Doctor of Philosophy

Diversified Probabilistic Graphical Models

by

Maoying Qiao

supervised

by

Prof. Dacheng Tao

the Centre for Quantum Computation and Intelligent Systems (QCIS)

the Faculty of Engineering and Information Technology (FEIT)

the University of Technology Sydney (UTS)

July, 2016

CERTIFICATE OF AUTHORSHIP/ORIGINALITY

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree except as fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Signature of Candidate

ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere gratitude to my supervisor Prof. Dacheng Tao for his continuous support of my Ph.D study. Thanks for his consistent patience and motivation, for his encouraging attitude and expert knowledage for my research. His strict academic attitude and deligent work style have played a model role for me and will continue to benefit me through my life. It is no exaggeration to say without his help steering my research direction, I would not have finished this thesis so smoothly and on time.

Besides my principal supervisor, I would like to thank Dr. Wei Bian. I want to thanks for his never bored discussion with my seemingly endless simple questions from research motivation, model development, algorithm implementation, and paper drafting. With these specific and detailed technical discussion, I have gained practical skills to effectively and efficiently develop and implement my research problems. I also gain deep understandings for my research areas. In addition, these discussion moments between us and with Qiang Li have been an inseparable part of my Ph.D life, and will always be cherished by myself.

I would like to thank Dr. Richard Yi da Xu for his advices on my study of Monte Carlo sampling methods. His help undoubtably has widened my research area. I also want to thank my colleages in Prof. Tao's group for their academic discussion and help. My special thanks goes to Ms. Jemima Moore for helping improve my English skills in both academic publications and presentations.

My sincere thanks also goes to China Scholarship Council (CSC), Univer-

sity of Technology Sydney (UTS) and the centre for Quantum Computation & Information System (QCIS). With the financial support from CSC-UTS joint scholarship, I could concentrate on my research study and did not have to worry about my living. QCIS has provided a excellent and supportive research environment for academic-related activities.

Last but not the least, my gratitude extends to my family who have been patiently encouraging and waiting for the finish of this thesis.

ABSTRACT

Probabilistic graphical models (PGMs) as diverse as Bayesian networks and Markov random fields have provided a fundamental framework to learn and reason using limited and noisy observations. Examples include, but are not limited to, hidden Markov models (HMMs), sequential graphical models, and probabilistic principal component analysis mixture models (PPCA-MM). PGMs have been used in a wide variety of applications such as speech recognition, natural language processing, web searching, and image understanding. However, one potential drawback of using PGMs with traditional learning and inference methods is that the learned parameters or inferred variables are easily trapped within local, clustered optima rather than distributed evenly across the whole space. Taking mixture models as an example, the learned mixing components might overlap. Consequently, the resulting models might show ambiguity when clustering is performed based on these overlapping mixing components. This phenomenon might limit PGM performance.

Although efforts have been made to explore a variety of priors to alleviate this potential drawback and to enhance PGM performance, diverse priors have yet to be fully explored and utilized. Diversity is a concept that encourages counterpart model parameters and variables to repel as much as possible and, in doing so, spread out model components and decrease overlapping. However, how to explicitly encode these priors into a PGM and how to solve the resulting diversified PGMs are two critical problems that must be solved. This thesis proposes a uni-

fied framework to constrain PGMs with diverse priors. Three different PGMs - HMMs, time-varying determinantal point processes (TV-DPPs), and PCA-MMs - are elaborated to demonstrate the proposed diversified PGM framework. For each PGM, three basic constituent framework elements are examined: which part of the traditional PGM is diversified, how to formulate the diversity, and how to solve the diversified version, e.g., parameter learning and inference. In addition, experiments are conducted using various application scenarios to verify the effectiveness of the proposed diversified PGMs.

***Keywords*:** Probabilistic graphical models (PGMs), diversity prior, determinantal point processes (DPPs), hidden Markov models (HMMs), time-varying DPPs, probabilistic principal component analysis (PPCA).

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

1.1 Background

1.1.1 Probabilistic graphical models

Probabilistic graphical models (PGMs) are used to represent conditional dependences between variables and are widely applied in information extraction, speech recognition, gene regulatory network modelling, and many other applications. A PGM can be classified into two main types depending on whether its edges are directed or undirected, namely Bayesian networks and Markov networks. Here we mainly focus on Bayesian networks, a typical example being of the form:

$$p(X, Z, \Theta) \propto f(\Theta)p(Z)p(X|Z, \Theta). \tag{1.1}$$

where Θ symbolizes model parameter variables, and X, Z the observable and latent random variables, respectively. $p(X, Z, \Theta)$ represents the joint distribution over X, Z, Θ, $f(\Theta)$ usually represents model parameter information (either in a deterministic form or in a probabilistic form as a prior distribution), and $p(X|Z, \Theta)$ and $p(Z)$ illustrate conditional dependences that Z has an indepen-

dent prior while X is dependent on Z and model parameter Θ.

Several specific PGM examples include, but are not limited to, hidden Markov models (HMMs), sequential graphical models, and mixture principal component analysis (PCA) models. On the one hand, these PGMs have successfully been applied to a variety of applications such as speech recognition (Rabiner, 1989), sequential modelling (Iwata et al., 2012), and image processing (Li and Tao, 2013). On the other hand, they exhibit one common limitation, namely that they need more than one independent model parameter sample counterpart. We explain this phenomenon and illustrate it with these model types below.

Classical HMMs can be thought of as the simplest dynamic Bayesian network and have three model parameter types: $\Theta = \{\pi, A, B\}$, where π is the initial state distribution parameters (commonly modelled with a multinomial distribution), A transition distributions, and B emission distributions. Since hidden states can generate different valued tokens, corresponding transition distribution and emission distribution counterparts are available. When the transition or emission distribution counterparts are similar, there is ambiguity when inferring hidden states. Efforts have been made to explicitly alleviate this ambiguity caused by emission distributions, but solutions for ambiguity caused by similar transition distributions for different states have yet to be described.

In the mixture model setting (such as Gaussian mixture models (GMM) and latent Dirichlet allocation (LDA) models), the model parameters associated with mixing components are $\Theta = \{\theta_k\}_{k=1}^{K}$ and the joint distribution over them is $p(\Theta) = \prod_k p(\theta_k)$, which denotes independence between mixing component distributions. This situation corresponds to the phenomenon described above, i.e., that mixing independent counterparts inferred from the same observations may produce similar values. In other words, undesirably large overlaps might occur between the mixing components, which should be avoided since more mixing

components may be required to cover the model space, thus increasing the risk of overfitting.

In contrast to the above two situations, in the sequential model setting, independent sampling at each single time stamp (which can be treated as independent counterparts) may lead to computational redundancy. For example, in an online news service system, news is first collected from an enormous number of news sources at each time stamp to form news bases, and then a diverse subset of this news is displayed on news browsers. However, due to the high overlap between neighbouring news bases caused by the variable news updating rates of different news sources, treating diverse subset sampling tasks independently along time stamps will lead to repetitive determinantal point process (DPP) sampling along subsequent news subsets, which is time-consuming and inefficient.

In summary, current popular graphical models have potential limitations that may lead to either ambiguity or redundancy in both models and applications. In this thesis we aim to overcome these limitations with diversity-encouraging priors. The diversity-encoding distribution, i.e., DPP, is reviewed below.

1.1.2 Determinantal point processes (DPPs)

A point process $\mathcal{P}$ on a discrete ground set $\mathcal{S} = \{1, 2, ..., N\}$ is called a determinantal point process (DPP) with a positive semi-definite matrix K indexed by the elements of $\mathcal{S}$ if, when X is a random subset drawn according to $\mathcal{P}$, for every $x \subseteq \mathcal{S}$,

$$\mathcal{P}(X \supseteq x) = \det(K_x), \tag{1.2}$$

for $K \preceq I$. Here, $K_x \equiv [K_{ij}]_{i,j \in x}$ is the restriction of K to the entries indexed by elements of x, and we adopt the convention that $\det(K_\emptyset) = 1$. K is referred to as the marginal kernel. DPP is clearly a probability measure over all $2^{|\mathcal{S}|}$ subsets

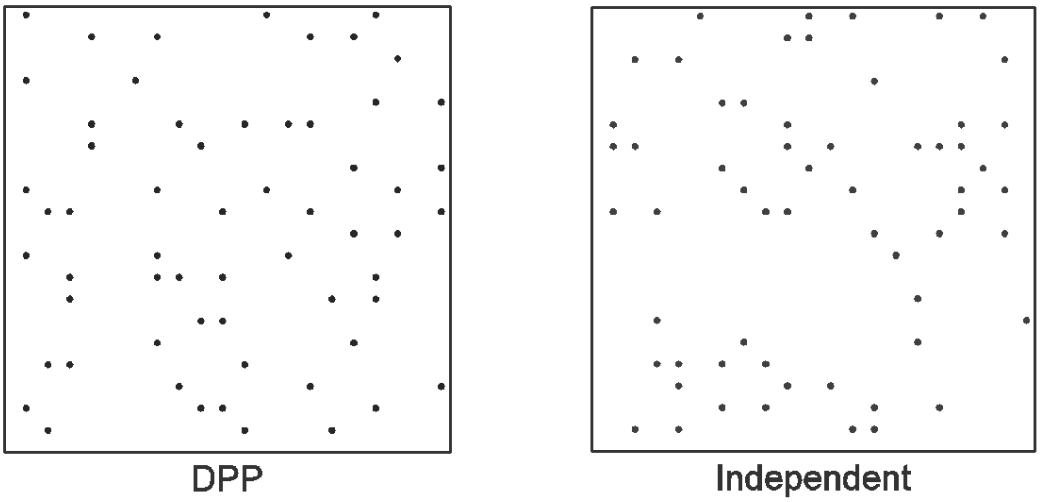


Figure 1.1: Illustration of diverse subset

of the ground set $\mathcal{S}$, with $|\cdot|$ denoting its cardinality.

An example subset sampled by DPP is illustrated in Figure 1.1. It can easily be seen that the independently sampled data points tend to cluster. In constrast, the ones sampled from a DPP are evenly distributed over the entire space.

The most relevant construction of a DPP for the purpose of modelling real data is via L-ensembles (Borodin and Rains, 2005). An L-ensemble defines a DPP through a positive semi-definite matrix L indexed by the elements of ground set $\mathcal{S}$ as

$$\mathcal{P}_L(X = x) = \frac{\det(L_x)}{\det(L + I)}, \tag{1.3}$$

where I is the $N \times N$ identity matrix. This definition has several practical advantages over the one defined with a marginal kernel. First, L-ensemble defines the atomic probabilities over every possible instance of X rather than the marginal probabilities of inclusion as given by marginal kernel K. Second, $\mathcal{P}_L$ has a closed-form normalization given by the identity $\sum_{x \subseteq \mathcal{S}} \det(L_x) = \det(L+I)$. The summation over exponentially counting subsets is equal to a tractable deter-

minant operator that can be exactly computed with polynomial time complexity. Third, unlike marginal kernel K, the eigenvalues of the positive semi-definite kernel L do not need to be upper bounded by one, making it far more useful for real-world data modelling. The relationship between the marginal DPP definition and L-ensemble construction is that a DPP defined with a marginal kernel K has an L-ensemble kernel $L = K(I - K)^{-1}$ (when the inverse exists), and an L-ensemble can be computed from a marginal kernel $K = L(I + L)^{-1}$.

Another important constructing representation of a DPP is based on the fact that a positive semi-definite kernel matrix L can be expressed as a Gram matrix (Kulesza and Taskar, 2010),

$$L_{ij} = q_i \phi_i^T \phi_j q_j, \tag{1.4}$$

where $q_i, q_j \in \mathbb{R}^+$ represent the qualities of elements i, j, and $\phi_i, \phi_j \in \mathbb{R}^n$, the unit length feature vectors, represent the similarity between elements i and j with $\phi_i^T \phi_j \in [-1, 1]$. With this decomposition, one can independently and simultaneously model the quality and diversity of a subset with a unified model. This encourages a DPP to choose subsets with elements of high quality as well as dissimilarity.

From above quality-diversity decomposition formulation, the diversity probability related to the similarity kernel K part can be separated from the quality term and is:

$$P_K(Y) \propto det(K_X) = vol^2(\{\phi(x_i)_{i \in X}\}).$$

It can be seen that the probability defined by a DPP relates to the squared $|Y|$-dimensional volume of the parallelepiped spanned by the selected items in the associated Hilbert space of K. It prefers diverse subsets, because the feature

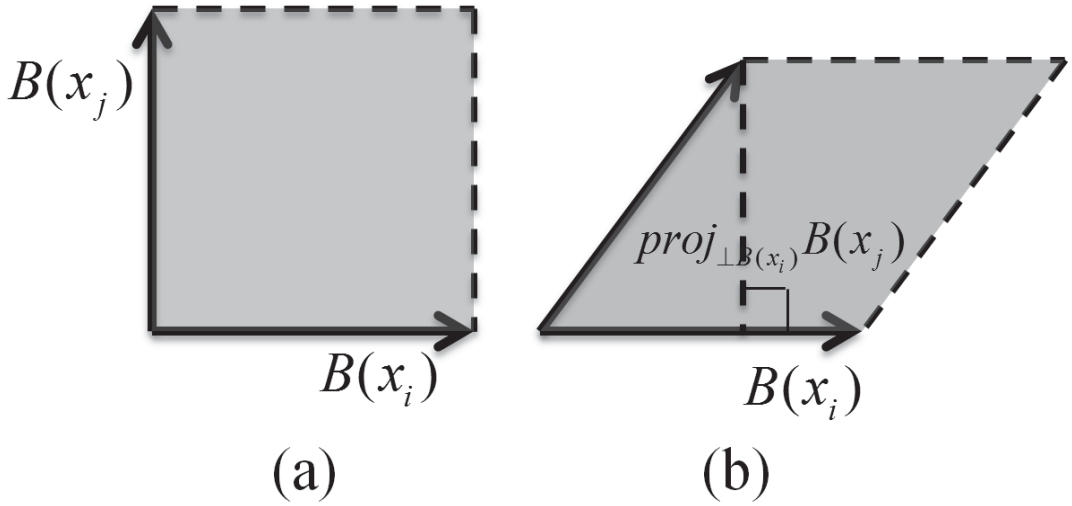


Figure 1.2: Demonstration of diversity captured by DPP.

vectors of these diverse items in the Hilbert space are more orthogonal, and hence span larger volumes, as shown in Figure 1.2.

In summary, DPPs provide a probability measure over every subset configuration on data points. Using the data's similarity matrix and a determinant operator, DPP assigns higher probabilities to those subsets with dissimilar items (Kulesza and Taskar, 2012). This corresponds to a phenomenon that naturally arises in physics (fermions, eigenvalues of random matrices) and in combinatorics (non-intersecting paths, random spanning trees) (Hough et al., 2006) and is used to capture the repulsion of particles (Amer-Yahia et al., 2014).

1.2 Related works

Since the graphical models described in this thesis such as HMMs, the sequential graphical model, and PPCA-MM are well known and described, here we focus on DPP-related issues that are relevant to this work. We classify these issues into three categories: DPP-related developments, diversity-extending models, and diversity-requiring applications.

1.2.1 DPP-related developments

Efficiency improvements

DPPs are convenient statistical tools for modelling diversity. Their popularity is in no small part due to their tractability. This reflects the fact that the DPP inference can be solved with polynomial time complexity and the tractable inference algorithms include, but are not limited to, normalization, sampling (Kulesza and Taskar, 2012), marginalization (Affandi, Fox and Taskar, 2013), generating a diverse set of objects (Kulesza and Taskar, 2011*b*), and maximizing a posterior (MAP, (Gillenwater et al., 2012*b*)).

The exact DPP inference/sampling algorithms are developed based on eigen-decomposition of DPP's kernel matrix. Therefore, the time complexity of DPP sampling is $\mathcal{O}(N^3)$ (Kulesza and Taskar, 2012), where N is the total number of possible items in a ground dataset. When N is too large, this time complexity significantly disadvantages DPP when considering real-time processing, and different approaches have been proposed to address this limitation for large-scale applications. One notable method is based on the low-rank kernel matrix, i.e., rank $D \ll N$. A dual representation is introduced with the Gram matrix (Kulesza and Taskar, 2012), and the time complexity is reduced to $\mathcal{O}(ND + D^3)$. Another example develops a fast approximation of eigen-decomposition of the kernel matrix (Wang et al., 2014), which is the most time-consuming DPP sampling operation.This approach employs the matrix ridge approximation (MRA) to speed up the eigen-decomposition step, and its time complexity is $\mathcal{O}(ND^2) + T_{multiply}(N^2D)$ with $T_{multiply}$ denoting the time complexity of matrix multiplication. This shows that in certain circumstances the proposed MRA-DPP is far more exact than the one approximated by the Nyström method (Affandi, Fox, Adams and Taskar, 2013). More recently, a fast DPP

sampling scheme based on Markov chain Monte Carlo (MCMC) techniques has also been proposed (Kang, 2013), with a ϵ-mixing time of $\mathcal{O}(N \log(N/\epsilon))$.

DPP variants

Several DPP variants have been developed that adapt the basic DPP from simple application scenarios to various complex ones such as k-DPP, structured DPP, Markov DPP, and sequential DPP.

k-DPP (Kulesza and Taskar, 2011*a*) is DPP modelling over subsets with fixed cardinality k, which is useful in many situations, such as diverse resource allocation with a fixed number of resources and search engines returning diverse retrieval results of fixed number. The only alteration to DPP in k-DPP is the normalization term, which summarizes over only k-sized subsets rather than each one in a ground dataset. Its other inference algorithms are directly derived from the ones used in basic DPP.

A tractable structured DPP (SDPP) (Kulesza and Taskar, 2010) has been derived to handle scenarios in which each data item is a structural element, which are ubiquitous in real-world applications such as chain structures for trajectory and news thread modelling and pictorial structures for human pose modelling. One difficulty of this extension compared to basic DPP is that its constructed ground dataset becomes exponentially large and is not easily handled. They overcome this difficulty by taking advantage of structure factorization and dual representation techniques.

