit theorem (CLT), so as soon as the effect of CLT kicks in, increasing document length further will do very little in improving this sufficient statistics. Increasing the document length will probably not hurt the performance, but to what extent it contributes relative to the number of documents awaits a longer and deeper analysis.

To offer this insights from empirical perspective, we vary the document length and the number of documents in synthesis data and examine the posterior of the clus-

| Method | Perplexity (***on words only***) | | | | Feature used |
|---|---|---|---|---|---|
| | PNAS | (K,M) | NIPS | (K,M) | |
| HDP | 3027.5 | $(-, 86)$ | 1922.1 | $(-, 108)$ | words |
| npTOT | 2491.5 | $(20, 145)$ | 1855.33 | $(14, 94)$ | words+timestamp |
| MC² without context | 1742.6 | $(40, 126)$ | 1583.2 | $(19, 61)$ | words |
| MC² with titles | – | – | 1393.4 | $(32, 80)$ | words+title |
| MC² with authors | – | – | 1246.3 | $(8, 55)$ | words+authors |
| MC² with timestamp | **895.3** | $(12, 117)$ | **984.7** | $(15, 95)$ | words+timestamp |

Table 4.1: Perplexity evaluation on PNAS and NIPS datasets. (K,M) is (#cluster,#topic). (Note: missing results are due to title and author information not available in PNAS dataset). npTOT is nonparametric topic over time (Dubey et al., 2013)

tering labels $z_j$. Fig. 4.4.2 shows this result.

### 4.4.2 Experiments with real-world datasets

We use two standard NIPS and PNAS text datasets, and the NUS-WIDE image dataset.

*NIPS* contains 1,740 documents with vocabulary size 13,649 (excluding stop words); timestamps (1987-1999), authors (2,037) and title information are available and used as group-level context. *PNAS* contains 79,800 documents, vocab size = 36,782 with publication timestamp (915-2005). For *NUS-WIDE* we use a subset of the 13-class animals [1] comprising of 3,411 images (2,054 images for training and 1357 images for testing) with off-the-shelf features including 500-dim bag-of-word SIFT vector and 1000-dim bag-of-tag annotation vector.

#### 4.4.2.1 Text modeling with document-level contexts

We use NIPS and PNAS datasets with 90% for training and 10% for held-out perplexity evaluation. We compare the perplexity with HDP (Teh et al., 2006) where no group-level context can be used, and npTOT (Dubey et al., 2013) where only

[1] downloaded from http://www.ml-thu.net/~jun/data/

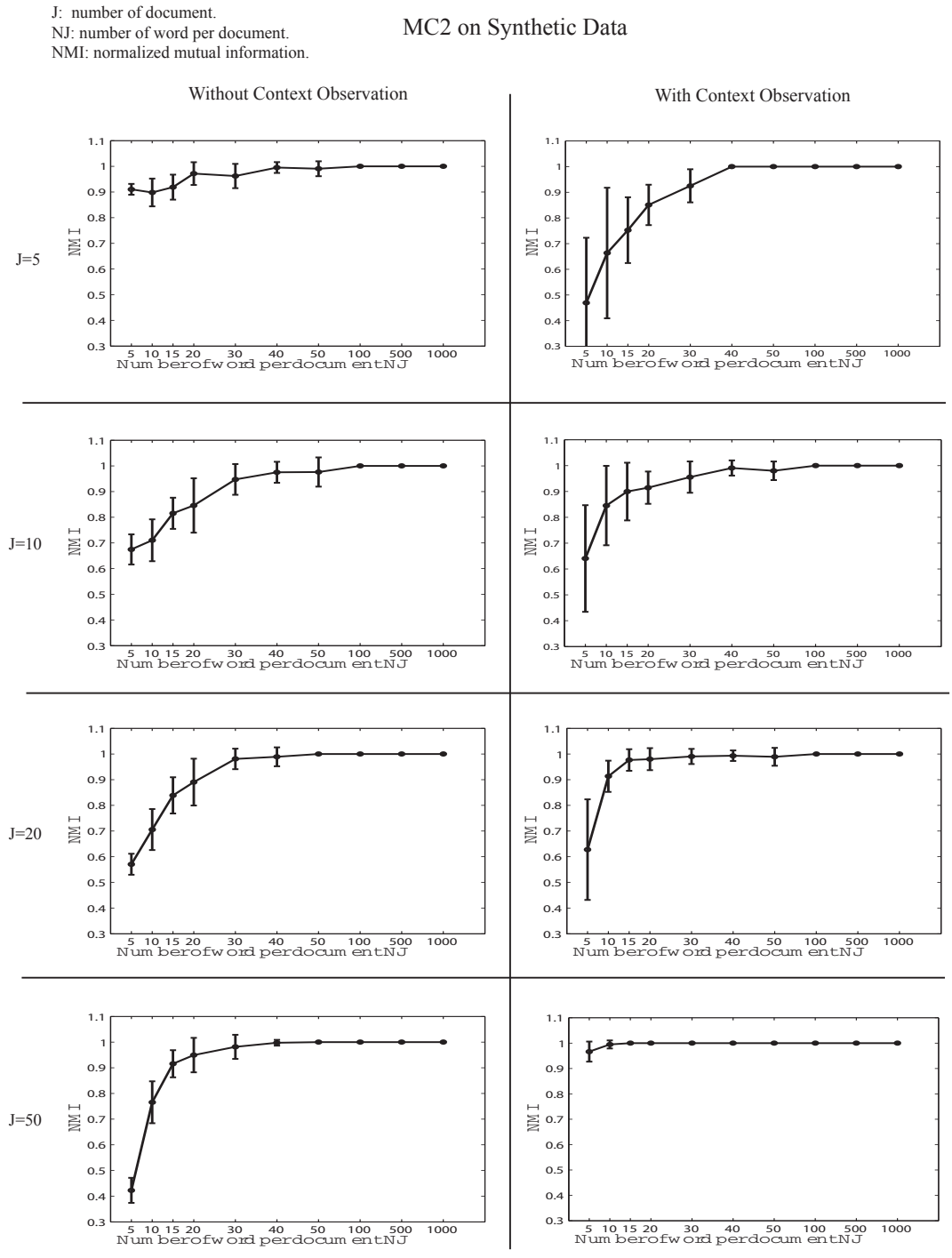



Figure 4.4.2: Performance of clustering labels $z_j$ inferred from the model by varying the number of document length $N_j$ (assumed to be same for all $j$) and the number of documents $J$.

timestamp information can be used. We note that unlike our model, npTOT requires replication of document timestamp for *every* word in the document, which is somewhat unnatural.

We use perplexity score (Blei et al., 2003) on held-out data as performance metric, defined as

$$\exp\left\{-\sum_{j=1}^{J}\log p\left(\mathbf{w}_j^{\text{test}} \mid \boldsymbol{x}^{\text{train}}, \mathbf{w}^{\text{train}}\right) / \left(\sum_j N_j^{\text{test}}\right)\right\}.$$

To ensure fairness and comparable evaluation, *only words* in held-out data is used to compute the perplexity. We use univariate Gaussian for timestamp and Multinomial distributions for words, tags and authors. We ran collapsed Gibbs for 500 iterations after 100 burn-in samples.

Table 4.1 shows the results where MC² achieves significant better performance. This shows that group-level context information during training provide useful guidance for the modelling tasks. Regarding the informative aspect of group-level context, we achieve better perplexity with timestamp information than with titles and authors. This may be explained by the fact that 1361 authors (among 2037) show up only once in the data while title provides little additional information than what already in that abstracts. Interestingly, without the group-level context information, our model still predicts the held-out words better than HDP. This suggests that inducing partitions over documents simultaneously with topic modelling is beneficial in this case.

Beyond the capacity of HDP and npTOT, our model can induce clusters over documents (value of $K$ in Table 4.1). Fig. 4.4.3 shows an example of one such document cluster discovered from NIPS data with authors as context.

Our proposed model also allows flexibility in deriving useful understanding into the data and to evaluate on its predictive capacity (e.g., who most likely wrote this article, which authors work in the same research topic and so on). Another possible usage is to obtain *conditional* distributions among context topics $\phi_k$ (s) and content topics $\psi_m$ (s). For example if the context information is timestamp, the model immediately yields the distribution over time for a topic, showing when the topic rises and falls. Fig. 4.4.4 illustrates an example of a distribution over time for a content

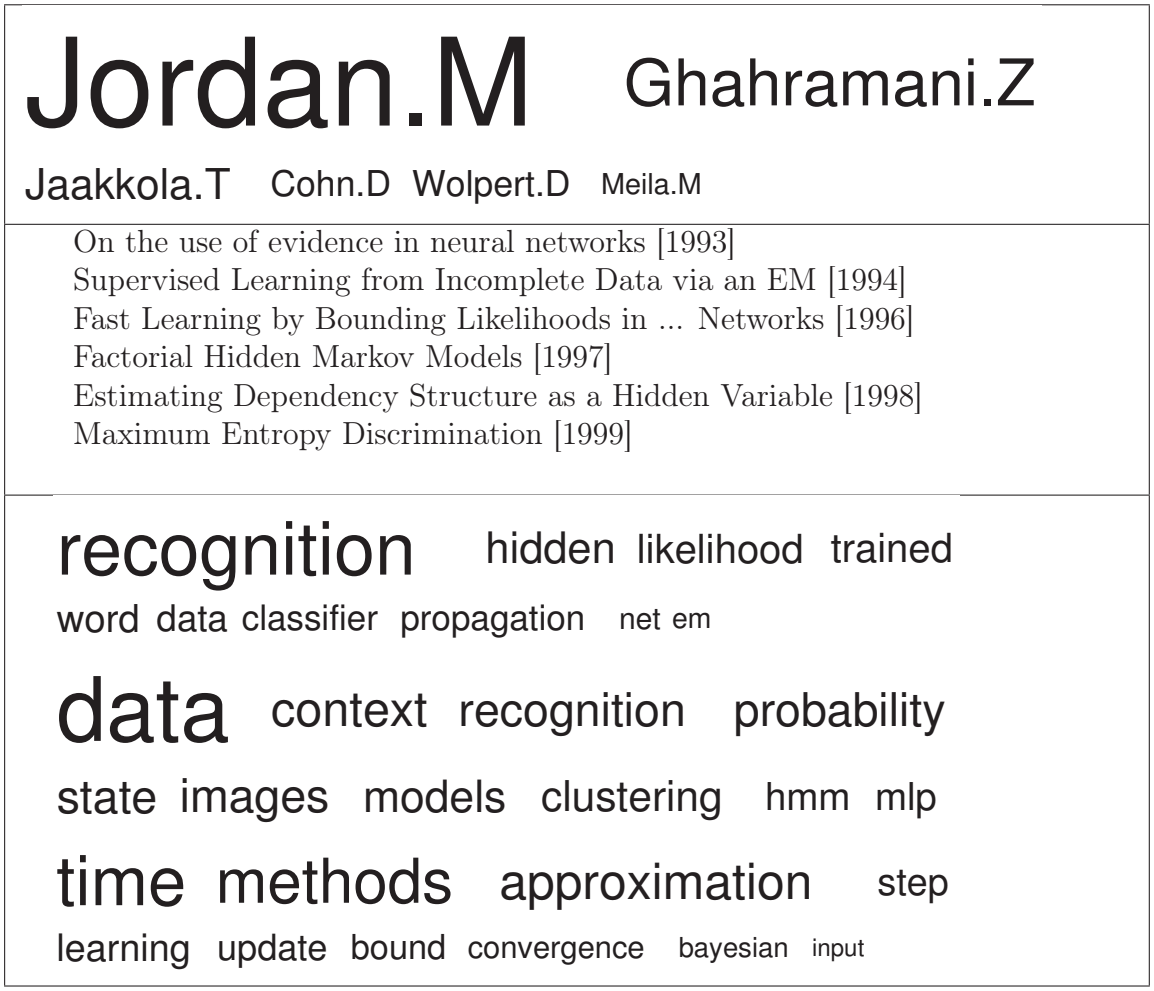



Figure 4.4.3: An example of document cluster from NIPS. Top: distribution over authors. Middle: examples of paper titles. Bottom: examples of word topics in this cluster.

topic discovered from PNAS dataset where timestamp was used as context. This topic appears to capture a congenital disorder known as *Albinism.* This distribution illustrates research attention to this condition over the past 100 years from PNAS data. To seek evidence for this result, we search the term "Albinism" in Google Scholar, using the top 50 searching results and plot the histogram over time in the same Fig. 4.4.4. Surprisingly, we obtain a very close match between our results and the results from Google Scholar as evidenced in the Fig. 4.4.4.

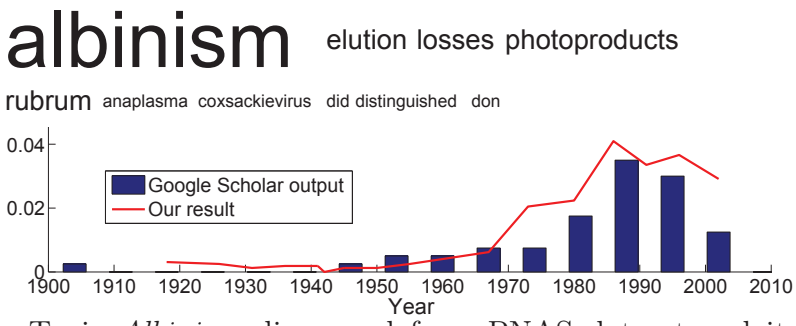



Figure 4.4.4: Topic *Albinism* discovered from PNAS dataset and its conditional distribution over time using our model; plotted together with results independently searched from Google Scholar using the top 50 hits.

| Method | Perplexity | Feature used |
|---|---|---|
| HDP | 175.62 | SIFT |
| MC² without context | 162.74 | SIFT |
| MC² with context | **152.32** | Tags+SIFT |

Table 4.2: NUS-WIDE dataset. Perplexity is evaluated on SIFT feature.

#### 4.4.2.2 Image clustering with image-level tags

We evaluate the clustering capacity of MC² using contexts on an image clustering task. Our dataset is NUS-WIDE described earlier. We use bag-of-word SIFT features from each image for its content. Since each image in this dataset comes with a set of tags, we exploit them as context information, hence each context observation $x_j$ is a bag-of-tag annotation vector.

First we perform the perplexity evaluation for this dataset using a similar setting as in the previous section. Table 4.2 presents the results where our model again outperforms HDP even when no context (tags) is used for training.

Next we evaluate the clustering quality of the model using the provided 13 classes as ground truth. We report performance on four well-known clustering evaluation metrics: Purity, Normalized Mutual Information (NMI), Rand-Index (RI), and Fscore (detailed in Rand (1971); Cai et al. (2011)). We use the following baselines for comparison:

- Kmeans and Non-negative Matrix Factorization (NMF)(Lee et al., 1999). For these methods, we need to specify the number of clusters in advance, hence

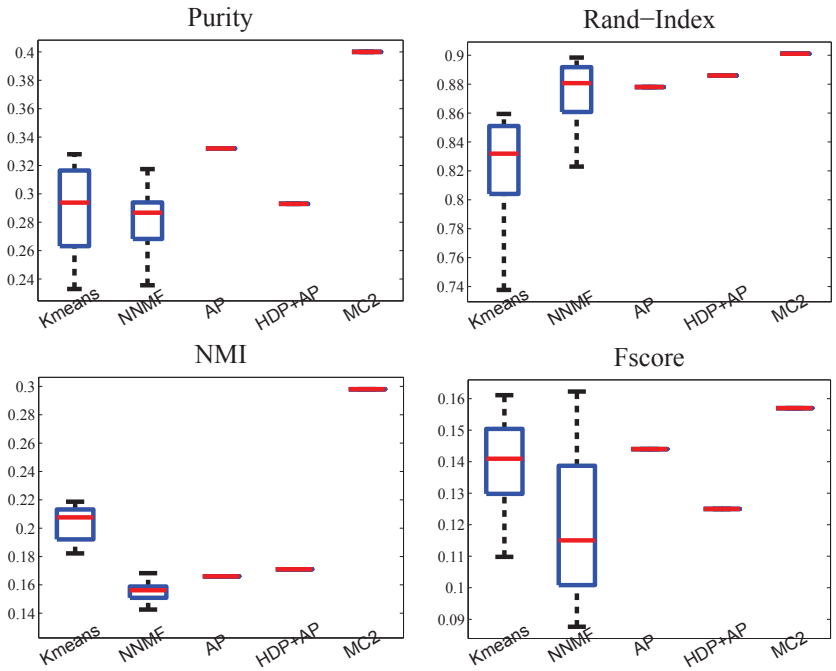



Figure 4.4.5: Clustering performance measured in purity, NMI, Rand-Index and F-score using NUS-WIDE dataset.

we vary this number from 10 to 40. We then report the min, max, mean and standard deviation.

- Affinity Propagation (AP) (Frey and Dueck, 2007): AP requires a similarity score between two documents and we use the Euclidean distance for this purpose.

- Hierarchical Dirichlet Process (HDP) + AP: we first run HDP using content observations, and then apply Affinity Propagation with similarity score derived from the symmetric KL divergence between the mixture proportions from two documents.

Fig. 4.4.5 shows the result in which our model consistently delivers highest performance across all four metrics. For purity and NMI, our model beats all by a wide margin.

To gain some understanding on the clusters of images induced by our model, we run t-SNE (Van der Maaten and Hinton, 2008), projecting the feature vectors (both content and context) onto a 2D space. For visual clarity, we randomly select 7 out

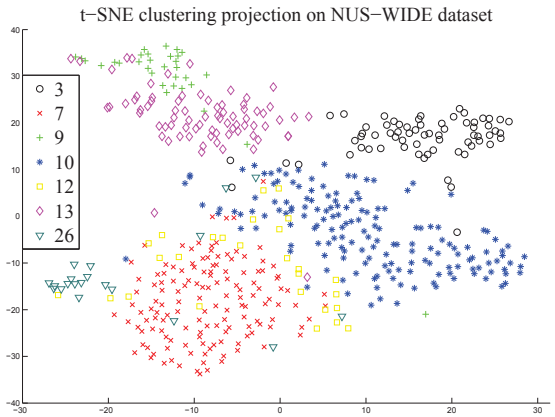



Figure 4.4.6: Projecting 7 discovered clusters (among 28) on 2D using t-SNE (Van der Maaten and Hinton, 2008).

of 28 clusters and display in Fig. 4.4.6 where it can be seen that they are reasonably well separated.

#### 4.4.2.3 Effect of partially observed and missing data

Missing and unlabelled data is commonly encountered in practical applications. Here we examine the effect of context observability on document clustering performance. To do so, we again use the NUS-WIDE 13-animal subset as described previously, then vary the amount of observing context observation $x_j$ with missing proportion ranges from 0% to 100%.

| Missing (%) | Purity | NMI | RI | F-score |
|---|---|---|---|---|
| 0 % | 0.407 | 0.298 | 0.901 | 0.157 |
| 25 % | 0.338 | 0.245 | 0.892 | 0.149 |
| 50 % | 0.320 | 0.236 | 0.883 | 0.137 |
| 75 % | 0.313 | 0.187 | 0.860 | 0.112 |
| 100 % | 0.306 | 0.188 | 0.867 | 0.119 |

Table 4.3: Clustering performance with different missing proportion of context observation $x_j$.

Table 4.3 reports the result. We make two observations: a) utilising context results in a big performance gain as evidenced in the difference between the top and bottom row of the table, and b) as the proportion of missing context starts to increase, the performance degrades gracefully up to 50% missing. This demonstrates the

robustness of model against the possibility of missing context data.

## 4.5 Closing Remark

This chapter has addressed the problem of multilevel clustering when data are organised into groups and the number of clusters in multilevel are unknown. We consider handling the availability of group-level context information to improve the clustering and modelling performance.

We have introduced MC$^2$ model for multilevel clustering. Our model provides a single joint model for utilising group-level contexts to form group clusters while discovering the shared topics of the group contents at the same time. We provide a collapsed Gibbs sampling procedure and perform extensive experiments on three real-world datasets in both text and image domains. The experimental results using our model demonstrate the importance of utilising context information in clustering both at the content and at the group level. Since similar types of contexts (time, tags, locations, ages, and genres) are commonly encountered in many real-world data sources, we expect that our model will also be further applicable in other domains.

Our model contains a novel ingredient in DP-based Bayesian nonparametric modelling: we propose to use a base measure in the form of a product between a context-generating prior $H$ and a content-generating prior $\mathrm{DP}(vQ_0)$. Doing this results in a new model with one marginal being the DPM and another marginal being the nDP mixture, thus establishing an interesting bridge between the DPM and the nDP. Our product base measure construction can be generalized to yield new models suitable for data presenting in more complicated nested group structures (e.g., more than 2-level deep).

# Chapter 5

# Topic Model Kernel with Features Extracted from Multilevel Models

Chapter 3 contains the infinite stream data segmentation and pattern extraction in an unsupervised manner. In chapter 4, our multilevel Bayesian nonparametric approaches present an unsupervised learning setting in which there is no label provided. In many situations where the group-level label (e.g., image category, document class) is provided, our task is to classify the groups given the individual observations which can be noise and in high dimension. For example, given a multilevel data of words organized into documents, each document is represented by a collection of words. Our aim is to summarizing and classifying the documents (document labels are available for supervised learning task). We aware that the word observations are in very high dimension (depending on the vocabulary size) and likely noise.

Therefore, there is a need in using the low-dimensional mixtures extracted from topic models embedded inside the high-dimensional data as an alternative approach to extract features for classification. Representing data by dimensional reduction of mixture proportion extracted from topic models is not only richer in semantics interpretation, but could also be informative for classification tasks. To make use of this feature properly, we propose the Topic Model Kernel (TMK), a high-dimensional mapping for classification the data generated from multilevel models. Inherit the property of Jensen-Shannon divergence, TMK is well representing the similarity and difference between probabilistic features. The applicability of our proposed

kernel is demonstrated in several classification tasks on real-world datasets. TMK outperforms existing kernels on the distributional features and gives the comparative results on non-probabilistic data types.

The rest of this chapter is organised as follows. The low-dimensional representation by topic models is presented in Section 5.1. Recall the multilevel probabilistic models of LDA (Blei et al., 2003), HDP (Teh et al., 2006) and MC2 (Nguyen et al., 2014) from Section 2.2 and Section 2.3 in Chapter 2, we further present the background material of Support Vector Machine (Cortes and Vapnik, 1995) and kernel methods in Section 5.2. Then, we describe the Topic Model Kernel in Section 5.3. Next, the experiment and analysis are performed in Section 5.4. Finally, we present the summary of the chapter in Section 5.5.

## 5.1 Topic Models for Feature Representation

Data representation is critical in data analysis tasks. Central to Support Vector Machines are kernels, which maps the input data to another dimensional spaces in which the linear separating hyperplanes are easier to construct. Given a mapping function $\phi$ and two data points $(x_i, x_j)$, the kernel function $k$ computes inner product $k\left(\phi(x_i), \phi(x_j)\right)$ without explicit computation of $k(\phi(x_i))$ and $k(\phi(y_i))$ separately. Several kernels have been introduced in literature that has examined appropriate kernels for a wide variety of data. Each dataset requires careful choice of the appropriate kernel for SVM classification. In this chapter we focus on a class of problem for SVM when the feature input can be conveniently represented in distributional forms. Such distributions constitute rich information one can exploit, as they are outputs from the probabilistic topic models (Blei et al., 2003) whose latent variables can be used as distributional representation for data. Examples include Probabilistic Latent Semantic Indexing (PLSI) (Hofmann, 1999), Latent Dirichlet Allocation (LDA) (Blei et al., 2003) or Hierarchical Dirichlet Processes (HDP) (Teh et al., 2006), which can produce multinomial distributions over topics given text data or raw pixels in images. This representation is not only *richer in semantics* than the original bag of words, but also (Blei et al., 2003) have demonstrated that the topic model features could be *more informative* for classification than the raw word feature as demonstrated in (Blei et al., 2003). Moreover, such derived features *occupy*

*only 0.04 percent* in space compared to a very large raw feature set of individual words.

The combinations of generative approaches (such as LDA, HDP) with discriminative ones (e.g., SVM) have recently shown to be very effective (Fritz and Schiele, 2008; Phung et al., 2012). Hence it is attractive to expose methods integrating these statistical models and discriminative classifiers. Furthermore, we are motivated by recent successful applications of Jensen-Shannon divergences to compute the similarities and distances when the data are drawn from probabilistic distributions (Antolín et al., 2009; Wartena and Brussee, 2008; Nguyen et al., 2013b).

In this chapter, firstly we make use of preprocessing raw data by topic models for extracting the latent feature in probabilistic space. The probabilistic feature is then utilised for classification task. We propose of a proper kernel originated from the Jensen-Shannon divergence (Endres and Schindelin, 2003), namely Topic Model Kernel (TMK) for optimizing the discriminative among these features. The source code is released at the first author webpage[1] (Nguyen et al., 2013c, 2015b). The recent advance in Bayesian nonparametric modelling, such as the HDP (Teh et al., 2006) which automatically determine the number of topics, make the proposed classification framework more attractive to real-world application. We conducted extensive experimental validation of the proposed TMK which outperforms other existing kernels on the *probabilistically derived features* and yields a comparative performance on *other data types* (non-distribution guarantee).

## 5.2 Background

To provide the context and lay the ground work, we briefly review three related body of work: support vector machine, kernel method and probabilistic topic models.

[1]source code is available at http://www.prada-research.net/∼tienvu/code/

### 5.2.1 Support Vector Machines and kernel method.

Support Vector Machines (SVM) (Cortes and Vapnik, 1995) is a very well-known supervised learning method for classification. The SVM optimization equation (Boser et al., 1992; Chang and Lin, 2011) for binary case is expressed as:

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i \\ \text{subject to} \quad & y_i \left( w^T \phi(x_i) + b \right) \geq 1 - \xi_i \\ & \xi_i \geq 0 \end{aligned} \tag{5.2.1}$$

where $(x_i, y_i)$ is a set of instance-label pairs; $x_i \in \mathcal{R}^n$ and $y \in \{1, -1\}^l$ and $\xi_i$ is a slack variable. For muticlass SVM, it aims to assign labels $y \in 1, 2, 3, ...m$ to each instance which is typically reduced the single multiclass issue into multiple binary classification tasks. A mapping function $\phi(x)$ here becomes $X \rightarrow \mathcal{R}^M$.

SVM is laying within a broader umbrella of kernel methods (Shawe-Taylor and Cristianini, 2004) that approaches the supervised learning problem by mapping the data into a high-dimensional feature space. The goal is to find a better representation by this mapping transformation. Because the mapping can be general, there are numerous existing kernels in literature, including Exponential Kernel, Laplacian Kernel, Inverse Multiquadric Kernel, Cauchy Kernel, and so on. Each kernel is taking into account for different 'genres' of the real-world data. Some examples of kernel functions are summarized below.

- Radial Basic Function Kernel (RBF): $k(x, y) = \exp(-\gamma \parallel x - y \parallel^2)$. It is recommended as the first choice for Support Vector Machine (Chang and Lin, 2011). The parameter $\gamma$ plays a crucial role in the classification performance.
- Linear Kernel: $k(x, y) = x^T y + c$ where $c$ is a constant.
- Polynomial Kernel: $k(x, y) = \left(\alpha x^T y + c\right)^d$ with polynomial degree $d$.
- Sigmoid Kernel: $k(x, y) = \tanh\left(\alpha x^T y + c\right)$ where slope parameter $\alpha$ needs to be adjusted for the best performance.
- Inverse Multiquadric Kernel: $k(x, y) = \frac{1}{\sqrt{||x-y||^2 + c}}$ where $c$ is a constant.

- Power Kernel: The Power kernel is also known as the (unrectified) triangular kernel. It is an example of scale-invariant kernel (Fleuret and Sahbi, 2003) and is also only conditionally positive definite (Boughorbel et al., 2005): $k(x, y) = -||x - y||^{-\beta}$ where $\beta$ is a parameter from $0 < \beta < 1$.

- Spline Kernel: the Spline kernel is given as a piece-wise cubic polynomial, as derived in the works by Gunn (1998). With $x, y \in \mathcal{R}^d$, we have:

$$k(x, y) = \prod_{i=1}^{d} \left( 1 + x_i y_i + x_i y_i \min(x_i, y_i) - \frac{x_i + y_i}{2} \min(x_i, y_i)^2 + \frac{\min(x_i, y_i)^3}{3} \right)$$

- Cauchy Kernel: the Cauchy kernel comes from the Cauchy distribution (Basak, 2008). It is a long-tailed kernel and can be used to give long-range influence and sensitivity over the high dimension space. The kernel is defined by the kernel function with smoothing parameter $\sigma$ as $k(x, y) = \frac{1}{1+\frac{||x-y||^2}{\sigma^2}}$.

Kernel selection is heavily dependent on the data types. For instance, the linear kernel is important in large sparse data vectors and it can be seen as the simplest of all kernels. Whereas, the Gaussian (or RBF) are general purpose kernels used when prior knowledge about data is not available. It decreases with distance and ranges between 0 (in the limit) and 1 (when $x = y$). The polynomial kernel is widely applied in natural language processing (Goldberg and Elhadad, 2008) while Spline Kernel is usually reserved for continuous-space image processing (Horbelt et al., 2000). Because classification accuracy heavily depends on kernel selection, researchers had proposed to have kernel functions based on a general purpose learning and domain specific. A specific data type requires a suitable kernel for their best performance as working with SVM classification. The most appropriate kernel must guarantee the smoothness amongst data within the same class and maintain distinction to others classes. In this chapter, we propose TMK for the probabilistic feature derived by topic models.

We are motivated by the importance of the low-dimensional features derived by topic models. In real-world applications, e.g., text analysis, the raw data always are represented in high-dimensional, which the dictionary size can be thousand or hundred thousand dimensions. Therefore, extracting the low-dimensional hidden feature embedded inside the raw data is essential for *richer in semantic* and *informative* for

classification.

We choose the four baseline kernels: RBF, Linear, Polynomial, and Sigmoid, for comparison with the proposed kernel. The four kernels, which are built-in in LibSVM (Chang and Lin, 2011), are being used extensively as a common choice for classification with SVM.

### 5.2.2 Probabilistic topic models

The discrete distribution features in practice can be the outcome from probabilistic topic models that has become popular in modern machine learning. At the first glance, the probabilistic mixture models, can be seen as mixture distribution, comprise an underlying set of distributions transforming the complex data into a group of simpler densities. Blei et al. (2003) introduce the topic model, Latent Dirichlet Allocation which is a class of topic models providing a simple way to analyse large volumes of unlabeled text. A 'topic' consists of the cluster of words that frequently occur together. There are $K$ topics $\phi_k$, $k \in \{1, ..., K\}$ which are discrete distributions over words. For example, a topic 'sport' may contain high probabilities to such words as 'athlete', 'tennis', 'championship'. Then, each document is assumed to be characterized by a mixture of topics. Our focus is on document feature representation, the mixture proportion (the latent variable $\pi_j$ on Fig. 2.2.5) which is a $k$-dimensional vector. Each element $k$-th of vector $\pi_j$ indicates how much the document $j$ contributes to the topic $k$-th. Traditionally, we need to input the number of topic $K$ for the model. However, Bayesian nonparametric models, such as Hierarchical Dirichlet Process (Teh et al., 2006), can identify the suitable number of $K$. The good model guarantees to return the posterior distribution of the underlying expressive factors for the observed data.

These topic models (e.g., LDA, HDP) are designed to work with a single data channel (e.g., word observations in a document). To accommodate the additional context information (e.g., timestamp, location) (Nguyen et al., 2014) have recently proposed the Multilevel Clustering with Context model (MC$^2$). To demonstrate our Topic Model Kernel, we consider the extracted feature from all of three settings: (1) traditional single observation in parametric (fixed number of topic), (2) single

observation in nonparametric (the number of cluster is automatically identified), and (3) multiple observations in nonparametric setting (e.g., word, timestamp, location, etc). For single observation, there are noticeably Latent Dirichlet Allocation (Blei et al., 2003) in parametric setting and Hierarchical Dirichlet Processes (Teh et al., 2006) in nonparametric configuration. For multiple observation, we consider the Multilevel Clustering with Context (MC$^2$) (Nguyen et al., 2014). The detailed generative processes and posterior inferences behind these prototypes can be found in the original papers (Blei et al., 2003; Teh et al., 2006; Nguyen et al., 2014). Essentially, the algorithm proceeds by looping iteratively through each of the data points and performing MCMC moves on the cluster indicators for each point.

For further details, we refer the readers to Sec. 2.2.3 for LDA, Sec. 2.3.3 for HDP and Sec. 4.3 for MC$^2$.

## 5.3 Topic Model Kernel

In this section, we firstly present the Kullback–Leibler divergence and Jensen–Shannon divergence in information theory. Then, we propose the Topic Model Kernel for classifying the probabilistic features with SVM.

### 5.3.1 Kullback–Leibler Divergence

The Kullback–Leibler (KL) divergence (Kullback and Leibler, 1951), introduced in information theory and probability theory is a non-symmetric measure of the similarity between two probability distributions. Its intuitive understanding arises from likelihood theory (Shlens, 2007) measuring the distance between the initialized probability parameter and the estimated distribution from its generated instances. The KL divergence from distribution $P$ to $Q$ for discrete case is defined as:

$$D_{KL}\left(P \parallel Q\right) = \sum_{i} P(i) \ln \frac{P(i)}{Q(i)}$$

and for continuous distributions as:

$$D_{KL}\left(P \parallel Q\right) = \int_{-\infty}^{+\infty} p(x) \ln \frac{p(x)}{q(x)} dx$$

where $p$ and $q$ denote the densities of the distributions P and Q. Moreno et al. (2003); Chan et al. (2004) have proposed a symmetric KL divergence kernel for classifying objects under the generative model of Gaussian mixture, a step toward classifying distribution data with SVM.

### 5.3.2 Jensen–Shannon Divergence

Based on the KL divergence, the Jensen-Shannon (JS) divergence (Endres and Schindelin, 2003) calculates the distance between two probability distributions $P$ and $Q$ as:

$$D_{JS}\left(P, Q\right) = \pi D_{KL}\left(P \parallel M\right) + \left(1 - \pi\right) D_{KL}\left(Q \parallel M\right)$$

where $M = \frac{1}{2}\left(P + Q\right)$ and $D_{KL}$ is the KL divergence discussed in Section 5.3.1. The lower bound of JS divergence is 0 when the two distributions are identical. Its square root (Endres and Schindelin, 2003) is proof as an asymptotic approximation to the well-known $\chi^2$ and being a metric with the triangle inequality property for two distributions. This distance can be seen (in the symmetric KL flavour) as the average distance between two random distributions to their empirical mean, with $\pi$ is set as 0.5 (Chan et al., 2004). Another interesting property of JS divergence is negative definite on $R_+ \times R_+$ (Topsoe, 2003) that will be useful when we verify for kernel validation.

### 5.3.3 Topic Model Kernel

The kernel function is basically a measurement criteria that compares the similarity between two points or vectors. But not all of the measurement distances or similarity functions yield proper attributes to be a valid kernel. The Topic Model Kernel

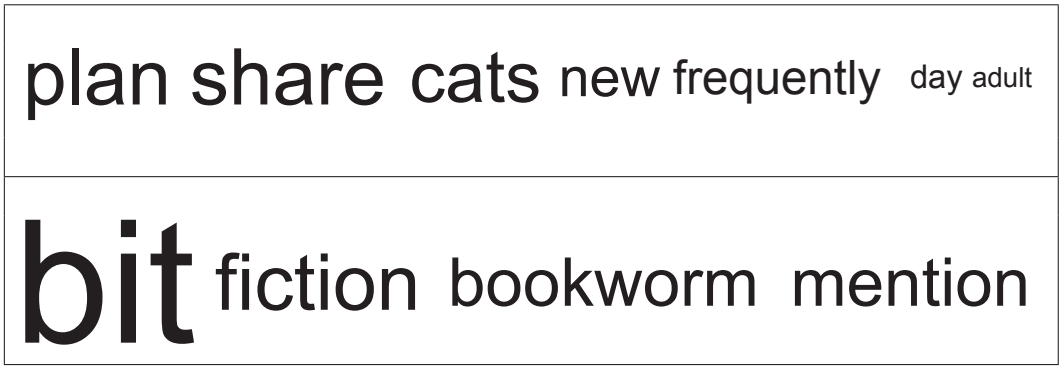



Figure 5.4.1: Two examples of LDA topic $\phi_k$ on LiveJournal data.

(TMK) is defined following:

$$K_{TMK}(X,Y) = \exp\left\{-\frac{1}{\sigma^2} \times D_{JS}(X,Y)\right\}$$
$$= \exp\left\{-\frac{1}{\sigma^2} \times \left[\frac{1}{2}\sum_i X(i) \ln \frac{X(i)}{M(i)} + \frac{1}{2}\sum_i Y(i) \ln \frac{Y(i)}{M(i)}\right]\right\} \quad (5.3.1)$$

By exponentiating the negative JS divergence, it leads to the positive definite kernel function $K_{TM}$ because (1) JS divergence is negative definite on $R_+ \times R_+$ (Topsoe, 2003), (2) let exponentiate the negative of JS divergence giving the positive definite kernel that projecting the divergence distance into the bounded range of 0 and 1. Thus, TMK satisfies the Mercer condition of $c^T K_{TM} c \geq 0$ with $K_{TM(i,j)} = k_{TM}(x_i, x_j)$ for the validity of the kernel. The variance $\sigma^2$ plays a role as a shape parameter to flexibly flat or widen the data.

## 5.4 Experiment Results and Analysis

Experiments are conducted using real-world data in various classification scenarios, including:

- The topic model features derived from single observation in parametric form of LDA or nonparametric counterpart as HDP.
- The extracted feature from multiples observation of MC$^2$ model.

- The generic features are obtained from other sources that we do not guarantee them fit into any type of distribution.
- We analyse the kernel performance on parameter space to verify our kernel's superiority on the probabilistic features.
- We demonstrate a possible way of classification as combined product of raw feature and topic model feature for better performance in classification.

We use the LibSVM (Chang and Lin, 2011) as a standard library to compare the proposed kernel with four baseline LibSVM built-in kernels: Radial Basic Function Kernel, Linear Kernel, Polynomial Kernel, and Sigmoid Kernel. The data will be scaled as recommended in LibSVM to ensure the best performance. We focus on the multi-class classification problem viewed as multiple binary classification problems.

The scores are reported at two types of parameter: the default parameter (set by LibSVM) and the optimal parameter by brute-force cross validation searching (as the default parameter sometimes cannot provide the best performance). For Topic Model Kernel, we empirically set the default parameter $\sigma^2$ is equal to the feature dimension size after observing TMK operations on several datasets. Throughout this experiment, the whole data is randomly splited into 10 sets, which comprise of training set and testing set such that the instances in testing is not appearing in training set.

### 5.4.1 Topic Model Features

LDA and HDP are used to model the single observation data (e.g., words in a document). We run LDA and HDP to extract the mixture proportions $\pi_j$ on Livejournal, Reuter21578, and LabelMe dataset. LDA is carried out on Live Journal and Reuter21578 datasets and HDP on LabelMe to extract the mixing proportion features, then use SVM for classification with the proposed kernel.

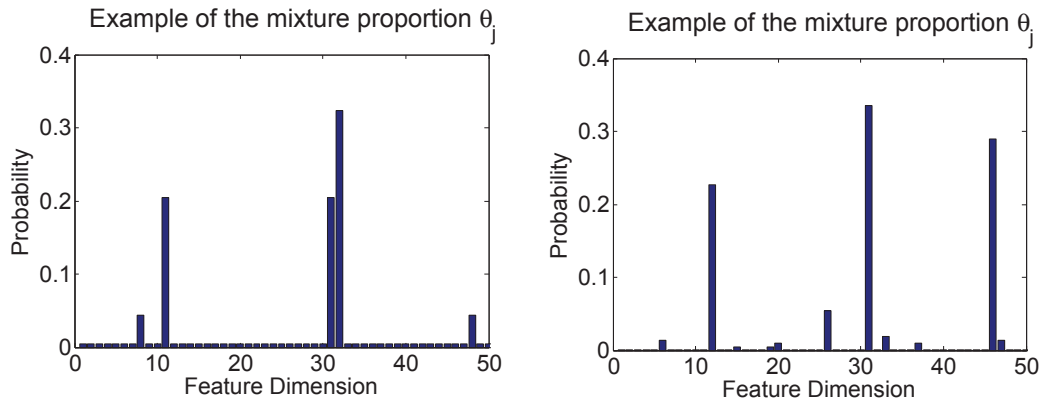


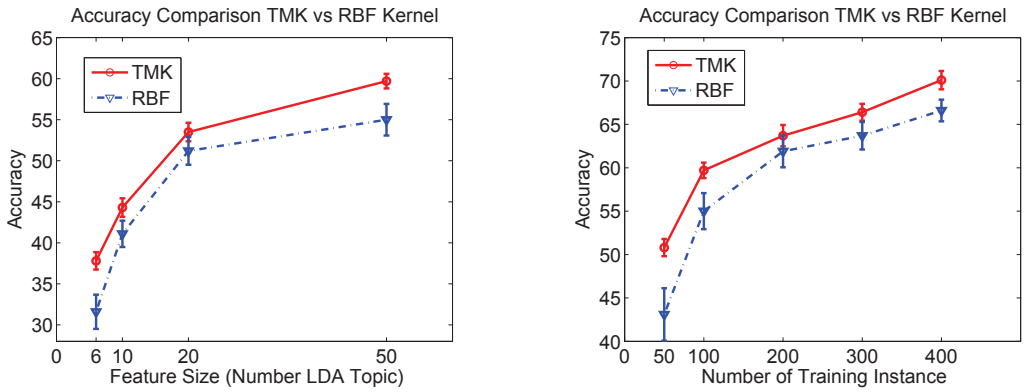

Figure 5.4.2: Two examples of the reduced feature $\pi_j$ by LDA from 65,483 to 50.

#### 5.4.1.1 Livejournal Dataset

**Data processing set up:** We crawled the communities listed in the Livejournal directory, retrieved August 2012. These communities are categorised by Livejournal into 10 categories from the 100 communities obtained, summarizing of 8,758 posts giving the vocabulary size of 65,483 which is the feature dimension of raw data. The task is to predict the category, given text data from user's posts. We treat each user post as a document and run LDA with fixed number of latent factors from $\{6, 10, 20, 50\}$. Latent Dirichlet Allocation is carried for the whole dataset with 1000 iteration Gibbs sampling. The examples of estimated topic $\phi_k$, about literature and life, are visualized in Fig. 5.4.1 and our LDA features are in Fig. 5.4.2 which reduced from original high dimension of 65,483 to 50.

**Classification set up:** We do the experiments progressively with increasing numbers of training instances from 10 to 400 (refer Fig. 5.4.3b) and varying the number of hidden factors $K$ (refer Fig. 5.4.3a). The optimal parameter (for the best performance) is achieved with 3 fold validation on training data sets. The performance is judged by averaging 10 random subsets of train/test datasets.

The results in Fig. 5.4.3 and Table 5.1 demonstrate the superiority of our kernel and clearly shows the effect of increasing the number of learned feature or number of training instances.




(a) With varied feature dimension. (b) With varied number of training data.

Figure 5.4.3: Experiments comparison between TMK and RBF kernels on LDA feature derived from Live Journal data.

#### 5.4.1.2 Reuter21578 Dataset

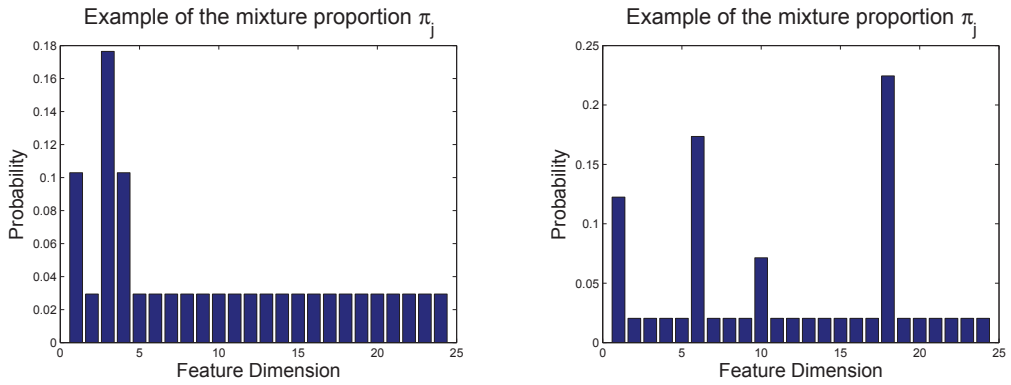
**Data processing set up:** Reuter21578 is a common dataset for text classification. It consists of documents appeared on the Reuters Newswire in 1987. There are totally 10 categories for classification. Similar to Live Journal data, we utilise posterior inference of LDA on Reuters21578 dataset (again using 1000 iterations of Gibbs sampling) to extract the mixing proportion feature $\pi_j$ in which the number of hidden factors is set as $K = 20$.

**Classification set up:** The accuracy comparison is displayed in Table 5.1. The Topic Model Kernel (TMK) outperforms four baseline kernels on this dataset in both cases of parameter (default and optimal). The number of training instances and testing instances are set as 100 instance for each category (totally 1000 instances for training and 1000 instances for testing). The final classification score is reported with standard deviation in 10 randomly experiment subsets. We observed that the optimal parameter for SVM along with the kernel feature, obtained by brute-force searching, slightly increases the accuracy about 1% on LDA feature, whereas in other types of data, the input parameters will make a significant effect on the accuracy (Hsu et al., 2003).

| Accuracy | Default Parameter | | Optimal Parameter | |
|---|---|---|---|---|
| Kernel | LiveJournal | Reuter21578 | LiveJournal | Reuter21578 |
| TMK | **58.10**±2.15 | **81.33**±0.20 | **58.70**±1.78 | **81.87**±0.17 |
| RBF | 54.90±4.93 | 79±0.55 | 55.00±4.85 | 79.40±0.55 |
| Linear | 54.40±5.29 | 78.27±0.13 | 54.90±4.28 | 79.07±0.13 |
| Polynomial | 52.60±6.65 | 77.93±0.10 | 54.20±5.20 | 78.93 + 0.10 |
| Sigmoid | 51.80±5.18 | 77.40±0.48 | 53.50±4.79 | 79.20±0.34 |

Table 5.1: Accuracy comparison of SVM classification on features derived from LDA.


Figure 5.4.4: LabelMe dataset: the learned topics $\phi_k$ by HDP.




a) Highway class. b) Buildings class.

Figure 5.4.5: Two examples $\pi_j$ of the HDP feature on LabelMe dataset.

#### 5.4.1.3 LabelMe Dataset

**Data processing set up:** LabelMe (Oliva and Torralba, 2001) is the well-known benchmark dataset for image annotations and object categorizations that contents a bunch of images and tags. The subset of 8 classes LabelMe is justified for this experiment including ground truth for 8 categories classification consisting of *tall buildings*, *inside city*, *street*, *highway*, *coast*, *open country*, *mountain*, and *forest* in totally 2688 images. To discard the noise and mistagging issues, top 30 high frequency tags are chosen giving a vocabulary size of 30. The Hierarchical Dirichlet Processes (Teh et al., 2006) is carried out to extract the topic assignment feature flexibly, each image is treated as a document while each tag is considered as a word $w_{ji}$ (refer Fig. 4.3.1) in the model. The collapsed Gibbs inference during 500 iterations are collected to compute the posterior. HDP automatically identifies 24 topics $\phi_k$, four of whom is displayed in Fig. 5.4.4 for visualization. The extracted feature $\pi_j$ by HDP is therefore under the dimension of 24 (see Fig. 5.4.5 for two examples) where two different classes are likely to have dissimilar features $\pi_j$.

**Classification set up:** The evaluation procedure is conducted alike the previous experiments that we splits the data into 10 training and testing subsets. In each subset, there are 800 and 800 instances for training and testing respectively (100 instances in each class). Then, we run 3 fold cross validation to get the optimal parameter for testing. The performances with default SVM parameter (the default

| Accuracy | TMK | RBF | Linear | Polynomial | Sigmoid |
|---|---|---|---|---|---|
| Default Parameter | 72.3±1.96 | 73.5±1.99 | **74.5±1.87** | 62.7±4.26 | 72.8±1.66 |
| Optimal Parameter | **76.1**±1.88 | 73.3±2.08 | 73.8±1.46 | 74.5±2.22 | 73.9±1.30 |

Table 5.2: Classification on LabelMe dataset from features learned by HDP.

$\sigma^2$ in TMK is set at feature size of 24) and optimal value are recorded in Table 5.2. Due to the sparsity of image tag and extracted feature, Linear kernel achieves the best performance at the default parameter. However, our TMK attains the best performance at the optimal parameter.

### 5.4.2 Topic model features from multiple observations model

Previously, we have illustrated experiments on the feature derived from LDA and HDP running on single observation dataset. In this section, we aim to learn the performance of the topic model feature extracted from model with utilising information from multiple observation (e.g., words, timestamp, authors in a document) for comparison.

#### NUS-WIDE dataset

**Data processing set up:** We set up to run Multilevel Clustering with Context (MC²) (Nguyen et al., 2014) on NUS-WIDE subset of the 13-class animals with 2054 images. We use the available image label for image classification task. The feature vector includes 1000-dim annotation and 500-dim bag-of-word SIFT. We consider two cases of experiment on MC2 model: (1) running multiple observations of both annotation and SIFT (multiple observation), and (2) running the model on single observation of SIFT only (single observation). The posterior inference of MC² returns 15 topics. The topic model feature for each image in this MC² is not directly obtain like LDA or HDP. We compute the mixture proportion feature for each image using the latent indicator (variable $l_{ji}$ in Fig. 4.3.1). The mixture proportion $\pi_j$ for an image $j$-th is a vector in $M$-dimension, where $M$ is the number of topic discovered

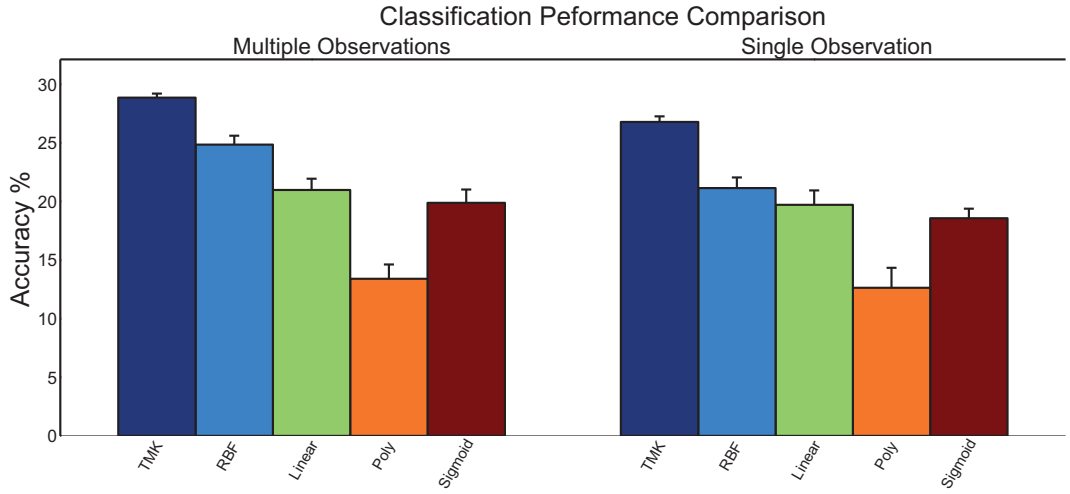



Figure 5.4.6: Classification accuracy comparison on NUS-WIDE dataset from featured learned by the MC2 in two settings: multiple observations (with context information) and single observation (without context information).

by MC². Each element $\pi_{jm}$ is computed in such a way similar to HDP (Teh et al., 2006).

**Classification set up:** The classification comparison between multiple kernels is displayed in Fig. 5.4.6 where we use 100 images per class (totally 1300 images) for training and 50 images per class (totally 650 images) for testing. Standard deviation error is calculated across 10 randomly experiments. Our kernel achieves the best performance among other kernels. Fig. 5.4.6 presents the scores at default parameter (optimal parameter yields similar performance). The classification performance from the feature obtained by multiple observation slightly better than single observation. The extracted features from multiple data-source model can be richer in semantics and more informative for classification than the feature from single data-source model counterpart. There are two proper reasons for this claim. The first reason is that the context information from multiple data source prevents the model from over-fitting to the single data source. Another reason is the additional data channel offering the multi-view information toward the data instances in different categories. Therefore, jointly modelling multiple data observation produces informative features which are improving performance for classification.

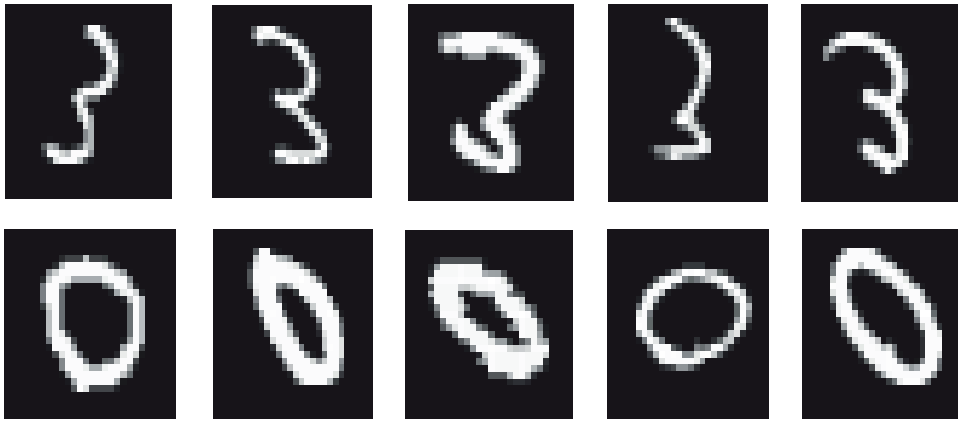


Figure 5.4.7: Examples of digit 3 and 0 in MNIST dataset.

### 5.4.3 Non-distributional data source

To highlight the applicability of the TMK, we show how the proposed kernel performs on the raw data of MNIST dataset *instead of extracting topic model features as previously*. This experiment is aiming to discover the wide applicability of TMK on such kind of non-distribution data.

#### MNIST Dataset

**Data processing set up:** The MNIST dataset (LeCun et al., 1998) is a well-known dataset of handwritten digits, referred as a standard benchmark for many tasks, especially in classification problem. The ready-to-use extracted feature is available at author website (http://yann.lecun.com/exdb/mnist/) with the classification performances and the state of the art result on 60,000 training and 10,000 testing instances. In this experiment setting, we do not aim to beat the state of the art result on MNIST, but we want to illustrate the classification comparison between the TMK versus others with SVM tool.