(Affandi, Fox and Taskar, 2013) developed a Markov DPP (MDPP) for the scenarios in which multiple diverse sets of items are sequentially requested in online applications. For example, a good news displaying service should provide every user with news at fixed-length time intervals that is not only diverse but also diverse between these intervals. Markov DPP achieves such tasks By enforcing

diversity over neighbouring variables; its corresponding graphical representation is shown in Figure 1.3. It derives that individual marginal distributions for each time interval and joint marginal distributions for pairwise time intervals, and subsequent conditional distributions following DPP distributions. Such elegant properties lead to efficient inference algorithms.

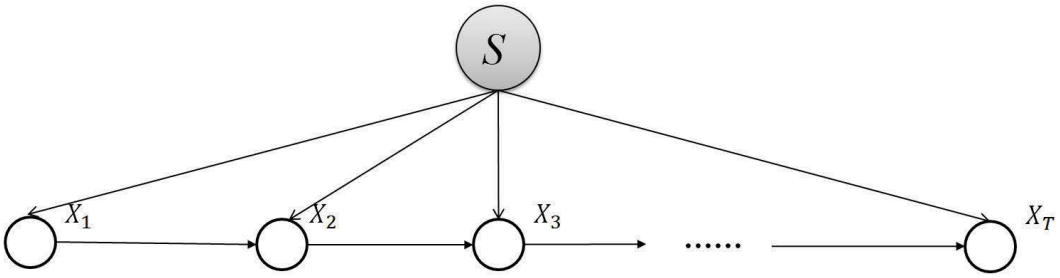


Figure 1.3: Graphical representation for Markov DPP

A sequential DPP was developed for real-time video summaries (Gong et al., 2014); its graphical representation is shown in Figure 1.4. By comparing the graphical representations of sequential DPP and MDPP, it can easily be seen that their biggest difference is the ground datasets at each time stamp, where individual DPPs of MDPP rely on a global ground dataset and the ones of sequential DPP depend only on local ground datasets. As a result, the inference algorithms for sequential DPP are likely to be far more efficient than those for MDPP.

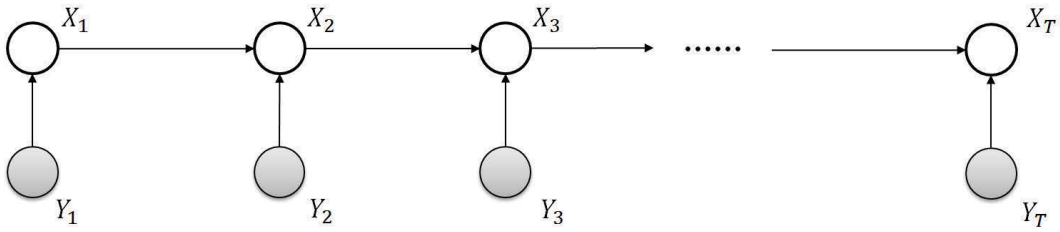


Figure 1.4: Graphical representation for sequential DPP

1.2.2 Diversity-extending models

Several studies have explored DPPs as priors/regularizers to develop news models. These can be divided into three main categories in terms of diversity functionality: diversity in latent variable models, diversity for variable selection models, and diversity for clustering models.

Latent variable models

Zou (Zou and Adams, 2012) introduced DPP priors into latent variable models (LVMs) such as generative latent Dirichlet allocation (LDA) and mixture models. This technique is motivated by the limitations of model parameter priors that the i.i.d. samples are often highly redundant. To reduce this redundancy, the i.i.d. assumption is replaced with a measure-valued DPP to introduce joint diversity over the parameter samples. When this strategy is imposed over topic parameters in the LDA context, diverse topic representations are encouraged that were well distributed across the topic space. The same semantic diversity is also achieved in mixture models. Most of our work is inspired by this strategy.

Snoek and Adams (Snoek and Adams, 2013) introduced diverse priors over sequential latent variables for neural spiking data, in which complex inhibitory and competitive interactions between neurons naturally exist. However, most common neural point processes such as Poisson processes and Gamma renewal processes cannot handle these interactions and correlations. In contrast, a diverse prior over latent diverse neuron embedding can effectively capture such inhibitory relationships.

Recently, Xie et al. (Xie et al., 2016) investigated diversified LVMs in the Bayesian learning paradigm rather than in a frequentist-style regularization framework as described above. Instead of directly employing a DPP, they proposed a diversity-promoting mutual angular distribution as a diverse prior

distribution. They also developed two efficient approximation posterior inference algorithms based on variational inference and MCMC sampling techniques for model inference.

Variable selection models

Some works directly construct a DPP via a correlation matrix of variables to build diverse relationships between them. For example, (Kojima and Komaki, 2014) and (Rocková and George, n.d.) exploited DPPs for variable selection in a linear regression scenario based on this strategy. The motivation behind this approach is to avoid a collinear predictor subset that results in an ill-conditioned predictor matrix. Instead, a diverse subset of predictors is selected to overcome this problem. Since the predictor correlation matrix is a type of similarity metric for predictors, it can be used directly as the similarity kernel in a DPP for variable selection. This strategy of encouraging diversity among spike-and-slab variables represents a far more straightforward and meaningful use of DPPs.

The same strategy was employed by (Xiong et al., 2016) but for video repair. In this case, latent inducing points compose a small video frame subset to represent all observed undamaged frames. This subset needs to be diverse and resistant to context-awareness and be artefact free to benefit video repairing task. Again, the inducing point correlation matrix provides a direct similarity metric that can be used as the kernel matrix for a DPP prior over these inducing points. Experimental results confirmed the effectiveness of this diversity-encouraging scheme for video repair.

Similarly, Batmanghelich et al. (Batmanghelich et al., 2014) developed a diversifying Bayesian sparsity regression model based on the same diversity strategy, i.e., feature covariance, but also proposed a novel strategy to integrate this diversity into the regression model. Unlike previous works that placed the

diversity-encouraging DPP as a prior over model parameters, this work imposed the DPP over model parameter posterior distributions during variational inference. Two real-world cases demonstrated the effectiveness of this strategy.

Clustering models

Clustering tasks aim to divide a ground dataset into groups in which data items are semantically similar and data items in different groups are dissimilar. Subsets with elements picked from different groups intuitively match subsets sampled from a DPP that repel each other. From this perspective, DPP provides heuristics for initial centroids of clustering tasks as in (Kang, 2013). This method has two advantages. First, it can automatically decide the number of clusters and no pre-defined number of clusters needs to be provided, which is a complex but important issue in clustering. Second, since these initial centroids from a DPP are well distributed across the ground dataset, it avoids the situation that two centroids are so close that the clusters expanded by them need to be merged.

Rather than directly applying a diverse DPP subset as initial centroids, Amar and Zoubin (Shah and Ghahramani, 2013) developed a novel determinantal clustering process (DCP) by reversing the diversity metric provided by a DPP into a similarity metric over all possible subsets of the ground dataset. By sampling such a partition of the ground dataset in a semi-supervised setting, DCP achieved a nonparametric Bayesian approach for clustering without specifying the cluster number.

1.2.3 Diversity-requiring applications

To provide insights into situations requiring diversity, here we provide several examples that frequently arise in real-world applications, from diverse summarization to diverse clustering seeds.

Summarization

Video summarization (Gong et al., 2014) is an efficient way to handle large amounts of daily video captures. The aim is to provide video searchers and browsers with succinct but informative frames to represent original content-redundant video clips. Diversity is obviously an important metric to measure the goodness of a video summary, and a diverse frame set can clearly cover far more information than repetitive key frame sets. A similar situation arises in document summarization (Lin and Bilmes, 2012)(Gillenwater et al., 2012*a*); automatically selecting sentence subsets to summarize core ideas provides readers with a quick way to extract useful information from large numbers of complex documents. Again, diversity is an essential characteristic of the sentence summarizing subsets, since it encourages the subsets to cover more details than just repetitive key topic sentences and provides more comprehensive perspectives for the reader.

Recommendation

Diversity also plays a key role in recommendation as well. For example, in online retail (McSherry, 2002)(Gillenwater et al., 2014), many factors dictate which product should be recommended to customers such as the rating from all customers, transaction records. However, the products selected by such factors may fall into fixed categories that are favoured by each customer. Therefore, diversity is a good strategy to explore a wider range of customer interests and ultimately to increase revenue. Another important example influencing every day's life is news recommendation (Abel et al., 2013). Many online services such as Google news try to provide real-time but personalized news package to each user. A satisfying news recommendation package should contain news that is not only highly related to the person's interests but that also exhibits diversity

to provide a good coverage of different aspects of the news events.

Image processing

Diversity has also been shown to be helpful for different image processing tasks such as image segmentation and human pose estimation. Image segmentation (Kim et al., 2011) (e.g., separating a given outdoor image into two parts - 'sky' and 'earth') is an essential step in most image-based applications such as image annotation and objects recognition. This task can be reformulated as a clustering problem but with the additional clue that the image segments are spatially non-overlapping. In these circumstances, a diverse image pixel subset is a good choice for the initial cluster centres. Also, pose estimation (Kulesza and Taskar, 2010) is an essential step in the automatically analysis of group activities in a human dominant images, where each person is usually represented with pictorial structures. Since people tend to occupy disjointed locations in space, a group of spatially diverse pictorial structures is preferred here.

Resource allocation (Guestrin et al., 2005)(Hartline et al., 2008)

Diversity helps to save resources for resource allocation tasks by picking proper locations. One example is sensor placement tasks for Gaussian processes. Given a limited budget, it is important to carefully choose the best locations to install sensors such that the measurements made by these sensors are as informative as possible about the entire space. Another example is in social network marketing. An efficient online marketing strategy is referred to as 'influence-and-exploit'. Specifically, marketing strategies should be placed on the node subset from one network that not only contains influential nodes with high-degree connections but also diversely spread out on the whole network.

Miscellanies

Two more miscellaneous but important scenarios requiring diversity are information retrieval (Kulesza and Taskar, 2011*a*) and clustering (Kang, 2013). In an information retrieval task, every search is devoted to selecting items that are most relevant to users' queries from a huge image or text dataset via an efficient ranking system. Most ranking strategies emphasize items of high quality and prioritize them in the rankings but ignore their relationships/connections. As a result, similar content may be listed at the top of retrieval results, which is suboptimal for users. In the worse situation in which query keywords are ambiguous, the retrieval subset containing similar items may be completely irrelevant to users. Therefore, in addition to a relevance criterion, diversity should be another important factor in ranking strategies for information retrieval. The diversity criterion plays a similar role in selecting initial clustering centres in a clustering task, and it is well known that the initial sample subset chosen as clustering centres determines the effectiveness of convergence of clustering algorithm, e.g., in k-means. Therefore, a sample subset with qualified individuals and a diversity relationship between them can represent the entire data population and is likely to improve clustering accuracy. Similarly, (Reichart and Korhonen, 2013) proposed a unified framework for verb clustering that made use of DPP to sample high quality and diverse verb subsets that flowed into hierarchical clustering as input seeds.

1.3 Motivations and research significance

1.3.1 Motivations and contributions

Inspired by the promising results achieved by current diversity-encoded models, in this thesis we integrate diverse priors into fundamental PGMs to alleviate their ambiguity/redundancy problems exhibited by them. However, ambiguity and redundancy problems. However, ambiguity and redundancy mean different things under different circumstances. For instance, in HMMs, ambiguity in hidden states may occur when state transition probabilities are similar, especially when other HMM parameters are fixed. In a sequential setting, repetitive overlapping neighbouring subset sampling will lead to computational redundancy for recommendation tasks. Another example is in mixture models, where inferred mixing components from observations may overlap, which may lead to label ambiguity in terms of clustering tasks. Therefore, this thesis is devoted to introducing different explicit diversities to these different circumstances to alleviate the disadvantages of ambiguity and redundancy. The main contributions are as follows:

- We formulate a diverse prior with probability measure-valued DPP and explicitly integrate it over state transition probabilities of HMMs to improve HMM performance. An EM framework is derived to solve the diversified HMMs. Experiments conducted over sequential labelling tasks, e.g., PoS tagging and OCR, confirm the effectiveness of the proposed diversified extension of HMMs.

- We design a new graphical model for sequential diverse subset sampling, referred to as time-varying determinantal point processes (TV-DPP), to meet the requirements of several different application scenarios such as recom-

mending sequentially diverse news subsets for news browsers and discovering diverse networks evolving in employment networks. This setting takes advantage of the huge overlap between neighbouring ground datasets and straightforwardly avoids computational redundancy along the timeline.

- We design a matrix-valued DPP diverse prior and integrate it into a probabilistic PCA mixture model (PPCA-MM) to encode desirable repulsion between mixing components. The learned/inferred mixing components are thus expected to be well distributed across the whole model space. This reduces the risk of overlapping between mixing components to the lowest point. An approximation algorithm based on Jensen's inequality is derived for learning and inference of the proposed diversified PPCA-MMs. Empirical results verify the effectiveness of the proposed diversity scheme.

1.3.2 Research significance

This thesis makes three main research contributions:

- We extend PGMs by integrating naturally existing and application-desirable diverse priors. It significantly enriches the PGM research area and also extends model application to intrinsically diversity-desirable applications.

- Three specific PGMs are diversified: hidden Markov models (HMMs), sequential models, and probabilistic PCA (PPCA), which play fundamental roles especially in sequential data processing and dimensionality reduction. Therefore, the diversified versions of these models will have a fundamental impact on areas relevant to these basic models.

- A single guiding framework for diversifying PGMs is summarized from the above three models that is helpful for developing future diversified PGMs.

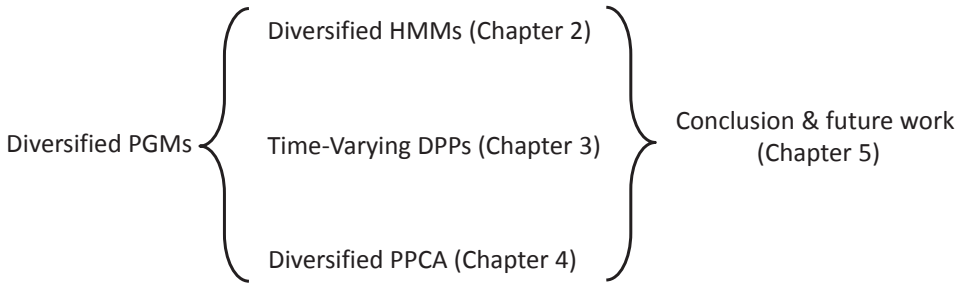


Figure 1.5: Thesis structure

This framework has three parts: (i) to identify which part of a PGM is reasonable to be diversified based on either modelling capacity or real-world application demands; (ii) how to design diverse priors and encode them into the original model formulations; and (iii) how to derive solving algorithms for the diversified models.

- The instantiated diversified models presented in this thesis are directly and successfully applied to various applications. They also establish novel frameworks for practical applications, e.g., for news recommendation. They are undoubtedly of important practical significance.

1.4 Thesis Structure

This thesis introduces diversity into traditional PGMs and illustrates it with three widely used PGMs. The remainder of this thesis elaborates these three examples. The structure of this thesis is shown in Figure 1.5, and it is organized as follows. Chapter 2 introduces diversifies hidden Markov models (HMMs) for sequential learning. Chapter 3 develops a time-varying determinantal point processes model for sequential diverse subset sampling. Chapter 4 accomplishes diversified PPCA-MM. Finally, Chapter 5 concludes this thesis and proposes

some future work.

Chapter 2

Diversified Hidden Markov Models for Sequential Labelling

Labelling of sequential data is a prevalent meta-problem for a wide range of real world applications. While the first-order hidden Markov models (HMMs), an important member of PGMs, provides a fundamental approach for unsupervised sequential labelling. The basic model does not show satisfying performance when it is directly applied to real world problems such as part-of-speech tagging (PoS tagging) and optical character recognition (OCR). Aiming at improving performance, important extensions of HMMs have been proposed in the literatures. One of the common key features in these extensions is the incorporation of proper prior information. In this chapter, we propose a new extension of HMMs, termed diversified hidden Markov models (dHMM), which utilizes a diversity-encouraging prior over the state-transition probabilities and thus facilitates more dynamic sequential labelings. Specifically, the diversity is modelled by a continuous determinantal point process (DPP), which can be applied to both unsupervised and supervised scenarios. Learning and inference algorithms for dHMM are derived. Empirical evaluations on benchmark datasets for unsu-

pervised PoS tagging and supervised OCR confirm the effectiveness of dHMM, with competitive performance to state-of-the-art methods.

2.1 Introduction

Sequential labeling is an important meta-problem in many real world applications, including natural language processing (NLP) tasks (Murveit and Moore, 1990) (Morwal et al., 2012), video analysis (Niu and Abdel-Mottaleb, 2005)(Liu and Chen, 2003), protein secondary structure(Krogh et al., 2012) (Asai et al., 1992). It has received considerable attentions in the past years. One of the fundamental models for sequential labelling is HMM, which assumes a âĂŸchainâĂŹ of discrete-valued latent states and each of them depends only on the immediate neighbouring states. Conditioning on this latent chain, the observations are probabilistically independent. Take PoS tagging task from NLP as an example, speech tags (NNS-Noun, plural; MD-Modal; etc) are discrete-valued latent state, while words (directors; are; etc) are observations.

However, the parameter-learning task in the classical HMM implemented with expectation-maximization (EM) algorithm performs unsatisfactorily in unsupervised setting for sequential labeling (Johnson, 2007). A key reason for this drawback is the well-known fact that maximum likelihood estimation (MLE) with mixture parameters has the tendency to converge towards a singular estimate at the boundary of the parameter space (Figueiredo and Jain, 2000)(Bishop et al., 2006), no matter how the observations are actually distributed (e.g. normally distributed or askew distributed). With improperly estimated parameters, the performance of inference of the latent states can be severely unsatisfied. Besides, the identifiability of parameters is another issue for HMM parameter learning with MLE implementation.

Therefore, a penalized MLE with properly chosen prior distribution over parameters is essential for the practical applications of HMM. For example, smoothing penalty (Wang and Schuurmans, 2005) and sparse penalty (Tao et al., 2012) (Li et al., 2014) (Yang et al., 2014) (Long et al., 2014) are two popular priors over either transition distribution parameters or emission distribution parameters for sequential labelling.