**Classification set up:** We randomly pick up 100 items for training and another 100 for testing set and run for 10 times. We do not run for the whole 60,000 training vs 10,000 testing due to (1) resource limitation when constructing the gram matrix of the huge data (2) our goal is to proof the efficiency of the TMK by comparing

| Kernel | Default Parameter | Optimal Parameter |
|---|---|---|
| TMK | 88.4±0.9 | **90.8**±2.25 |
| RBF | 82.2±1.9 | 89.3±2.5 |
| Linear | **91.3**±0.5 | 88.4±2.7 |
| Polynomial | 83.7±2.6 | 88.7±2.9 |
| Sigmoid | 79.6±3.4 | 85.8±2.7 |

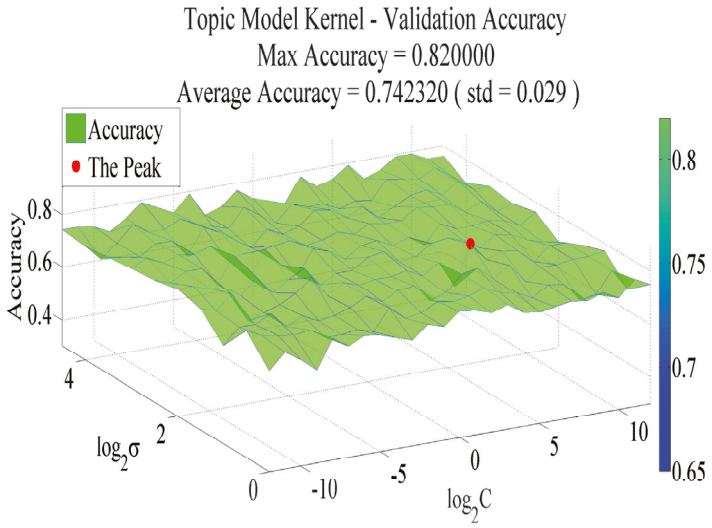
Table 5.3: Classification comparison on raw feature of MNIST dataset.

with other kernels, not strike the state of the art result. The feature dimension of each gray image is 784 (28×28 pixels) at which the pixel value ranges from 0 to 255 (refer Fig. 5.4.7 for examples). We note that this kind of raw image data is not pledged to drawn from any type distribution when use with Topic Model Kernel for classification. The accuracy is displayed in Table 5.3, although Linear kernel perform very well with default parameter, our kernel achieves the best result in optimal parameter (with brute force searching on validation set). The detailed performance on parameter space of MNIST dataset is discussed in the next section.

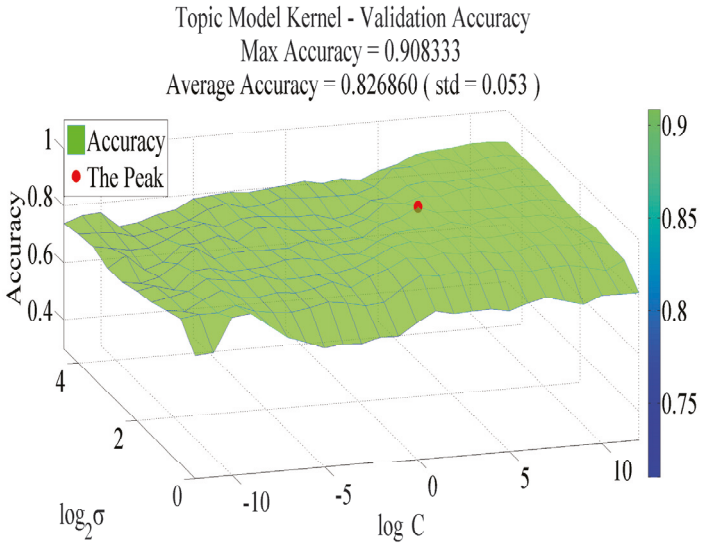
### 5.4.4 Parameter selection analysis

We now move on to our characterization of performance on various axes of parameters. To demonstrate the TMK is more robust on the parameter space, we record the accuracy planes with parameter of $C$ in equation 5.2.1 for SVM and TMK parameter $\sigma$ shown in Fig. 5.4.8a). We get the peak accuracy of 0.82 on by 3 fold cross validation at which the optimal parameter is further used for testing. The average accuracy with standard deviation is used to evaluate the preeminent of TMK when the data is drawn from distribution. Topic Model Kernel accomplishes the best in the way that it get the highest score on average accuracy (0.74), lowest standard deviation (0.029), and the TMK's peak (0.82) is the highest among four baseline kernel's peaks (refer Table 5.4). Detailed visualization performances of the baseline kernels on HDP feature are illustrated in Fig. 5.4.9. We observe that RBF, Linear, and Sigmoid kernels are quite stable than Polynomial kernel.

Further, we would like to see the performance on the non-distribution feature when varying the parameters of TMK. Although it is not really stable (with high standard deviation and lower average accuracy on the grid), it performs pretty well with




(a) HDP feature on LabelMe dataset.




(b) Raw feature on MNIST dataset.

Figure 5.4.8: Topic Model Kernel cross validation accuracy by brute-force parameter searching.

| Kernel | LabelMe: HDP Feature | | | MNIST: Raw Feature | | |
|---|---|---|---|---|---|---|
| | Peak | Average | Std | Peak | Average | Std |
| TMK | **0.82** | **0.74** | **0.029** | **0.91** | **0.83** | 0.053 |
| RBF | 0.77 | 0.70 | 0.033 | 0.90 | 0.67 | 0.252 |
| Linear | 0.75 | 0.70 | 0.034 | 0.88 | 0.80 | **0.029** |
| Polynomial | 0.75 | 0.35 | 0.236 | 0.88 | 0.44 | 0.295 |
| Sigmoid | 0.76 | 0.69 | 0.041 | 0.85 | 0.43 | 0.304 |

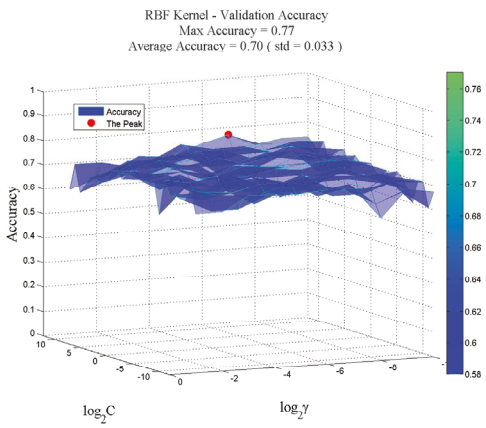
Table 5.4: Cross validation accuracy on parameter space comparisons of probabilistic feature of HDP versus non-probabilistic feature (or raw feature).

comparable accuracy to other kernels at a certain area (can be obtained by cross validation).

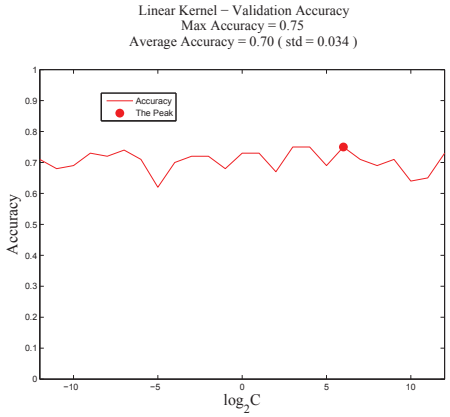
### 5.4.5 Improved classification performance with feature combination

To analyse the classification performance under different feature kinds, we compare performances with various features including raw feature, extracted feature by HDP, extracted feature by MC2. Here, we use the MC2 with single observation (without context information) to be fair classification comparison with HDP and raw feature. In addition, we want to improve the classification by using feature produced by combining these individual features. The mixture proportion extracted (from raw data) by topic models offers an additional view to the data. It captures information from statistical perspective, representing proportion over underlying topics (Blei et al., 2003). By focusing on the underlying topics, the topic model features ignore the noise information from the data. Two documents in the same class would likely to have similar mixture proportions.

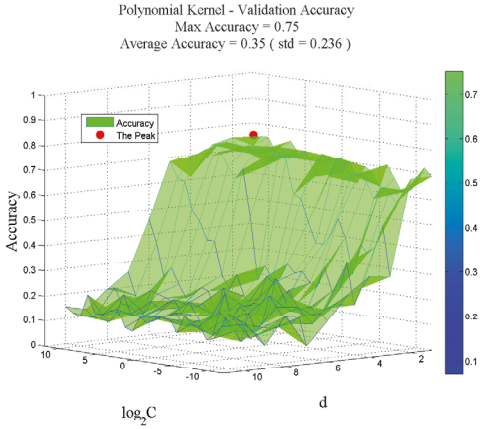
We perform experiments on NUSWIDE dataset (similar to Section 5.4.2) with various features for comparison (refer Table 5.5). HDP and MC2 models produce features which get similar performance for classification. MC2 feature slightly excesses HDP feature in classification but it is not distinction. We use MC2 model with annotation as the additional context information (multiple observations case). The raw feature itself attains better classification performance compared to features extracted by HDP and MC2. The possible reason for it is that the raw feature



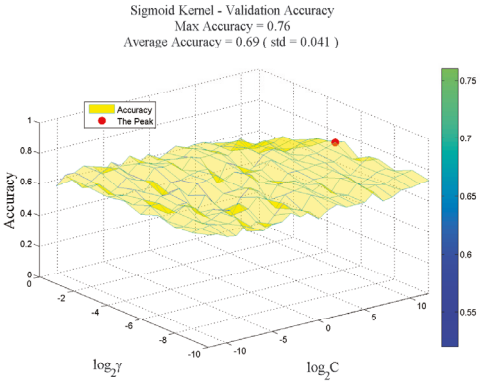

(a) RBF kernel: parameter $\gamma$ and $C$




(b) Linear kernel: parameter $C$




(c) Polynomial kernel: parameter $d$ and $C$




(d) Sigmoid kernel: parameter $\gamma$ and $C$

Figure 5.4.9: Accuracy on parameter space of four baseline kernels on HDP feature of LabelMe dataset.

| Feature | RBF | Linear | Poly | Sigmoid | TMK |
|---|---|---|---|---|---|
| Raw | **32.98±0.3** | 30.82±0.2 | 9.72±0.6 | 32.85±0.3 | 32.75±0.2 |
| HDP | 29.29±0.7 | 27.95±0.4 | 19.17±1.3 | 28.48±0.5 | **30.20±0.5** |
| MC2 | 29.31±1.1 | 27.57±1.5 | 18.22±1.0 | 27.09±0.3 | **31.95±0.3** |
| Raw+HDP | 33.09±0.4 | 31.31±0.3 | 10.12±1.2 | 32.86±0.4 | **37.73±0.3** |
| Raw+MC2 | 32.80±0.3 | 32.15±0.3 | 9.71±0.8 | 32.71±0.3 | **37.74±0.3** |

Table 5.5: Classification with different features on NUSWIDE dataset.

dimension is 500 while HDP and MC2 features are only 15. The higher dimension feature contains richer information toward the data. Furthermore, the features combined by Raw+HDP and Raw+MC2 achieve the best performance for classification. The joined features are better than the raw feature itself and the topic model feature (HDP or MC2) individually. Here, we do not include experiment from LDA because the performance of LDA can be seen from HDP. HDP is a Bayesian nonparametric counterpart of LDA, e.g., HDP (Teh et al., 2006) automatically identifies the suitable number of topics while LDA does not. If we fix the number of topic, HDP extracted feature should be similar to LDA extracted feature. Our proposed kernel demonstrates its superior performance on these features comparing to other baseline kernels.

## 5.5 Closing Remark

In this chapter, we have introduced the Topic Model Kernel (TMK) for data classification. Our task is classifying group-level given the individual observations in each group. Hence, we rely on the probabilistic feature, extracted from multilevel models (e.g., LDA, HDP, MC2), representing for each group to perform classification. These extracted features are more condensed, richer in semantic, and more informative for classification than the raw feature. The experimental results show the feasibility of the proposed kernel on not only the probabilistic data but also the generic types of data (non probabilistic).

The significant applications of this work in real-world data are examined on the probabilistic features derived from recent topic models of LDA, HDP, and MC$^2$. Further, we show that the probabilistic feature extracted from multiple observation

model is better than from single observation model. Moreover, we explore that combining raw feature with the extracted feature from probabilistic model would increase the performance. Detail analysis of the performance w.r.t. the parameter space is also provided.

In the next chapter, we continue our investigation into multilevel analysis for supervised learning problem where we observe the continuous outcome from multilevel data.

# Chapter 6

# Bayesian Nonparametric Multilevel Regression

The majority of our works in the previous chapters, e.g., Chapter 3 and Chapter 4, have been formulated in an unsupervised setting (a.k.a. clustering) where the label of data (in multilevel structure) is unknown. As we have highlighted previously, the underlying problem with these clustering models is that the number of clusters is not provided and can be varied with the data observed. Therefore, the model needs to infer and adapt these number of clusters from the data. Then, in Chapter 5, we have considered the case when we observe the group-level label (or categorical outcome) for *classification* using the features learned by multilevel models. However, to our knowledge, no previous work has addressed the case when we observe the continuous outcome from multilevel data – a setting we call *Bayesian nonparametric multilevel regression*.

Regression is at the cornerstone of statistical analysis. Multilevel regression, on the other hand, receives little research attention, though it is prevalent in economics, biostatistics and healthcare just to name a few. In this chapter, we present a Bayesian nonparametric framework for multilevel regression where *individuals* including observations and outcomes are organised into *groups*. Furthermore, our approach exploits additional group-specific context observations to improve the modelling performance. From the theoretical perspective, we use Dirichlet Process with product-space base measure in a nested structure to model group-level context distribution

and the regression distribution to accommodate the multilevel structure of the data. The proposed model simultaneously partitions groups into a cluster and perform regression. We provide collapsed Gibbs sampler for posterior inference.

Our contributions in this chapter includes: (1) a novel model for multilevel regression which handles group-level context and partition groups into clusters, (2) a novel application on regression prediction on individuals from unseen group which has not been observed during training, (3) we perform extensive experiments on econometric panel data and healthcare longitudinal data to demonstrate the effectiveness of the proposed model.

The remainder of this chapter is organised as follows. Section 6.1 presents our overview on the problem of multilevel regression. Next, we describe the multilevel regression problem and related works in Section 6.2. This is followed by our framework in Section 6.3. The application and experiment of our model is demonstrated in Section 6.4. Finally, Section 6.5 concludes our chapter with the closing remarks.

## 6.1 Overview

As we have mentioned earlier about multilevel data in Section 2.4, real data is complex and hardly conform to simple flat structure or a well-defined regular pattern. Multilevel, or hierarchical and nested, data structure persists in almost every day analysis tasks. Patients organised in different cohorts in multiple hospitals; economic activities of a city nested within a state, which is in turn influenced by national economic status and so on. Multilevel analysis (Hox, 2010; Leyland and Goldstein, 2001b; Snijders, 2011) is an approach to analyse group contexts as well as the individual outcomes. In multilevel analysis, multilevel regression are commonly used in econometrics (panel data), biostatistics and sociology (longitudinal data) for regression estimation. Examples include panel data measures GDP observations over a period of time tracking in multiple states of the USA or longitudinal studies on a collection of patients' admissions to a hospital. To the best of our knowledge, almost no work of multilevel regression has attempted to model group context information to form 'optimal' cluster of groups to be regressed together. The main challenge is how to model the optimal or 'correct' clustering to leverage shared statistical

strengths across groups.

In this chapter, we consider the multilevel regression problem in multilevel analysis where *individuals* including observations and outcomes are organised into *groups*. Our modelling assumption is that individuals exhibit similar regression behaviours should be grouped and perform regression task together to leverage on their shared statistical strengths. For example, children with the same parents tend to be more alike in their physical and mental characteristics than individuals chosen at random from large population. Particularly, we focus on the multilevel regression problem for predicting individuals in *unseen groups*, the groups do not appear in the training set. For example, in health research - relied on patient's history of electronic medical record (EMR) - patient history records can be empty for patients have not admitted to a hospital before. Predicting individuals in unseen groups using multilevel regression presents another contribution of our work.

Traditional *single* regression method often treats hierarchical data as flat independent observations. Hence, it tends to mis-specify the regression coefficients, leading to poor fitting in overall populations. The well-known approach to multilevel regression is the Linear Mixed Effect model (McLean et al., 1991; Pinheiro and Bates, 2000). However, it is not well applicable for predicting individuals from unseen groups because the random effect is fixed to the given training groups.

Another way to multilevel regression is via multitask learning where each data group is treated as a *task* and individual seen as *example*s. Multi-task regression aims to improve generalization performance of related tasks by joint learning (Caruana, 1997; Argyriou et al., 2008). A few works have attempted to partition related tasks into task-groups (Kang et al., 2011; Passos et al., 2012). Bayesian nonparametric approach is used to overcome the difficulty in defining the degree of relatedness among tasks (Gupta et al., 2013). For testing and evaluation, previous works use a proportion of examples in each task for training and the rest is further used for testing. Given a testing example, the task which the example belonged to, is identified from the hierarchical structure of the data. Nevertheless, given a testing example from unseen task, there is no proper way to perform prediction.

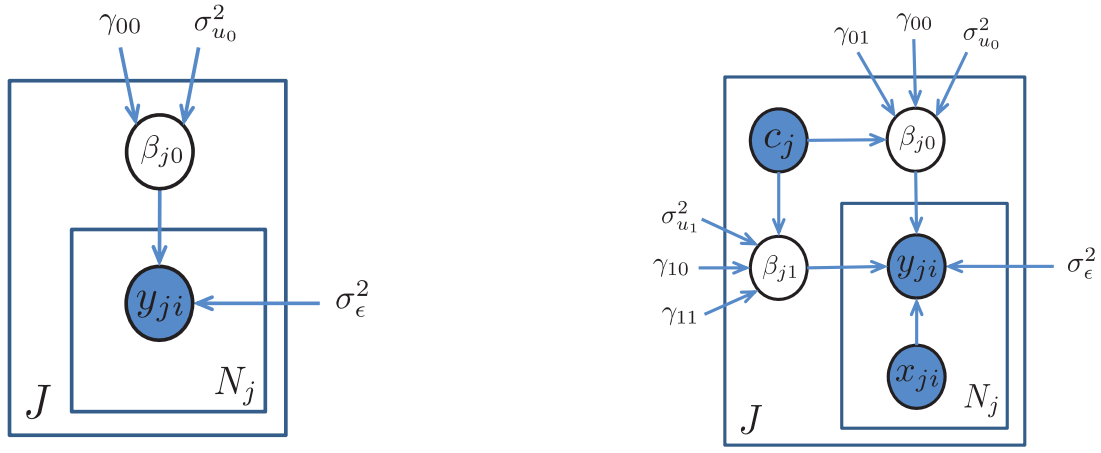
Addressing this gap, we present a *Bayesian Nonparametric Multilevel Regression* (BNMR) model. The proposed framework uses a Dirichlet Process as a product

base-measure of group-context distribution and regression distribution to discover the unknown number of group clusters and do regression jointly. The group cluster is estimated based on the group-context observations and regression outcome of individuals. The goal is making the related groups strengthen each other in regression while unrelated groups do not affect themselves. In addition, simultaneously clustering groups and performing regression can prevent from overfitting to each training group. By using group-context information, the proposed model can assign the unseen group into an existing group-cluster for regression.

## 6.2 Multilevel Regression and Further Related Background

Regression has a long tradition in statistics and machine learning. Within the scope of this chapter, we focus on the regression task that can perform *multilevel regression* where the data presented in groups. Observations in the same group are generally not independent, they tend to be more similar than observations from different groups. Standard single level models are not robust for multilevel data as it assumes the observations across groups are independence. This motivates the need for special multilevel regression framework. Dealing with grouped data, a popular setting known as multilevel analysis (Snijders, 2011; Hox, 2010) has a board applications from multilevel regression (Gelman et al., 2003) to multilevel document modelling and clustering (Nguyen et al., 2014).

We consider a pair of outcome and observation in hierarchical structure ($y_{ji} \in \mathcal{R}, \boldsymbol{x}_{ji} \in \mathcal{R}^d$) where $y_{ji}$ is an outcome (or response) and $\boldsymbol{x}_{ji}$ is an observation for trial $i$ in group $j$. The multilevel models are the appropriate choice that can be used to estimate the intraclass correlation and regression in the multilevel data. Specifically, we consider Linear Mixed Effects models which are extensions of linear regression models for data that are organised in groups. To provide a better context, we recall the basic intercept-only model (cf. Section 2.4).




(a) Intercept-only model.

(b) Linear Mixed Effects model.

Figure 6.2.1: Graphical representation. Left: Intercept Only. Right: Linear Mixed Effects model.

### 6.2.1 Intercept only model

The *intercept-only* model (null model, baseline model) uses only the intercept to explain the data. In this model, the outcome variable $y_{ji}$ in group $j$ is estimated as:

$$y_{ji} = \beta_{j0} + \epsilon_{ji} \tag{6.2.1}$$

where $\epsilon_{ji} \sim \mathcal{N}(0, \sigma^2_\epsilon)$. The variance of error for each individual is denoted as $\sigma^2_\epsilon$ . To make all individuals in the same group share the same parameter, the regression coefficient $\beta_{j0}$ is assumed as: $\beta_{j0} = \gamma_{00} + u_{j0}$ where $u_{ji} \sim \mathcal{N}(0, \sigma^2_u)$ and $\sigma^2_u$ is the variance of error in group level. Therefore, the single equation for the intercept-only model becomes:

$$y_{ji} = \gamma_{00} + u_{j0} + \epsilon_{ji}.$$

In practice, the intercept-only model above is often used as a baseline comparison for evaluation multilevel regression models.

### 6.2.2 Linear Mixed Effects model

As discussed previously in Section 2.4, the LME model (McLean et al., 1991) describes the relationship between a response variable and independent variables in multilevel structure, with coefficients that can vary with respect to one or more grouping variables. A mixed-effects model consists of two parts, fixed effects and random effects. Fixed-effects terms are usually the conventional linear regression part, and the random effects are associated with individual experimental units drawn randomly from population. The random effects have prior distributions whereas fixed effects do not. Linear Mixed Effects model can represent the covariance structure related to the grouping of data by associating the common random effects to observations in the same group. The standard form of a linear mixed-effects model is following:

$$y_{ji} = \boldsymbol{\beta}_{j0} + \boldsymbol{x}_{ji}^T \boldsymbol{\beta}_{j1} + \epsilon_{ji} \qquad \epsilon_{ji} \sim \mathcal{N}(0, \sigma_\epsilon^2)$$

where the regression coefficients for group $j$: $\boldsymbol{\beta}_{j0}$ and $\boldsymbol{\beta}_{j1}$ are computed:

$$\boldsymbol{\beta}_{j0} = \gamma_{00} + \gamma_{01} c_j + u_{j0} \qquad u_{j0} \sim \mathcal{N}(0, \sigma_{u0}^2)$$

$$\boldsymbol{\beta}_{j1} = \gamma_{10} + \gamma_{11} c_j + u_{j1} \qquad u_{j1} \sim \mathcal{N}(0, \sigma_{u1}^2)$$

Therefore, the final form to predict the individual outcome variable $y_{ji}$ using individual explanatory variables $x_{ji}$ and group explanatory variable $c_j$ is followed:

$$y_{ji} = \underbrace{\gamma_{00} + \gamma_{01} c_j + \gamma_{10} x_{ji} + \gamma_{11} c_j x_{ji}}_{\text{fixed effects}} + \underbrace{u_{j0} + u_{j1} x_{ji} + \epsilon_{ji}}_{\text{random effects}}.$$

Fixed effects have levels that are of primary interest and would be used again if the experiment were repeated. Random effects have levels that are not of primary interest, but rather are thought of as a random selection from a much larger set of levels. We present the graphical representation of LME model in Fig. 6.2.1b. The common parameter estimation methods for linear mixed effect include Iterative Generalized Least Squares (Goldstein, 1986) and Expectation Maximization algorithm (Raudenbush, 1992).

### 6.2.3 Linear Regression

Regression is an approach for modelling the relationship between a scalar *outcome* variable $y$ and one or more *explanatory* variables denoted $\boldsymbol{x}$. In linear regression, data are modelled using linear predictor functions, and unknown model parameters are estimated from the data. Given a data collection $\left\{y_i \in \mathcal{R}, \boldsymbol{x}_i \in \mathcal{R}^d\right\}_{i=1}^N$ of $N$ units, linear regression model assumes the relationship between the outcome variable $y_i$ and the $d$-dimension vector of observation $\boldsymbol{x}_i$ is linear. Hence, the model takes the form: $y_i = \boldsymbol{x}_i^T\boldsymbol{\beta} + \epsilon_i$ where $\epsilon_i$ is a residual or error term, $\boldsymbol{\beta}$ is a regression coefficient, including *intercept* and *slope* parameters. The solution for $\boldsymbol{\beta}$ is: $\hat{\boldsymbol{\beta}} = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{Y}$ where $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^N$ and $\boldsymbol{Y} = \{y_i\}_{i=1}^N$.