Different from the early works, we have explored the usage of diversity prior over a joint distribution of rows of HMM‘s transition matrix, in order to make these transition distributions more distinct when decoding the sequential latent states. In the cases, where the transition probabilities become similar, an HMM model approaches to a static mixture model. We can We can understand this intuition by considering an extreme case where each rows of a transition probability matrix are identical. This leads to the same state transitions regardless of the state we are currently in. Suppose we have a k-state HMM, parameterised by (π, A, B), i.e., the initial probability π, transition probability matrix A and emission probability B, if the rows of A are identical and given by vector a, the joint probability of the hidden states and observations over a sequence of length T can be calculated as:

$$\begin{aligned} P(X, Y|\pi, A, B) &= P(x_1|\pi) \prod_{t=2}^{T} P(x_t|A(x_{t-1}, :)) P(y_t|x_t, B) \\ &= P(x_1|\pi) \prod_{t=2}^{T} P(x_t|a) P(y_t|x_t, B) \end{aligned}$$

It can be seen that the joint probability becomes an independent product of variables at individual time point, and thus the HMM model becomes a static mixture model, i.e., the data become exchangeable. In contrast, a prior that encouraging diversity is able to reserve the dynamics property of HMMs. To the best of our knowledge, this is the first paper to apply diversity prior over

HMM parameters. We do so by incorporating a recently introduced determinantal point processes (DPP) (Kulesza and Taskar, 2012) methodology, which essentially defines a Probability Mass Function (PMF) that assigns higher probability to a diverse subset of data. Inspired by the work of (Zou and Adams, 2012), we propose a diversified HMM (dHMM) by extending the basic HMM with determinantal-driven diversity.

Specifically, our contributes can be summarized in the following:

- We extend the HMM to dHMM by incorporating a diversity-encouraging prior over transition distributions, with which we intend to mitigate the problem of singular estimate in HMM.

- The use of a prior does not change the E-step of the EM algorithm in a fundamental way. However, for the M-step of the estimation of dHMM parameters, we derive a new formulation to incorporate the continuous DPP prior over the transition probabilities.

- We demonstrate the effectiveness of dHMM under both the unsupervised and supervised settings by applying dHMM to benchmark sequential labelling problems, including: part-of-speech tagging (PoS tagging) and optical character recognition (OCR).

The rest of this chapter is organized as follows. Section 2.2 reviews related literatures on progresses of statistical HMMs for sequential labelling. Section 2.3 introduces our proposed model in detail. Firstly, we briefly review basic hidden Markov models (HMMs). Then how diversity-encouraging prior is encoded into HMMs and how an induced maximum a posterior (MAP) objective problem is solved are presented. Both simulated and real-world experiments are conducted in section 2.4. Finally, summary of this chapter is discussed in Section 2.5.

2.2 Related Work

HMM is a fundamental statistic model for modelling sequential dataset and of significant importance in many other fields, from speech recognition (Rabiner, 1989), handwriting recognition (Hu et al., 1992), video analysis (Liu and Chen, 2003), gesture recognition (Tao et al., 2012), gene sequence prediction (Krogh et al., 2012), to optical character recognition (OCR) (Feng and Manmatha, 2006) and part-of-speech tagging (PoS tagging)) (Gael et al., 2009). This section presents related work of HMMs.

We briefly review the development of HMM for sequential labeling. It is well known that the HMM parameters contain three parts: (1) initial state distribution, (2) transition distributions and (3) emission distributions of either discrete or continuous. Various extensions of HMM have been proposed by incorporating proper prior information into either one part or all three parts of these parameters.

Supervised sparse HMM (Tao et al., 2012) was proposed to improve the expressive power for sequential surgical gesture classification and skill evaluation. It assumes that the emission distributions sparsely and linearly constitute elements from dictionary of basic surgical motions, no matter the observations are discrete, Gaussian or factor analysed. Training dataset is needed for dictionary learning for each gesture together with an HMM grammar describing the transitions among different gestures. With learned dictionaries and grammar, the testing motion data is represented and classified.

Supervised large margin continuous density HMM (CD-HMM) for automatic speech recognition was proposed by Sha and Saul in (Sha and Saul, 2006). The real-valued observations (such as acoustic feature vectors) are modelled through Gaussian mixture models. Inspired by support vector machines, margin maximization is applied as training objective function which is defined over a param-

eter space of positive semi-definite matrices. This optimization problem can be solved efficiently with simple gradient-based methods.

Unsupervised learning is a more difficult but important problem, as it eliminates the need for expensive manual annotation. It was demonstrated from the work of (Wang and Schuurmans, 2005) that, smoothing HMM parameters can achieve significant improvements for PoS tagging. Two strategies have been applied: the first one is to smooth the emission distributions by computing observed word similarities. The second one is to specify a stationary distribution for hidden states to constrain the transition distributions.

Unsupervised sparse HMM based on Bayesian framework also has been explored in several literatures. Unlike supervised sparse HMM (Tao et al., 2012) where the emission distributions are learned from sparse representation, unsupervised sparse models add priors on transition distributions. (Bicego et al., 2007) introduces a negative Dirichlet prior on the transition distributions, which strongly encourages sparseness of the model. Then, maximum a posteriori (MAP) probability estimation of HMM parameters is devised under a modified expectation maximization algorithm. Manuele et al. evaluated the proposed technique on a 2D shape classification task. In (Goldwater and Griffiths, 2007), for PoS tagging, rather than performing MAP parameters estimation followed by inferring hidden states, Goldwater and Griffiths directly identify a distribution over latent variables, without ever fixing particular values for model parameters. This is achieved by integrating over all possible values of parameters under a Bayesian approach. The integrating over parameter space permits the usage of appropriate linguistic priors. For example, the symmetric Dirichlet prior may prefer equal, uniform or sparse multinomial distributions according to different settings of its hyper-parameters.

There exist other important extensions of HMM for determining the number

of hidden states which is a key issue in every clustering task. Non-parametric Bayesian method is one popular solution for this problem. For instance, Qi et al. (Qi et al., 2013) applied hierarchical Dirichlet processes prior over transition matrix to model the number evolution of dynamic community structures. Here we fix this parameter from experience and we refer to (Gael et al., 2009) (Teh et al., 2006) for interested readers.

From these previous works, proper prior information (Fang et al., 2015) encoded into HMM leads to visible performance increment. Either smooth prior or sparse prior is somewhat reasonable from a technical view. From an intuitive view, different states should have different transition distributions, otherwise, HMM will finally fall to a 'static' mixture model. To this end, a diversity-encouraging prior is demanded and also required by many real-world sequential applications. In next section, we show how this kind of prior is encoded into HMMs.

2.3 Diversified Hidden Markov Models

In this section, our proposed dHMM and its MAP solution are presented. The graphical model of the proposed dHMM is illustrated in Fig. (2.1). In order to make this paper self-contained, we illustrate all of its steps as well as background knowledge leading to our new work: (1) We first briefly review the basic HMM models. (2) Then, the probability product kernel is introduced as a basic building block for DPP. (3) Our dHMM is subsequently represented. (4) Finally, we detail inference steps for solving the proposed dHMM.

2.3.1 Kernels for probability diversity

We prepare a probability kernel, which allows us to apply DPP on HMM's transition probabilities. In this work, the Probability Product Kernel, which is proposed in (Jebara et al., 2004), defines the kernel function between distributions P_i and P_j of discrete variables, which are parameterized by A_i, A_j respectively:

$$\begin{aligned} K(A_i, A_j; \rho) &=< P(x|A_i)^\rho, P(x|A_j)^\rho > \\ &= \sum_{x \in X} P(x|A_i)^\rho P(x|A_j)^\rho \end{aligned}$$

for $\rho > 0$ and x runs through all possible values of discrete variable X. The kernel is computed by summing up the products between the two distributions in terms of x.

For distributions $P(x|A_i)$ and $P(x|A_j)$, the less 'correlation' between them, the more 'diversity' we can gain through the determinant of the kernels. To remove the scale effects of different probabilistic measurements, the normalized correlation kernel function is applied:

$$\tilde{K}(A_i, A_j; \rho) = \frac{K(A_i, A_j; \rho)}{\sqrt{K(A_i, A_i; \rho)}\sqrt{K(A_j, A_j; \rho)}} \tag{2.1}$$

The final continuous DPP kernel as a building block in our proposed model is: $\tilde{K}_A = [\tilde{K}(A_i, A_j)]_{i,j \in \{1,2,\ldots,d\}}$, where $\tilde{K}_A$ is $d \times d$ matrix, and $A_{i\cdot} \in \mathbb{R}_+^d$ with $\sum_j A_{ij} = 1$ for probability measure.

2.3.2 Log-likelihood function of Hidden Markov Models

Hidden Markov models assume what are being observed are generated by a Markov process with unobserved hidden states. It is especially known for their applications in temporal sequential pattern recognition.

In Fig. (2.1), a hidden Markov model is a k-state Markov chain observed at discrete time points $t = 1, 2, ..., T$. Let $\{A_1, A_2, ..., A_k\}$ be the finite state space. One state A_i can be transfered to all other states $\{A_j, j \in \{1, 2, ...k\}\}$ with probability distributions parameterized by transition matrix $A \equiv [a_{ij}], i, j \in \{1, 2, ..., k\}$. We use $X = \{X_1, X_2, ..., X_T\}$ as state variables, and $X_t = A_i$ means HMM is staying on state A_i at time step t. $P(X_t = A_j | X_{t-1} = A_i)$ denotes the transition probability from A_i to A_j, which is equal to A_{ij}. In unsupervised setting, hidden variables cannot be observed directly, which are represented by hollow circles in Fig. (2.1). In contrast, filled circles denote observations $Y = \{Y_1, Y_2, ..., Y_T\}$. Each chain observation is parameterized by i.i.d emission distribution B given hidden states. For each hidden state, its probability distribution is only dependent on its former state in the first-order HMM. The joint probability distribution over hidden variables and observations is parameterized by $\lambda = (\pi, A, B)$ (Rabiner, 1989). The likelihood is as follows.

$$
\begin{aligned}
P(X_1, ..., X_T, Y_1, ..., Y_T | \lambda) &= P(X_1; \pi) \prod_{t=2}^{T} P(X_t | X_{t-1}; A) \\
&\quad \times \prod_{t=1}^{T} P(Y_t | X_t; B) \\
s.t. \quad & \sum_{i=1}^{k} \pi_i = 1, \pi_i \geq 0, i \in \{1, 2, ..., k\} \\
& \sum_{j=1}^{k} A_{ij} = 1, A_{ij} \geq 0, \quad i, j \in \{1, 2, ..., k\} \\
& B: \textit{ probability measure}
\end{aligned}
$$

The linear constraints are required by discrete probability measure. The last statement above means that parameters B of emission distributions should also satisfy the requirement of probability measure in either continuous or discrete

space, decided by various applications.

Since supervised HMM is a special case of unsupervised HMM with known hidden state for training period and known parameters for test period, here we just demonstrate the solutions for unsupervised case. Three basic problems of HMM are identified in (Rabiner, 1989), namely, adjusting parameters given observations $max_\lambda P(Y|\lambda)$ for unsupervised setting (or $max_\lambda P(Y, X|\lambda)$ for supervised setting), computing log likelihood $logP(Y|\lambda)$ for unsupervised setting and inferring hidden states $max_X P(Y, X|\lambda)$ given both parameters and observations for both unsupervised setting and supervised setting during test period.

For unsupervised learning, these three problems are closely associated with the likelihood, which is computed by marginalizing out the hidden variables from the joint distribution. The log likelihood for one sequential observation is:

$$L(Y;\lambda) = logP(Y|\lambda) = log\sum_X P(X, Y|\lambda) \tag{2.2}$$

With Markov assumption of HMM and Jensen's inequality, the lower bound of the intractable formula in Eq.(2.2) is:

$$\begin{aligned} L(Y;\lambda) \geq & \sum_{X_1} q(X_1) logP(X_1|\pi) \\ & + \sum_{i=1}^{T}\sum_{X_i} q(X_i) logP(Y_i|X_i, B) \\ & + \sum_{i=2}^{T}\sum_{X_{i-1},X_i} q(X_{i-1}, X_i) logP(X_i|X_{i-1}, A) \\ & - \sum_X q(X) logq(X) \end{aligned} \tag{2.3}$$

where $\{q(X_i)\}_{i=1}^T$ and $\{q(X_{i-1}, X_i)\}_{i=2}^T$ are marginally unary and pairwise distributions of hidden variables.

Traditional HMM is solved under EM framework. As noted, it usually produces flatten emission distributions and meaningless transition matrix. In the next subsection, we detail how to encode the diversity-encouraging prior into the transition distributions and how to solve the three basic problems identified by the traditional HMM.

2.3.3 Proposed Diversified HMM

With all the previous concepts in hand, we can now proceed with our proposed diversified HMM (dHMM).

The HMM's transition distributions $P(X_t|X_{t-1})$ obey multinomial distribution parameterized by $\{A_{ij}\}_{i,j=1}^{k}$, where t is the time index and k is the number of hidden states. The corresponding normalized correlation kernel function for rows of A based on Eq.(2.1) is as:

$$\tilde{K}(A_{i\cdot}, A_{j\cdot}) = \frac{\sum_{x=1}^{k}(A_{ix}A_{jx})^{\rho}}{\sqrt{\sum_{x=1}^{k} A_{ix}^{2\rho}}\sqrt{\sum_{x=1}^{k} A_{jx}^{2\rho}}} \tag{2.4}$$

And the corresponding diversity prior of transition parameter matrix of HMM modelled by determinantal point processes (DPP) is:

$$P_{\tilde{K}}^{k}(A) \propto \det(\tilde{K}_A) \tag{2.5}$$

where $\tilde{K}_A$ is $|A| \times |A|$ kernel matrix, $A_i \in \mathbb{R}_+^k$ with $\sum_j A_{ij} = 1$. A is a k-size subset from the $k-1$ simplex. $P_{\tilde{K}}^{k}$ symbols kDPP. For all experiments based on our proposed dHMM, we set $\rho = 0.5$.

The graphical model of our proposed model is illustrated in Fig. (2.1). The bottom chain structure is a standard first-order HMM. Applying conventional symbols of graphical models, hollow circles indicate hidden states, while filled

circles are symbols of observations. Similar to (Zou and Adams, 2012), we draw a double-struck plate to denote the DPP prior placed on the state transition matrix A. Higher the probability of DPP is, the more diverse of the rows of an HMM's transition matrix is.

Unsupervised setting

To model the unsupervised sequential labelling, Maximum A Posterior (MAP) problem needs to be solved since it incorporates diversity-encouraging prior over parameters of rows the transition matrix. The new objective function is formulated as:

$$\begin{aligned} max_{\lambda} \;\; & L(Y;\lambda) + \alpha log|\tilde{K}_A| \\ & s.t. \sum_{i=1}^{k} \pi_i = 1, \pi_i \geq 0, i \in \{1,2,...k\} \\ & \sum_{j=1}^{k} A_{ij} = 1, \;\; A_{ij} \geq 0, \;\; i,j \in \{1,2,...,k\} \\ & B: \; probability \; measure \end{aligned} \tag{2.6}$$

where $\lambda = (\pi, A, B)$ is the parameters of our proposed dHMM and we adopt the same symbols with the traditional HMM. Note that we ignore the normalization constant of the DPP prior distribution, since it is irrelated to measuring the diversity of parameters of rows of transition matrix, as well as estimating parameters of initial distribution and emission distributions. And $\alpha > 0$ is used to balance the weights between measurements of likelihood and diversity-encouraging prior. When $\alpha = 0$, no diversity-encouraging prior will distract the estimation of transition matrix from Maximum Likelihood Estimation (MLE) learning. With α goes up, the weight of diversity-encouraging prior increases, and the diversity-encouraging prior will dominate the estimation of the parameters of transition matrix.

Supervised setting

For modelling supervised sequential labelling, as the hidden states are given during training period, parameters $\lambda = (\pi, A, B)$ can be learned in a count manner. Specifically, $\pi = A_i$ is the ratio between the frequency of state A_i and the total number of sequences. A_{ij} is the proportion of the pairwise states $(X_{t-1} = A_i, X_t = A_j)$ among all pairwise states appearing in the training sequences. B can be learned in a discriminative manner, since the observations are independent given hidden states. Obviously, the learned parameters fit the training dataset best, rather than the test dataset. To generalize the counting-computed parameters of transition matrix A_0 by incorporating diversity-encouraging prior, we construct the new objective function as below:

$$\begin{aligned} max_\lambda \;\; & L(Y, X; \lambda) + \alpha log|\tilde{K}_A| - \alpha_A||A - A_0||_2^2 \\ s.t. & \sum_{i=1}^{k} \pi_i = 1, \pi_i \geq 0, i \in \{1, 2, ...k\} \\ & \sum_{j=1}^{k} A_{ij} = 1, \;\; A_{ij} \geq 0, \;\; i, j \in \{1, 2, ..., k\} \\ & B: \; probability \; measure \end{aligned} \tag{2.7}$$

where A_0 is the trained parameters by $\lambda_0 = max_\lambda L(Y, X; \lambda)$ with $\lambda_0 = (\pi_0, A_0, B_0)$, α_A is used to control how far the final A can drift from A_0.

2.3.4 Solutions

In this subsection, we mainly focus on how to learn parameters from unsupervised setting with objective function Eq.(2.6) and supervised setting with objective function Eq.(2.7).

Unsupervised setting

Traditionally, Expectation-Maximization (EM) framework (Fang et al., 2014) is applied to learn HMM parameters. Here, our procedure is only different with traditional EM in M-step. This is because in a MAP setting, the diversity-encouraging prior term, i.e., $\log(\lambda)$ is irrelated to the hidden states X, which can be taken out of the integration in E-step.

E-step:

Given old parameters $\lambda^{old} = (A^{old}, B^{old}, \pi^{old})$, we apply forward-backward algorithm to do inference for hidden variables.

In the forward pass of HMM chain, it inductively summarizes all information before time step t into marginal distribution over each hidden variable X_t and all past observation variables $\{Y_1, ..., Y_t\}$, namely,

$$\begin{aligned} \alpha(X_t) &\propto P(X_t, Y_1, Y_2, ..., Y_t; \lambda^{old}) \\ \alpha(X_{t+1}) &\propto \left(\sum_{X_t} \alpha(X_t) P(X_{t+1}|X_t) \right) \times P(Y_{t+1}|X_{t+1}) \end{aligned} \tag{2.8}$$

Similarly, in the backward pass, it summaries information over all future observation variables after time step t, $\{Y_{t+1}, ..., Y_T\}$, namely,

$$\begin{aligned} \beta(X_t) &\propto P(Y_{t+1}, ..., Y_T|X_t; \lambda^{old}) \\ \beta(X_{t-1}) &\propto \sum_{X_t} (\beta(X_t) P(X_t|X_{t-1}) P(Y_t|X_t)) \end{aligned} \tag{2.9}$$

The initializations for both forward and backward pass are:

$$\begin{aligned} \alpha(X_1) &\propto P(X_1|\pi^{old}) \times P(Y_1|X_1, B^{old}) \\ \beta(X_T) &= 1 \end{aligned}$$

The conditionally marginal probabilities for hidden variables (required by likelihood in Eq.(2.3)) and likelihood can be computed by combining the forward and backward summarizations. The unary and pairwise hidden states distributions as well as the normalization are formulated as:

$$\begin{aligned} q(X_t) &\propto \alpha(X_t)\beta(X_t) \\ q(X_{t-1}, X_t) &\propto \alpha(X_{t-1})P(X_t|X_{t-1})P(Y_t|X_t)\beta(X_t) \\ P(Y_1, Y_2, ..., Y_T) &= \sum_{X_t} \alpha(X_t)\beta(X_t) \end{aligned}$$

M-step: In this step, dHMM optimizes the below objective function to update the parameters $\lambda^{old} = (\pi^{old}, A^{old}, B^{old})$ given N training sequences.

$$\begin{aligned} & max_{\pi,A,B} L(\pi, A, B|Y, X) \\ =& \sum_{n=1}^{N} \left(\sum_{t=2}^{T_n} \sum_{X_{nt}, X_{n,t-1}} q(X_{nt}, X_{n,t-1}) log P(X_{nt}|X_{n,t-1}, A) \right) \\ +& \sum_{n=1}^{N} \left(\sum_{t=1}^{T_n} \sum_{X_{nt}} q(X_{nt}) log P(Y_{nt}|X_{nt}, B) \right) \\ +& \sum_{n=1}^{N} \left(\sum_{X_{n1}} q(X_{n1}) log P(X_{n1}|\pi) \right) + \alpha log(|\tilde{K}_A|) \\ & s.t. \sum_{i=1}^{k} \pi_i = 1, \pi_i \geq 0, i \in \{1, 2, ..., k\} \\ & \sum_{j=1}^{k} A_{ij} = 1, \quad A_{ij} \geq 0, i, j \in \{1, 2, ..., k\} \\ & B: \textit{ probability measure} \end{aligned}$$

where, X_{nt}, Y_{nt} denote hidden state and observation of the nth sequence at time

step t respectively, T_n denotes the length of the n-th sample sequence. Note that, the last term of Eq.(2.3), which is the entropy of marginal distribution $q(X)$, is irrelated to model parameters λ, is simply ignored in M-step.

As both log function and log det function are concave, we directly apply the Lagrange multipliers method to solve the maximization problem in Eq.(2.6). The Lagrange function is given below:

$$\begin{aligned}\Lambda(\pi, A, B, \beta) =& L(\pi, A, B|Y, X) \\ & -\beta_0(\sum_{i=1}^{k}\pi_i - 1) - \sum_{i=1}^{k}\beta_i(\sum_{j=1}^{k}A_{ij} - 1)\end{aligned}$$

where $\beta_i, i \in \{0, 1, ..., k\}$ is the Lagrange multipliers.

As the gradients for both π and B are the same with traditional HMM, we just list the results. For π,

$\pi_i, i \in \{1, 2, ..., k\}$

$$\pi_i = \frac{\sum_{n=1}^{N} q(X_{n1} = i)}{N}$$

For emission distribution, in our experiments, it obeys either Gaussian distribution or multinomial distribution. For Gaussian distribution, $Y_t|X_t = A_i \sim \mathcal{N}(B.\mu_i, B.\sigma_i), i \in \{1, 2, ..., k\}$, the updating parameters for B are:

$B.\mu_i$

$$\mu_i = \frac{\sum_{n=1}^{N}\sum_{t=1}^{T_n} Y_{nt} q(X_{nt} = A_i)}{\sum_{n=1}^{N}\sum_{t=1}^{T_n} q(X_{nt} = A_i)} \tag{2.10}$$

$B.\sigma_i^2$

$$\sigma_i^2 = \frac{\sum_{n=1}^{N}\sum_{t=1}^{T_n} q(X_{nt} = A_i)(Y_{nt} - \mu_i)^2}{\sum_{n=1}^{N}\sum_{t=1}^{T_n} q(X_{nt} = A_i)} \tag{2.11}$$

For multinomial distribution, $Y_t|X_t = A_i \sim \mathcal{M}ulti(B.b_i), i \in \{1, 2, ..., k\}$, and the updating parameters for B are:

$B.b_i$

$$b_{ij} = \frac{\sum_{n=1}^{N}\sum_{t=1}^{T_n} q(X_{nt} = A_i)\delta(Y_{nt} = j)}{\sum_{j=1}^{k}\sum_{n=1}^{N}\sum_{t=1}^{T_n} q(X_{nt} = A_i)\delta(Y_{nt} = j)}$$

where $\delta(\cdot)$ is the Dirac delta function. The same results are applied when the emission distributions obey the Bayes Bernoulli distribution since Bernoulli distribution is a special case of multinomial distribution. Namely, $Y_t = y_j|X_t = A_i \propto b_{ij}^{y_j}(1 - b_{ij}^{1-y_j})$ with $y_j \in \{0, 1\}$ and parameters $0 \leq b_{ij} \leq 1$.

When handling with transition matrix A, the situations for traditional HMM and dHMM are different. Let Λ_A be the Lagrange function related to parameters A, namely,

$$\begin{aligned}\Lambda_A = &\sum_{n=1}^{N}\sum_{t=2}^{T_n}\sum_{X_{n,t-1},X_{n,t}} (q(X_{n,t-1}, X_{n,t})log(P(X_{n,t}|X_{n,t-1}, A))) \\ &+ \alpha log|\tilde{K}_A| - \sum_{i=1}^{k}\beta_i(\sum_{j=1}^{k} A_{ij} - 1)\end{aligned} \tag{2.12}$$

The gradients corresponding to parameters A are computed by

$$\nabla_{A_{ij}}\Lambda_A = 0 \tag{2.13}$$

When $\alpha = 0$, the updates for A are the same with traditional HMM:

$$A_{ij} = \frac{\sum_{n=1}^{N}\sum_{t=2}^{T_n} q(X_{n,t-1} = i, X_{nt} = j)}{\sum_{n=1}^{N}\sum_{t=2}^{T_2}\sum_{j} q(X_{n,t-1} = i, X_{nt} = j)}$$

When $\alpha > 0$, the solution for Eq.(2.13) has no closed form. We iteratively maximize Eq.(2.12) with projected gradient ascend method. First the gradients

are computed by:

$$\partial L_{A_{ij}} = \sum_{n=1}^{N} \sum_{t=2}^{T_n} \frac{q(X_{n,t-1}, X_{nt})}{A_{ij}} + \alpha \nabla_{A_{ij}} log|\tilde{K}_A| \tag{2.14}$$

with $\nabla_{A_{ij}} log|\tilde{K}_A| = \frac{1}{2} \sum_{m=1}^{k} \left([\tilde{K}_A^{-1}]_{mi} \frac{\sqrt{(}A_{mj})}{\sqrt{(}A_{ij})} \right)$. The updates for A are

$$A^{new} = A^{old} + \gamma \cdot \partial L_A \tag{2.15}$$

where γ is the step size, and here we apply adaptive step in our implementation.

Then we project all rows of A onto the $k-1$ probability simplex by finding a nearest point in the simplex for A^{new}, equally, we try to solve the following optimization problem:

$$\begin{aligned} min_{a_i} ||a_i - A_{i\cdot}^{new}||^2 \quad i = 1, ..., k \\ s.t. a_i^T \mathbf{1} = 1, \quad a_i \geq 0 \end{aligned} \tag{2.16}$$

We refer readers to Algorithm 1 in (Wang and Carreira-Perpinán, 2013) for more details.

The overall procedure for updating transition parameter matrix is summarized in Algorithm 2.1.

The rows of input A^{old} is initialized by samples from Dirichlet distribution and as shown, our stop criterion is based on the likelihood contributed by parameters A. The most time-consuming step is to compute the gradients, which are obtained by matrix inversion operation. Fortunately, we usually maintain a small transition matrix to be manipulated.

Algorithm 2.1: updating A for dHMM

Require: Initialization A^{old}, initial step γ, error threshold δ

$A^{new} = A^{old}$, L_A^{old}
repeat
  Compute gradient ∂L_A
  find the suitable step size γ
  $A^{new} \leftarrow A^{old} + \partial L_A \times \gamma$
  Project onto the probability simplex:
  $A^{new} \leftarrow ProjSimplex(A^{new})$ (Algorithm 1 in (Wang and Carreira-Perpinán, 2013))
  compute L_A^{new} with A^{new}
  if $|L_A^{new} - L_A^{old}| < \delta$ **then**
    break;
  end if
  $A^{old} \leftarrow A^{new}, L_A^{old} \leftarrow L_A^{new}$
until $|L_A^{new} - L_A^{new}| < \delta$
return A^{new}

Supervised setting

To solve the optimization problem in Eq.(2.7), again the projected gradient ascend method is applied. And the gradients with regard to A_{ij} are computed as:

$$\partial L_{A_{ij}} = \sum_{n=1}^{N} \sum_{t=2}^{T_n} \frac{\delta(X_{n,t-1} = i, X_{n,t} = j)}{A_{ij}} + \alpha \nabla_{A_{ij}} log|\tilde{K}_A| - 2\alpha_A(A_{ij} - A_{0ij}) \tag{2.17}$$

From above, the pairwise hidden states are counted rather than inferred in the supervised setting. $\nabla_{A_{ij}} log|\tilde{K}_A|$ is the same as in the unsupervised setting. Again, we iteratively update A with the initialization A_0 by gradient ascend method until converged.

Finally, we apply Viterbi algorithm to find the most likely hidden state sequences by solving the problem $max_X P(X, Y|\lambda)$ for unlabelled sequential observations under both unsupervised and supervised settings.

Convergence analysis

In the unsupervised setting, we maximize $L(Y;\lambda) + \alpha \log|\tilde{K}_A|$ which is lower bounded by $\mathcal{L}(q,\lambda) + \alpha \log|\tilde{K}_A|$ (Bishop et al., 2006). The E-step is the same with the general EM algorithm for HMMs. With fixed $\lambda^{old} = (\pi^{old}, A^{old}, B^{old})$, in E-step, the exact posterior distributions of hidden states are derived by maximizing the lower bound of the likelihood $\mathcal{L}(q,\lambda)$, i.e., $\mathcal{L}(q^\star,\lambda^{old}) \geq \mathcal{L}(q,\lambda^{old})$, where $q^\star$ is the optimal posterior distributions. In M-step, $\mathcal{L}(q^{old},\lambda)+\alpha \log|\tilde{K}_A|$ is maximized by using the gradient ascend algorithm, i.e., $\mathcal{L}(q^{old},\lambda^\star) + \alpha \log|\tilde{K}_{A^\star}| \geq \mathcal{L}(q^{old},\lambda^{old})+\alpha \log|\tilde{K}_{A^{old}}|$, where $\lambda^\star$ corresponds to the local maximum of the objective function. Therefore, we can conclude that the EM optimization produces a sequence of objective value that converges to a local maximum.

Similarly, in the supervised setting, the sequence of the objective value produced by the gradient ascend method will also converge to a local maximum.

Algorithm complexity analysis

From the above parameter learning and inference procedure, the computation for each related equation is simple but relevant to all N sequence observations. In addition, the computation for transition matrix in the proposed dHMM is more complex than the traditional HMMs, and its time complexity is $\bigcirc(k^3)$ with k the number of hidden states. Therefore, when it comes to large-scale datasets with either big N or big k, the proposed model can be accelerated with different techniques such as parallel computing in blockwise and matrix inversion approximation Soleymani (2012).

2.4 Experimental results

In this section, we demonstrate the effectiveness of our proposed diversified HMM (dHMM) by conducting experiments on both simulated and real-world datasets.

2.4.1 Toy experiments

For simulated dataset, $\{1, 2, 3, 4, 5\}$ as state space, where the cardinality of the state space is $k = 5$. For the ground-truth initial hidden state distribution, it is set as $\pi = (0.0101, 0.0912, 0.2421, 0.0652, 0.5914)$. The transition matrix A is shown in the first column of Fig. (2.2a). The emission probabilities are chosen to be single mode Gaussian distributions. The parameters of means and variances for k Gaussian distributions are set as $B.\mu = (1, 2, 3, 4, 5)$, and $B.\sigma_i = 0.025, i \in \{1, 2, 3, 4, 5\}$.

300 observation sequences were randomly generated from the ground-truth parameters above. For simplicity, we equally set length of all sequences as six, namely, $T_n = 6, n \in \{1, 2, ..., 300\}$. The EM framework represented in Sec 2.3.4 was applied to learn parameters $\lambda = (\pi, A, B)$ for both the traditional HMM and dHMM. The parameters of mean and variance of the emission distributions were initialized with samples from Gaussian distribution and Gamma distribution respectively. Initial state distribution and rows of transition matrix were sampled from a Dirichlet distribution $Dir(\eta_i)$, where the concentration parameters are set as $\eta_i = 3, i \in \{1, 2, 3, 4, 5\}$. For diversified HMM, the balance parameter is set as $\alpha = 1$. The learned parameters are shown in Fig.(2.2).

dHMM vs. HMM on Toy dataset

Since no label information for the learned parameters, alignment between learned parameters and the ground-truth is applied for visualization of the intuitive com-

parison. The rows of learned transition matrix are aligned by minimizing the distance between the learned transition matrix and the ground-truth transition matrix. The learned initial distribution and emission distributions are also aligned with the ground-truth ones accordingly.

In Fig.(2.2a), each column corresponds to one 5×5 transition matrix and each row corresponds to one state's transition distribution. The first column denotes the ground-truth transition distribution, and the last two columns are transition matrices learned from the traditional HMM and diversified HMM respectively. Comparing to the result of traditional HMM (the middle column), the diversity-encouraging prior takes effect as illustrated in the third column: The transition distributions of different states are mutually distinct. Until now, it is still hard to decide which model infers the hidden states better, since different hidden structures inferred by different models may inherit different meanings.

In Fig.(2.2b), each row illustrates the other two kinds of parameters $(\pi, B.\mu, B.\sigma)$ - the first column denotes the state initial distribution while the other two columns denote the means and covariances of the five emission Gaussian distributions. The first row shows the ground-truth. The second row shows the MLE result learned from the traditional HMM. The last row shows the MAP result learned from the proposed dHMM. From the figure, the traditional HMM identifies only two groups of patterns: The states $1, 2, 3, 4$ are in the first group, which have quite similar emission distributions in terms of their Gaussian means and variances. The state 5 is in the second group, which has very different Gaussian parameters comparing to the others. In this case, hidden states $1, 2, 3, 4$ are difficult to be differentiated, which can lead to ambiguous labelling results. In contrast, our proposed diversified HMM shows superiority in terms of its higher discriminative ability for differentiating the hidden states involved. This is revealed in the third row of Fig.(2.2b), which shows that different hidden states

induce distinct emission components. From the results obtained, the claim that the proposed diversity-encouraging prior over rows of transition matrix indirectly increases the discrimination of hidden states is justified.

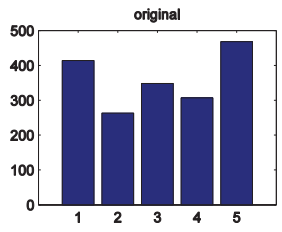
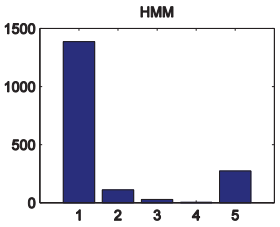
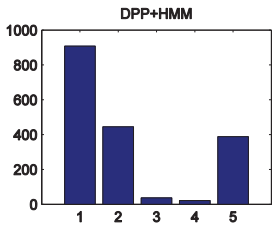
We also compared the proposed diversified HMM with traditional HMM in terms of sequential labelling accuracy. Given the learned parameters, the most likely sequential labels are inferred for each observed sequence by Viterbi algorithm. Both the inferred state frequencies and labelling accuracies of sequential labelling for different models are summarized in Table (2.1). The histograms of the sequential labels (i.e., the frequencies of hidden states in the given dataset) inferred from ground-truth parameters, parameters learned from traditional HMM and parameters learned from diversified HMM are shown in the second row of Table (2.1). The ground-truth histogram distributes almost equally amongst the five states, while the statistic of hidden states inferred from traditional HMM tends to be highly biased which favors one dominant state. This problem is somewhat rectified by the proposed diversified HMM. Shown in the third column, the histogram of hidden states inferred from dHMM shows more resemblance to the ground-truth than the histogram inferred from traditional HMM.

To compute the 1-to-1 accuracy of sequential labelling, labels inferred by parameters learned from both the traditional HMM and diversified HMM are aligned to the ground-truth by Hungarian algorithm. As shown in the third row of Table (2.1), the proposed dHMM outperforms traditional HMM by a large margin.

More explanations on dHMM's superiority over HMM

Further, in this subsection, the superiority of our proposed dHMM over traditional HMM is statistically illustrated especially in the case where the emission distributions are almost flatten. Under this situation, the hidden states are am-

Table 2.1: Comparison of state frequencies and accuracies between dHMM and HMM

	ground-truth	HMM	dHMM
state histograms			
labeling accuracies	1	0.4117	**0.4728**

biguous and less discriminative, which leads to that traditional HMM identifies less hidden states than the ground-truth, i.e. the learned transition matrix contains more similar rows than the ground-truth transition matrix does. Not only intuitively but also experimentally, our proposed dHMM mitigates this issue to some extent.