### 6.2.4 Bayesian Linear Regression

Bayesian linear regression is an approach to linear regression in which the statistical analysis is undertaken within the context of Bayesian inference with a prior distribution for parameter $\boldsymbol{\beta}$. In this setting, the regression errors (or residual) is assumed to follow a normal distribution $\epsilon_i \sim \mathcal{N}\left(0, \sigma^2\right)$. Given a data point $\boldsymbol{x} \in \mathcal{R}^d$ and its respond variable $y$, the likelihood of Bayesian linear regression model with parameter $\boldsymbol{\beta}$ is defined as:

$$p\left(y \mid \boldsymbol{x}, \boldsymbol{\beta}, \sigma^2\right) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left\{-\frac{1}{2}||y - \boldsymbol{x}^T\boldsymbol{\beta}||^2\right\}.$$

Posterior probability distributions of the model's parameter under conjugate prior distribution $\boldsymbol{\beta} \sim \mathcal{N}\left(0, \Sigma_0\right)$ is estimated following:

$$p\left(\boldsymbol{\beta} \mid \boldsymbol{x}_{1:N}, y_{1:N}, \Sigma_0, \sigma\right) \propto \mathcal{N}\left(\mu_n, \Sigma_n\right) \tag{6.2.2}$$

where the posterior mean $\mu_n = \Sigma_n\left\{\boldsymbol{X}\sigma^{-1/2}\boldsymbol{Y}\right\}$, and posterior covariance $\Sigma_n = \left(\Sigma_0^{-1} + \boldsymbol{X}\sigma^{-1/2}\boldsymbol{X}^T\right)^{-1}$(Bishop, 2006). We provide details of derivation to obtain

the posterior distribution of $\boldsymbol{\beta}$:

$$\begin{aligned} p\left(\boldsymbol{\beta} \mid x_{1:N}, y_{1:N}, \Sigma, \sigma^2\right) \propto & \prod_{i=1}^{N} p\left(y_i \mid x_i, \sigma^2, \boldsymbol{\beta}\right) p\left(\boldsymbol{\beta} \mid 0, \Sigma\right) \\ = & \frac{1}{\sqrt{2\pi}\sigma} \frac{1}{(2\pi)^{N/2}|\Sigma|^{Nd/2}} \\ & \times \exp\left\{\sum_{i=1}^{N}\left(-\frac{1}{2\sigma^2}||y_i - x_i^T\boldsymbol{\beta}||^2\right) - \frac{N}{2}\boldsymbol{\beta}^T\Sigma^{-1}\boldsymbol{\beta}\right\} \\ p\left(\boldsymbol{\beta} \mid x_{1:N}, y_{1:N}, \Sigma, \sigma^2\right) \propto & \exp\left\{\underbrace{-\frac{1}{2\sigma^2}||\boldsymbol{Y} - \boldsymbol{X}^T\boldsymbol{\beta}||^2 - \frac{1}{2}\boldsymbol{\beta}^T\Sigma^{-1}\boldsymbol{\beta}}_{A}\right\}. \end{aligned} \tag{6.2.3}$$

We manipulate the term $A$ inside the exponent as follows

$$\begin{aligned} A &= -\frac{1}{2}\left[\frac{1}{\sigma^2}\boldsymbol{Y}^T\boldsymbol{Y} - \frac{2}{\sigma^2}\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} + \frac{1}{\sigma^2}\boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\beta}^T\Sigma^{-1}\boldsymbol{\beta}\right] \\ &= -\frac{1}{2}\left[\frac{1}{\sigma^2}\boldsymbol{Y}^T\boldsymbol{Y} - \frac{2}{\sigma^2}\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\beta}^T\left(\frac{1}{\sigma^2}\boldsymbol{X}^T\boldsymbol{X} + \Sigma^{-1}\right)\boldsymbol{\beta}\right]. \end{aligned}$$

Denote $\Sigma_n^{-1} = \left(\frac{1}{\sigma^2}\boldsymbol{X}^T\boldsymbol{X} + \Sigma^{-1}\right)$, we continue the above equation:

$$A = -\frac{1}{2}\left[\frac{1}{\sigma^2}\boldsymbol{Y}^T\boldsymbol{Y} - \frac{2}{\sigma^2}\boldsymbol{Y}^T\boldsymbol{X}\left(\Sigma_n\Sigma_n^{-1}\right)\boldsymbol{\beta} + \boldsymbol{\beta}^T\Sigma_n^{-1}\boldsymbol{\beta}\right].$$

Denote $\mu_n = \frac{1}{\sigma^2}\boldsymbol{Y}^T\boldsymbol{X}\Sigma_n$, we express the above equation further:

$$\begin{aligned} A &= -\frac{1}{2}\left[\frac{1}{\sigma^2}\boldsymbol{Y}^T\boldsymbol{Y} - 2\mu_n\Sigma_n^{-1}\boldsymbol{\beta} + \boldsymbol{\beta}^T\Sigma_n^{-1}\boldsymbol{\beta} + \mu_n^T\Sigma_n^{-1}\mu_n - \mu_n^T\Sigma_n^{-1}\mu_n\right] \\ &= -\frac{1}{2}\left[\frac{1}{\sigma^2}\boldsymbol{Y}^T\boldsymbol{Y} - \mu_n^T\Sigma_n^{-1}\mu_n\right] + \frac{1}{2}\left(\boldsymbol{\beta} - \mu_n\right)^T\Sigma_n^{-1}\left(\boldsymbol{\beta} - \mu_n\right). \end{aligned} \tag{6.2.4}$$

We note that $\boldsymbol{Y}$ is a constant, $\Sigma_n$ and $\mu_n$ can be computed from $\boldsymbol{X}$ and $\boldsymbol{Y}$. Hence, from Eq. 6.2.3, it can be recognized that the term $A$ has the form of multivariate Gaussian distribution with mean $\mu_n$ and covariance $\Sigma_n$ as

$$p\left(\boldsymbol{\beta} \mid x_{1:N}, y_{1:N}, \Sigma, \sigma\right) \propto \mathcal{N}\left(\boldsymbol{\beta} \mid \mu_n, \Sigma_n\right).$$

The likelihood for predicting new explanatory $\boldsymbol{x}_{\text{new}}$ with new response $y_{\text{new}}$ is com-

puted:

$$\begin{aligned} p\left(y_{\text{new}} \mid \boldsymbol{x}_{\text{new}}, \mu_n, \Sigma_n\right) &= \int_{\boldsymbol{\beta}} p\left(y_{\text{new}} \mid \boldsymbol{x}_{\text{new}}, \boldsymbol{\beta}, \sigma^2\right) p\left(\boldsymbol{\beta} \mid \boldsymbol{x}_{1:N}, y_{1:N}, \Sigma_0, \sigma\right) d\boldsymbol{\beta} \\ &= \int_{\boldsymbol{\beta}} \mathcal{N}\left(y_{\text{new}} \mid \boldsymbol{x}_{\text{new}}^T \boldsymbol{\beta}, \sigma^2\right) \mathcal{N}\left(\boldsymbol{\beta} \mid \mu_N, \Sigma_N\right) d\boldsymbol{\beta} \\ &= \mathcal{N}\left(\boldsymbol{x}_{\text{new}}^T \mu_n, \sigma_n^2(\boldsymbol{x}_{\text{new}})\right) \end{aligned} \tag{6.2.5}$$

where $\sigma_n^2(\boldsymbol{x}_{\text{new}}) = \sigma^2 + \boldsymbol{x}_{\text{new}}^T \Sigma_n \boldsymbol{x}_{\text{new}}$. The above result is obtained by the convolution of two Gaussian distributions.

### 6.2.5 Bayesian Nonparametrics

To further provide a context for this chapter, we summarise the key aspects of Bayesian nonparametric modelling from Section 2.3 below.

A *Dirichlet Process* (Ferguson, 1973) $\text{DP}(\gamma, H)$ is a distribution over discrete random probability measure $G$ on $(\Theta, \mathcal{B})$. Sethuraman (1994) provides an alternative constructive definition which makes the discreteness property of a draw from a Dirichlet process explicit via the stick-breaking representation: $G = \sum_{k=1}^{\infty} \beta_k \delta_{\phi_k}$ where $\phi_k \overset{\text{iid}}{\sim} H, k = 1, \ldots, \infty$ and $\boldsymbol{\beta} = (\beta_k)_{k=1}^{\infty}$ are the weights constructed through a 'stick-breaking' process. As a convention, we hereafter write $\boldsymbol{\beta} \sim \text{GEM}(\gamma)$. Dirichlet Process has been widely used in Bayesian mixture models as the prior distribution on the mixing measures, resulting in a model known as the *Dirichlet Process Mixture model* (DPM) (Antoniak, 1974).

Dirichlet Process can also be constructed hierarchically to provide prior distributions over multiple exchangeable groups. One particular attractive approach is the *Hierarchical Dirichlet Processes* (HDP) (Teh et al., 2006) which posits the dependency among the group-level DPM by another Dirichlet process.

Another way of using DP to model multiple groups is to construct random measure in a nested structure in which the DP base measure is itself another DP. This formalism is the *Nested Dirichlet Process* (Rodriguez et al., 2008), specifically $G_j \overset{\text{iid}}{\sim} U$ where $U \sim \text{DP}(\alpha \times \text{DP}(\gamma H))$. Modelling $G_j$ (s) hierarchically as in HDP and

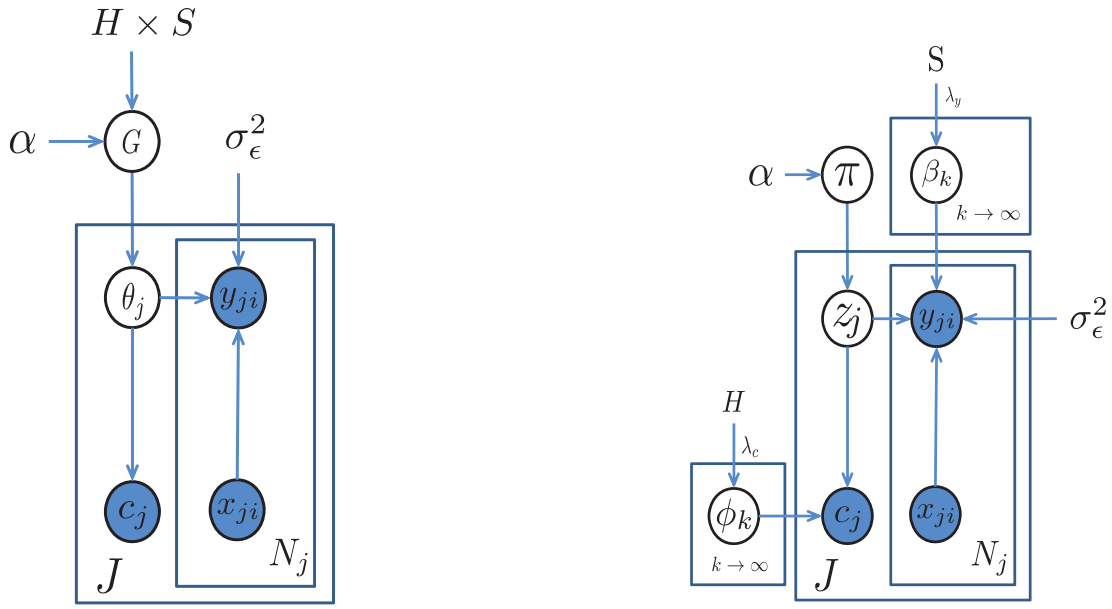



Figure 6.3.1: Bayesian Nonparametric Multilevel Regression graphical model. Left: Stochastic process view. Right: Stick-breaking view. There are $J$ groups with group-level explanatory variable $c_j$, each group has $N_j$ individuals including explanatory variable $x_{ji}$ and response variable $y_{ji}$.

nestedly as in nDP yields different effects. HDP focuses on exploiting statistical strength across groups via sharing atoms $\phi_k$ (s), but it does not partition groups into clusters. Whereas, nDP emphasises on inducing clusters on both observations and distributions, hence it partitions groups into clusters. Finally we note that this original definition of nDP in (Rodriguez et al., 2008) does not force the atoms to be shared across clusters of groups, but this can be achieved by introducing a DP prior for the nDP base measure (Nguyen et al., 2014).

## 6.3 Bayesian Nonparametric Multilevel Regression

In this section, we describe our proposed framework for the Bayesian Nonparametric Multilevel Regression (BNMR). Our goal is to simultaneously clustering the *groups* and estimating regression for *individuals*. The fundamental assumption is that when the groups are related, the group-level explanatory variable (or group-context obser-

vation) is induced in the same distribution component (e.g., Gaussian distribution). Firstly, we aim to use the related groups to strengthen regression estimation for improving regression performance (prevent from overfitting to each group) while unrelated groups do not influence themselves. Second, the induced group-context distribution can be used to identify cluster for new groups (based on group-context observations in new groups).

Iteratively modelling and clustering group context and individual regression would gain benefit and mutually promote each other. First, good groups clustering will produce good regression estimation (e.g., we assume individuals in the same group-cluster have similar regression behavior). Second, the good regression estimation in return provides important information for the group-clustering process previously.

### 6.3.1 Model representation

We consider data presented in a two-level structure. Denote by $J$ the number of groups, we assume that the groups are exchangeable. The collection of $\{c_j\}_{j=1}^{J}$ represents group-level explanatory or group-level context (e.g., age of the patient, population of the state). Each group $j$ contains $N_j$ exchangeable explanatory variable and response variable, represented by $\left\{\boldsymbol{x}_{ji} \in \mathcal{R}^d, y_{ji} \in \mathcal{R}\right\}_{i=1}^{N_j}$.

We now describe the generative process of BNMR (c.f Fig. 6.3.1). Denote by $H$ the base measure for generating group-context distribution and $S$ is a base measure for generating regression coefficients. We use a product base measure of $H \times S$ to draw a DP mixture for jointly clustering groups and regression individuals. Particularly, we have:

$$G \sim \mathrm{DP}\left(\alpha, H \times S\right) \qquad \left(\theta_j^c, \theta_j^y\right) = \theta_j \overset{\text{iid}}{\sim} G$$

Each realization $\theta_j$ includes a pair $\left(\theta_j^c, \theta_j^y\right)$ where $\theta_j^c$ is then used to generate the group-level explanatory observation $c_j$ and $\theta_j^y$ is further used to draw the individual response variables $y_{ji}$ following:

$$c_j \sim F\left(\theta_j^c\right) \qquad y_{ji} \sim \mathcal{N}\left(\boldsymbol{x}_{ji} \times \theta_j^y, \sigma_\epsilon^2\right)$$

where $\sigma_\epsilon$ is a standard deviation of residual error.

#### Stick-breaking representation

We further derive the stick-breaking representation for BNMR (cf. Fig. 6.3.1) where all of the random discrete measures are characterized by a distribution over integers and a countable set of atoms.

The random measure $G$ has the form: $G = \sum_{k=1}^{K} \pi_k \delta_{(\phi_k, \boldsymbol{\beta}_k)}$ where $\boldsymbol{\pi} \sim \mathrm{GEM}(\alpha)$, $\phi_k \overset{\text{iid}}{\sim} H(\lambda_c)$, and $\boldsymbol{\beta}_k \overset{\text{iid}}{\sim} S(\lambda_y)$. Next, we draw an indicator cluster for each group $z_j \overset{\text{iid}}{\sim} \boldsymbol{\pi}$ and generate group-context explanatory variable $c_j \sim F\left(\phi_{z_j}\right)$. Accordingly, the response variables in group $j$ given the cluster $z_j = k$ is drawn as $y_{ji} \sim \mathcal{N}\left(\boldsymbol{x}_{ji}^T \boldsymbol{\beta}_k, \sigma_\epsilon^2\right)$.

### 6.3.2 Inference

We derive collapsed Gibbs sampling for BNMR. Due to the conjugacy property, we would integrate out $\phi_k$,$\beta_k$, and $\boldsymbol{\pi}$. The remaining latent variable $\boldsymbol{z}$ and hyper-parameter $\alpha$ will be sampled.

**Sampling $z_j$.** The conditional distribution for sampling $z$ is:

$$p\left(z_j = k \mid c_j, \{y_{ji}, \boldsymbol{x}_{ji}\}_{i=1}^{N_j}\right) \propto p\left(z_j = k \mid z_{-j}, \alpha\right) \\ \times p\left(c_j \mid z_j = k, c_{-j}, z_{-j}, H\right) \times p\left(y_{ji} \mid \boldsymbol{x}_{ji}, z_j = k, S\right).$$

The first expression $p\left(z_j = k \mid z_{-j}, \alpha\right)$ is the Chinese Restaurant Process (CPR) with concentration parameter $\alpha$. The second term is the predictive likelihood of group-context observation under component (or topic) $k$. This can be analytically computed due to conjugacy of likelihood distribution and prior distribution $H$. The last term is the likelihood contribution from regression observations (including explanatory and response variables) in group $j$ following Eq. 6.2.5.

**Sampling concentration parameter $\alpha$.** We sample the concentration parameter $\alpha$ following (Escobar and West, 1995). Assuming $\alpha \sim \text{Gamma}(\alpha_1, \alpha_2)$ with the auxiliary variable $t$: $p(t \mid \alpha, K) \propto \text{Beta}(\alpha_1 + 1, J)$ where $J$ is the number of groups and $\frac{\pi_t}{1-\pi_t} = \frac{\alpha_1+K-1}{J(\alpha_2-\log t)}$

$$\begin{aligned} p(\alpha \mid t, K) \sim & \pi_t \text{Gamma}(\alpha_1 + K, \alpha_2 - \log(t)) \\ & + (1 - \pi_t)\text{Gamma}(\alpha_1 + K - 1, \alpha_2 - \log(t)). \end{aligned}$$

We integrate out the regression coefficient $\boldsymbol{\beta}_k$ for collapsed Gibbs inference. However, for visualization and analysis of the regression coefficient $\boldsymbol{\beta}_k$ can be re-computed as $p(\boldsymbol{\beta}_k \mid \boldsymbol{x}_i, \boldsymbol{y}_i, z_i = k, \Sigma_0)$ following Eq. 6.2.2.

Given unseen groups of data include $\left\{\boldsymbol{x}_{ji}^{\text{Test}}, c_j^{\text{Test}}\right\}$, we wish to estimate $\left\{y_{ji}^{\text{Test}}\right\}$. We observe that if $\boldsymbol{\beta}_k$ and $\sigma_\epsilon^2$ are known, then $y_{ji}^{\text{Test}}$ will be distributed by $\mathcal{N}\left(\boldsymbol{\beta}_k^T x_{ji}^{\text{Test}}, \sigma^2\mathbf{I}\right)$.

$$\hat{y}_{ji}^{\text{Test}} \propto \sum_{z_j^{\text{Test}}=1}^{K} \left[\boldsymbol{\beta}_{z_j}^T \boldsymbol{x}_{ji}^{\text{Test}}\right] \times p\left(z_j^{\text{Test}} \mid c_j^{\text{Test}}\right)$$

where $p\left(z_j^{\text{Test}} \mid c_j^{\text{Test}}\right) \propto p\left(z_j^{\text{Test}} \mid \boldsymbol{\pi}\right) p\left(c_j^{\text{Test}} \mid \phi_{z_j^{\text{Test}}}\right)$.

## 6.4 Experiment

We demonstrate the proposed framework on multilevel regression task, especially for regression individuals in unseen groups of data. Throughout this section, unless explicitly stated, the training and testing sets are randomly split, and repeated 10 times. The variables $\boldsymbol{x}_{ji}$ and $y_{ji}$ is centralized to have the mean of 0 as recommended in regression tasks (Hox, 2010). Our implementation is using Matlab. For synthetic and Econometric panel data, each iteration took about 1-2 seconds and it took 30-35 seconds for Heathcare dataset. All experiments were converged quickly within 30 iterations of collapsed Gibbs sampling. Initialization for concentration parameter $\alpha = 1$, $\alpha \sim \text{Gamma}(1, 2)$. The conjugate distribution for group-level context is NormalGamma. We use four baseline methods for comparing the regression performance on individuals of unseen groups as follows:

1. Naive Estimation: using the overall average of individuals outcome in training groups $\hat{y}_{\text{new}}^{\text{Test}} = \frac{1}{J}\sum_{j=1}^{J}\frac{1}{N_j}\sum_{i=1}^{N_j} y_{ji}^{\text{Train}}$ as the predicted value.

2. No-Group MultiTask Learning (NG-MTL) (Argyriou et al., 2008): where all tasks are considered in a single group.

3. No-Group MultiTask Learning With Context (NG-MTL-Context): where all tasks are considered in a single group, and context is treated as another explanatory variable.

4. LME: $y_{ji} = \gamma_{00} + \gamma_{01}c_j + \gamma_{01}\boldsymbol{x}_{ji} + \gamma_{11}c_j\boldsymbol{x}_{ji} + \epsilon_{ji}$, we ignore random variables $u_{j0}$ and $u_{j1}$ from original LME for predicting unseen groups because we do not have $u_j$(s) for unseen groups. ($u_j$ is representing for group $j$ given in training set).

In this experiment, the regression performance is evaluated using two metrics: Root Mean Square Error (RMSE), and Mean Absolute Error (MAE). The regression performance is better when it has lower error in both RMSE and MAE.

- Root Mean Square Error: The RMSE is a quadratic scoring rule which measures the average magnitude of the error. Since the errors are squared before they are averaged, the RMSE gives a relatively high weight to large errors. This means the RMSE is most useful when large errors are particularly undesirable.

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|^2}.$$

- Mean Absolute Error: The MAE measures the average magnitude of the errors in a set of forecasts, without considering their direction. The MAE is a linear score which means that all the individual differences are weighted equally in the average.

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|.$$

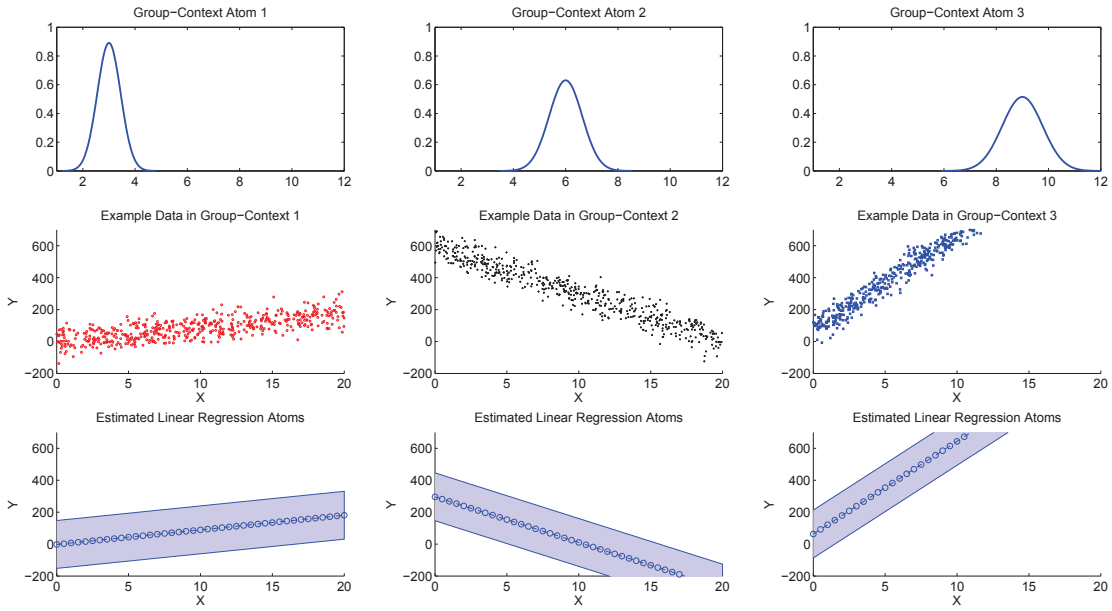



Figure 6.4.1: Synthetic experiment for BNMR.

| Methods\Metrics | RMSE | MAE |
|---|---|---|
| Naive Estimation | 343.3 (11.3) | 278.9 (8.1) |
| NG-MTL | 332.6 (6.9) | 284.0 (4.1) |
| NG-MTL-Context | 230.9 (8.7) | 180.1 (9.2) |
| LME | 190.1 (8.5) | 152.9 (5.4) |
| BNMR | **118.0 (34.0)** | **56.0 (9.7)** |

Table 6.1: Regression performances on synthetic experiment. The lower is the better. Standard deviation is in a parenthesis.

The RMSE will always be larger or equal to the MAE; the greater difference between them, the greater the variance in the individual errors in the sample. If the RMSE is equal to MAE, then all the errors are of the same magnitude.

### 6.4.1 Synthetic experiment

Our goal is to investigate BNMR's ability to recover the true group clusters and number of regression atoms. We first create three univariate Normal distributions $\phi_k(\mathrm{s})$ with different variances (Fig. 6.4.1) for generating group-context observations. Conditional on these context distribution, we initialize three linear regression atoms $\boldsymbol{\beta}_k(\mathrm{s})$ with standard deviation for residual error $\sigma^2 = 50$. Then, we randomly

sample $J = 200$ groups, each group comprises a group-context $c_j$ and $N_j = 20$ pairs of observation $(\boldsymbol{x}_{ji}, y_{ji})$.

The model recovers correctly the ground truth atoms. Visualizations of the group-context distribution and generated data are plotted in Fig. 6.4.1. For evaluation, we split data into 70% number of groups for training and the rest (30% groups) for testing. The performance comparison is displayed in Table 6.1 so that our model gains great improvement in regression than the baseline methods.

### 6.4.2 Econometric panel data: GDP prediction

The Panel Data (Munnell and Cook, 1990) includes 48 states (ignoring Alaska and Hawaii) and 17 years of GDP collection from 1970 to 1986. There are nine divisions in the United States, e.g., New England, Mid-Atlantic, Pacific, and so on (Fig. 6.4.2). Each division contains from 3 to 8 states. We list down nine divisions which is displayed in Fig. 6.4.2:

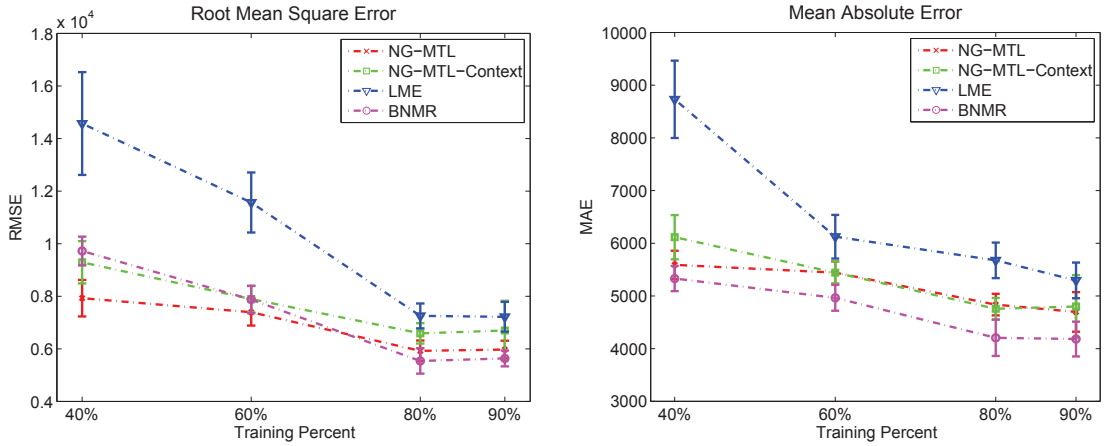
- D1: New England (Connecticut, Maine, Massachusetts, New Hampshire, Rhode Island, and Vermont)
- D2: Mid-Atlantic (New Jersey, New York, and Pennsylvania)
- D3: East North Central (Illinois, Indiana, Michigan, Ohio, and Wisconsin)
- D4: West North Central (Iowa, Kansas, Minnesota, Missouri, Nebraska, North Dakota, and South Dakota)
- D5: South Atlantic (Delaware, Florida, Georgia, Maryland, North Carolina, South Carolina, Virginia, and West Virginia)
- D6: East South Central (Alabama, Kentucky, Mississippi, and Tennessee)
- D7: West South Central (Arkansas, Louisiana, Oklahoma, and Texas)
- D8: Mountain (Arizona, Colorado, Idaho, Montana, Nevada, New Mexico, Utah, and Wyoming)
- D9: Pacific (California, Oregon, and Washington)


Figure 6.4.2: US map of 48 states in 9 divisions. Maps have been modified from the Census Regions and Divisions www.census.gov/geo/maps-data/maps/pdfs/reference/us_regdiv.pdf.

The explanatory variable $\boldsymbol{x}_{ji}$ for each year $i$ in a state $j$ includes 11 dimensions, such as public capital stock, highways and streets capital stock, water and sewer facilities capital stock, employees on non-agricultural payrolls, unemployment rate, and so on. The response variable $y_{ji}$ is a GDP.

We consider the state population (Wyoming has the lowest population of 0.57 millions and the highest population of 38 millions belongs to California, as of 2012) is an explanatory variable for group level. Population is one of the key factor determining the GDP (Maddison, 2010; Kitov, 2005). Hence, states which alike number of population tend to have similar GDP outcome than other states in different number of population. We model the context distribution using univariate Gaussian distribution. The mean and precision for group context distribution are $(\mu, \tau) \sim \text{NormalGamma}(4, 0.25, 0.01, 1)$ and the standard deviation for regression residual error is set as $\sigma_\epsilon = 7000$.

We split the data into training set and testing set such that the states in the testing set do not appear in the training. We vary the proportion of training states from 40% to 90% and perform prediction on the rest. The number of state clusters are




(a) RMSE evaluation.

(b) MAE evaluation.

Figure 6.4.3: Regression performance comparison for panel data.

identified as $K = 3$ (indicating low, mid, and high population). The regression performance of BNMR versus NG-MTL, NG-MTL-Context and LME are plotted in Fig. 6.4.3. We do not include the scores of Naive Estimation into the figure because of its poor performance in this dataset. This poor performance of Naive Estimation can be explained by the high variance in the outcome (e.g., the GDPs of California and Texas are 10-20 times higher than GDPs of Vermont and Delaware). The proposed method achieves the best regression performance in term of RMSE (Fig. 6.4.3a) and MAE (Fig. 6.4.3b) scores. The more state we observe, the more accuracy in prediction we achieve.

### 6.4.3 Healthcare longitudinal data: prediction patient's readmission interval

Meaningful use, improved patient care and competition among providers are a few of the reasons electronic medical records are succeeding at hospitals. Readmission interval prediction could be used to help the delivery of hospital resource-intensive and care interventions to the patients. Ideally, models designed for this purpose would provide close estimation of the admission interval for the next admission. Very often, patients come to a hospital without any existed electronic medical records because they may have not been admitted before. This fact causes problem for existing multilevel regression approaches. We aim to use the proposed framework

to improve performance for predicting readmission interval on new patients.

Our data collected from regional hospital (ethics approval 12/83.). Our main interest is in the chronic Polyvascular Disease (PolyVD) cohort. The collected data includes 209 patients with 3207 admissions in total. We consider the readmission interval within less than 90 days between two consecutive admissions. We treat a patient as a *group* consisting of multiple admissions as *individuals*. The feature for each admission $\boldsymbol{x}_{ji}$ (in patient $j$) includes *External Factor Code*, and *Diagnosis Code* in 289 dimension.

The readmission interval outcome $y_{ji}$ indicates how many days between this admission to the next admission. We use patient's age as a group-context $c_j$. We assume that patients within the same 'age region' would have the similar effects on diseases and readmission gap. For example, under the same diseases, patients in the age of 40-50 would be readmitted to a hospital differently from patients in the age of 70-80 because the prevalence of most chronic diseases increases with age (Denton and Spencer, 2010).

The mean and precision for context distribution are $(\mu, \tau) \sim \text{NG}(40, 0.25, 0.2, 1.1)$ and the standard deviation for regression residual error is specified as $\sigma_\epsilon = 24$.

The data is split with 147 patients (70%) for training and the rest of 62 patients are used for testing (as unseen patients). The posterior inference results in $K = 6$ patient clusters. The univariate Normal distribution of age is plotted in Left Fig. 6.4.4 where we discover the patient's age distribution. In addition, we visualize the two conditional regression coefficients ($\boldsymbol{\beta}_k$) on two patient's group of age 50 and 78 respectively. The estimated $\boldsymbol{\beta}_k$(s) also reveal the correlation among disease codes to patient age clusters (Middle Fig. 6.4.4). There are several disease codes, such as *Inflammatory disorders of scrotum* (feature dimension 287), affecting on the elder of 78 rather than the younger of 50 (resulting zero value in vector regression coefficient).

Our model uses group-level explanatory variable to identify patient's clusters, then do regression using the regression coefficients produced by the patients in the same cluster. Thus, we prevent from overfitting on each training patient and obtain better prediction on testing patients than the three baseline methods (Right Fig. 6.4.4).

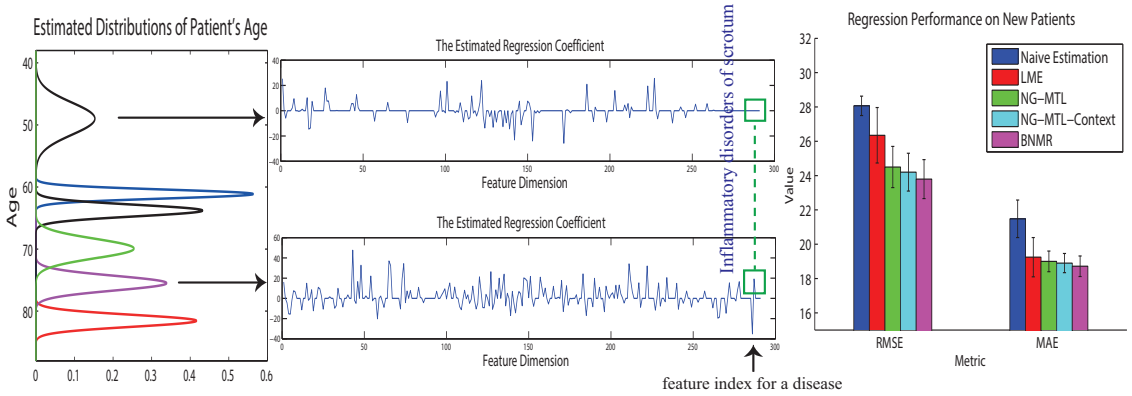



Figure 6.4.4: Regression on HealthData with BNMR. Left: The estimated patient's age distributions. Middle: Two examples of the learned regression coefficient ($\boldsymbol{\beta}_k$), discovering the correlation of disease code versus patient age *(*e.g., *Inflammatory disorders of scrotum* affects elder group of 78, not the group of 50). Right: Regression performance comparison on new patients.

## 6.5 Closing Remarks

In this chapter, we have addressed the problem of multilevel regression where individuals including observations and outcomes are organised into groups. The need of multilevel regression for individuals in unseen groups is commonly encountered in many data domains from econometrics panel data and healthcare longitudinal data domains. Our proposed Bayesian Nonparametric Multilevel Regression model provides an integrated model for clustering groups and does regression for individuals. The unknown number of group cluster and regression coefficients are identified using Bayesian nonparametric setting. By clustering group, the estimated regression coefficients are more generalised and do not overfit to each training group.

We have also presented a *novel* approach for multilevel regression where prediction target is for individuals in new groups. We then conduct two real-world applications of the proposed model on econometric data (for GDP prediction) and healthcare dataset (for patients' readmission prediction). We expect that our model will also be further applicable in other domains.

# Chapter 7

# Scalable Multilevel Clustering with Nested Kmeans

In previous chapters, we have developed various methods and algorithms for multilevel modelling with Bayesian nonparametric models including abnormality detection in video surveillance (Chapter 3), multilevel clustering with group-level context (Chapter 4), classification at group-level in multilevel setting (Chapter 5), and multilevel regression (Chapter 6). Despite the success and flexibility of the Bayesian nonparametric frameworks, their applicability is restricted. For large-scale datasets, which have become more ubiquitous in today analysis tasks, these algorithms might be problematic and unpractical to use.

In this chapter, we consider novel scalable multilevel clustering setting where the data are organised into groups. The task is finding the unknown number of clusters nested at multilevel – a setting that, to our knowledge, we are the first to formulate and solve using Bayesian nonparametric tool. When the data are large, several traditional probabilistic inference algorithms that rely on MCMC principle are no longer applicable due to their associated computational cost. To address this issue, we propose in this chapter a new multilevel clustering framework termed as *Nested Kmeans* (nKmeans). Our solution roots in the recent principle of small variance asymptotic analysis (Kulis and Jordan, 2012). Our resulting framework can nestedly cluster data points within group and groups themselves. Furthermore, the number of local clusters within each group and the number of global clusters are also

induced thanks to the inherent property of Bayesian nonparametric. Experiments on synthetic and real-world datasets demonstrate the advantages of the proposed Nested Kmeans.

Our contributions in this chapter are the followings: (1) We provide an alternative inference for nested Dirichlet Process using the concept of Chinese Franchise Restaurant-Bus. (2) We derive the hard assignment in the limit (let the variance of data approach zero) for scalable inference upon the proposed Chinese Franchise Restaurant-Bus. (3) We derive the objective function and algorithm complexity for further analysis. (4) We perform extensive experiments using synthetic and image clustering on large-scale dataset with millions of data points.

This chapter is organised as follows. We present an overview on multilevel clustering as well as the need for scalable multilevel clustering in Section 7.1. Next, we provide the related background in multilevel clustering and small variance asymptotic techniques for Bayesian nonparametric models in Section 7.2. Further, we describe the proposed framework and algorithm in Section 7.3. Finally, we demonstrate extensive experiments in Section 7.4 to validate our framework.

## 7.1 Overview

With simplicity and efficiency, K-means is undoubtedly one of the most popular clustering algorithms over the past 50 years (Jain, 2010). Given a collection of data points, K-means partitions the data into non-overlapping clusters that minimises the inter-distance within clusters efficiently. It is a *flat* clustering algorithm which acts on single data point level. However, K-means cannot identify the unknown number of cluster itself that requires an input from user. Moreover, when data is presented hierarchically or nestedly such as in the presence of multiple groups of data points, it is not well applicable to use K-means. It requires a task to simultaneously cluster data points within each group as well as clustering these groups into clusters at the higher level. To our knowledge, developing a Kmeans-style algorithm to simultaneously cluster at multiple levels and automatically identify a suitable number of clusters remains an open problem.

We refer to this task as *multilevel clustering* (Nguyen et al., 2014; Hox, 2010; Xie and Xing, 2013) which aims to jointly group *individuals* (at a lower level) and *individual-units* (at a higher level). Examples to this individual-unit data structure include words are organised into documents, or students organised into classes. One crucial issue in clustering is to choose appropriate number of clusters. Recent development in Bayesian nonparametric (BNP) has elegantly addressed this issue with its flexibility to infer the unknown number of clusters from the data. There are notably the hierarchical Dirichlet Process (HDP) (Teh et al., 2006) for sharing statistical strength among groups, and the nested Dirichlet Process (nDP) (Rodriguez et al., 2008) for inducing clusters on both observations and distributions, thus partition groups into clusters.

Despite the success and flexibility of the Bayesian nonparametric frameworks, their applicability are restricted on large-scale dataset due to lacking of scalable inference in rich probabilistic models. Therefore, a recent thread of research, namely small variance asymptotic of BNP model has received much attention (Jiang et al., 2012; Roychowdhury et al., 2013; Kulis and Jordan, 2012) which aims to provide scalability, but still maintain the main properties of Bayesian nonparametric modelling. So far, asymptotic analysis has been derived only to a few BNP models. To the best of our knowledge, there is no such study for small variance asymptotic of multilevel clustering problem in literature.

This chapter performs asymptotic analysis for nDP and results in the Nested Kmeans (nKmeans) algorithm. The new algorithm addresses the scalable Bayesian nonparametric multilevel clustering problem: jointly cluster the data observations and their observation-groups in large scale dataset. To develop scalable algorithm for multilevel data, we propose a Gibbs inference for Chinese Franchise Restaurant-Bus (CFRB) metaphor on the nDP (Rodriguez et al., 2008; Nguyen et al., 2014). Further, we derive the hard-assignment procedure in the small-variance limit. The resulting Nested Kmeans algorithm obtains the objective function like K-means with three penalties: for number of document clusters, for number of total word-topic, for number of local topic per cluster. Our new nKmeans algorithm preserves the advantages of the nested Dirichlet Process in nonparametric multilevel clustering while improve the computational time to be deterministic in the scale of the classic Kmeans.

## 7.2 Additional Related Background

We begin this section by reviewing the related works of the small variance asymptotic in Bayesian nonparametric. Next, we present the background of the nested Dirichlet process on which we will further develop our approach.

### 7.2.1 Bayesian Nonparametric Multilevel Clustering

Recall from Section 2.3.1 from Chapter 2 that Dirichlet process (Ferguson, 1973) has been widely used in Bayesian mixture models as the prior distribution on the mixing measures due to its discrete property. Each is associated with an atom $\phi_k$ in the stick-breaking representation (Sethuraman, 1994) $G = \sum_{k=1}^{\infty} \beta_k \delta_{\phi_k}$ where $\phi_k \overset{\text{iid}}{\sim} H, k = 1, \ldots, \infty$ and $\boldsymbol{\beta} = (\beta_k)_{k=1}^{\infty}$ where the weights constructed through a stick-breaking process. A likelihood kernel $F(\cdot)$ is used to generate data $x_i \mid \phi_k \overset{\text{iid}}{\sim} F(\cdot \mid \phi_k)$, resulting in a model known as the Dirichlet Process Mixture model (DPM) (Antoniak, 1974). Hierarchical Dirichlet Processes (Teh et al., 2006) is introduced to discover the sharing statistical strength over multiple exchangeable groups which posits the dependency among the group-level DPM by another Dirichlet process.

The nested Dirichlet Process (Rodriguez et al., 2008) constructs a random measure in a nested structure in which the DP base measure is itself another DP to model multiple groups of data. modelling data hierarchically as in HDP and nestedly as in nDP yields different effects. HDP focuses on exploiting statistical strength across groups via sharing atoms $\phi_k$ (s), but it does not partition groups into clusters. Whereas, nDP emphasises on inducing clusters on both observations and distributions, hence it partitions groups into clusters. The original definition of nDP in (Rodriguez et al., 2008) does not force the atoms to be shared across clusters of groups, but this can be achieved in (Nguyen et al., 2014) by simply introducing a DP prior for the nDP base measure. This chapter follows the latter definition of nDP for sharing atoms across group clusters (Nguyen et al., 2014).

### 7.2.2 Small variance asymptotic in Bayesian nonparametric

As discussed in Section 2.3.2 in Chapter 2, recent works of small variance asymptotic analysis are motivated by the connection between Gaussian Mixture Model (GMM) and k-means: as the variances of Gaussian goes to zero, the GMM becomes k-means. The asymptotic derivation to DPM and HDP are introduced in (Kulis and Jordan, 2012), opening the line of work in BNP and hard clustering methods. For the generic case of distributions (non Gaussian case), small variance derivation is proposed in (Jiang et al., 2012) which is suitable for discrete-data problems. More recently, the asymptotic work of (infinite) HMM (Roychowdhury et al., 2013) and Dependent Dirichlet Process Mixture (Campbell et al., 2013) offer scalable analysis for sequential data. All of the previous works focus on clustering the data points at a single level. Such as DPM, HDP assign words to topics (or word clusters). Similarly, HMM and DDPM assign data point (at a single level data structure) to a state in a sequence. To our knowledge, there is no such asymptotic BNP work for multilevel clustering analysis.

## 7.3 Framework

We are motivated by (1) the difficulty in deriving asymptotic from the previous sampling scheme (the impact of stick-breaking prior is not maintained) and (2) Chinese Restaurant Franchise for deriving asymptotic HDP. Therefore, we propose the Chinese Franchise Restaurant-Bus (CFR-B) sampling scheme for NDP. The resulting Nested Kmeans (nKmeans) algorithm are obtained by taking small variance limit of the CFR-B. Finally, we derive the objective function for nKmeans and provide the complexity analysis.

### 7.3.1 Chinese Franchise Restaurant-Bus

In order to yield a non-trivial small variance asymptotic assignment for NDP, we propose a Chinese Franchise Restaurant-Bus for Nested DP (Rodriguez et al., 2008). We note that the original paper for NDP (Rodriguez et al., 2008) employed a trun-

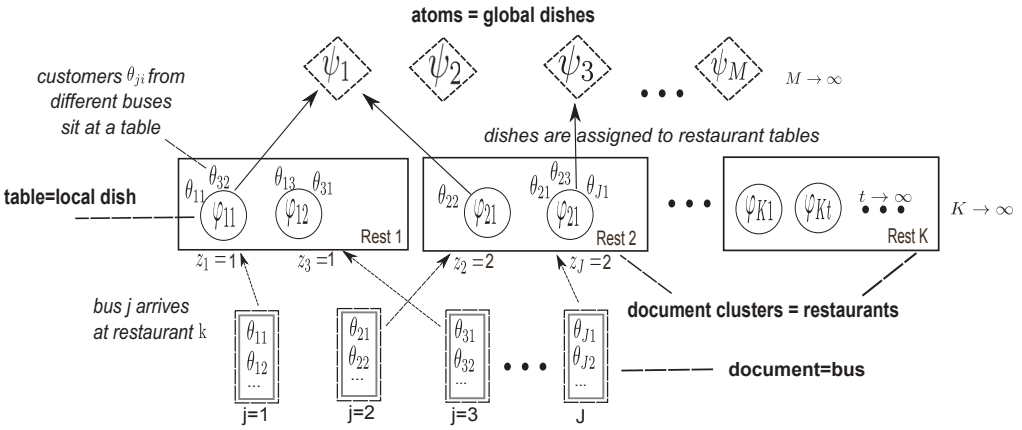



Figure 7.3.1: Chinese Franchise Restaurant-Bus representation. A document $j$-th is seen as a bus arriving at a restaurant $k$-th (number of restaurant can go to infinity). Each customer $\theta_{ji}$ inside the bus will seat at a table with other customers (may come from other buses). At each table, a local dish ($\varphi_{kt}$) is served taking from a global menu ($\psi_m$).

cated Gibbs inference approach. Our urn characterisation through the CFR-B in this chapter presents a new inference scheme for NDP. In the CFR-B, the metaphor of the Chinese restaurant process (CRP) and Chinese restaurant franchise (CRF) is extended to allow multiple documents grouped into an infinite number of clusters.

In Fig. 7.3.1, we have a series of restaurants ($k = 1, 2, .., \infty$) with infinite number of table in each restaurant ($\varphi_{k1}, \varphi_{k2}, ...\varphi_{k\infty}$), sharing global menu of dish ($\psi_1, \psi_2, ...\psi_\infty$) together. Each bus is a document that carrying customers (words) will arrive at a restaurant (e.g., bus $b_1$ arrives at restaurant $k = 1$). Each customer (e.g., $\theta_{11}$) can choose one table (e.g., $\varphi_{11}$) in the restaurant $k$-th (e.g., restaurant 1) following CRP with other customers who previously come from the same or different buses.

The CFR-B and CRF (Teh et al., 2006) are similar in the way that a customer behaves following other customers in the same restaurant. "The rich get richer" property of CRP and CRF is still held on the CFR-B. The principle differences between CFR-B and CRF (Teh et al., 2006) are as follows. On one hand, the CRF formulates each *document* as a *restaurant* and the number of restaurants is the number of documents which is *fixed*. On the other hand, a *document* in CFR-B is a *bus*. *Restaurant* in CFR-B is built for clustering documents and the number of restaurants is growing *flexible*. In CRF, customers in different documents do not influence their table-preference likelihood. On the contrary, different documents in

the same restaurant affect each other in CFR-B.

We denote customer $\theta_{ji}$ is associated with one table $\varphi_{kt}$, $z_j = k$ is assignment of the bus $j$-th to a restaurant $k$-th; $t_{ji}$ is assignment of customer $i$-th in bus $j$ to table $t$ in restaurant $k$-th. The variable $\varphi_{kt}$ represent a table-specific choice of dishes (the dish served at table $t$ in restaurant $k$). The index $d_{kt}$ is also introduced to map a global dish $m$ to table $t$ in restaurant $k$.

For inference purpose, we need to keep track of count statistics. To do so, we maintain the counts of customers in each table in a restaurant, counts of tables in each restaurant, and counts of dishes used by each bus. We use notation $n_{kt}$ to denote the number of customers in restaurant $k$ sitting at table $t$. $o_{km}$ is a count of the number table in restaurant $k$-th choosing dish $m$ and $o_{k*}$ is the total number of tables in restaurant $k$-th. Finally, the notation $\mathbf{y}_j$ capture the vector statistics of all customer in bus $j$-th choosing dishes (e.g., the count how many time the global dish $m$-th is used by the bus $j$-th).

### 7.3.2 Model representation

Following the Polya urn scheme (Blackwell and MacQueen, 1973) and the Chinese Restaurant Franchise (Teh et al., 2006), the conditional distribution for $\theta_{ji}$ given $z_j = k$, $\{\theta_{j'i'} \mid z_{j'} = k, \forall (j'i'), (j'i') \neq (ji)\}$ and $Q_0$ where $Q_k$ is integrated out:

$$\theta_{ji} \mid z_j = k, \theta_{j'i'}, v, Q_0 \sim \sum_{t=1}^{o_{k*}} \frac{n_{kt}}{i-1+v}\delta_{\varphi_{kt}} + \frac{v}{i-1+v}Q_0$$

where $\theta_{ji}$ is a customer $i$-th in bus $j$-th, arriving at restaurant $k$-th. $o_{k*}$ is the total number of tables in restaurant $k$-th. $v$ is a concentration parameter. $n_{kt}$ is the count number of customers seating at table $t$-th in restaurant $k$-th (note that customers may come from multiple buses). If the first term in the above expression (the summation) is selected (it means he seats at table $t$-th in restaurant $k$-th), we set $\theta_{ji} = \varphi_{kt}$ and put $t_{ji} = t$ for the $t$-th is chosen. Otherwise, the second term is selected, we set up a new table $t_{ji} = o_{k*} + 1$ and draw a new local dish for the new table (in restaurant $k$-th) $\varphi_{kt_{ji}} \sim Q_0$, and $\theta_{ji} = \varphi_{kt_{ji}}$.

$Q_0$ is distributed by Dirichlet process, we can integrate it out and the conditional distribution of local dish $\varphi_{kt}$ as similar to the CRF (Teh et al., 2006):

$$\varphi_{kt} \mid \varphi_{k1}, ..., \varphi_{kt-1}, \eta, S \sim \sum_{m=1}^{M} \frac{o_{*m}}{o_{**} + \eta} \delta_{\psi_m} + \frac{\eta}{o_{**} + \eta} S.$$

If we choose the local dish $\varphi_{kt}$ from the a collection of previous dish $\{\psi_1, ...\psi_M\}$, we set $\varphi_{kt} = \psi_m$ and $d_{kt} = m$. If the new dish is chosen, we draw a new dish $\psi_{M+1} \sim S$, set $d_{kt} = M + 1$, and increase $M$ accordingly.

The prior likelihood for sampling document cluster index is followed CRP (Blackwell and MacQueen, 1973). We can write:

$$z_j \mid z_1, ...z_{j-1}, \alpha \sim \sum_{k=1}^{K} \frac{q_k}{j - 1 + \alpha} \delta_k + \frac{\alpha}{j - 1 + \alpha} \delta_{K+1}$$

where $q_k$ is the count number of documents belonged to cluster $k$-th.

### 7.3.3 Graphical representation and generative process

Firstly, we generate a mixture weight for document (bus) clustering: $\pi \sim \text{GEM}(\alpha)$. Then, each document $j$ will be indexed by $z_j \sim \text{Mult}(\pi)$. The number of document cluster (restaurant) is identified as $K = |\{z_1, z_2, ...z_J\}|$[1] which can go to infinity.

For each document cluster $k$, we draw a specific mixture weight $\beta_k \sim \text{GEM}(v)$, which later on will influence on generating table indicator $t_{ji}$. Next, we sample table indicator $t_{ji}$ for every customer in every bus (or word in document) $t_{ji} \mid z_j = k \sim \text{Mult}(\beta_k)$. The number of table is taken in restaurant $k$ is denoted as $o_k = |\{t_{ji} \mid \forall (j, i), z_j = k\}|$.

We now sample a mixture weight for dish: $\epsilon \sim \text{GEM}(\eta)$. For each restaurant $k$, a table $t$ will be assigned to a dish $m$: $d_{kt} \sim \text{Mult}(\epsilon)$. The number of dish is $M = |\{d_{kt} \mid \forall (k, t)\}|$. Finally, the data observation is draw as $x_{ji} \sim F\left(\psi_{d_{z_j, t_{ji}}}\right)$.

[1] cardinality of a set does not count repeating elements.

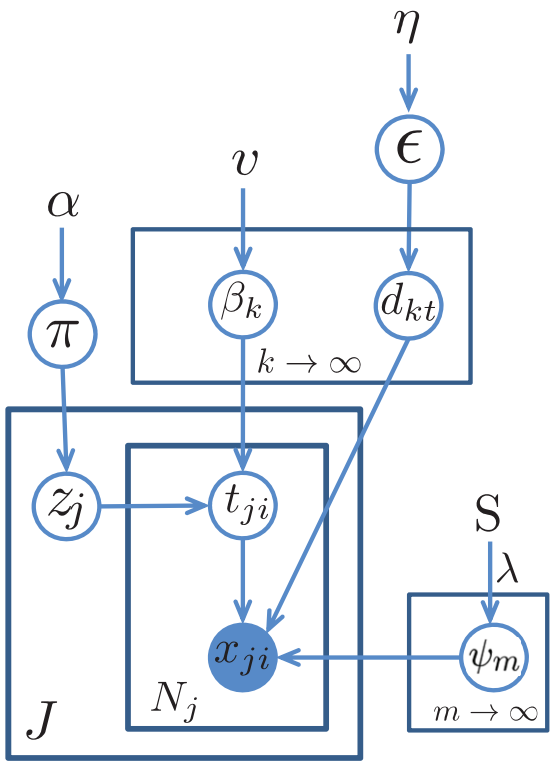



Figure 7.3.2: Graphical representation of CFR-B

### 7.3.4 Gibbs sampler for CFR-B

We next present the Gibbs sampler for our model using Chinese Franchise Restaurant-Bus metaphor. Let recall the random variables of interest. The observed data is $x_{ji}$, drawn from $F(\theta_{ji})$, which is a customer $i$-th from the bus $j$-th. There are three latent variables $z_j, t_{ji}$, and $d_{kt}$ which we need to sample from, which form the Gibbs state space. The index $z_j = k$ is an assignment of the restaurant $k$-th that the bus $j$-th arrives. The latent variable $t_{ji} = t$ indicates the table of customer $x_{ji}$ in the restaurant $z_j = k$. The hidden variable $d_{kt}$ can be seen as the mapping of table $t$-th (in restaurant $k$-th) to global dish $m$-th. Hence, a customer $x_{ji}$ will be served by the dish $d_{z_j,t_{ji}}$. We have three hyperparameters $\alpha, v$, and $\eta$. Using the sufficient statistic defined in Section 7.3.1, detail sampling of these variables are presented below.

**Sampling $t_{ji}$.** We sample the table indicator for customer $i$-th in bus $j$-th which arrives at restaurant $k$-th

$$p\left(t_{ji}=t \mid z_j=k\right) \propto \begin{cases} n_{kt}^{-ji} \times f_{d_{kt}}^{-x_{ji}}\left(x_{ji}\right) & \text{used } t \\ v \times o_{*m}^{-ji} \times f_{m}^{-x_{ji}}\left(x_{ji}\right) & \text{new } t\text{,used } m \\ v \times \eta \times f_{m^{new}}^{-x_{ji}}\left(x_{ji}\right) & \text{new } t\text{,new } m \end{cases} \tag{7.3.1}$$

where the probability density function $f$ is generating data $x_{ji}$ (e.g., Multinomial, Gaussian).

**Sampling $d_{kt}$.** We assign table $t$-th in restaurant $k$-th to global dish $m$-th

$$p\left(d_{kt}=m \mid ..\right) \propto \begin{cases} o_{*m} \times f_{m}^{-x_{ji}}\left(\mathbf{x}_{kt}\right) & \text{used } m \\ \eta \times f_{m^{new}}^{-x_{ji}}\left(\mathbf{x}_{kt}\right) & \text{new } m \end{cases} \tag{7.3.2}$$

where $\mathbf{x}_{kt}$ is a set of data points who customers arriving at table $t$-th in restaurant $k$-th (can be from multiple buses).

**Sampling $z_j$.** We sample the restaurant that a bus $j$-th will arrive

$$p\left(z_j=k \mid\right) \propto \begin{cases} r_{k}^{-j} \times h_{k}^{-y_j}\left(\mathbf{y}_j\right) & \text{used } k \\ \alpha \times h_{k^{new}}^{-y_j}\left(\mathbf{y}_j\right) & \text{new } k. \end{cases} \tag{7.3.3}$$

The likelihood density function $h$ is denoted for a predictive likelihood for the bus $j$-th to restaurant $k$-th, using the Multinomial-Dirichlet conjugacy property.

### 7.3.5 Asymptotic hard-assignment for CFR-B

In this section, we derive the hard-assignment by evaluating the small variance limit behaviors for the sampling Eq.s [7.3.1,7.3.2,7.3.3] in previous section. The asymptotic formula for Gaussian distribution is a squared Euclidean function (Kulis and Jordan, 2012). This can be further generalized to other exponential family distributions and their corresponding Bregman divergences as the asymptotic counterparts

(Banerjee et al., 2005) (e.g., Multinomial distribution has asymptotic form as Kullback–Leibler (KL) divergence ). For ease of understanding and interpretation, we simply present the Gaussian case (other distributions can be straightforwardly accommodated). The likelihood for density function is assumed following Gaussian distribution: $f_m(x_{ji}) = \mathcal{N}(x_{ji} \mid \mu_m, \sigma^2 I_d)$.