All ground-truth parameters of HMM take the same setting as above except the variances of the emission distributions $B.\sigma_i, i \in \{1,2,3,4,5\}$. The variances of the Gaussian distributions are gradually enlarged to 'flatten' the emission distributions. In here, we used a sequence of variance parameters $\{B_t.\sigma_i, i \in 1,...,5\}_{t=1}^{T}, T = 50$, where $B_t.\sigma_i = 0.025 + 0.1 \times (t-1)$. For each t, we generated the experimental sequences by the same method described above. The experimental results are averaged over 10 runs with independent initializations.

We apply averaged Bhattacharyya distance over all pairwise of rows of transition matrix as diversity measure. Higher Bhattacharyya distance means more diversity of rows of transition matrix. The quantized diversities of rows of transition matrix is shown in Fig. (2.3). The green line shows the diversity of the ground-truth transition matrix whose value is 0.531. The red curve below the

green line and the blue curve above the green line show the diversities of the transition matrices learned by traditional HMM and proposed dHMM respectively. The effectiveness of diversity-encouraging prior is obvious: The dHMM consistently outperforms the HMM, no matter what the parameters of variances are set.

Higher diversity of transition matrix implies more inferred hidden states. One example of the histogram of the inferred states is shown in Fig.(2.4), in which the number of states is identified by omitting the labels whose frequencies are below certain threshold σ_F. In our case, we set the threshold as $\sigma_F = 50$ indicated by the black line. We show that the dHMM (coloured as Green) identifies all five states, while the HMM (coloured as Red) identifies only two states, since the frequencies of states $2, 3, 4$ are below the threshold σ_F.

We summarize the statistical results in Fig.(2.5). From the left part of the curve, the emission Gaussian distributions start with low variance. The hidden states can be easily identified and the dHMM performs on par with HMM. However, along with the increasing of the variance, the emission Gaussian distributions are becoming increasingly 'flattened', which make the states severely ambiguous and hard to identify. Shown in the right half part of the curve, the advantage of our dHMM is becoming more obvious as it identifies more hidden states than the traditional HMM does.

2.4.2 Real-world experiments

In this section, our proposed diversified HMM (dHMM) is applied to solve real-world sequential labelling problems: PoS tagging under unsupervised setting and OCR under supervised setting.

PoS tagging

Part-of-Speech tagging (PoS tagging) (Mitkov, 2003) has been used in the linguistics community for a long time. The task is to automatically assign contextually appropriate grammatical descriptors to words in texts. In fact, PoS tagging usually produces low level semantic information, which can serve as a precursor towards more abstract levels of analysis, e.g., text indexing and retrieval, as nouns and adjectives are better candidates for index terms than adverbs or pronouns are.

The Penn Treebank Wall Street Journal (WSJ) corpus ((Marcus et al., 1993)) is one of the most widely used datasets for evaluating performance of the statistical language models. The training corpus, tagged by gold standard PoS tags, is utilized to evaluate proposed diversified HMM (dHMM) for Part-of-Speech (PoS) tagging task under unsupervised setting. The vocabulary size of the corpus is around $10K$. In Table (2.2), the PoS tags appeared in the WSJ corpus are listed. Detailed definitions of the abbreviation of tags are annotated in (Marcus et al., 1993). The tags are preprocessed to reduce the hidden state size from 46 to 15 by combining similar tags. The idx column shows the indexes of the reduced tag set. The PoS column shows the semantic title of the tags. The $frequency$ column shows the frequencies of all tags. From this statistics, 25% tags account for nearly 85% words. All of 3828 sentences are used in our experiment, and the sequential length is between $2 \sim 250$. An example sentence with true sequential PoS tags is illustrated in Fig. 2.6, where the true tags lay behind the corresponding words. Naturally, the transition distribution for different tags are different. Take tags /NNP and /VB as an example, the /NNP has higher probability to be followed or following the same /NNP tag. By contrast, /VB is usually followed by /DT or /IN, and follows /MD, /TO or /RB. This discriminative prior information considered by our model is significantly helpful

Table 2.2: Summary of PoS tags of WJS corpus

idx	PoS	frequency	idx	PoS	frequency
1.	NNP	9408	6.	VBD	3043
1.	NNPS	244	6.	VBN	2134
1.	NNS	6047	6.	VBP	1321
1.	NN	13166	6.	VBG|NN	1
1.	SYM	1	7.	DT	8165
2.	,	4886	7.	PDT	27
2.	–	712	7.	WDT	445
2.	”	693	8.	IN	9959
2.	:	563	8.	CC	2265
2.	.	3874	8.	TO	2179
2.	$	724	9.	FW	4
2.	(	120	10.	WRB	178
2.	)	126	10.	RB	2829
2.	LS	13	10.	RBS	35
2.	#	16	10.	RBR	136
3.	CD	3546	11.	UH	3
4.	JJS	182	12.	WP	241
4.	JJ	5834	12.	WP$	14
4.	JJR	381	12.	PRP	1716
5.	MD	927	12.	PRP$	766
6.	VBZ	2125	13.	POS	824
6.	VB	2554	14.	EX	88
6.	VBG	1459	15.	RP	107

for sequential labelling, which is verified and demonstrated in detail below.

The number of hidden states is set as $k = 15$ as enumerated in Table (2.2). Afterwards, the initial state distribution π is a 15-dimension vector and the transition distribution is parameterized by A which is a 15×15 matrix. The words in the vocabulary are treated as observations and the emission distributions are parameterized by B which is a $k \times V$ matrix, where V is the size of vocabulary. The π, each row of A and each row of B are randomly initialized by samples from the Dirichlet distribution. We apply the 1-to-1 accuracy measure to quantize

our experimental results. Similar with the simulated experiment, the Hungarian algorithm is utilized to map the inferred labels to the ground-truth ones.

First, we test the effectiveness of diversity-encouraging prior over rows of transition matrix in terms of prior weights, namely, α values. The labelling accuracies with regard to αs are plotted in Fig.(2.7). The setting $\alpha = 0$ corresponds to the setting of traditional HMM. Traditional HMM gets an accuracy of 0.4475, while our proposed dHMM achieves the best accuracy of 0.4688 with $\alpha = 100$. Larger weight (α) will overemphasize the diversity-encouraging prior over rows of transition matrix and will lead to decreasing the sequential labelling accuracy, as shown with a sharp drop when α increases to 1000 in Fig.(2.7).

Then, we qualitatively demonstrate the effectiveness of our diversity prior by comparing the transition parameters matrix learned from dHMM ($\alpha = 100$) to the parameters learned from the baseline - traditional HMM. The diversity measurements (Bhattacharyya distance) between tag 1 and the other tags are shown in Fig. (2.8). The proposed dHMM identifies that tag 1 (NOUN) is most different from tag 11 (Interjection), while HMM identifies that tag 1 (NOUN) is most different from tag 5 (MODAL). Since the frequency of tag 11 (Interjection) is only 3, intuitively, the transition distribution of this tag is quite different from the transition distributions of other tags. The same situation is applied to tag 9 (Foreign word, whose frequency is 4). From Fig. (2.8), the proposed dHMM identifies more accuracy result, which indicates that the transition distributions of both tags 11 and 9 are most different from tag 1 (NOUN), than the result learned from the traditional HMM.

Finally, we show how the diversity-encouraging prior indirectly rectifies the emission distributions learned from traditional HMM to fit the dataset better, as illustrated in Fig. (2.9). Here, we choose $\alpha = 100$ to explain the behaviour of the proposed dHMM. Three curves are plotted. The statistics of the 'ground-

truth' curve is obtained through the inferred hidden labels by the true parameters. [1] From the 'ground-truth' curve, small portion of tags explain majority of the words, which is pointed out as skewed long-tail distribution (Johnson, 2007). 'dHMM' curve learned by our proposed diversified HMM reflects this phenomenon especially for the less frequent 10 tags and shows promising result for unsupervised sequential labeling task. To some extent, the diversified transition prior latently adjusts the flatten distributions for the less frequent 10 tags obtained from traditional HMM to a better trend approaching the true distributions.

OCR

Optical character recognition (OCR) is a task of converting images of typewritten or printed texts (which are the common forms of scanned data, e.g., passport, receipts) into computer-readable texts. It can be applied to many real-world applications, such as efficient data entry for business documents, automatic number plate recognition. To achieve this aim, one has to perform both character segmentation and recognition tasks. In this work, we assume every character has been segmented out and stored in its own image patch, so that we focus on the recognition task, which is referred to as sequential labelling here.

We apply the OCR dataset processed by (Taskar et al., 2003). They select clean subset from the handwritten words collected by Rob Kassel at the MIT Spoken Language Systems Group. By removing the first capitalized letters, Ben et al. rasterized and normalized images of the rest lowercase letters into 16×8 images. There are in total 6877 words containing $1 \sim 14$ letters. Three word

[1] obtained by counting the starting tags, the pairwise tags, and the tag-word pairs of each sentence through the whole corpus, these three statistics are corresponding to physical meanings of the HMM parameters $\lambda = (\pi, A, B)$: initial distribution, transition distribution and emission distributions.

Table 2.3: Examples of OCR dataset

		mbraces
		ommanding
		olcanic

examples are listed in Table (2.3). The first two columns show two different handwritten patterns from different persons for the word listed in the third column. Apparently, each letter has different probability being transferred to other letters. As highlighted in Table (2.3), letter 'm' has high probability to be followed by 'm', 'a' or 'b', while letter 'n' will prefer to be transferred to 'd', 'g' or 'i'. Intuitively, we suppose our proposed model will take effect on this dataset and we verify it in the following.

We apply the true number of lowercase English letters as the size of the hidden state space, namely $k = 26$. Accordingly, the initial state distribution π is a 26-d vector , and the transition matrix A is a 26×26 matrix. Each observed letter image is reshaped into a binary 1×128 vector. For the emission distributions, Naive Bayes assumption is applied and each dimension of binary vector is independently modelled by Bernoulli distribution, parameterized by $p_d, d \in 1, 2, ..., 128$, measuring the probability of that the current pixel value is equal to 1. Finally, emission distributions B is modelled by $26 * 128 = 3328$ parameters. In supervised setting, the parameters $\lambda = (\pi, A_0, B)$ are learned by MLE from the training set. All of our experiments are run with 10-fold cross

validation.

Like PoS experiment, we first test the effectiveness of our proposed diversity-encouraging prior with a range of αs. The test accuracies are shown in Fig. (2.10). The results are given by the averages across 10 runs. Another parameter α_A, which tries to drag A to A_0, has been chosen through the 1-to-1 accuracy criterion and is fixed as $1e5$. $\alpha = 0$ corresponds to the traditional HMM and it gets an accuracy of 0.7102 while our proposed dHMM obtains an accuracy of 0.7203 with $\alpha = 10$. That an increasing trend is gained demonstrates the effectiveness of dHMM though larger α will decrease the performance.

Our proposed model is compared to three baseline algorithms for supervised sequential labeling: Naive Bayes, traditional HMM and Optimized HMM (Krevat and Cuzzillo, 2006). The average accuracies with standard deviations are shown in Fig. (2.11). Naive Bayes algorithm ignores the relationship between neighbor letters and achieves the lowest accuracy of 62.7% with a standard deviation of 1.1%. Incorporating one-order chain structure of letters, HMM achieves a 70.6% accuracy rate with a standard deviation of 1.3%. With other tricks, the optimized HMM obtains limited improvement. By contrast, by adding diversity-encouraging prior over the rows of transition matrix of traditional HMM, our proposed dHMM achieves an accuracy of 72.06 with a standard deviation of 2.2% which apparently gains a significant margin.

Finally, a qualitative demonstration of the diversity is shown in Fig. (2.12). The transition matrix A is trained from the setting of $\alpha = 10$, $\alpha_A = 1e5$. Fig.(2.12a) (Fig. (2.12b)) shows the diversity measurements (Bhattacharyya distance) between transition distribution of character 'x' ('y') and transition distribution of the other 25 letters. From the curves, the total trends almost are the same everywhere between traditional HMM and our proposed dHMM, except that dHMM heights the pairwise diversities between transition distributions of

('x','g'), ('x','j') and ('y','f'), which we conclude contributes to the improvement of test accuracies in some extent .

2.5 Summary

Based on the methodology of traditional HMM, a diversified HMM (dHMM) for sequential labelling was proposed in this chapter. Instead of explicitly constraining the parameters associated with observations, we placed a diversity-encouraging prior over parameters of transition distributions, modelled by determinantal point processes (DPP), which is an essential part of a traditional HMM. To facilitate this variation of HMM, a new maximum a posterior (MAP) scheme was proposed under both unsupervised setting and supervised setting. For unsupervised setting, maximum a posterior with marginal likelihood was solved based on an EM framework that is similar to the one for the traditional HMMs, but with a modified M-step. For supervised setting, maximum a posterior with joint likelihood was trained directly through a gradient descend method. We verified the effectiveness of the proposed dHMM through both simulated and real-world datasets (e.g., unsupervised PoS tagging and supervised OCR).

Future work will involve with the development of a non-parametric extension to dHMM, which simultaneously learns the number of hidden states, as well as all HMM parameters. We will carry out a theoretical study into the effectiveness of the number of states as well as diversity-encouraging prior over rows of transition matrix under our setting with regard to labelling accuracy.

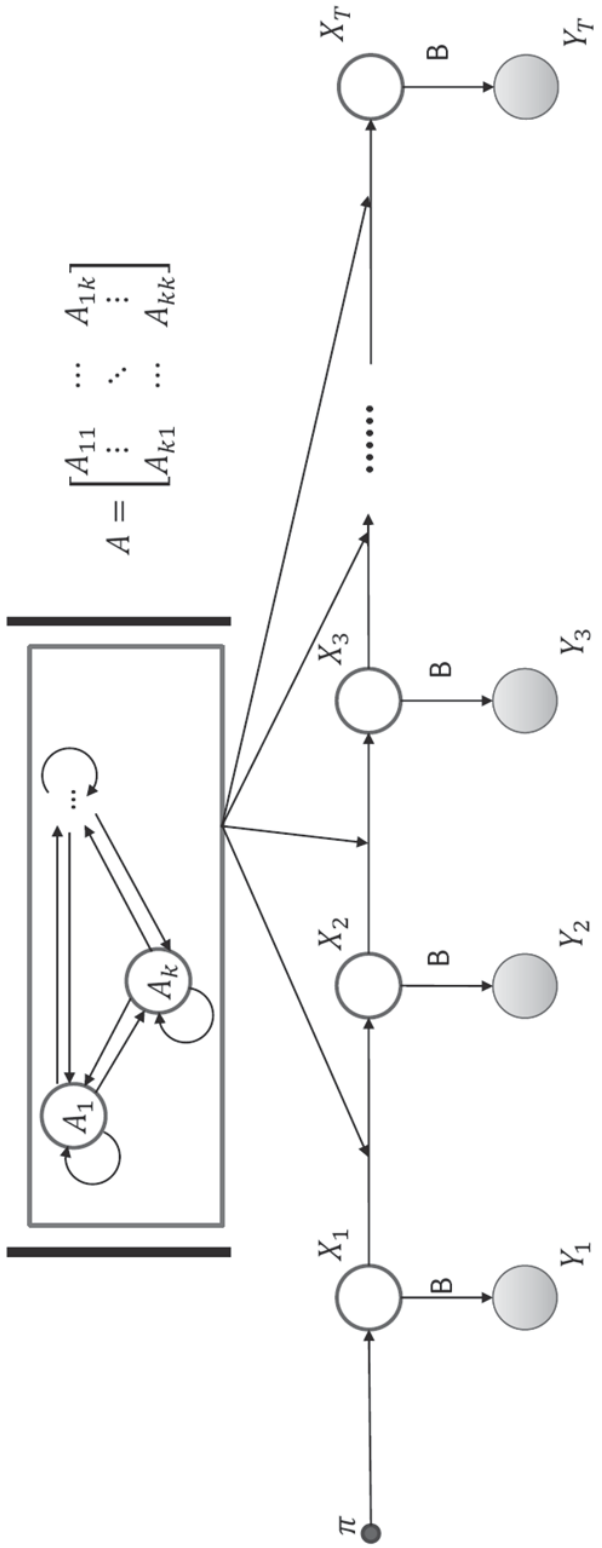
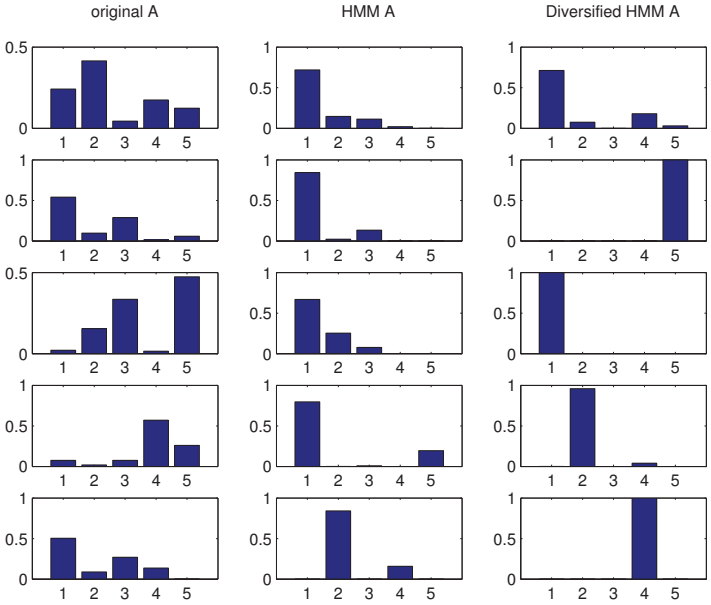
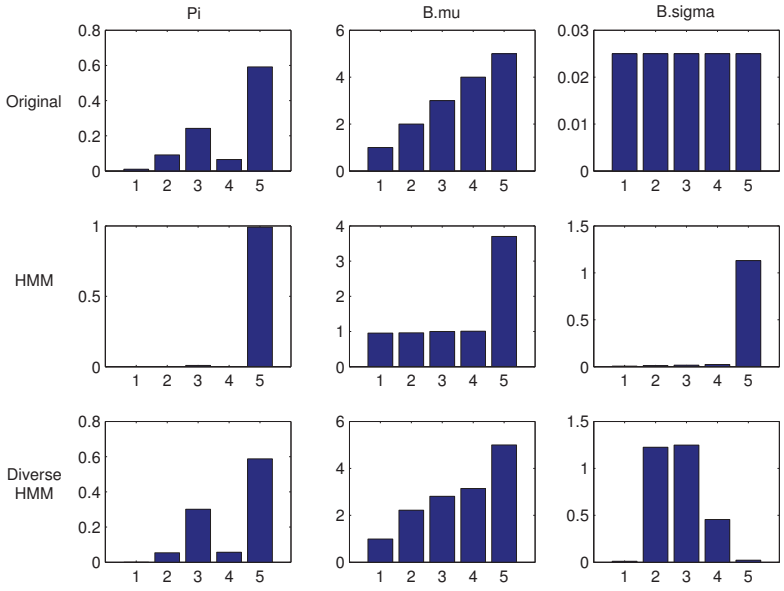


Figure 2.1: Graphical model of diversified HMM



(a) Transition matrix A



(b) Initial distribution π and emission distribution B

Figure 2.2: Parameters of ground-truth, learned by proposed dHMM and by traditional HMM

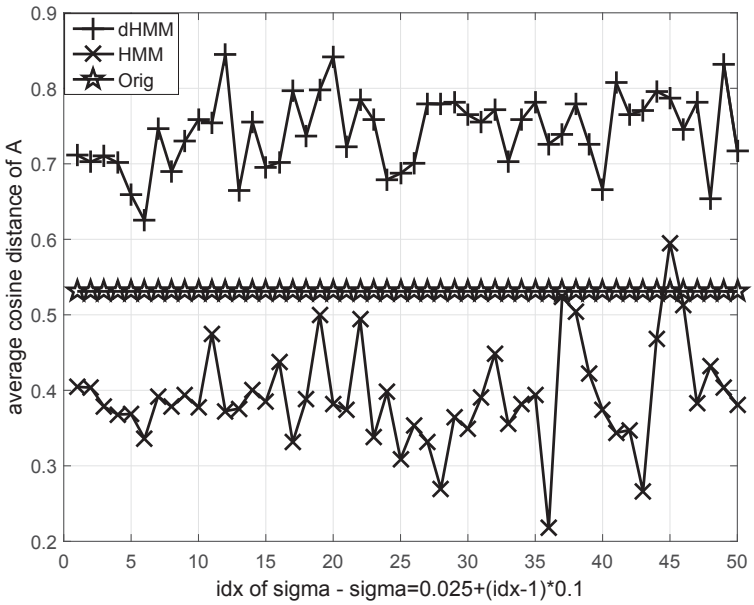


Figure 2.3: Diversities of transition matrix of ground-truth, dHMM-learned and HMM-learned with regard to the parameter of variance of the Gaussian emission distributions

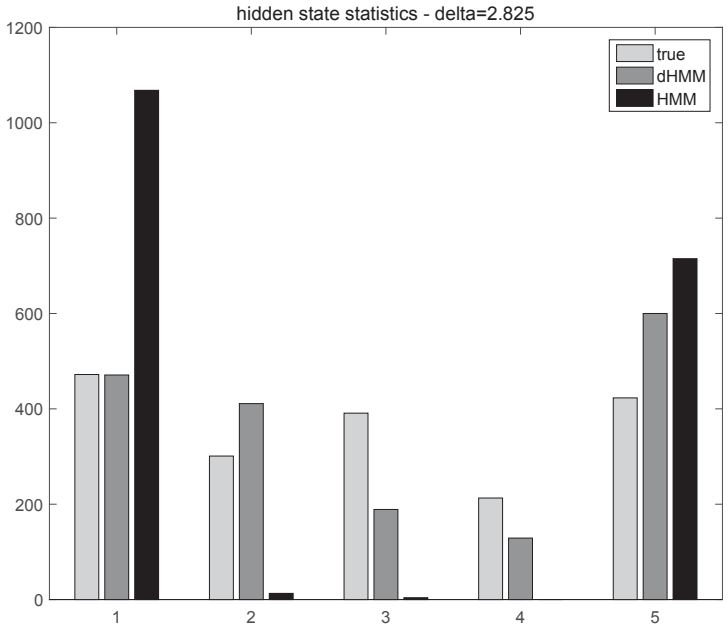


Figure 2.4: Histograms of hidden states inferred from parameters of ground-truth, dHMM-learned and HMM-learned

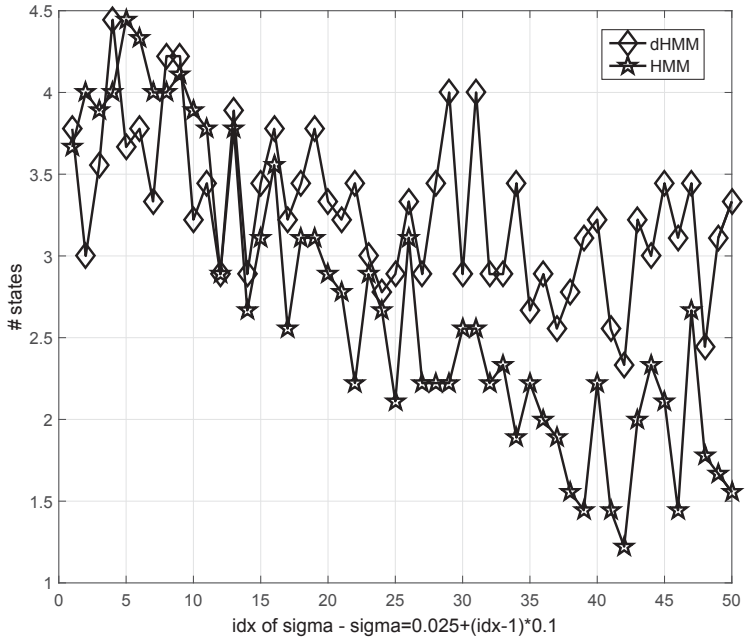


Figure 2.5: Number of hidden states inferred by model parameters of dHMM-learned and HMM-learned with regard to the variance of Gaussian emission distributions

Pierre/NNP Vinken/NNP ,/, 61/CD years/NNS old/JJ ,/,
will/MD join/VB the/DT board/NN as/IN a/DT
nonexecutive/JJ director/NN Nov./NNP 29/CD ./.

Figure 2.6: Sentence example with PoS tags

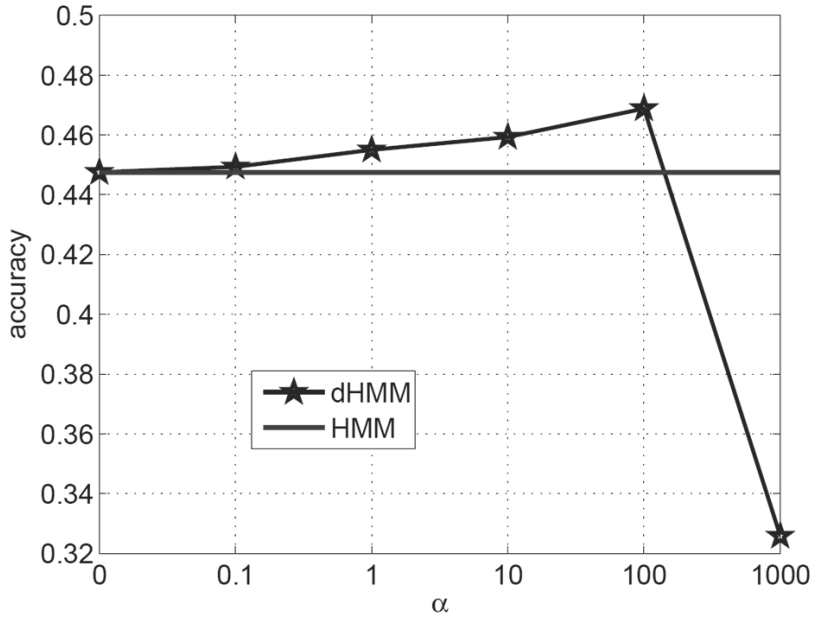


Figure 2.7: Effectiveness of α for PoS tagging

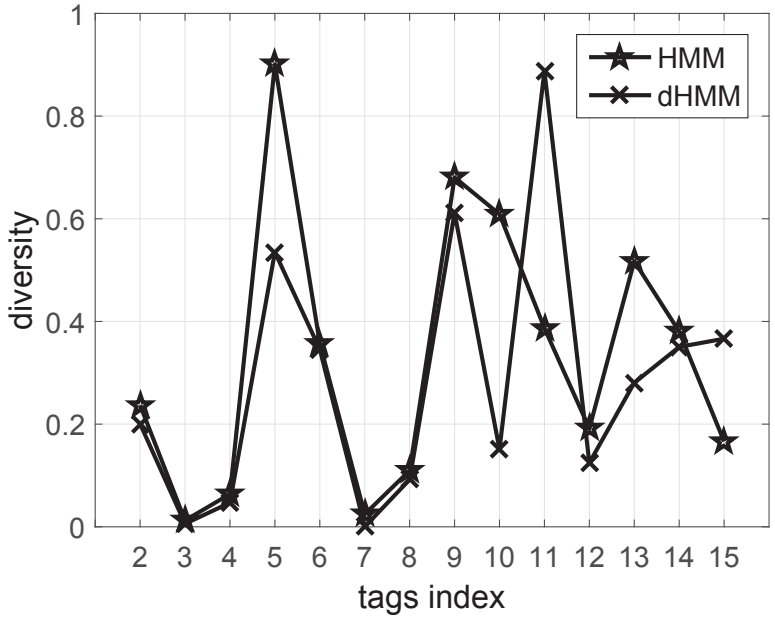


Figure 2.8: Transition diversity comparison between dHMM and HMM for tag '1' and all other tags

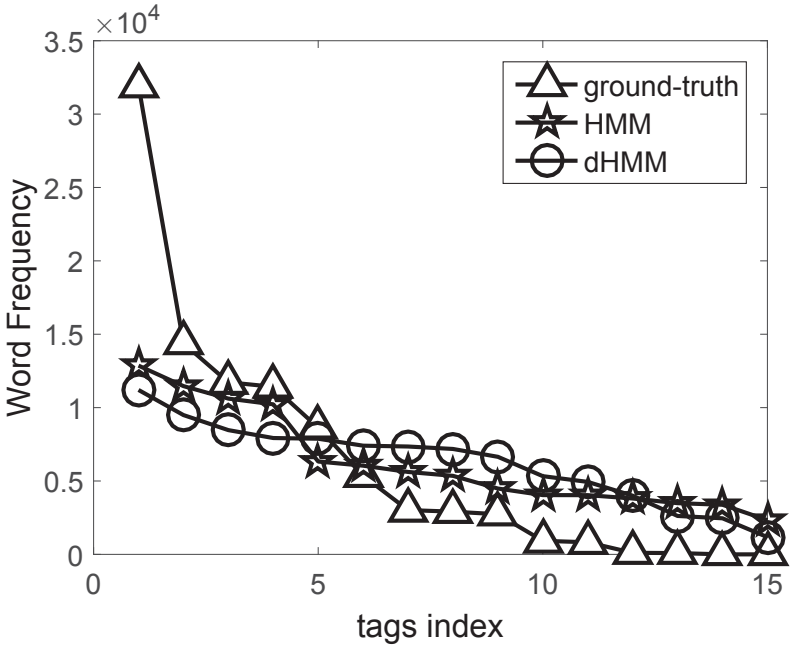


Figure 2.9: Histogram comparisons among ground-truth, HMM and dHMM

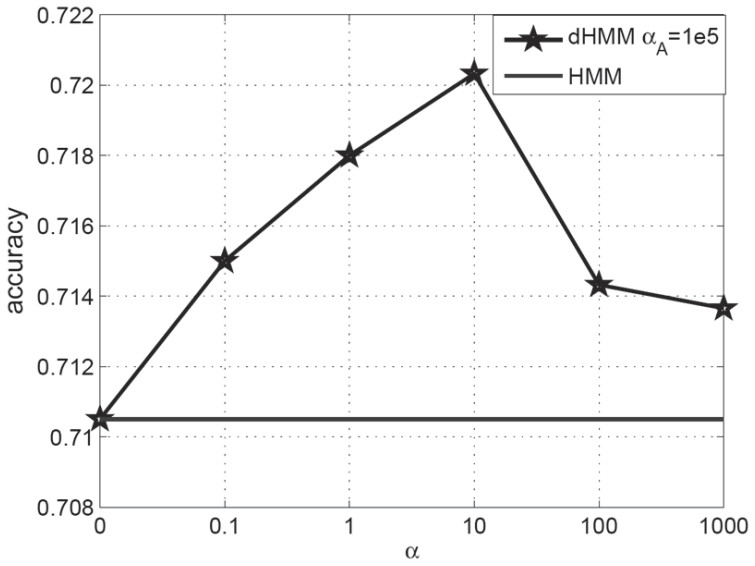


Figure 2.10: Effectiveness of α for OCR

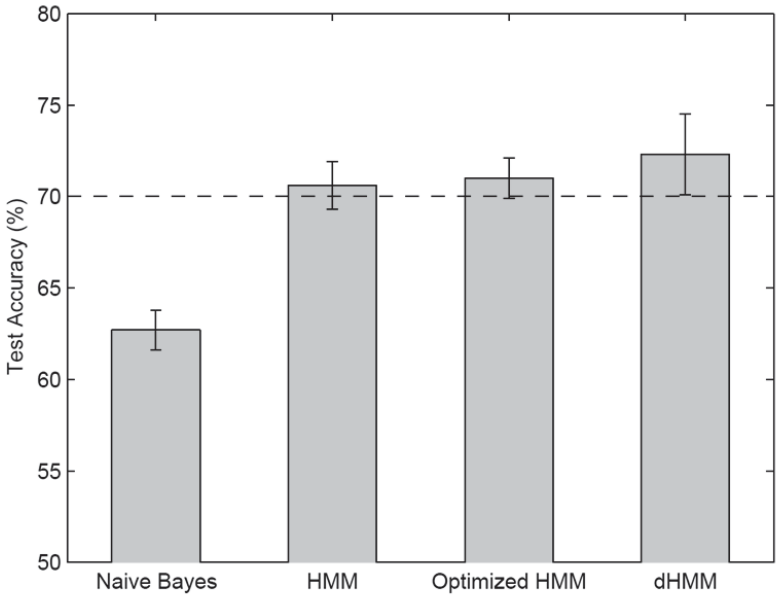
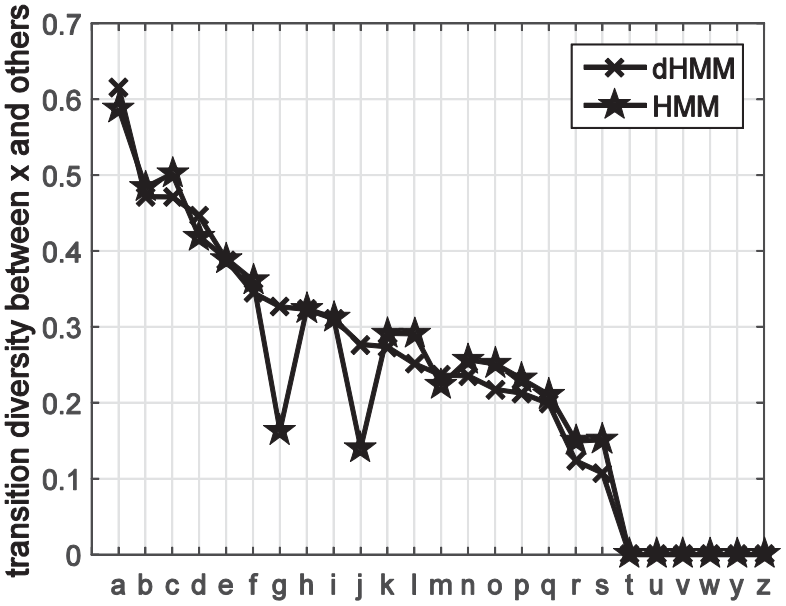
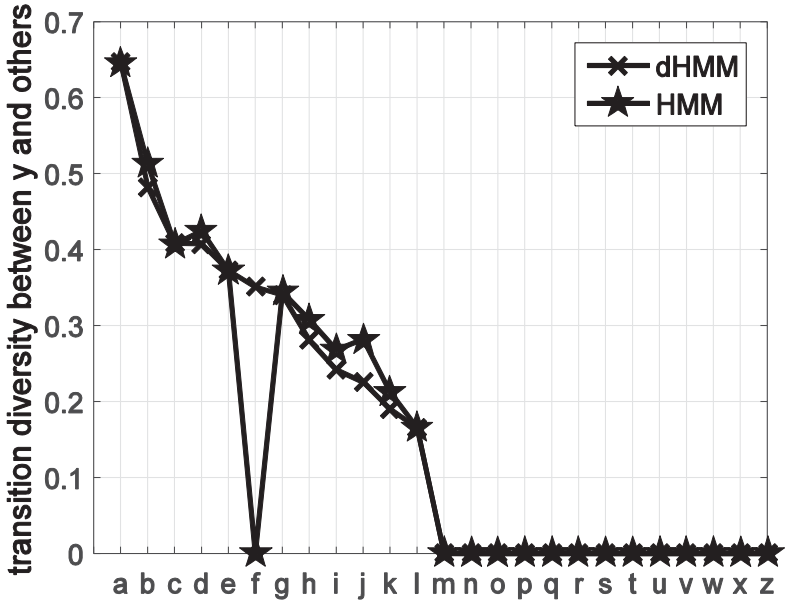


Figure 2.11: Test accuracies of different classifiers



(a) letter x



(b) letter y

Figure 2.12: Transition diversity comparison between dHMM and HMM

Chapter 3

Fast Sampling for Time-Varying Determinantal Point Processes

Determinantal point processes (DPPs) are stochastic models which assign each subset of a base dataset with a probability proportional to the subset's degree of diversity. It has been shown that DPPs are particularly appropriate in data subset selection and summarization tasks, e.g., news display, video summarization. DPPs prefer diverse subsets while other conventional models cannot offer. However, DPPs inference algorithms have a polynomial time complexity which makes it difficult to handle large and time-varying datasets, especially when real-time processing is required. To address this limitation, we developed a fast sampling algorithm for DPPs which takes advantage of the nature of the time-varying setting (e.g. news corpora updating, communication network evolving), where the data changes between time stamps are relatively small. The proposed algorithm is built upon the simplification of marginal density functions over successive time stamps and the sequential Monte Carlo (SMC) sampling technique. Evaluations on both a real-world news dataset and the Enron Corpus dataset confirm the efficiency of the proposed algorithm.