- Hard assigning table $t_{ji}$ for customer $x_{ji}$, $\hat{\gamma}(t_{ji})$ is:

$$\lim_{\sigma \to 0} p(t_{ji} \mid .) \propto \begin{cases} ||x_{ji} - \mu_{d_{kt_{ji}}}||^2 & \text{used } t \\ \lambda_l + ||x_{ji} - \mu_m||^2 & \text{new } t, \text{used } m \\ \lambda_g + \lambda_l & \text{new } t, \text{new } m. \end{cases} \tag{7.3.4}$$

*Proof.* From Eq. 7.3.1, we denote:

$$\begin{aligned} A &= \frac{n_{kt}^{-ji}}{n_{k*}^{-ji} + v} \times f_{d_{kt}}^{-x_{ji}}(x_{ji}) \\ B &= \frac{v}{n_{k*}^{-ji} + v} \times \frac{o_{*m}^{-ji}}{o_{**} + \eta} \times f_m^{-x_{ji}}(x_{ji}) \\ C &= \frac{v}{n_{k*}^{-ji} + v} \times \frac{\eta}{o_{**} + \eta} \times f_{m^{new}}(x_{ji}). \end{aligned}$$

We only consider terms that would not be constants after we do the asymptotic analysis: $A \propto n_{kt}^{-ji} \times f_{d_{kt}}^{-x_{ji}}(x_{ji})$, $B \propto v \times q_{*m}^{-ji} \times f_m^{-x_{ji}}(x_{ji})$, and $C \propto v \times \eta \times f_{m^{new}}(x_{ji})$. □

Substitute $v = \exp\left(-\frac{\lambda_l}{2\sigma}\right)$ and $\eta = \left(\frac{\sigma}{\sigma+\rho}\right)^{-d/2} \exp\left(-\frac{\lambda_g}{2\sigma}\right)$ into $B$ and $C$ (similar to Section 3.1 in (Kulis and Jordan, 2012)), we have the following:

$$\begin{aligned} A &= n_{kt}^{-ji} (2\pi\sigma)^{-d/2} \exp\left(-\frac{1}{2\sigma}||x_{ji} - \mu_{d_{kt}}||^2\right) \\ B &= o_{*m}^{-ji} (2\pi\sigma)^{-d/2} \exp\left(-\frac{1}{2\sigma}\left[||x_{ji} - \mu_m||^2 + \lambda_l\right]\right) \\ C &= (2\pi\sigma)^{-d/2} \exp\left(-\frac{1}{2\sigma}\left[\lambda_l + \lambda_g + \frac{\sigma}{\rho+\sigma}||x_i||^2\right]\right). \end{aligned}$$

In the limit of $\sigma \to 0$, only the smallest of these value

$\{\underbrace{||x_{ji} - \mu_{d_{kt_{ji}}}||^2, ...}_{\text{old t}}, \underbrace{\lambda_l + ||x_{ji} - \mu_m||^2, ...}_{\text{new t, old m}}, \underbrace{\lambda_l + \lambda_g}_{\text{new t, new m}}\}$ will receive a non-zero value (the remaining terms are zero).

- Hard assigning local dish index $d_{kt}$ (in restaurant $k$-th) to global dish $m$-th:

$$\lim_{\sigma \to 0} p\left(d_{kt} \mid .\right) = \hat{\gamma}\left(d_{kt}\right) \propto \begin{cases} ||\mathbf{x}_{kt} - \mu_m||^2 & \text{used } m \\ \lambda_g & \text{new } m. \end{cases} \tag{7.3.5}$$

*Proof.* Denote $\eta = \left(\frac{\sigma}{\sigma+\rho}\right)^{-d/2} \exp\left(-\frac{\lambda_g}{2\sigma}\right)$. Ignore the constant term, the asymptotic behavior of Eq. 7.3.2 will be increasingly dominated by the smallest value of $\{||\mathbf{x}_{kt} - \mu_1||^2, ...||\mathbf{x}_{kt} - \mu_M||^2, \lambda_g\}$. □

- Hard assigning document cluster index $z_j$:

$$\hat{\gamma}\left(z_j\right) \propto \begin{cases} KL\left(\mathbf{y}_j || \tau_k^{-j}\right) & \text{used } k \\ \lambda_\alpha & \text{new } k \end{cases} \tag{7.3.6}$$

where $y_j$ is a vector of the sufficient statistic of all customer in bus $j$-th choosing dishes. $\tau_k^{-j}$ is a vector of the sufficient statistic of all cusomers in choosing dishes $(1, ...M)$ from all buses arriving at restaurant $k$-th, excluding bus $j$-th.

*Proof.* The density kernel $h$ (Eq. 7.3.6) is Multinomial distribution (Nguyen et al., 2014), its asymptotic derivation will result in KL divergence (Banerjee et al., 2005). Jiang et al. (2012) have proved that the asymptotic behavior of these exponential family distributions will be increasingly dominated by the smallest value of $\{D_\phi\left(y_j, \tau_1\right), ..., D_\phi\left(y_j, \tau_k\right), \lambda_\alpha\}$ where $D_\phi$ is a Bregman divergence of the density kernel $\phi$. In our case, $D_\phi$ is a KL divergence. The hyperparameter $\alpha$ is transformed into $\lambda_\alpha$ similar to the way described in Lemma 3.1 in (Jiang et al., 2012). □

### 7.3.6 Objective function

We follow the probability of random partition function over a Chinese Restaurant Process (CRP) used in (Teh et al., 2011; Pitman, 1995; Broderick et al., 2013b).

**Lemma 7.1.** *Denote $z$ is a partition over some set $S$ with $n$ elements. $|z|$ is the number of cells in $z$. Each cell $c \in z$ is a subset of $S$. $|c|$ denotes the size of this subset. $z$ follows a Chinese Restaurant Process distribution $z \sim CRP_S(\alpha)$ i*

$$p(z|\alpha, S) = \frac{\Gamma(\alpha+1)}{\Gamma(\alpha+|S|)}\alpha^{|z|}\prod_{c\in z}\Gamma(|c|). \tag{7.3.7}$$

The joint log conditional distribution of $t, d, z, x$ in Chinese Franchise Restaurant Bus metaphor can be written as:

$$\begin{aligned}\log p\left(t, d, z, x \mid \alpha, v, \eta, \lambda\right) = \log p\left(z \mid \alpha\right) + \log p\left(d \mid t, z, \eta\right) + \log p\left(x \mid t, d, z, \lambda\right) \\ + \log p\left(t \mid d, z, v\right)\end{aligned} \tag{7.3.8}$$

We then compute the log probabilities separately following. The conditional probability of $z$ (restaurant indicator for bus) follows a CRP with concentration parameter $\alpha$, or $z \sim CRP_J(\alpha)$, following Lemma 7.1 is defined as:

$$\begin{aligned} p\left(z \mid \alpha\right) &= \alpha^{K-1}\frac{\Gamma\left(\alpha+1\right)}{\Gamma\left(\alpha+J\right)}\prod_{k=1}^{K}\Gamma\left(r_k\right) \\ \log p\left(z \mid \alpha\right) &= \left(K-1\right)\log\alpha + \log\frac{\Gamma\left(\alpha+1\right)}{\Gamma\left(\alpha+J\right)} + \log\sum_{k=1}^{K}\Gamma\left(r_k\right) \end{aligned} \tag{7.3.9}$$

where $r_k = \sum_{j=1}^{J}\mathbb{I}\left(z_j = k\right)$ is number of bus belong to cluster $k$-th.

The conditional likelihood of the observation $x$ is following:

$$\begin{aligned} p\left(x \mid t, d, z, \lambda\right) &= p\left(x \mid t, d, z, \psi\right) \times p\left(\psi \mid \lambda\right) \\ &= \prod_{j=1}^{J} \prod_{i=1}^{N_j} p\left(x_{ji} \mid \psi_{d_{z_j, t_{ji}}}\right) \times \prod_{m=1}^{M} p\left(\psi_m \mid \lambda\right) \\ \log p\left(x \mid t, d, z, \lambda\right) &= \log \sum_{j=1}^{J} \sum_{i=1}^{N_j} p\left(x_{ji} \mid \psi_{d_{z_j, t_{ji}}}\right) + \log \sum_{m=1}^{M} p\left(\psi_m \mid \lambda\right) \end{aligned} \tag{7.3.10}$$

where $\lambda$ is the hyperparameter for observation $x$. We assume the kernel likelihood generating observation $x_{ji}$ is Gaussian density. Then, we have $p\left(x_{ji} \mid \psi_{d_{z_j, t_{ji}}}\right) = \mathcal{N}\left(x_{ji} \mid \mu_{d_{z_j, t_{ji}}}, \sigma^2 I_d\right)$ and $p\left(\psi_m \mid \lambda\right) = \mathcal{N}\left(\mu_m \mid 0, \rho^2 I_d\right)$ as in (Kulis and Jordan, 2012). The Eq. 7.3.10 is expressed as:

$$\begin{aligned} \log p\left(x \mid t, d, z, \lambda\right) = \log \sum_{j=1}^{J} \sum_{i=1}^{N_j} \frac{1}{2\pi\sigma^{d/2}} \exp\left(-\frac{1}{2}||x_{ji} - \mu_{d_{z_j, t_{ji}}}||^2\right) \\ + \log \sum_{m=1}^{M} \frac{1}{2\pi\rho^{d/2}} \exp\left(-\frac{1}{2}||\mu_m||^2\right). \end{aligned}$$

The conditional likelihood of the observation $d$ is followed the random partition function (Lemma 7.1) over a CRP with concentration parameter $\eta$:

$$\begin{aligned} p\left(d \mid z_j = k, \eta\right) &= \eta^{M-1} \frac{\Gamma\left(\eta + 1\right)}{\Gamma\left(\eta + c_{**}\right)} \prod_{m=1}^{M} \Gamma\left(c_{km}\right) \\ \log p\left(d \mid z_j = k, \eta\right) &= \left(M - 1\right) \log \eta + \log \frac{\Gamma\left(\eta + 1\right)}{\Gamma\left(\eta + c_{k*}\right)} + \log \sum_{m=1}^{M} \Gamma\left(c_{km}\right) \end{aligned} \tag{7.3.11}$$

where $c_{km} = \sum_{j=1}^{J} \mathbb{I}\left(z_j = k\right) \sum_{i=1}^{N_j} \mathbb{I}\left(d_{k, t_{ji}} = m\right)$ and $c_{k*} = \sum_{m=1}^{M} c_{km}$.

The conditional likelihood of the observation $t$, which is influenced by (1) the Chinese Restaurant Process with concentration parameter $v$ and (2) conditional probability given the bus assignment $z_j$ and table-dish mapping $d_{kt}$, is following:

$$p\left(t \mid d, z, v\right) = p\left(t \mid z, v\right) \times p\left(t \mid d, z\right).$$

We have $n_{kt}$ is number of elements in this set$\{t_{ji} = t \mid z_j = k, \forall j\}$ and $n_{k*}$ is the

total number of customer in restaurant $k$-th, equivalently the count of this set $\{t_{ji} \mid \forall (j,i), z_j = k\}$. $o_{k*} = |\{t_{ji} \mid z_j = k\}|$ is the number of table in restaurant $k$-th. In each restaurant $k$-th, the table assignment $t$ follows CRP with concentration parameter $v$, $t \sim CRP_{n_{k*}}(v)$. Therefore, the first argument is a random partition function (Lemma 7.1) within each restaurant $k$.

$$\begin{aligned} p(t \mid z, v) &= \prod_{k=1}^{K} \mathbb{I}(z_j = k) \prod_{j=1}^{J} \prod_{i=1}^{N_j} p(t_{ji} \mid z_j = k, v) \\ &= \prod_{k=1}^{K} v^{o_{k*}-1} \frac{\Gamma(v+1)}{\Gamma(v+n_{k*})} \prod_{t=1}^{o_{k*}} \Gamma(n_{kt}). \end{aligned}$$

The second term of conditional likelihood is calculated:

$$\begin{aligned} p(t \mid d, z) &= \prod_{j=1}^{J} p(z_j = k \mid t_{j*}, d, \{t_{j'*} \mid z_{j'} = k\}) \\ &= \prod_{j=1}^{J} \int_{\tau_k} p\left(z_j = k \mid \boldsymbol{\tau}_k^{-j}\right) p\left(\boldsymbol{\tau}_k^{-j} \mid d, \{t_{j'*} \mid z_{j'} = k\}\right) d\tau_k \\ &= \prod_{j=1}^{J} \underbrace{\frac{\Gamma\left(\sum_{m=1}^{M} c_{z_j,m}^{-j}\right)}{\Gamma\left(\sum_{m=1}^{M} c_{z_j,m}^{-j} + y_{jm}\right)} \prod_{m=1}^{M} \frac{\Gamma\left(c_{z_j,m}^{-j} + y_{jm}\right)}{\Gamma\left(c_{z_j,m}^{-j}\right)}}_{p\left(\boldsymbol{y}_j \mid \boldsymbol{\tau}_{\mathbf{z_j}}^{-\mathbf{j}}\right)} \end{aligned}$$

where $\mathbf{y}_j$ is a vector count of the empirical distribution that using dish $m$-th in bus $j$-th, each element is computed as $y_{jm} = \sum_{i=1}^{N_j} \mathbb{I}\left(d_{z_j t_{ji}}, m\right)$. $\boldsymbol{\tau}_k$ is a vector count of empirical distribution using dish $m$-th in restaurant $k$-th, each element is calculated as $\tau_{km} = \sum_{j=1}^{J} \mathbb{I}(z_j = k) \sum_{i=1}^{N_j} \mathbb{I}\left(d_{z_j,t_{ji}} = m\right)$ and $c_{km}^{-j}$ is similar to $c_{km}$ excluding document $j$ as $c_{km}^{-j} = \sum_{\forall j' \neq j} \mathbb{I}(z_{j'} = k) \sum_{i=1}^{N_{j'}} \mathbb{I}\left(d_{z_{j'},t_{j'i}} = m\right)$.

The predictive likelihood of $p\left(\boldsymbol{y}_j \mid \boldsymbol{\tau}_{\mathbf{z_j}}^{-\mathbf{j}}\right)$ has a Multinomial distribution form given

the data vector of $\mathbf{y}_j$ and the parameter $\boldsymbol{\tau}_{z_j}^{-j}$.

$$\begin{aligned}\log p\left(t \mid d, z, v\right) &= \log p\left(t \mid z, v\right) + \log p\left(t \mid d, z\right) \\ &= \sum_{k=1}^{K}\left(o_{k*}-1\right)\log v + \sum_{k=1}^{K}\log\frac{\Gamma\left(v+1\right)}{\Gamma\left(v+n_{k*}\right)} \\ &+ \sum_{k=1}^{K}\log\sum_{t=1}^{o_{k*}}\Gamma\left(n_{kt}\right) + \sum_{j=1}^{J}\log p\left(\boldsymbol{y}_j \mid \boldsymbol{\tau}_{z_j}^{-j}\right).\end{aligned} \tag{7.3.12}$$

In order to apply small-variance asymptotic, we allow the variance of the above probabilities to go to zero. We represent this log likelihood in exponential family form following (Jiang et al., 2012) as:

$$p\left(\boldsymbol{y}_j \mid \boldsymbol{\tau}_{z_j}^{-j}\right) = \exp\left\{-d_\phi\left(\mathbf{y}_j \mid \boldsymbol{\tau}_{z_j}\right)\right\} b_\phi\left(\mathbf{y}_j\right) \tag{7.3.13}$$

where $d_\phi$ is a Bregman divergence corresponding to distribution $\phi$ (e.g., Bregman divergence of Multinomial distribution is KL divergence).

To retain the impact of hyperparameters $\alpha$ and to scale the variance appropriately in both types of probability (in Eq. 7.3.10 and Eq. 7.3.13), we define constant $\lambda_c > 0$ such that $\alpha = \exp\left(-\frac{\lambda_c}{2\sigma^2}\right)$ and $\hat{\beta} = \frac{\lambda_c}{2\sigma^2}$. Lemma 3.1 of Jiang et al. (2012) has proposed a proper way to scale the variance of exponential family distribution with the new parameter $\hat{\beta}$ as:

$$p\left(\boldsymbol{y}_j \mid \boldsymbol{\tau}_{z_j}^{-j}\right) = \exp\left\{-\hat{\beta} d_{\hat{\phi}}\left(\mathbf{y}_j \mid \boldsymbol{\tau}_{z_j}\right)\right\} b_{\hat{\phi}}\left(\mathbf{y}_j\right)$$

where $\hat{\phi} = \hat{\beta}\phi$ to scale the probability (as $\hat{\beta} \to \infty$, in the limit the variance goes to zero and the mean is the same as original form). Then, the log likelihood is written as:

$$\log p\left(\boldsymbol{y}_j \mid \boldsymbol{\tau}_{z_j}^{-j}\right) = -\frac{\lambda_c}{2\sigma^2}\sum_{j=1}^{J}\left\{d_{\hat{\phi}}\left(\mathbf{y}_j \mid \boldsymbol{\tau}_{z_j}\right) + \log b_{\hat{\phi}}\left(\mathbf{y}_j\right)\right\}.$$

In addition, in order to retain the impact of hyperparameters $\eta$ and $v$ in the limit, we can define some constants $\lambda_l$,$\lambda_g > 0$ such that $v = \exp\left(-\frac{\lambda_l}{2\sigma^2}\right)$, and $\eta = \exp\left(-\frac{\lambda_g}{2\sigma^2}\right)$ . The joint conditional distribution of $z, x, t, d$ will become the sum of Eq. 7.3.9,

Eq. 7.3.10, Eq. 7.3.12, and Eq. 7.3.11 in logarithm space.

$$\log p = -\frac{1}{2\sigma^2}\left\{(K-1)\,\lambda_c + (o_{k*}-1)\,\lambda_l + (M-1)\,\lambda_g - 2\sigma^2 C\right\} - \frac{1}{2\sigma^2}\left\{\sum_{j=1}^{J}\sum_{i=1}^{N_j} |||x_{ji} - \mu_{d_{z_j,t_{ji}}}||^2 + \lambda_c \sum_{j=1}^{J} d_{\hat{\phi}}\left(\mathbf{y}_j \mid \boldsymbol{\tau}_{z_j}\right)\right\}.$$

As $\sigma^2 \to 0$, we only consider the terms which are remaining in the limit:

$$\lim_{\sigma\to 0} -2\sigma^2 \log p = (K-1)\,\lambda_c + \sum_{k=1}^{K}(o_{k*}-1)\,\lambda_l + (M-1)\,\lambda_g + \sum_{j=1}^{J}\sum_{i=1}^{N_j} |||x_{ji} - \mu_{d_{z_j,t_{ji}}}||^2 + \lambda_c \sum_{j=1}^{J} KL\left(\mathbf{y}_j || \boldsymbol{\tau}_{z_j}\right)$$

We now consider maximizing the joint probability which is equivalent to minimise the objective function (under some constraints from the model) defined as:

$$\min_{t_{ji},d_{kt},z_j} \sum_{j=1}^{J}\sum_{i=1}^{N_j} ||x_{ji} - \mu_{d_{z_j,t_{ji}}}||^2 + \lambda_c \sum_{j=1}^{J} KL\left(\mathbf{y}_j || \boldsymbol{\tau}_{z_j}\right) + \lambda_l \sum_{k=1}^{K}(o_{k*}) + \lambda_g M + \lambda_c K \tag{7.3.14}$$

such that

$$\sum_{k=1}^{K}\sum_{j=1}^{J}\delta\left(z_j,k\right) \geq 1 \quad \sum_{k=1}^{K}\sum_{\forall t_{ji}|z_j=k}\sum_{u=1}^{o_{k*}}\delta\left(t_{ji},u\right) \geq 1 \sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{t=1}^{o_{k*}}\delta\left(d_{kt},m\right) \geq 1$$

where $\delta$ is a Dirac delta function (zero everywhere except at zero), $K \geq z_j \geq 1$, $o_{z_{j*}} \geq t_{ji} \geq 1$ and $M \geq d_{kt} \geq 1$.

The constraints can be intuitively understood as: (1) the value of each index can not excess the number of active component and (2) for each active component (restaurant, table, dish), there is existing at least one member for it. The above objective function in Eq. 7.3.14 has a kmeans-like objective form with penalized terms as used in previous works (Roychowdhury et al., 2013; Kulis and Jordan, 2012) which has been proved to monotonically decrease to a local convergence (Kulis and Jordan,

2012). We have three penalty parameters: $\lambda_c$ for controlling the number of restaurant, $\lambda_g$ for the number of global dish, $\lambda_l$ for the number of local dish. The first term is a distance of data point to a centre. The second term controls the number of local table in each restaurant. We note that $o_{km}$ is the number of table in restaurant $k$-th taking dish $m$-th. To minimise $o_{k*}$, we need to control the number of table $t_{ji}$ in each restaurant. The third and the last term manage the number of global dish $M$ and number of restaurant $K$, respectively.

### 7.3.7 Algorithm and computational analysis

To our knowledge, the existing optimization methods (e.g., convex optimization, quadratic programming, etc) are hard or unable to handle such a complicated objective function with constraints in Eq. 7.3.14. However, the hard assignment procedure in Section 7.3.5 can be directly applied to derive the algorithm for optimizing this objective function.

The high-level algorithm for learning the proposed Nested Kmeans is described in Algorithm 7.1 (interested readers can refer to the supplement for details). After initialization, we perform the hard-assignment iteratively to all customers, tables all restaurants, and all buses. Firstly, we assume the bus $j$-th arriving at restaurant $k$-th, then we assign all of the customer $x_{ji}$(s) to table $t_{ji}$(s) in that restaurant. Next, we will assign a dish to a table which can be a new dish or a used dish. Finally, a bus $j$-th will reassign to go to an existing restaurant $k$-th or a new restaurant.

We present the computational complexity of the above algorithm. We demonstrate that our approach scales linearly in the number of data points which can make scalability to large data sets. Denotes: $J$ is the number of documents, $N$ is the average number of words per document, $K$ is the (current) number of document clusters (or restaurants), $M$ is the (current) number of topics (or dishes), and $D$ is the dimension of the word feature (e.g., $x_{ji}$), $U$ is the average number of words taking a dish (in multiple documents), and $Q$ is the average number of dishes served in a restaurant. The majority of computations in the proposed Nested Kmeans algorithm arises from the following calculations per iteration: Computing all table indicators $t_{ji}$ having complexity O($J \times N \times M \times D$), dish indicators $d_{kt}$ includes

**Algorithm 7.1** High-level algorithm for Nested Kmeans

**Input**: $\{x_{ji}\}$: input data; $\lambda_l$: local dish penalty; $\lambda_g$: global dish penalty; $\lambda_\alpha$: penalty for clustering.

Initialize $K = 1, M = 1, r_k = 1, z_j = 1, t_{ji} = 1$.
**while** (~converged) **do**
  **for** each document $j = 1, ..., J$ **do**
    **for** each word $i = 1, ..., N_j$ **do**
      Assigning customer-table indicator $t_{ji}$ (Eq. 7.3.4)
    **end for**
  **end for**
  **for** each cluster $k = 1, ...K$ **do**
    Assigning local dish $d_{kt}$ to global dish (Eq. 7.3.5)
  **end for**
  **for** all bus $j = 1, ..., J$ **do**
    Assigning bus to restaurant $z_j$ (Eq. 7.3.6)
  **end for**
  Updating global dish $\psi_1, \psi_2, ..., \psi_M$
**end while**

**Output**: Global dishes $\psi_1, \psi_2, ..., \psi_M$, cluster index $z_j$, customer-table indicator $t_{ji}$, table-dish indicator $d_{kt}$.

complexity $\mathrm{O}(K \times M \times U \times D)$ and document cluster indicators $z_j$ possess complexity $\mathrm{O}(J \times K \times Q)$.

We assume the number of $K, M, D, Q \ll J, N$ for large scale data. We aware that this complexity involves feature dimension $D$, in practice we can ignore this term by vectorizing implementation (e.g., to compute Euclidean distance or KL divergence). In large scale dataset, where the total number of words (in all document) is much more than number of topics $M$, we finally obtain the complexity $\mathrm{O}(J \times N)$. nKmeans algorithm scales linearly in the number of data points which scalability is consequently feasible.

Though asymptotic analysis provides great way for scalability, its parameter selection (for penalty terms) is still under heuristic manner, used in most of the previous works (Jiang et al., 2012; Roychowdhury et al., 2013; Kulis and Jordan, 2012; Broderick et al., 2013b). We follow the simple farthest-first approach (Kulis and Jordan, 2012) to find penalties (lambdas) based on the input desired number of clusters, number of global dishes, number of local dishes per cluster.

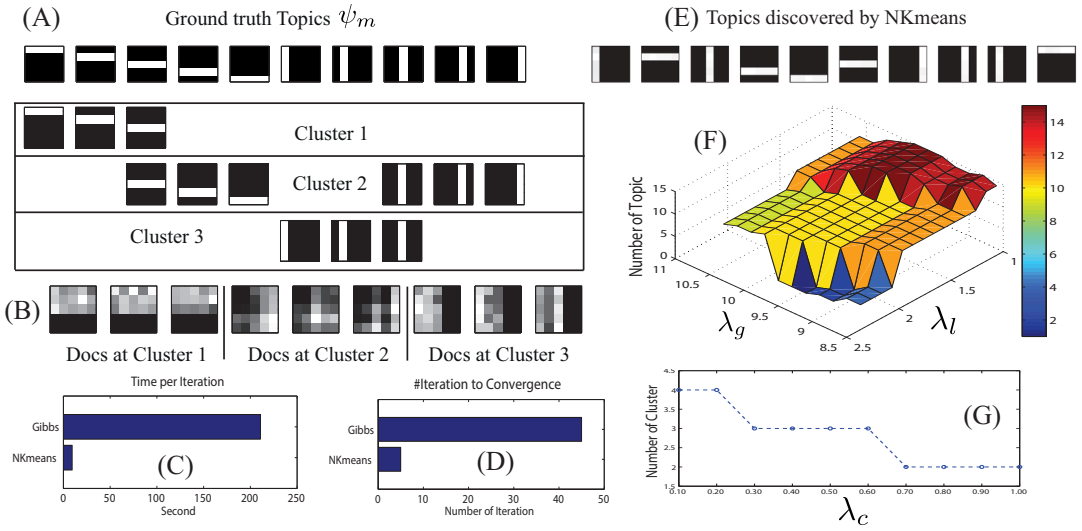



Figure 7.4.1: Synthetic experiment. Our algorithm takes totally 50 secs and collapsed Gibbs NDP takes 9450 secs for convergence. (A): Ground truth global topics and local topics in each cluster. (B): Examples of three documents in each cluster. (C): Running time per iteration comparison. (D): Number of iteration to convergence comparison. (E): Topics learned by nKmeans. (F): Number of topics as a function of $\lambda_g$ and $\lambda_l$. (G): Number of clusters as a function of $\lambda_c$.

## 7.4 Experiments

The main goal of this experiment is to highlight the scalability advantages (speed and accuracy) of our model and applications in real-world data. We first evaluate the model via synthetic studies. Then, we demonstrate the proposed algorithm enjoys many properties of Bayesian nonparametric modelling and multilevel clustering while we gain better performance than traditional probabilistic approach on image dataset. The Nested KMeans is significantly faster than the Gibbs sampler (for fixed accuracy), or can achieve higher accuracy for a fixed computational budget.

Throughout this section, all implementations are in Matlab environment on a Window machine Core i7 3.0GHz, 16GB Ram for fair comparison.