3.1 Introduction

Determinantal point processes (DPPs) naturally model repulsive interaction where diverse subsets are preferred. It arises as an important tool in random matrix theory (Ginibre, 1965), physics (fermions, eigenvalues of random matrices) (Macchi, 1975) and in Combinatorics (non-intersecting paths, random spanning trees) (Hough et al., 2006). Recently, it has been introduced to the machine learning field, and shown to be particularly valuable in many data mining applications such as text summarization (Kulesza and Taskar, 2011*b*), document thread revealing (Gillenwater et al., 2012*a*), topic model (Zou and Adams, 2012), information retrieval (Kulesza and Taskar, 2011*a*), pose estimation (Kulesza and Taskar, 2012), and neural inhibition (Snoek and Adams, 2013).

DPPs have exact inference/sampling algorithms, and they are developed based on eigen-decomposition of DPPs' kernel matrix. Therefore, the time complexity of DPPs sampling is $\mathcal{O}(N^3)$ (Kulesza and Taskar, 2012), where N is the total number of items in the dataset. Such time complexity makes DPPs infeasible for real-time processing, especially when N is large. In order to improve the efficiency of DPP inference algorithms, different approaches have been proposed. Dual representation is introduced when the kernel matrix is low-rank (e.g., rank $D \ll N$) based on Gram matrices (Kulesza and Taskar, 2012), and the time complexity is reduced to $\mathcal{O}(ND + D^3)$. Nyström approximation and Matrix Ridge Approximation (MRA) (Wang et al., 2014) have also been introduced to enhance the efficiency of eigen-decomposition of a kernel matrix, which is the most time-consuming operation for DPP sampling. Most recently, a scheme based on an Markov Chain Monte Carlo (MCMC) technique for DPP sampling has been proposed by (Kang, 2013), and its ϵ-mixing time is $\mathcal{O}(N \log(N/\epsilon))$.

The above approaches aim to improve the efficiency of DPPs' computation by focusing on its inference algorithms. However, they do not take into con-

sideration the structure of the data itself, which potentially can be exploited as a valuable source of efficiency improvement (Rakthanmanon et al., 2013). One such data structure can be often found in a setting where there are sequential changes occurring in a dataset, but the changes in each time interval are relatively small when compared to the entries of the entire dataset. This structure has been seen in many online services (Abel et al., 2013). Taking online news services as an example, the services strive to sequentially display a diverse subset of news sampled from the most recently updated news corpora from many third-party sources. It is easily observed that any real-time updates of the news corpora are relatively small in a short time interval.

In this chapter, we derive a fast sampling algorithm to improve the efficiency of time-varying DPPs for handling large-scale datasets. In a time-varying setting, records are collected from many information sources at each time stamp, and the whole dataset is built sequentially. Because the update rates of different information sources vary, only a small proportion of them have new information feeds during a short time interval. Such relatively small changes are utilized to develop a fast sampling scheme for time-varying DPPs (TV-DPPs). We improve sequential DPP sampling efficiency by incorporating a simplified computation of successive marginal density functions into an SMC framework. Our contributions are summarized as follows.

- We propose a novel time-varying determinantal point processes (TV-DPPs) setting to accomplish real-time diverse subset sampling task with making use of successive, proportionally small updates of information sources between two successive time stamps, with respect to the overall large-scale dataset.

- We embed a simplification computation over successive marginal density functions into the framework of sequential Monte Carlo sampling tech-

niques to obtain a fast DPP sampling algorithm for a sequentially collected large-scale dataset.

- We evaluate the accuracy and efficiency of the proposed algorithm on two real-world scenarios, including news recommendation and Enron event discovery.

The rest of this chapter is arranged as follows. Section 3.2 presents related works. Section 3.3 introduces the background of the sequential Monte Carlo method. The proposed TV-DPPs setting and its fast sampling algorithm are detailed in Section 3.4, and the experimental results on two real-world datasets are reported in Section 3.5. Section 3.6 concludes this chapter and provides discussions on future work.

3.2 Related Work

Directly applying DPP to complex application scenarios has limitations. For example, structural elements are ubiquitous in many application domains such as chain structures for trajectories and news threads, and pictorial structures for human poses. Based on factorization of structures, both quality and similarity can be directly constructed. However, due to high-dimensional variables in structures, the size of a base set will become exponentially large. Taking advantage of dual representation, (Kulesza and Taskar, 2010) derives a tractable structure DPP.

Another example is for sequential application scenarios, and here we list two instances. One instance is an online news service system. It tries to provide every user with sequential news subsets, where diversity not only applies to news articles at any individual point in timeline, but also needs to be addressed temporally. In order to fulfil such requirements, Markov DPP is developed (Affandi,

Fox and Taskar, 2013; Liu et al., 2016). It models a sequence of random sets which come from a large-scale base set, and maintains two kinds of margin DPPs: one at each single time stamp and the other for pairwise time stamps, and one kind of conditional DPPs for the transitional probability distributions. Therefore, existing inference algorithms for DPPs can be directly applied. However, the inference algorithms could not be effective for a large-scale dataset, since the size of kernel matrices for each DPP is as large as the cardinality of the base set, which presents itself as a major bottleneck. The other instance of sequential application scenarios is large-scale video summarization. Recently, (Gong et al., 2014) developed a sequential DPP which incorporates the concept of diversity for extracting succinct subsets to summarize large-scale videos. Instead of taking whole frames as a base set as Markov DPPs do, the sequential DPP employs a divide-and-conquer strategy. It first partitions the whole video into disjoint yet consecutive segments, and then maintains margin DPPs for a union of two neighbouring segments. Consequently, the inference for a sequential DPP is more efficient. This setting is different from what we have proposed in this chapter, because of the following two reasons: (1) each segment is sequentially collected from a single source (i.e., the same video) rather than synchronously collected from different information sources, and (2) its divisions do not overlap.

Sequential subset selection techniques based on other criteria rather than diversity have also been developed in different areas, and we refer interested readers to works (Chen and Hsu, 1991; Tollefson et al., 2014; Rao et al., 2003; Liu and Tao, 2016; Xu et al., 2015).

3.3 Review of Sequential Monte Carlo

Sequential Monte Carlo (SMC) sampling techniques can be used to sample from a sequence of distributions $\pi_1(x_1), ..., \pi_t(x_t)$ for variables $X_{1:t} = \{x_1, x_2, ..., x_t\}$, where $\{1, \ldots, t\}$ are time indexes. It improves sequential important sampling with introducing auxiliary variables and artificial backward Markov kernels with density $\{L_k(x_{k+1}, x_k)\}_{k=1}^{t-1}$. It applies importance sampling (IS) technique (Del Moral et al., 2006; Wu et al., 2013; Kantas et al., 2009), between an artificial joint distribution $\widetilde{\pi}_t(X_{1:t}) = \widetilde{\gamma}(X_{1:t})/Z_t$ and a proposed joint importance distribution $\eta_t(X_{1:t}) = \eta_1(x_1)\prod_{k=2}^{t} K_k(x_{k-1}, x_k)$, where Z_t is a normalization constant, and $\widetilde{\gamma}_t(X_{1:t}) = \widetilde{\gamma}_t(x_t)\prod_{k=1}^{t-1} L_k(x_{k+1}, x_k)$, to collect samples. Note that both joint distributions - the artificial one and the proposed joint one - are decomposed in a sequential multiplying manner. The backward Markov kernels $\{L_k(x_{k+1}, x_k)\}_{k=1}^{t-1}$ in the artificial joint distribution play roles as backward conditional distributions $\widetilde{\gamma}(x_{k-1}|x_k)$. Similarly, forward Markov kernels $\{K_k(x_{k-1}, x_k)\}_{k=2}^{t}$ in the proposed joint importance distribution play roles as forward conditional distributions, i.e., $\eta(x_k|x_{k-1})$. The samples from the proposed importance distribution are usually biased from the "true" distribution. Importance sampling (IS) is applied to correct the discrepancy between them by weighting the samples with the matching degree to the "true" distribution. The likelihood ratios are usually applied as matching criteria, and are formulated as

$$\begin{aligned} w_t(X_{1:t}) &= \frac{\widetilde{\gamma}_t(X_{1:t})}{\eta_t(X_{1:t})} \\ &= w_{t-1}(X_{1:t-1})\widetilde{w}_t(x_{t-1}, x_t), \end{aligned} \tag{3.1}$$

where

$$\widetilde{w}_t(x_{t-1}, x_t) = \frac{\widetilde{\gamma}_t(x_t)L_{t-1}(x_t, x_{t-1})}{\widetilde{\gamma}_{t-1}(x_{t-1})K_t(x_{t-1}, x_t)}. \tag{3.2}$$

From Eq. (3.1), it can be easily seen that the importance weights are calculated in a recursive form. At a given time stamp, the joint weight over the time period $1 \sim t$ - $w_t(X_{1:t})$ - is computed by multiplying two components: One is the joint weight accumulated to $t-1$, i.e., $w_{t-1}(X_{1:t-1})$, and the other is the incremental weight from the local pairwise joint distributions with consecutive time stamps, i.e. $\widetilde{w}_t(x_{t-1}, x_t)$. Clearly, at each time stamp, instead of computing the whole joint weight from the beginning, SMC updates it with local incremental weights. Thus, the overall computation is efficient. From the definition in Eq. (3.2), the incremental weight relates to not only the marginal distribution, but also the backward and forward Markov kernels. How to choose these two kernels are a crucial step to achieve high approximation results. The marginal distributions $\{\pi_t(x_t)\}$ can be approximated by the sequentially sampled N particle-weight pairs $\{X_{1:t}^{(i)}, w_t^{(i)}\}_{i=1}^N$ (Ohsaka et al., 2014),

$$\pi_t^N = \sum_{i=1}^{N} w_t^{(i)} \delta(x_t^{(i)}), \tag{3.3}$$

where δ is the Dirac Delta function.

3.4 Time Varying Determinantal Point Processes

Sequential data is ubiquitous in real-world scenarios. We represent sequence data with variables $X_{1:t} = \{x_1, x_2, ..., x_t\}$, and their corresponding probability measures are denoted as $\pi_1(x_1), ..., \pi_t(x_t)$, where t is time indexes. We focus on sequentially sampling from separate distributions $\{\pi_i\}_{i=1}^t$, rather than from a joint distribution $\Pi(X_{1:t})$. This is suitable for some applications that require real-time results at each single time stamp. For example, a news provider needs to sample a diverse news subset from all its information sources in order to display

what is currently happening to its clients. Since its clients change from time to time, only diverse subsets at single time stamps are essential. No temporal diversity is needed to be considered. Under this case we sequentially sample subsets by starting with π_1, then π_2 and so on.

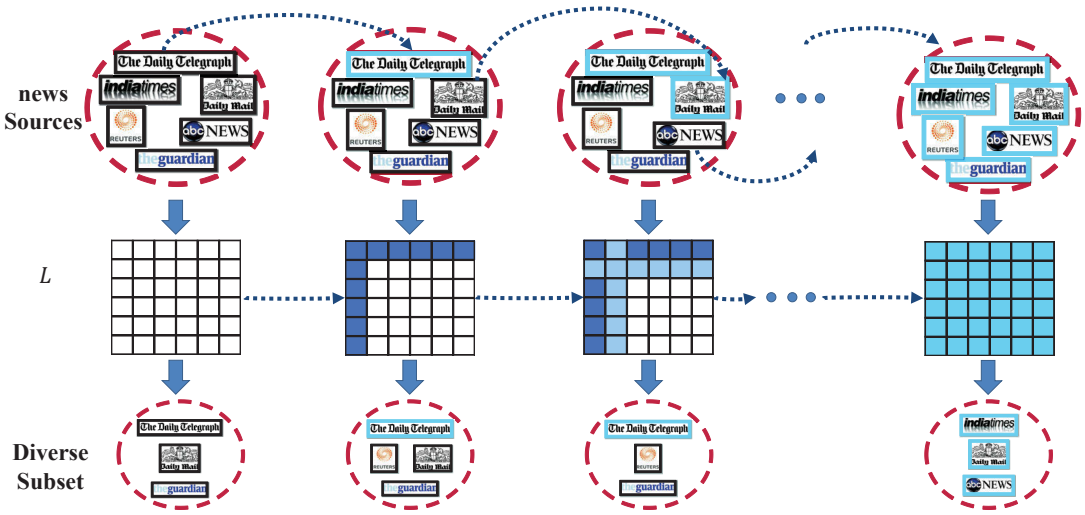
A time-varying structure assumes the neighbouring ground datasets are largely overlapped. In other words, the difference between them is subtle. Consequently, the probability measures built on the ground datasets are almost the same. Such a property makes it feasible to exploit sequential Monte Carlo (SMC) sampling technique to design a fast sampling algorithm for time-varying samples.

A time-varying determinantal point process (TV-DPP) integrates DPPs into the time-varying setting, whose marginal distribution at each time stamp t obeys Determinantal Point Processes (DPPs) (Kulesza and Taskar, 2012). Formally,

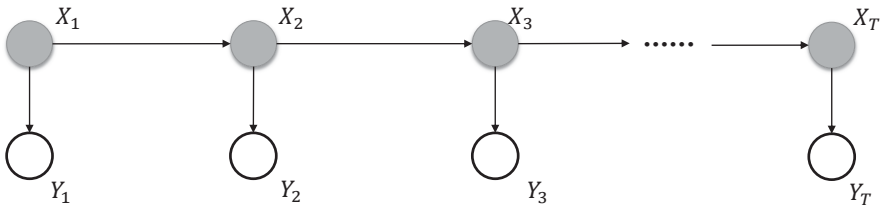
$$\pi(X = x_t) = \frac{\det(L_{t,x_t})}{\det(L_t + I_t)}. \tag{3.4}$$

Here, I_t is the identity matrix of the same dimension with L_t. $x_t \subseteq \mathcal{S}_t$, where $\mathcal{S}_t$ is the ground dataset at time stamp t. L_{t,x_t} is a submatrix of an L-ensemble kernel matrix L_t, whose indexes are restricted to the elements in x_t. Typically, $L_t = X_t X_t^T$ is constructed from data features $X_t = (x_{t1}, ..., x_{tN_t})^T, x_{ti} \in R^D$. Due to the time-varying structure, the ground dataset X_t differs from X_{t+1} by only a few elements. As a result, L_t slightly differs from L_{t-1} by a few rows and columns, which implies $\pi_{t-1} \approx \pi_t$. TV-DPPs are given an illustrative example in Figure 3.1a, and its graphical representation is shown in Figure 3.1b.

We apply the SMC framework introduced in Background section of this chapter to achieve the sequential sampling task. We make use of a fast DPP sampler (Kang, 2013) to collect samples at $t = 1$. We employ MCMC kernels proposed in (Kang, 2013) as the forward Markov kernels $\{K_t(x_{t-1}, x_t)\}$, where they are invariant to distributions π_t and make use of a standard Metropolis-Hasting al-



(a) Illustration of time varying DPPs with news dataset.



(b) Graphical representation for TV-DPPs.

Figure 3.1: Time Varying DPPs: In the first diagram, the first row represents the news updating process along time stamps. Six different news sources are schematically listed, i.e. 'The Daily Telegraph', 'Daily Mail', 'ABC NEWS', 'The Guardian', 'Reuters' and 'Indiatimes'. From time to time, only a small portion of the news sources are updated. The arrows make which news sources are updating clear: It starts at a news source with old news and points to the same source with new headlines -bordered in cyan - at the next time stamp. The second row shows the evolution of DPP marginal kernel L along with the news updates. The difference between two successive L-s is highlighted with different colours and is apparently tiny. The third row shows explanatory diverse subsets outputted by TV-DPPs. In the second diagram, the solid circles represent the observations, which correspond to the news dataset shown in the first row of the above figure, and the hollow circles represent the variables obeying the DPP distribution, one example of which can be found in the third row of the above figure. One important truth is that given the observations $\{X_1, X_2, ..., X_T\}$, the variables $\{Y_1, Y_2, ..., Y_T\}$ are independent.

gorithm. Regarding the unequal cardinality case, i.e., $|X_{t-1}| \neq |X_t|$, we can dynamically maintain the elements in samples during the Metropolis-Hasting algorithm: (1) when the cardinality increases, the selected elements to be included in a new sample should choose from the new, enlarged ground dataset; and (2) when the cardinality decreases, a new sample transited from the old one should first exclude the elements those do not appear in the new ground dataset. Based on the choice for forward Markov kernel, one good approximation for optimal backward Markov kernel L_{t-1} in Eq.(3.2) is given by (Del Moral et al., 2006)

$$L_{t-1}(x_t, x_{t-1}) = \frac{\pi_t(x_{t-1})K_t(x_{t-1}, x_t)}{\pi_t(x_t)}. \tag{3.5}$$

From the artificial joint distribution, one marginal distribution $\pi_t(x_t)$ can be computed by:

$$\pi_t(x_t) \propto \int \tilde{\gamma}_t(x_t)L_{t-1}(x_t, x_{t-1})dx_{t-1}. \tag{3.6}$$

Finally, the unnormalized incremental weight is derived as follow via substituting Eq.(3.5) and Eq.(3.6) into Eq. (3.2),

$$\widetilde{w}_t(x_{t-1}, x_t) = \frac{\tilde{\gamma}_t(x_{t-1})}{\tilde{\gamma}_{t-1}(x_{t-1})}. \tag{3.7}$$

Clearly the incremental weight relates to two successive marginal distributions π_{t-1} and π_t over the sampled particles $\{x_{t-1}^{(i)}\}_{i=1}^N$ at time stamp $t-1$. In conclusion, under the SMC framework (3.1), we sequentially move sampling particles forward with transitional kernel $K_t(x_{t-1}, x_t)$, and compute their weights at each time stamp t: $\{w_t^{(i)}\}_{i=1}^N$ with particle weights at time stamp $t-1$: $\{w_t^{(i)}\}_{i=1}^N$ as well as the incremental weights from time stamp $t-1$ to time stamp t: $\{\widetilde{w}_t(x_{t-1}^{(i)}, x_t^{(i)})\}_{i=1}^N$.