### 7.4.1 Synthetic example

We begin with a toy data to verify the algorithm consistency in recovering the correct document clusters and word topics. Using a bar topics (Griffiths and Steyvers, 2004), we initialize 10 global word topics (or global dishes in CFRB metaphor) including 5 horizontal and 5 vertical bars (see Fig 7.4.1-A). These topics are assigned into 3 clusters. Two clusters can take the same topics (e.g., cluster 1 shares the 3rd topic with cluster 2). Each document including a collection of words which will assign at a cluster. We then generate 600 documents which are organised into 3 clusters (200 documents per cluster). In each document, we draw 200 words following Multinomial distribution (each word observation vector is in the dimension of 25). An example of the generated documents arriving in different clusters can refer to Fig. 7.4.1-B.

We use KL divergence as a distance function in Nested Kmeans since data observation is Multinomial distributed. Multinomial-Dirichlet conjugacy is also utilised in probabilistic Gibbs inference for fair comparison. Our algorithm takes 50 seconds (5 iterations, 10 seconds per iteration) to recover correctly 10 number of word topics (cf. 7.4.1-E) and 3 document clusters. The document cluster assignments and word indicators are entirely matched to the ground truth. nKmeans convergence speed is compared to Gibbs inference which takes 9000 seconds (45 iterations with averagely 210 seconds per iteration) for convergence. The detailed computations are visualized in Fig. 7.4.1-(C,D) where our algorithm achieves great improvement in speed than probabilistic counterpart.

In Fig. 7.4.1-(F,G), we illustrate the effects of the input parameters $\lambda_g, \lambda_l, \lambda_c$ on learning the number of topics and number of clusters. The smaller value of lambdas results in more number of cluster and topic. We find that the algorithm depends heavily on the value of penalties (similar effects were also observed in previous works (Jiang et al., 2012; Roychowdhury et al., 2013; Kulis and Jordan, 2012)). We follow the heuristic approach described in Kulis and Jordan (2012) to choose lambdas.

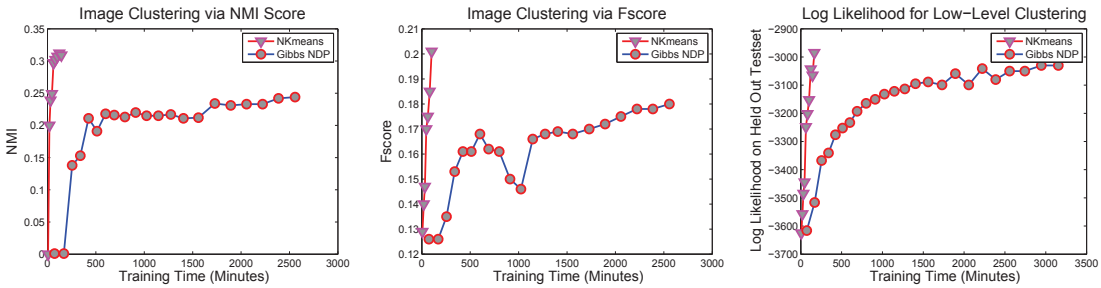



Figure 7.4.2: Scalable multilevel clustering performance comparison. Left: Image clustering by NMI. Middle: Image clustering by Fscore. Right: Log likelihood evaluated on held-out data.

### 7.4.2 Image clustering on Fifteen Scenes Category dataset

We use Fifteen Scenes image dataset[2] to demonstrate the effectiveness of the proposed algorithm on multilevel clustering of images and local features. There are totally 4,485 images and two million SIFT (128-dim) vectors are extracted which takes 2.6GB for storage in local machine. This data is sufficiently large for our scalability evaluation since nKmeans performs clustering to all of millions local descriptors and thousands of images simultaneously.

The images are organised into 15 categories (e.g., bedroom, kitchen, MITcoast, and so on). The provided image categories will further be used as a ground truth for evaluation. We treat an image as a document (totally 4,485 documents). On each image, we extract SIFT (Lowe, 2004) descriptors, considered as words $x_{ji}$ for a document $j$. Each word of the raw SIFT descriptor (dimension of 128) is further used by our model. In a probabilistic perspective, the oriented gradients of SIFT feature contain the *sufficient statistic* (by magnitude) regarding to different orientations. Therefore, it makes sense to assumed SIFT feature follow Multinomial distribution and use with its Bregman divergence as KL divergence (Banerjee et al., 2005). In fact, we experimentally find KL divergence distance achieve slightly better clustering performance than Euclidean distance on SIFT feature.

The data is split into 3 sets: training 50%, validation 20%, and testing 30%. The validation set is used to select parameters. The document clustering evaluation is run on training set (1,300 images and one million local descriptors). The word

[2]http://www-cvr.ai.uiuc.edu/ponce_grp/data/

clustering is evaluated by running the model on training set, and computing the predictive log likelihood on held-out testing set.

Our non-probabilistic algorithm returns 22 image clusters (e.g., restaurants) and 60 SIFT topics (e.g., dishes) with the following parameters: $\lambda_g = 4.67$,$\lambda_l = 0.39$, and $\lambda_c = 0.03$. These parameters are chosen by running experiment on validation set from following the heuristic farthest-first approach used in Kulis and Jordan (2012).

We compare our nKmeans performance against the probabilistic Gibbs inference. Each nKmeans iteration takes roughly from 400-500 seconds. It converges within 30 iterations (totally 3.5-4 hours). Collapsed Gibbs sampling slowly consumes 5,000-9,000 seconds per iteration. It requires about 500 iterations to be converged that is equivalent to 970 hours (40 days). For a fixed computational budget, we compare the image clustering performance of nKmeans and Gibbs inference. The performance according to time (up to 45 hours running on training set) is recorded and displayed in Fig. 7.4.2. We achieve better clustering scores at two levels (image clustering and SIFT grouping) within the fixed computational milestones. Because we do not have ground truth for each SIFT feature (word level), we use predictive log likelihood per word on held-out test set given training set ( $\log p\left(x_{ji}^{Test} \mid x_{ji}^{Train}\right)$) to make quantitative comparisons for clustering word level. The best word grouping will give high probability to predict new unseen word. This fact is widely used in evaluating topic modelling performance (D. Blei, 2007; Wallach et al., 2009).

We use the Purity, NMI, RI, Fscore (details in Rand (1971); Cai et al. (2011)) to compute evaluation for image clustering task which we have ground truth of 15 categories. The following baseline clustering methods are used for comparison:

- k-means and Non-negative Matrix Factorization (NNMF) (Lee et al., 1999): we use Matlab built-in function. We alter the number of clusters from 5 to 25. The min, max, mean, and standard deviation is reported.
- Affinity Propagation (AP) (Frey and Dueck, 2007): the Euclidean distance is used to compute the similarity score.
- DPMeans (Kulis and Jordan, 2012): We implement DPMeans algorithm, then use Euclidean distance as asymptotic distance for Gaussian distribution

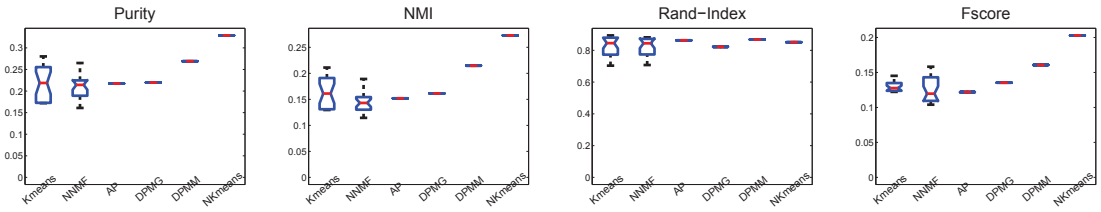



Figure 7.4.3: Image clustering comparison with four evaluation criteria: Purity, NMI, RI, and Fscore.

(DPMG), and KL distance as asymptotic distance for Multinomial distribution (DPMM), respectively. We observe that KL divergence version performs better than Euclidean version.

Fig. 7.4.3 presents image clustering performance that our model consistently gains the highest performance across all four criteria. In addition, the baseline methods only perform clustering at one level which cannot do multilevel clustering. In contrast, nKmeans can do clusters at two levels, then it offers flexibility for data grouping.

## 7.5 Closing Remark

In this chapter, we have addressed the problem of scalable multilevel clustering where observations are organised into groups. Our aim is to discover the labels at multilevel of individuals and groups for large scale applications. As the amount of available data grows super-abundantly, traditional approaches for multilevel clustering using probabilistic models could be computationally problematic. Thus, developing scalable approach for multilevel clustering plays an important role for real-world applications. In addition, it is hard, or impossible, to do model selection to select the number of clusters for large-scale data. Therefore, our Bayesian nonparametric approach is promising as it can identify the suitable number of clusters (at multilevel) from the data.

To this end, we have proposed a scalable Bayesian nonparametric multilevel clustering algorithm, namely Nested Kmeans. It was built on the theory of small variance asymptotic technique for the nested Dirichlet Process to yield a deterministic com-

putational time as fast as the classic K-means for a rich probabilistic framework. The nKmeans algorithm gains great scalability than the probabilistic counterpart while preserving the benefits of Bayesian nonparametric modelling. Experiments on synthetic data and large-scale image clustering task have demonstrated the advantages of our proposed model.

# Chapter 8

# Conclusion and Future Work

This thesis has presented a systematic investigation into multilevel data analysis under the recently emerged theory of Bayesian nonparametrics. Our broad objective was to advance the knowledge base in Bayesian nonparametrics and multilevel modelling literature. At the same time we have sought to develop novel methods and applications across domains in text analytics, abnormality detection in video surveillance, image analysis and information retrieval.

## 8.1 Summary

Our contributions from this thesis can be broadly categorised into two main themes. The first includes the development of novel approaches for modelling multilevel data. The applications include segmenting, browsing and extracting latent features from the multilevel structured video data for abnormality detection in Chapter 3 and for group-level classification in Chapter 5. The second part is the theoretical contributions to multilevel modelling using Bayesian nonparametrics for multilevel clustering presented in Chapter 4, multilevel regression in Chapter 6, and scalable multilevel clustering in Chapter 7.

Specifically, Chapter 3 contributes our first study into the abnormality detection in video surveillance using Bayesian nonparametric theory where video frames and

frame blocks are organised into multilevel structure. We propose to use the Infinite Hidden Markov model grounded in the theory of Dirichlet process, to segment video stream into infinite number of coherent sections for multi-model abnormality detection at video frame level. We have segmented the motion from video surveillance into two sections of day time (from 8am to 9pm) and night time (from 10pm to 7am) where the motion is more crowded and active on day time than night time. Then, we computed the principle and residual subspaces for abnormality detection on each segmented sections respectively. Furthermore, we built an interactive system for a user to browse and filter abnormal events in which the factors were learned using Bayesian Nonparametric Factor Analysis. Using spatial filtering, the proposed framework can find abnormality at multilevel, video frames level and frame blocks level (e.g., which video frame is abnormal, and then in this video frame which specific frame block is abnormal).

In Chapter 4, we delved deeper into multilevel clustering problem to jointly cluster data at multilevel using the theory of Bayesian nonparametrics. We introduced the Multilevel Clustering with Group-Level Context (MC$^2$) which can perform multilevel clustering and utilise group-level context if available. To our knowledge, MC$^2$ is the first model in Bayesian nonparametric for multilevel modelling that can rigorously model observations at multiple levels. Using the Dirichlet Process as the building block, our model constructs a product base-measure with a nested structure to accommodate content and context observations at multiple levels. In particular, our key contributions from this chapter are the following:

- We introduce the MC$^2$ model for nonparametric multilevel clustering where the number of clusters at multilevel are not known in advance. Our MC$^2$ possesses a principle mechanism to accommodate group-level context observation for improving modelling and clustering performance.
- We provide a Polya-urn view of the model and an efficient collapsed Gibbs inference procedure.
- A detailed theoretical analysis of marginalization property is also provided.
- We perform extensive experiments including image clustering and text modelling.

MC² outperformed other baselines methods in various experiments including image clustering and text modelling. We also demonstrate that MC² can utilise completely missing or partially missing context observation because the context information is not always available for learning.

Next, Chapter 5 presented our work on extracting latent representation and performs classification for groups where data are organised in multilevel with individuals and groups. For each group, we observe the individuals which are in high dimension and noisy. Because there is the difficulty in using the raw observations at individual level (high-dimensional and noisy), we propose to extract the low-dimensional feature embedded inside the data for each group, using probabilistic frameworks for multilevel data such as LDA (Blei et al., 2003), HDP (Teh et al., 2006) and our recent proposed MC² (Nguyen et al., 2014). After obtaining probabilistic feature from multilevel models, we proposed the Topic Model Kernel to perform document classification with Support Vector Machine (Cortes and Vapnik, 1995). The proposed kernel is not only outperforming baseline kernels on probabilistic feature, but also achieves comparable performances on generic features (non-probabilistic feature).

In Chapter 6, we introduced the Bayesian Nonparametric Multilevel Regression (BNMR) for modelling and predicting continuous outcome, where data observation are organised into multilevel structure. Our BNMR also employs the group-level context information to induce the group clusters to strengthen the regression performance by borrowing information across similar groups. Thus, it allows predicting for individuals on unseen (or new) groups which is known to be a difficult problem in multilevel analysis. The proposed model is the first Bayesian nonparametric framework for multilevel regression for predicting unseen grouped data. We derived model representation and collapsed Gibbs sampler for posterior inference. We have performed extensive experiments on econometric panel data and healthcare longitudinal data to demonstrate the effectiveness of the proposed model.

Our next major contribution contained in Chapter 7, we formalised the novel method for multilevel clustering, emphasising on the scalability and speed, namely Nested K-means. We introduced the concept of Chinese Restaurant Franchise-Bus upon which our result is derived using the principle of the recent small variance asymptotic analysis framework (Kulis and Jordan, 2012). This resulted in an algorithm that can nestedly cluster data points within group and groups themselves. Furthermore,

the number of local clusters within each group and the number of global clusters are also automatically induced due to the inherent property of Bayesian nonparametric. We validated the proposed model on scalable multilevel clustering where we achieve extremely high performance for clustering at both levels comparing with the Gibbs sampling on the speed axes.

## 8.2 Future Directions

In this section, we discuss some possible future extensions to the work presented in this thesis.

The first extension is originated from Chapter 3 namely *multilevel abnormality detection*. Traditional anomaly detection research focuses on a single level of data points. Often the most interesting or unusual things in the data are not only appearing at individual points, but also become apparent in groups when the data are considered at multilevel setting. Multilevel abnormalies exist in many real-world problems. For example, in video surveillance, we consider which video frame the abnormal event happens and at which specific block in the video frame is anomalous. Our aim is to discover anomalous behaviors from groups of data. For this purpose, one might wish to extend our work to model the grouped data using probabilistic model to detect various types of anomalies at multilevel.

The second possible direction from Chapter 4 is *multilevel clustering with multiple/multilevel contexts*. A natural extension is to consider the multilevel clustering task where the context observations can be observed repeatedly at different levels. The first extension could consider the case where we observe more than one kind of context observations at group-level. For example, given a collection of documents, each document contains a collection of words considered as *content* and each document has author, timestamp and title information which are considered as *multiple contexts*. Particularly, denote by $J$ the number of groups, $N_j$ be the number of observations in group $j$, our data observations contain a collection of content words $\mathbf{w}_j = \{w_{j1}, w_{j2}, \ldots, w_{jN_j}\}$ and group-level (multiple) context $\boldsymbol{x}_j = \{x_{j1}, x_{j2}, ..., x_{jT_j}\}$ where $T_j$ is the number of context in group $j$. The second plausible extension includes the case where we observe the *multilevel contexts* comprising of the group-level

context (e.g., author or timestamp information) and individual-level context (e.g., part-of-speech tagging or annotation for word). To be more concrete, one can have a collection of content word $\mathbf{w}_j = \{w_{j1}, w_{j2}, \ldots, w_{jN_j}\}$, group-level multiple context $\boldsymbol{x}_j = \{x_{j1}, x_{j2}, ..., x_{jT_j}\}$, and individual-level context $\boldsymbol{s}_j = \{s_{j1}, s_{j2}, ..., s_{jN_j}\}$. We note that the context observations of $\boldsymbol{x}$ and $\boldsymbol{s}$ appear at multilevel. All of these contexts at individual level and group-level carry useful knowledge to improve multilevel clustering and modelling performance which can be readily accommodated using the theory we have developed in this thesis.

The third possible extension is to develop alternative scalable inference technique for MC². Traditional Markov Chain Monte Carlo sampler for MC² could be unsuitable large scale dataset. Therefore, there is a need to research into more efficient and scalable methods for multilevel learning. It seems readily doable to again employ the small variance asymptotic technique in Chapter 7 to derive the scalable version for our proposed MC². Another alternative option is developing streaming variational Bayes (Broderick et al., 2013a) for MC² to allow a single pass through each data point that is suitable for a large scale dataset.

# Bibliography

L. S. Aiken, S. G. West, and R. R. Reno. *Multiple regression: Testing and interpreting interactions*. Sage, 1991.

E. B. Andersen. Sufficiency and exponential families for discrete sample spaces. *Journal of the American Statistical Association*, 65(331):1248–1255, 1970.

C. Andrieu, N. De Freitas, A. Doucet, and M. Jordan. An introduction to MCMC for machine learning. *Machine learning*, 50(1):5–43, 2003.

J. Antolín, J. Angulo, and S. López-Rosa. Fisher and jensen–shannon divergences: Quantitative comparisons among distributions. application to position and momentum atomic densities. *The Journal of chemical physics*, 130:074110, 2009.

C. Antoniak. Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *The Annals of Statistics*, 2(6):1152–1174, 1974.

A. Argyriou, T. Evgeniou, and M. Pontil. Convex multi-task feature learning. *Machine Learning*, 73(3):243–272, 2008.

A. Banerjee, S. Merugu, I. S. Dhillon, and J. Ghosh. Clustering with bregman divergences. *The Journal of Machine Learning Research*, 6:1705–1749, 2005.

K. Barnard, P. Duygulu, D. Forsyth, N. De Freitas, D. Blei, and M. Jordan. Matching words and pictures. *The Journal of Machine Learning Research*, 3:1107–1135, 2003. ISSN 1532-4435.

J. Basak. A least square kernel machine with box constraints. In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pages 1–4. IEEE, 2008.

M. Bayes and M. Price. An essay towards solving a problem in the doctrine of chances. by the late rev. mr. bayes, frs communicated by mr. price, in a letter

to john canton, amfrs. *Philosophical Transactions (1683-1775)*, pages 370–418, 1763.

M. Beal, Z. Ghahramani, and C. Rasmussen. The infinite hidden Markov model. In *Advances in Neural Information Processing Systems (NIPS)*, volume 1, pages 577–584. MIT, 2002.

D. Beeferman and A. Berger. Agglomerative clustering of a search engine query log. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 407–416. ACM, 2000.

J. O. Berger. *Statistical decision theory and Bayesian analysis*. Springer Science & Business Media, 1985.

J. A. Bilmes et al. A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. *International Computer Science Institute*, 4(510):126, 1998.

C. M. Bishop. *Pattern recognition and machine learning*. springer New York, 2006.

M. Bishop and E. Thompson. Maximum likelihood alignment of dna sequences. *Journal of molecular biology*, 190(2):159–165, 1986.

D. Blackwell and J. MacQueen. Ferguson distributions via Pólya urn schemes. *The annals of statistics*, 1(2):353–355, 1973.

D. Blei and M. Jordan. Variational inference for dirichlet process mixtures. *Bayesian Analysis*, 1(1):121–143, 2006.

D. Blei and J. Lafferty. Dynamic topic models. In *Proc. Int. Conf. on Machine learning ICML'06*, pages 113–120. ACM New York, NY, USA, 2006.

D. Blei and J. Lafferty. A correlated topic model of science. *Annals of Applied Statistics*, 1(1):17–35, 2007.

D. Blei, A. Ng, and M. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003. ISSN 1533-7928.

J. A. Bondy and U. S. R. Murty. *Graph theory with applications*, volume 290. Macmillan London, 1976.

B. Boser, I. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. pages 144–152, 1992.

S. Boughorbel, J.-P. Tarel, and N. Boujemaa. Conditionally positive definite kernels for svm based image recognition. In *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, pages 113–116. IEEE, 2005.

S. Boykin and A. Merlino. Machine learning of event segmentation for news on demand. *Communications of the ACM*, 43(2):35–41, 2000.

T. Broderick, N. Boyd, A. Wibisono, A. C. Wilson, and M. I. Jordan. Streaming variational bayes. In *Advances in Neural Information Processing Systems*, pages 1727–1735, 2013a.

T. Broderick, B. Kulis, and M. Jordan. Mad-bayes: Map-based asymptotic derivations from bayes. In *Proceedings of The 30th International Conference on Machine Learning*, pages 226–234, 2013b.

S. Budhaditya, D. Pham, M. Lazarescu, and S. Venkatesh. Effective anomaly detection in sensor networks data streams. In *Data Mining, 2009. ICDM'09. Ninth IEEE International Conference on*, pages 722–727. IEEE, 2009.

H. Bui, D. Phung, and S. Venkatesh. Hierarchical hidden markov models with general state hierarchy. In D. L. McGuinness and G. Ferguson, editors, *Procs. of the National Conference on Artificial Intelligence (AAAI)*, pages 324–329, San Jose, California, USA, 2004. AAAI Press / The MIT Press.

R. P. Cabeen and D. H. Laidlaw. White matter supervoxel segmentation by axial dp-means clustering. In *Medical Computer Vision. Large Data in Medical Imaging*, pages 95–104. Springer, 2014.

D. Cai, X. He, and J. Han. Locally consistent concept factorization for document clustering. *Knowledge and Data Engineering, IEEE Transactions on*, 23(6):902–913, 2011.

T. Campbell, M. Liu, B. Kulis, J. P. How, and L. Carin. Dynamic clustering via asymptotics of the dependent dirichlet process mixture. In *Advances in Neural Information Processing Systems*, 2013.

E. J. Candes, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.

R. Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997.

G. Chamberlain. Multivariate regression models for panel data. *Journal of Econometrics*, 18(1):5–46, 1982.

A. B. Chan, N. Vasconcelos, and P. J. Moreno. A family of probabilistic kernels based on information divergence. *Univ. California, San Diego, CA, Tech. Rep. SVCL-TR-2004-1*, 2004.

C.-C. Chang and C.-J. Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

M. Chavent, Y. Ding, L. Fu, H. Stolowy, and H. Wang. Disclosure and determinants studies: An extension using the divisive clustering method (div). *European Accounting Review*, 15(2):181–218, 2006.

X. Chen, M. Zhou, and L. Carin. The contextual focused topic model. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 96–104. ACM, 2012.

G. Claeskens and N. L. Hjort. *Model selection and model averaging*, volume 330. Cambridge University Press Cambridge, 2008.

C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

J. M. D. Blei. Supervised topic models. *Proc. Neural Information Processing Systems. NIPS 2007*, 2007.

A. Darwiche. *Modeling and reasoning with Bayesian networks*, volume 1. Cambridge University Press, 2009.

S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407, 1990.

A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society - Series B (Methodological)*, pages 1–38, 1977.

F. T. Denton and B. G. Spencer. Chronic health conditions: changing prevalence in an aging population and some implications for the delivery of health care services. *Canadian Journal on Aging*, 29(1):11, 2010.

P. Diaconis, D. Ylvisaker, et al. Conjugate priors for exponential families. *The Annals of statistics*, 7(2):269–281, 1979.

A. V. Diez-Roux. Multilevel analysis in public health research. *Annual review of public health*, 21(1):171–192, 2000.

A. Dubey, A. Hefny, S. Williamson, and E. P. Xing. A nonparametric mixture model for topic modeling over time. In *SDM*, pages 530–538. SIAM, 2013.

T. Duong, H. Bui, D. Phung, and S. Venkatesh. Activity recognition and abnormality detection with the Switching Hidden Semi-Markov Model. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 838–845, San Diego, 20-26 June 2005. IEEE Computer Society.

M. Ebden. Gaussian processes for regression: A quick introduction. *The Website of Robotics Research Group in Department on Engineering Science, University of Oxford*, 2008.

D. Edwards. *Introduction to graphical modelling*. Springer, 2000.

P. K. Elango and K. Jayaraman. Clustering images using the latent dirichlet allocation model. *University of Wisconsin*, 2005.

D. M. Endres and J. E. Schindelin. A new metric for probability distributions. *IEEE Transactions on Information theory*, 2003.

A. Eriksson and A. van den Hengel. Efficient computation of robust low-rank matrix approximations in the presence of missing data using the l1 norm. In *Procs. of IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2010. best paper award.

M. Escobar and M. West. Bayesian density estimation and inference using mixtures. *Journal of the american statistical association*, 90(430):577–588, 1995.

B. S. Everitt, D. J. Hand, et al. *Finite mixture distributions*, volume 9. Chapman and Hall London, 1981.

X. Fan, Y. Zeng, and L. Cao. Non-parametric power-law data clustering. *arXiv preprint arXiv:1306.3003*, 2013.

T. Ferguson. A Bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1(2):209–230, 1973.

J. R. Finkel, T. Grenager, and C. D. Manning. The infinite tree. In *In Association for Computational Linguistics (ACL)*, volume 45, page 272. Citeseer, 2007.

F. Fleuret and H. Sahbi. Scale-invariance of support vector machines based on the triangular kernel. In *3rd International Workshop on Statistical and Computational Theories of Vision*, 2003.

E. Fox, E. Sudderth, M. Jordan, and A. Willsky. Developing a tempered HDP-HMM for systems with state persistence. Technical report, MIT Laboratory for Information and Decision Systems, 2007.

E. B. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky. An hdp-hmm for systems with state persistence. In *Proceedings of the 25th international conference on Machine learning*, pages 312–319. ACM, 2008.

C. Fraley and A. E. Raftery. How many clusters? which clustering method? answers via model-based cluster analysis. *The computer journal*, 41(8):578–588, 1998.

B. Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 315:972–976, February 2007. ISSN 1095-9203. doi: 10.1126/science.1136800.

M. Fritz and B. Schiele. Decomposition, discovery and detection of visual categories using topic models. pages 1–8, 2008.

A. Gelfand, A. Kottas, and S. MacEachern. Bayesian nonparametric spatial modeling with dirichlet process mixing. *Journal of the American Statistical Association*, 100(471):1021–1035, 2005.

A. Gelman and J. Hill. *Data analysis using regression and multilevel/hierarchical models*. Cambridge University Press, 2006.

A. Gelman, J. Carlin, H. Stern, and D. Rubin. *Bayesian Data Analysis*. Chapman & Hall/CRC, 2003.

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.

W. R. Gilks, S. Richardson, and D. J. Spiegelhalter. Introducing Markov chain monte carlo. In *Markov chain Monte Carlo in practice*, pages 1–19. Springer, 1996.

M. Girolami and A. Kabán. On an equivalence between plsi and lda. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 433–434. ACM, 2003.

Y. Goldberg and M. Elhadad. splitsvm: fast, space-efficient, non-heuristic, polynomial kernel computation for nlp applications. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, pages 237–240. Association for Computational Linguistics, 2008.

H. Goldstein. Multilevel mixed linear model analysis using iterative generalized least squares. *Biometrika*, 73(1):43–56, 1986.

H. Goldstein. *Multilevel statistical models*, volume 922. John Wiley & Sons, 2011.

S. Goldwater, T. L. Griffiths, and M. Johnson. Contextual dependencies in unsupervised word segmentation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 673–680. Association for Computational Linguistics, 2006.

K. C. Gowda and T. Ravi. Divisive clustering of symbolic objects using the concepts of both similarity and dissimilarity. *Pattern Recognition*, 28(8):1277–1282, 1995.

T. Griffiths and Z. Ghahramani. Infinite latent feature models and the Indian buffet process. *Advances in Neural Information Processing Systems*, 18:475, 2006.

T. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(90001):5228–5235, 2004.

W. E. L. Grimson, C. Stauffer, R. Romano, and L. Lee. Using adaptive tracking to classify and monitor activities in asite. In *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 22–29, 1998.

S. R. Gunn. Support vector machines for classification and regression. *ISIS technical report*, 14, 1998.

S. Gupta, D. Phung, and S. Venkatesh. A nonparametric Bayesian Poisson Gamma model for count data. In *Proceedings of International Conference on Pattern Recognition (ICPR)*, pages 1815–1818, 2012.

S. Gupta, D. Phung, and S. Venkatesh. Factorial multi-task learning : A bayesian nonparametric approach. In *Proceedings of International Conference on Machine Learning (ICML)*, Atlanta, USA, June 16-21 2013.

N. Hjort. Nonparametric Bayes estimators based on beta processes in models for life history data. *The Annals of Statistics*, 18(3):1259–1294, 1990. ISSN 0090-5364.

M. Hoffman, D. Blei, and F. Bach. Online learning for latent dirichlet allocation. *Advances in Neural Information Processing Systems*, 23:856–864, 2010.

T. Hofmann. Probabilistic latent semantic indexing. In *Proc. ACM SIGIR Int. Conf. on Research and Development in Information Retrieval*, pages 50–57, New York, NY, USA, 1999. ACM. ISBN 1-58113-096-1. doi: http://doi.acm.org/10.1145/312624.312649.

T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42(1):177–196, 2001.

S. Horbelt, A. Munoz, T. Blu, and M. Unser. Spline kernels for continuous-space image processing. In *Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on*, volume 6, pages 2191–2194. IEEE, 2000.

B. Horn and B. Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3): 185–203, 1981.

J. Hox. *Multilevel analysis: Techniques and applications*. Routledge, 2010.

C.-W. Hsu, C.-C. Chang, C.-J. Lin, et al. A practical guide to support vector classification, 2003.

J. Huang, Z. Liu, and Y. Wang. Joint video scene segmentation and classification based on hidden markov model. In *Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on*, volume 3, pages 1551–1554. IEEE, 2000.

U. Iurgel, R. Meermeier, S. Eickeler, and G. Rigoll. New approaches to audio-visual segmentation of tv news for automatic topic retrieval. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, volume 3, pages 1397–1400. IEEE, 2001.

A. K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651–666, 2010.

J. L. W. V. Jensen. Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta Mathematica*, 30(1):175–193, 1906.

J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 119–126. ACM, 2003.

K. Jiang, B. Kulis, and M. I. Jordan. Small-variance asymptotics for exponential family dirichlet process mixture models. In *Advances in Neural Information Processing Systems*, pages 3167–3175, 2012.

S. C. Johnson. Hierarchical clustering schemes. *Psychometrika*, 32(3):241–254, 1967.

M. I. Jordan. Graphical models. *Statistical Science*, pages 140–155, 2004.

M. I. Jordan and Y. Weiss. Probabilistic inference in graphical models. *Handbook of neural networks and brain theory*, 2002.

Z. Kang, K. Grauman, and F. Sha. Learning with whom to share in multi-task feature learning. pages 521–528, 2011.

E. Kijak, G. Gravier, P. Gros, L. Oisel, and F. Bimbot. Hmm based structuring of tennis videos using visual and audio cues. In *Multimedia and Expo, 2003. ICME'03. Proceedings. 2003 International Conference on*, volume 3, pages III–309. IEEE, 2003.

I. Kitov. Gdp growth rate and population". *Economics Bulletin*, 28(9):A1, 2005.

P. Kohli, P. H. Torr, et al. Robust higher order potentials for enforcing label consistency. *International Journal of Computer Vision*, 82(3):302–324, 2009.

D. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

B. Kulis and M. I. Jordan. Revisiting k-means: New algorithms via bayesian nonparametrics. In *Proceedings of the 29th International Conference on Machine Learning (ICML)*, Edinburgh, UK, 2012.

S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.

J. Lafferty and L. Wasserman. Challenges in statistical machine learning. 2006.

P. S. Laplace. Memoir on the probability of the causes of events. *Statistical Science*, pages 364–378, 1986.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

D. Lee and H. Seung. Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing Systems*, 13, 2001.

D. D. Lee, H. Seung, et al. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

A. H. Leyland and H. Goldstein. Multilevel modelling of health statistics. 2001a.

A. H. Leyland and H. Goldstein. *Multilevel modelling of health statistics*. Wiley, 2001b.

P. Liang, S. Petrov, M. I. Jordan, and D. Klein. The infinite pcfg using hierarchical dirichlet processes. In *EMNLP-CoNLL*, pages 688–697, 2007.

J. Liu. The collapsed gibbs sampler in bayesian computations with applications to a gene regulation problem. *Journal of the American Statistical Association*, 89 (427):958–966, 1994.

A. Y. Lo. On a class of Bayesian nonparametric estimates: I. density estimates. *The Annals of Statistics*, 12(1):351–357, 1984.

D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

D. A. Luke. *Multilevel modeling*, volume 143. Sage, 2004.

W. Luo, D. Phung, V. Nguyen, T. Tran, and S. Venkatesh. Speed up health research through topic modeling of coded clinical data. In *2nd International Workshop on Pattern Recognition for Healthcare Analytics*. https://sites.google.com/site/iwprha2/proceedings, 2014.

S. MacEachern. Dependent nonparametric processes. In *ASA Proceedings of the Section on Bayesian Statistical Science*, pages 50–55, 1999.

D. MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

J. MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, page 14. California, USA, 1967.

A. Maddison. Statistics on world population, gdp and per capita gdp, 1-2008 ad. *Historical Statistics*, 2010.

T. Masada, S. Kiyasu, and S. Miyahara. Clustering images with multinomial mixture models. pages 343–348, 2007.

R. A. McLean, W. L. Sanders, and W. W. Stroup. A unified approach to mixed linear models. *The American Statistician*, 45(1):54–64, 1991.

A. Merlino, D. Morey, and M. Maybury. Broadcast news navigation using story segmentation. In *Proceedings of the fifth ACM international conference on Multimedia*, pages 381–391. ACM, 1997.

N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

T. Minka and J. Lafferty. Expectation-propagation for the generative aspect model. In *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, pages 352–359. Morgan Kaufmann Publishers Inc., 2002.

P. J. Moreno, P. Ho, and N. Vasconcelos. A kullback-leibler divergence based kernel for svm classification in multimedia applications. *Advances in neural information processing systems*, 16:1385–1393, 2003.

A. H. Munnell and L. M. Cook. How does public infrastructure affect regional economic performance? In *Is There a Shortfall in Public Capital Investment? Proceedings of a Conference*, 1990.

K. P. Murphy. Conjugate bayesian analysis of the gaussian distribution. 1:16, 2007.

K. P. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

B. Muthén and T. Asparouhov. Multilevel regression mixture analysis. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 172(3):639–657, 2009.

M. Natrella. Nist/sematech e-handbook of statistical methods. 2010.

J. F. Navarro, C. S. Frenk, and S. D. White. A universal density profile from hierarchical clustering. *The Astrophysical Journal*, 490(2):493, 1997.

R. Neal. Markov chain sampling methods for Dirichlet process mixture models. *Journal of computational and graphical statistics*, 9(2):249–265, 2000.

T. Nguyen, D. Phung, G. Sunil, and S. Venkatesh. Interactive browsing system for anomaly video surveillance. In *Proc. of IEEE Eight International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, pages 384 – 389, Melbourne, Australia, April 2013a.

T. Nguyen, D. Phung, X. Venkatesh, S. Nguyen, and H. Bui. Bayesian nonparametric multilevel clustering with group-level contexts. In *Proc. of International Conference on Machine Learning (ICML)*, pages 288–296, Beijing, China, 2014.

T. C. Nguyen, D. Phung, S. Gupta, and S. Venkatesh. Extraction of latent patterns and contexts from social honest signals using hierarchical Dirichlet processes. In *2013 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 47–55, 2013b.

T.-V. Nguyen, T. Tran, P. Vo, and B. Le. Efficient image segmentation incorporating photometric and geometric information. In *Proceedings of the International MultiConference of Engineers and Computer Scientists*, volume 1, 2011.

T.-V. Nguyen, N. Pham, T. Tran, and B. Le. Higher order conditional random field for multi-label interactive image segmentation. In *Computing and Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), 2012 IEEE RIVF International Conference on*, pages 1–4. IEEE, 2012a.

T. V. Nguyen, D. Phung, S. Rana, D. S. Pham, and S. Venkatesh. Multi-modal abnormality detection in video with unknown data segmentation. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 1322–1325, Tsukuba, Japan, November 2012b. IEEE.

T. V. Nguyen, D. Phung, , and S. Venkatesh. Topic model kernel: An empirical study towards probabilistically reduced features for classification. In *Neural Information Processing*, pages 124–131. Springer Berlin Heidelberg, 2013c.

V. Nguyen, D. Phung, D.-S. Pham, and S. Venkatesh. Bayesian nonparametric approaches to abnormality detection in video surveillance. *Annals of Data Science*, pages 1–21, 2015a.

V. Nguyen, D. Phung, and S. Venkatesh. Topic model kernel classification with probabilistically reduced features. *Journal of Data Science*, 13(2), 2015b.

V. Nguyen, D. Phung, S. Venkatesh, and H. Bui. A bayesian nonparametric approach to multilevel regression. In *Proc. of Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, pages 330–342, 2015c.

K. Ni, L. Carin, and D. Dunson. Multi-task learning for sequential data via iHMMs and the nested Dirichlet process. In *Proc. of Int. Conf. on Machine learning (ICML)*, page 696. ACM, 2007.

J. R. Norris. *Markov chains*. Number 2. Cambridge university press, 1998.

J. Novovičová and A. Malík. Application of multinomial mixture model to text classification. In *Pattern Recognition and Image Analysis*, pages 646–653. Springer, 2003.

A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001.

P. Orbanz and Y. W. Teh. Bayesian nonparametric models. 2010.

J. Paisley and L. Carin. Nonparametric factor analysis with Beta process priors. In *Procs. of the International Conference on Machine Learning (ICML)*, pages 777–784. ACM, 2009.

A. Passos, P. Rai, J. Wainer, and H. Daume. Flexible modeling of latent task structures in multitask learning. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 1103–1110, 2012.

J. Pearl. *Probabilistic reasoning in intelligent systems; Networks of Plausible Inference*. California: Kaufmann, 1988.

E. J. Pedhazur and F. N. Kerlinger. Multiple regression in behavioral research. 1982.

D. Pham, S. Rana, D. Phung, and S. Venkatesh. Generalized median filtering - a robust matrix decomposition perspective. *Preprint*, 2011.

D. Phung. *Probabilistic and Film Grammar Based Methods for Video Content Analysis*. PhD thesis, Curtin University of Technology, Australia, 2005.

D. Phung, X. Nguyen, H. Bui, T. Nguyen, and S. Venkatesh. Conditionally dependent Dirichlet processes for modelling naturally correlated data sources. Technical report, Pattern Recognition and Data Analytics, Deakin University, 2012.

D. Phung, T. C. Nguyen, S. Gupta, and S. Venkatesh. Learning latent activities from social signals with hierarchical Dirichlet process. In G. Sukthankar, C. Geib, D. V. Pynadath, H. Bui, and R. P. Goldman, editors, *Handbook on Plan, Activity, and Intent Recognition*, pages 149–174. Elsevier, 2014.

J. C. Pinheiro and D. M. Bates. *Mixed-effects models in S and S-PLUS*. Springer, 2000.

E. J. G. Pitman. Sufficient statistics and intrinsic accuracy. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 32, pages 567–579. Cambridge Univ Press, 1936.

J. Pitman. Exchangeable and partially exchangeable random partitions. *Probability Theory and Related Fields*, 102(2):145–158, 1995.

J. Pitman. Poisson–Dirichlet and GEM invariant distributions for split-and-merge transformations of an interval partition. *Combinatorics, Probability and Computing*, 11(05):501–514, 2002. ISSN 1469-2163.

L. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

L. Rabiner and B.-H. Juang. An introduction to hidden markov models. *ASSP Magazine, IEEE*, 3(1):4–16, 1986.

W. M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850, 1971.

C. E. Rasmussen. The infinite gaussian mixture model. In *NIPS*, volume 12, pages 554–560, 1999.

C. E. Rasmussen. Gaussian processes for machine learning. 2006.

S. W. Raudenbush. Hierarchical linear models, 1992.

S. W. Raudenbush and A. S. Bryk. *Hierarchical linear models: Applications and data analysis methods*, volume 1. Sage, 2002.

L. Ren, D. Dunson, and L. Carin. The dynamic hierarchical Dirichlet process. In *Proc. of Int. Conf. on Machine Learning (ICML)*, pages 824–831. ACM, 2008.

L. Rigouste, O. Cappé, and F. Yvon. Inference and evaluation of the multinomial mixture model for text clustering. *Information processing & management*, 43(5): 1260–1280, 2007.

A. Rodriguez, D. Dunson, and A. Gelfand. The nested Dirichlet process. *Journal of the American Statistical Association*, 103(483):1131–1154, 2008.

M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pages 487–494. AUAI Press, 2004.

C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics (TOG)*, 23(3): 309–314, 2004.

A. Roychowdhury, K. Jiang, and B. Kulis. Small-variance asymptotics for hidden markov models. In *Advances in Neural Information Processing Systems*, pages 2103–2111, 2013.

R. Schlaifer and H. Raiffa. Applied statistical decision theory. 1961.

F. Scholz. Maximum likelihood estimation. *Encyclopedia of Statistical Sciences*, 1985.

J. Sethuraman. A constructive definition of Dirichlet priors. *Statistica Sinica*, 4(2): 639–650, 1994.

J. Shawe-Taylor and N. Cristianini. *Kernel methods for pattern analysis*. Cambridge Univ Pr, 2004.

J. Shlens. Notes on kullback-leibler divergence and likelihood theory. *System Neurobiology Laboratory, Salk Institute for Biological Studies, California*, 2007.

T. A. Snijders. *Multilevel analysis*. Springer, 2011.

R. Srinivasan. *Importance sampling: Applications in communications and detection*. Springer Science & Business Media, 2002.

T. Starner and A. Pentland. Real-time american sign language recognition from video using hidden markov models. In *Motion-Based Recognition*, pages 227–243. Springer, 1997.

T. S. Stepleton, Z. Ghahramani, G. J. Gordon, and T. S. Lee. The block diagonal infinite hidden markov model. In *International Conference on Artificial Intelligence and Statistics*, pages 552–559, 2009.

E. B. Sudderth. *Graphical models for visual object recognition and tracking*. PhD thesis, Citeseer, 2006.

E. B. Sudderth, A. Torralba, W. T. Freeman, and A. S. Willsky. Describing visual scenes using transformed objects and parts. *International Journal of Computer Vision*, 77(1-3):291–330, 2008.

B. G. Tabachnick, L. S. Fidell, et al. Using multivariate statistics. 2001.

Y. Teh. A hierarchical bayesian language model based on pitman-yor processes. In *Proc. of Int. Conf. on Computational Linguistics (ACL)*, pages 985–992. Association for Computational Linguistics, 2006.

Y. Teh and M. Jordan. Hierarchical Bayesian nonparametric models with applications. In N. Hjort, C. Holmes, P. Müller, and S. Walker, editors, *Bayesian Nonparametrics: Principles and Practice*, page 158. Cambridge University Press, 2009.

Y. Teh, M. Jordan, M. Beal, and D. Blei. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2006.

Y. Teh, D. Gorur, and Z. Ghahramani. Stick-breaking construction for the Indian buffet process. In *Proc. of the Int. Conf. on Artificial Intelligence and Statistics (AISTAT)*, volume 11, 2007.

Y. W. Teh, C. Blundell, and L. T. Elliott. Modelling genetic variations with fragmentation-coagulation processes. *Advances in neural information processing systems*, 2011.

F. Topsoe. Jenson-shannon divergence and norm-based measures of discrimination and variation. *Preprint*, 2003.

V. F. Turchin. On the computation of multidimensional integrals by the monte-carlo method. *Theory of Probability & Its Applications*, 16(4):720–724, 1971.

L. Van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(2579-2605):85, 2008.

J. Van Gael, Y. Saatci, Y. Teh, and Z. Ghahramani. Beam sampling for the infinite hidden Markov model. In *Proc. of Int. Conf. on Machine learning (ICML)*, pages 1088–1095. ACM, 2008.

A. Vlachos, Z. Ghahramani, and A. Korhonen. Dirichlet process mixture models for verb clustering. In *Proceedings of the ICML workshop on Prior Knowledge for Text and Language*, 2008.

M. J. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1-2): 1–305, 2008.

H. Wallach, I. Murray, R. Salakhutdinov, and D. Mimno. Evaluation methods for topic models. In *Procs. of Int. Conference on Machine Learning (ICML)*, pages 1105–1112. ACM, 2009.

X. Wang and E. Grimson. Spatial latent Dirichlet allocation. In *Advanes in Neural Information Processing Systems (NIPS)*, 2007.

X. Wang and A. McCallum. Topics over time: a non-markov continuous-time model of topical trends. In *Proc. ACM SIGKDD Int. Conf. on Knowledge Ddiscovery and Data Mining*, pages 424–433, New York, NY, USA, 2006. ACM. ISBN 1-59593-339-5. doi: http://doi.acm.org/10.1145/1150402.1150450.

X. Wang, X. Ma, and W. Grimson. Unsupervised activity perception in crowded and complicated scenes using hierarchical Bayesian models. *IEEE Trans on Pattern Analysis and Machine Intelligence (PAMI)*, pages 539–555, 2008. ISSN 0162-8828.

C. Wartena and R. Brussee. Topic detection by clustering keywords. In *Database and Expert Systems Application, 2008. DEXA'08. 19th International Workshop on*, pages 54–58. IEEE, 2008.

F. Wood, S. Goldwater, and M. J. Black. A non-parametric bayesian approach to spike sorting. In *Engineering in Medicine and Biology Society, 2006. EMBS'06. 28th Annual International Conference of the IEEE*, pages 1165–1168. IEEE, 2006.

C. Wooters and M. Huijbregts. The icsi rt07s speaker diarization system. In *Multimodal Technologies for Perception of Humans*, pages 509–519. Springer, 2008.

C. J. Wu. On the convergence properties of the em algorithm. *The Annals of statistics*, pages 95–103, 1983.

D. Wulsin, S. Jensen, and B. Litt. A hierarchical dirichlet process model with multiple levels of clustering for human eeg seizure modeling. *Proceedings of the 29th International Conference on Machine learning*, 2012.

L. Xie, S. Chang, A. Divakaran, and H. Sun. Learning hierarhical hidden markov models for unsupervised structure discovery from video. Technical report, Technical report, Columbia University, New York, 2002.

P. Xie and E. P. Xing. Integrating document clustering and topic modeling. In *Proceedings of the International Conference on Uncertainty in Artificial Intelligence*, 2013.

W.-F. Xuan, B.-Q. Liu, C.-J. Sun, D.-Y. Zhang, and X.-L. Wang. Finding main topics in blogosphere using document clustering based on topic model. 4:1902–1908, 2011.

J. Zhang, Y. Song, C. Zhang, and S. Liu. Evolutionary hierarchical dirichlet processes for multiple correlated time-varying corpora. *Proceedings of the 29th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1079–1088, 2010.

W. Zhang, D. Zhao, and X. Wang. Agglomerative clustering via maximum incremental path integral. *Pattern Recognition*, 46(11):3056–3065, 2013.

X. Zhou, J. Zhang, and B. Kulis. Power-law graph cuts. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, pages 1144–1152, 2015.

## TIEN VU NGUYEN

**From:** Hideo Saito <saito@hvrl.ics.keio.ac.jp>
**Sent:** Friday, 17 April 2015 10:01 AM
**To:** TIEN VU NGUYEN
**Cc:** Linda O'Gorman
**Subject:** Re: Request IAPR Copyright Permission for Thesis Publication

Dear Tien Vu Nguyen,

I was the publication chair of ICPR2012, who is also taking care of the copyright of the proceedings of ICPR2012 on behalf of the organizing committee of ICPR2012. The copyright of ICPR2012 proceedings is owned by the organizing committee of ICPR2012.

I am happy to approve that you will use your paper that you have published at ICPR 2012

Multi-modal Abnormality Detection in Video with Unknown DataSegmentation

for your thesis publication.

Best Regards,

Hideo Saito, Ph.D.
Professor
Dept. of Information and Computer Science, Keio University
3-14-1 Hiyoshi Kohoku-ku Yokohama, 223-8522, Japan
Tel: +81-45-566-1753, Fax: +81-45-566-1747
http://www.hvrl.ics.keio.ac.jp/

>
> ---------- Forwarded message ----------
> From: TIEN VU NGUYEN <tvnguye@deakin.edu.au>
> Date: Tue, Apr 14, 2015 at 9:51 PM
> Subject: Request IAPR Copyright Permission for Thesis Publication
> To: "logorman@alumni.duke.edu" <logorman@alumni.duke.edu>
>
>
> Dear Linda J. O'Gorman
> My name is Tien Vu Nguyen, a PhD student at Deakin University, Australia.
>
> I would like to use my paper for thesis publication that I have
> published at ICPR 2012
>
> http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6460383&tag=1
>
> I want to obtain the copyright license but I am unable to do from the
> website. The website says gplease contact the publisher of this
> content directly c h as the attached file.
>
> Can you please help me with the copyright license?
>
> Regards,

**Title:** Interactive browsing system for anomaly video surveillance

**Conference Proceedings:** Intelligent Sensors, Sensor Networks and Information Processing, 2013 IEEE Eighth International Conference on

**Author:** Tien-Vu Nguyen; Phung, D.; Gupta, S.; Venkatesh, S.

**Publisher:** IEEE

**Date:** 2-5 April 2013



## Thesis / Dissertation Reuse

**The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:**

*Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:*

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

*Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:*

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK CLOSE WINDOW

# SPRINGER LICENSE
# TERMS AND CONDITIONS

Oct 22, 2014

This is a License Agreement between Vu Nguyen ("You") and Springer ("Springer") provided by Copyright Clearance Center ("CCC"). The license consists of your order details, the terms and conditions provided by Springer, and the payment terms and conditions.

**All payments must be made in full to CCC. For payment instructions, please see information listed at the bottom of this form.**

| | |
|---|---|
| License Number | 3494470854117 |
| License date | Oct 22, 2014 |
| Licensed content publisher | Springer |
| Licensed content publication | Springer eBook |
| Licensed content title | Topic Model Kernel: An Empirical Study towards Probabilistically Reduced Features for Classification |
| Licensed content author | Tien-Vu Nguyen |
| Licensed content date | Jan 1, 2013 |
| Type of Use | Thesis/Dissertation |
| Portion | Full text |
| Number of copies | 1 |
| Author of this Springer article | Yes and you are the sole author of the new work |
| Order reference number | None |
| Title of your thesis / dissertation | Bayesian Nonparametric Multilevel Modelling and Applications |
| Expected completion date | Apr 2015 |
| Estimated size(pages) | 200 |
| Total | 0.00 USD |
| Terms and Conditions | |

## Introduction

The publisher for this copyrighted material is Springer Science + Business Media. By clicking "accept" in connection with completing this licensing transaction, you agree that the following terms and conditions apply to this transaction (along with the Billing and Payment terms and conditions established by Copyright Clearance Center, Inc. ("CCC"), at the time that you opened your Rightslink account and that are available at any time at http://myaccount.copyright.com).

## Limited License

With reference to your request to reprint in your thesis material on which Springer Science and Business Media control the copyright, permission is granted, free of charge, for the use indicated in your enquiry.

Licenses are for one-time use only with a maximum distribution equal to the number that you identified in the licensing process.

Springer

# PERMISSION LETTER

**April 15, 2015**

**Springer reference**

*Annals of Data Science*
April 2015
Date: 07 Apr 2015
**Bayesian Nonparametric Approaches to Abnormality Detection in Video Surveillance**
Vu Nguyen, Dinh Phung, Duc-Son Pham, Svetha Venkatesh

DOI: 10.1007/s40745-015-0030-3
Print ISSN: 2198-5804
Online ISSN: 2198-5812
Journal no. 40745

**Material to be reused:** Complete Article

**Your project**

**Requestor:** Tien Vu Nguyen
E-mail: tvnguye@deakin.edu.au
**University:** Deakin University, Australia
**Purpose:** Dissertation/Thesis

With reference to your request to reuse material in which Springer Science+Business Media controls the copyright, our permission is granted free of charge under the following conditions:

**Springer material**

- represents original material which does not carry references to other sources (if material in question refers with a credit to another source, authorization from that source is required as well);
- requires full credit (Springer and the original publisher, book/journal title, chapter/article title, volume, year of publication, page, name(s) of author(s), original copyright notice) to the publication in which the material was originally published by adding: "With permission of Springer Science+Business Media";
- may not be altered in any manner. Abbreviations, additions, deletions and/or any other alterations shall be made only with prior written authorization of the author and/or Springer Science+Business Media;
- Springer does not supply original artwork or content.

**This permission**

- is non-exclusive;
- is valid for one-time use only for the purpose of defending your thesis and with a maximum of 100 extra copies in paper. If the thesis is going to be published, permission needs to be reobtained.
- includes use in an electronic form, provided it is an author-created version of the thesis on his/her own website and his/her university's repository, including UMI (according to the definition on the Sherpa website: http://www.sherpa.ac.uk/romeo/);
- is subject to courtesy information to the co-author or corresponding author;
- is personal to you and may not be sublicensed, assigned, or transferred by you to any other person without Springer's written permission;
- is only valid if no personal rights, trademarks, or competitive products are infringed.

This license is valid only when the conditions noted above are met.

# SPRINGER LICENSE
# TERMS AND CONDITIONS

Apr 20, 2015

This is a License Agreement between Vu Nguyen ("You") and Springer ("Springer") provided by Copyright Clearance Center ("CCC"). The license consists of your order details, the terms and conditions provided by Springer, and the payment terms and conditions.

**All payments must be made in full to CCC. For payment instructions, please see information listed at the bottom of this form.**

| | |
|---|---|
| License Number | 3613040102698 |
| License date | Apr 20, 2015 |
| Licensed content publisher | Springer |
| Licensed content publication | Springer eBook |
| Licensed content title | A Bayesian Nonparametric Approach to Multilevel Regression |
| Licensed content author | Vu Nguyen |
| Licensed content date | Jan 1, 2015 |
| Type of Use | Thesis/Dissertation |
| Portion | Full text |
| Number of copies | 3 |
| Author of this Springer article | Yes and you are the sole author of the new work |
| Order reference number | None |
| Title of your thesis / dissertation | Bayesian Nonparametric Multilevel Modelling and Applications |
| Expected completion date | Apr 2015 |
| Estimated size(pages) | 265 |
| Total | 0.00 USD |
| Terms and Conditions | |

## Introduction

The publisher for this copyrighted material is Springer Science + Business Media. By clicking "accept" in connection with completing this licensing transaction, you agree that the following terms and conditions apply to this transaction (along with the Billing and Payment terms and conditions established by Copyright Clearance Center, Inc. ("CCC"), at the time that you opened your Rightslink account and that are available at any time at http://myaccount.copyright.com).

## Limited License

With reference to your request to reprint in your thesis material on which Springer Science and Business Media control the copyright, permission is granted, free of charge, for the use indicated in your enquiry.

Licenses are for one-time use only with a maximum distribution equal to the number that you identified in the licensing process.