There are two time-consuming steps in the above sampling procedure. One

is the particle moving-forward step, which actually samples from a marginal DPP distribution, and a fast sampling algorithm (Kang, 2013) is applied as we have already introduced. The other step is responsible for updating the incremental weights, and it accomplishes this by computing the likelihood ratios between two consecutive DPP distributions. Due to the possibility of large-scale ground datasets, direct computing the likelihoods is impractical. We show how to make use of the proposed time-varying setting (slight changes between two successive distributions) to accelerate this computation. Specifying the general likelihoods with DPP distributions, the incremental particle weights for TV-DPPs are computed with the following determinantal ratio

$$\begin{aligned} \widetilde{w}_t(x_{t-1}, x_t) &= \frac{\det(L_{t,x_{t-1}})/\det(L_t + I)}{\det(L_{t-1,x_{t-1}})/\det(L_{t-1} + I)} \\ &\propto \det(L_{t,x_{t-1}})/\det(L_{t-1,x_{t-1}}). \end{aligned} \tag{3.8}$$

Since the term $\det(L_{t-1} + I)/\det(L_t + I)$ is constant over all particles $\{x^{(i)}\}_{i=1}^N$, we simply ignore it and focus on the computation for unnormalized incremental weights.

From time-varying structure, the difference between L_t and L_{t-1} has been small. Therefore, many elements of the set $\left\{\det\left(L_{t,x_{t-1}^{(i)}}\right)/\det\left(L_{t-1,x_{t-1}^{(i)}}\right)\right\}_{i=1}^N$ are equal to 1, which is a significant speed-up factor in our work. For the rest elements of $\left\{\det\left(L_{t,x_{t-1}^{(i)}}\right)/\det\left(L_{t-1,x_{t-1}^{(i)}}\right)\right\}$, whose values are not equal to 1, we compute them as follows. We drop particle indexes (i) for clarity. Let L^{cc} denote the shared submatrix between $L_{t,x_{t-1}}$ and $L_{t-1,x_{t-1}}$. Then, $L_{t,x_{t-1}}$ can be decomposed as $\begin{bmatrix} L^{cc} & L^{ct} \\ L^{tc} & L^{tt} \end{bmatrix}$, where L^{ct}, L^{tc} and L^{tt} are the rest submatrices for $L_{t,x_{t-1}}$. Similarly, we decompose $L_{t-1,x_{t-1}}$ with symbols $L^{cc}, L^{c,t-1}, L^{t-1,c}, L^{t-1,t-1}$. Let $k_1 = |L^{cc}|$ and $k_2 = |L^{tt}|$ (or $k_2 = |L^{t-1,t-1}|$) be the cardinalities for shared submatrix and dissimilitude respectively, and $k_1 \gg k_2$, based on the time-varying

structure. Then, the determinant ratio in Eq. (3.8) can be computed efficiently by applying the determinant formula of partitioned block matrices which is multiplicative in the Schur complement, without computing the nominator and denominator explicitly. By cancelling the shared $\det(L^{cc})$, the weight can be computed as follows.

$$\widetilde{w}_t(x_{t-1}, x_t) \propto \frac{\det(L^{tt} - L^{tc}(L^{cc})^{-1}L^{ct})}{\det(L^{t-1,t-1} - L^{t-1,c}(L^{cc})^{-1}L^{c,t-1})}, \tag{3.9}$$

where $L^{tt}, L^{cc}, L^{tc}, L^{t-1,t-1}, L^{t-1,c}$ are submatrix of $L_{t,x_{t-1}}$ or $L_{t-1,x_{t-1}}$. It is easily observed that the complexity has reduced from $\mathcal{O}((k_1 + k_2)^3)$ to k_2^3. However, when the changes occur between time intervals $t-1$ and t become large, namely, k_2 growing up, the complexity will increase at an exponential rate. The worst situation for our setting is an extreme case where two contiguous data sets are completely non-overlapping, and then our model degenerates to the case of individually sampling DPP.

As shown in the above formula, the computation of incremental weight relates to matrix addition and multiplication, as well as the inverse for the shared submatrix $(L^{cc})^{-1}$. Based on our assumption - neighbouring shared submatrix (e.g. L^{cc}_{t-1} and L^{cc}_t) are slightly different with several elements - we are able to efficiently update the inverse of shared submatrix by repeatedly applying both matrix block inverse formula and matrix inverse lemma. We elaborate this update step by step. Suppose the shared submatrix of L^{cc}_{t-1} and L^{cc}_t is L^{comm}, and the unshared ones are A, A^T, B. There are two cases for updating the inverse from L^{cc}_{t-1} to L^{cc}_t. One case is when elements are added, i.e., $L^{comm} = L^{cc}_{t-1}$. Therefore, $L^{cc}_t = \begin{bmatrix} L^{comm} & A \\ A^T & B \end{bmatrix}$. Since $(L^{comm})^{-1}$ is currently known, the inverse of L^{cc}_t can be expanded by first applying matrix block inverse formula (Golub

and Van Loan, 2012):

$$
\begin{aligned}
(L_t^{cc})^{-1} &= \begin{bmatrix} L^{comm} & A \\ A^T & B \end{bmatrix}^{-1} \\
&= \begin{bmatrix} (L^{comm} - AB^{-1}A^T)^{-1} & -(L^{comm})^{-1}A(B - A^T(L^{comm})^{-1}A)^{-1} \\ -B^{-1}A^T(L^{comm} - AB^{-1}A^T)^{-1} & (B - A^T(L^{comm})^{-1}A)^{-1} \end{bmatrix}.
\end{aligned} \tag{3.10}
$$

It is easily observed that most inverses are either known (i.e., $(L^{comm})^{-1}$) or easily to be computed due to its small sizes (e.g., $(B - A^T(L^{comm})^{-1}A)^{-1}$), except the inverse of a correction form of L^{comm}. Here matrix inverse lemma is applied to convert computing the inverse of a correction form to computing the correction of the original matrix, namely,

$$
\begin{aligned}
&(L^{comm} - AB^{-1}A^T)^{-1} \\
=&(L^{comm})^{-1} + (L^{comm})^{-1}A(B - A^T(L^{comm})^{-1}A)^{-1}A^T(L^{comm})^{-1}
\end{aligned} \tag{3.11}
$$

The other case is when elements are deleted, i.e., $L^{comm} = L_t^{cc}$, and $L_{t-1}^{cc} = \begin{bmatrix} L^{comm} & A \\ A^T & B \end{bmatrix}$. Since $(L_{t-1}^{cc})^{-1}$ is known, we set $(L_{t-1}^{cc})^{-1} \equiv \begin{bmatrix} E & F \\ F^T & G \end{bmatrix}$. Equally,

$$
\begin{bmatrix} L^{comm} & A \\ A^T & B \end{bmatrix} = \begin{bmatrix} E & F \\ F^T & G \end{bmatrix}^{-1}. \tag{3.12}
$$

Again, applying matrix block inverse formula to the left hand of above equation, we obtain $(E - FG^{-1}F^T)^{-1} = L_t^{cc}$, and therefore,

$$
(L_t^{cc})^{-1} = (E - FG^{-1}F^T). \tag{3.13}
$$

Note: Usually the variance of the incremental weight has an increasing tendency

along the timeline, resulting in a potential degeneracy of the particle approximation. Routinely, the degeneracy is measured by the Effective Sample Size (ESS) criterion (Sahlin, 2011). When the ESS is smaller than a predefined threshold $\alpha \cdot N$ (α is a predefined ratio), we re-sampling the particles with a multinomial distributions parameterized by the normalized particle weights. To make the re-sampled particles more diverse, we further randomly move the equally weighted particles with an MCMC kernel of stationary distribution π_t (Gilks and Berzuini, 2001). The whole process of SMC is illustrated in Figure 3.2.

To sum up, we start the SMC process by initializing particles $\{x_1^{(i)}\}$ of marginal distribution π_1. We sample $\{x_1^{(i)} \sim \pi_1(X = x_1^{(i)})\}_{i=1}^N$ using a fast DPPs sampling algorithm proposed in (Kang, 2013). Then, at each time t, we update these samples from $\{x_{t-1}^{(i)}\}_{i=1}^N$ using the above customized SMC sampling scheme. The proposed fast sampling algorithm for time-varying DPPs (TV-

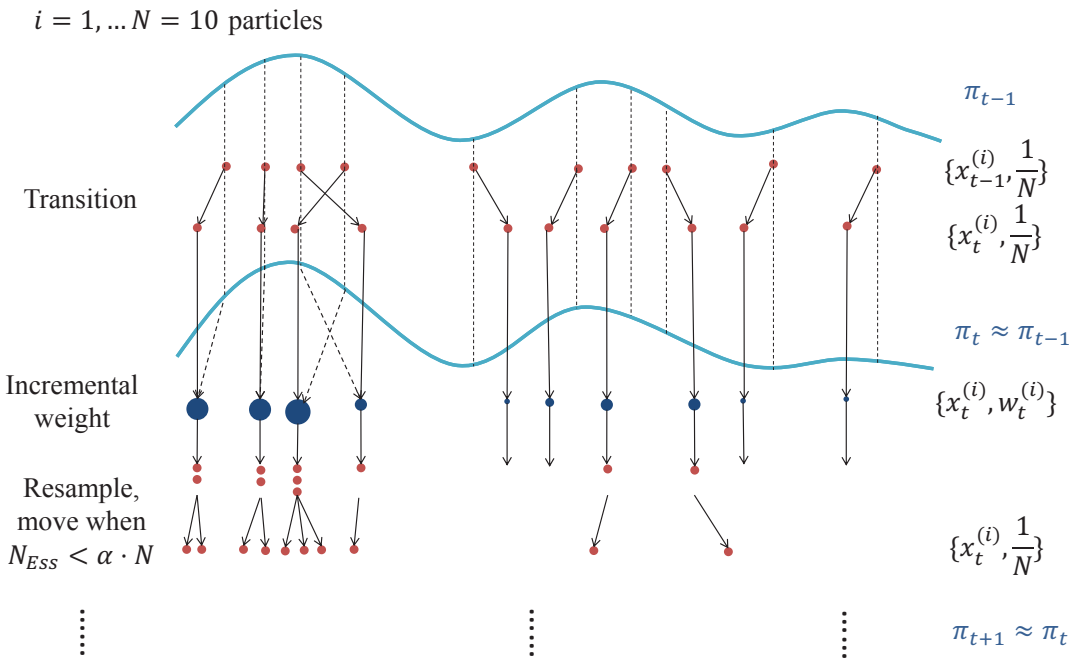


Figure 3.2: Illustration of Sequential Monte Carlo: At time stamp $t-1$, 10 particles in red with equal weights are given, i.e. $\{x_{t-1}^{(i)}\}_{i=1}^{10}$. At this stage, two computations will be done - One is computing the incremental weights; the other is computing the particles for the next time stamp. For the incremental weight of each particle at time $t-1$, according to Eq. (3.7), it is simply the likelihood ratio between time stamps t and $t-1$. The corresponding relationship is denoted by the dashed line connecting two neighbour distributions and the weight for each particle is illustrated by size of blue solid circle. For the particle's location at next time stamp t, usually, a Markov transition kernel is used to qualify the transition job between two slightly different neighbour distributions. The transition relationship is indicated by solid line with arrow. For the particle's weight at time stamp t, it is gained by multiplying the weight at time stamp $t-1$ by the incremental weight. To alleviate the degeneracy of the algorithm which is measured by effective sample size (ESS), a re-sampling step is applied when $N_{ESS} < \alpha \cdot N$. High weighted particles will re-birth as several equal weighted particles, while particles with low weights may disappear. To increase the samples' diversity, a move step is followed. Once particle' locations and weights at t are prepared, it will recursively carry out the whole above procedure.

DPPs) is shown in Algorithm 2 [1, 2].

Algorithm 3.1: Fast sampling for TV-DPPs

Data: $\{L_i\}_{i=1}^{t}$

Result: $\{\{x_k^{(i)}, W_k^{(i)}\}_{i=1}^{N}\}_{k=1}^{t}$

$\{x_1^{(i)} \sim \det(L_{1,x_1^{(i)}})\}_{i=1}^{N}$ by (Kang, 2013) or (Kulesza and Taskar, 2012) ;

$\{W_1^{(i)} = 1/N\}_{i=1}^{N}$;

$k = 1$;

repeat

k = k+1 ;

$\{\widetilde{w}_k(x_{k-1}^{(i)}, x_k^{(i)}) = \det(L_{k,x_{k-1}^{(i)}}) / \det(L_{k-1,x_{k-1}^{(i)}})\}_{i=1}^{N}$;

$\{W_k^{(i)} = W_{k-1}^{(i)} \widetilde{w}_k^{(i)} / \sum_{i=1}^{N}(W_{k-1}^{(i)} \widetilde{w}_k^{(i)})\}_{i=1}^{N}$;

$\{x_k^{(i)} \sim K(x_{k-1}^{(i)}, x_k^{(i)})\}_{i=1}^{N}$;

$N_{k,ESS} = \{\sum_{i=1}^{N}(W_k^{(i)})^2\}^{-1}$ (Sahlin, 2011);

if $N_{k,ESS} < \alpha \cdot N$ **then**

$\{x_k^{(i)} \sim Multi(W_k^{(1)}, ..., W_k^{(N)})\}_{i=1}^{N}$;

$\{W_k^{(i)} = 1/N\}_{i=1}^{N}$;

move $\{x_k^{(i)}\}_{i=1}^{N}$ by π_k invariant MCMC kernel $K_{\pi_k}(x_k^{(i)}, \cdot)$ (Kang, 2013);

until $k = t$;

3.5 Experimental Results

In this section, we report experimental results of our proposed fast sampling for TV-DPPs when applied to a real-world news dataset and the Enron Corpus (Diesner and Carley, 2005).

[1] We use $W_k^{(i)}$ and $\widetilde{w}_k^{(i)}$ to replace $W_k(x_{1:k}^{(i)})$ and $\widetilde{w}_k(x_{k-1}^{(i)}, x_k^{(i)})$ for simplicity

[2] $Multi(W_k^{(1)}, ..., W_k^{(N)})$ stands for the multinomial distribution parameterized by $\{W_k^{(i)}\}_{i=1}^{N}$

Table 3.1: Summary of news categories for different news media sources

	A	D1	D2	G	I	R
sport	√	√	√	√	√	
news	√	√	√		√	
politics	√			√		√
business	√	√	√		√	√
sciencetech		√				
femail		√				
tvshowbiz		√				
technology	√		√		√	√
global				√		
world				√		√
entertainment					√	
markets						√
others	√	√	√		√	

3.5.1 News recommendation

We collect the news dataset from six news websites with different topics as resources during the period of 12: 16am 2 Jun, 2014 ∼8: 13am 5 Jun, 2014. The websites and topics are summarized in Table 3.1 [3] . The topic titles are directly named from the categories of the news website.

There are 33 topics in each time stamp. At the beginning $t = 1$, we collect each resource's latest news corpora $\mathcal{S}_{t=1} = \{d_{t=1,k}\}_{k=1}^{33}$. At next time stamp, there are n topics are updated. We recorded the latest news corpora for these n topics as well as the news for unchanged topics as $\mathcal{S}_{t=2} = \{d_{t=2,k}\}_{k=1}^{33}$. In this manner, we record the sequential news corpora $\mathcal{S}_{t:T} = \{d_{t,k}\}_{k=1}^{33},_{t=3}^{T}$. Herein, we choose $n = 5, T = 1000$ for experimental demonstration.

We extract normalized TF-IDF (short for Term Frequency-Inverse Document Frequency) feature vectors (Wu and Luk., 2008; Xuan et al., 2015) to represent news articles. All news corpora $\mathcal{S}_{1:T}$ are employed to compute the IDF. We

[3] A-ABC, D1-Dailymail, D2-Dailytelegraph, G-Guardian, I-Indiantimes, R-Reuters

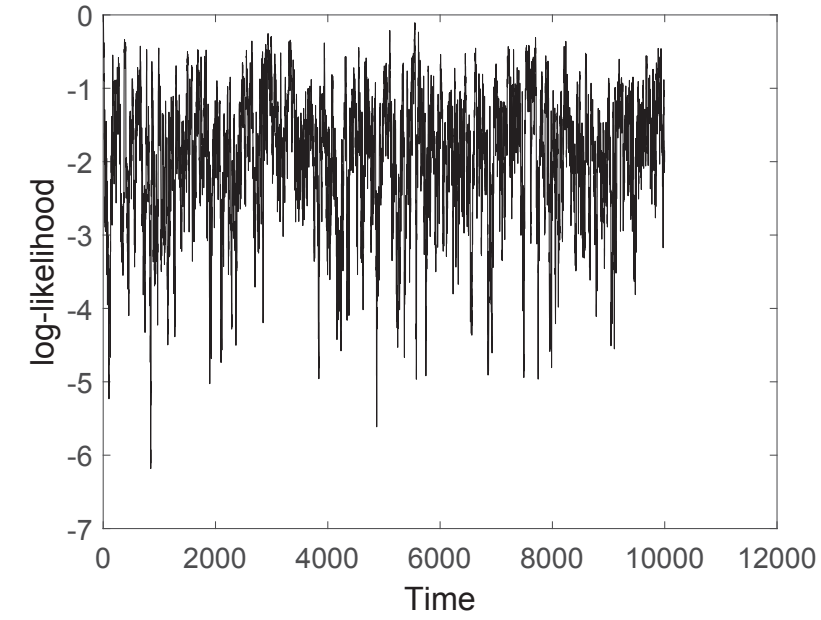
apply cosine similarity (Gillenwater et al., 2012*a*) to construct kernel matrix L. $L(d_i, d_j) = \text{cos-sim}(d_i, d_j)$ and $\text{cos-sim}(d_i, d_j)$ is defined as

$$\frac{\sum_{w \in W} \text{tf}_{d_i}(w)\, \text{tf}_{d_j}(w)\, \text{idf}^2(w))}{\sqrt[2]{\sum_{w \in W} \text{tf}^2_{d_i}(w)\, \text{idf}^2(w)}\, \sqrt[2]{\sum_{w \in W} \text{tf}^2_{d_j}(w)\, \text{idf}^2(w)}}, \tag{3.14}$$

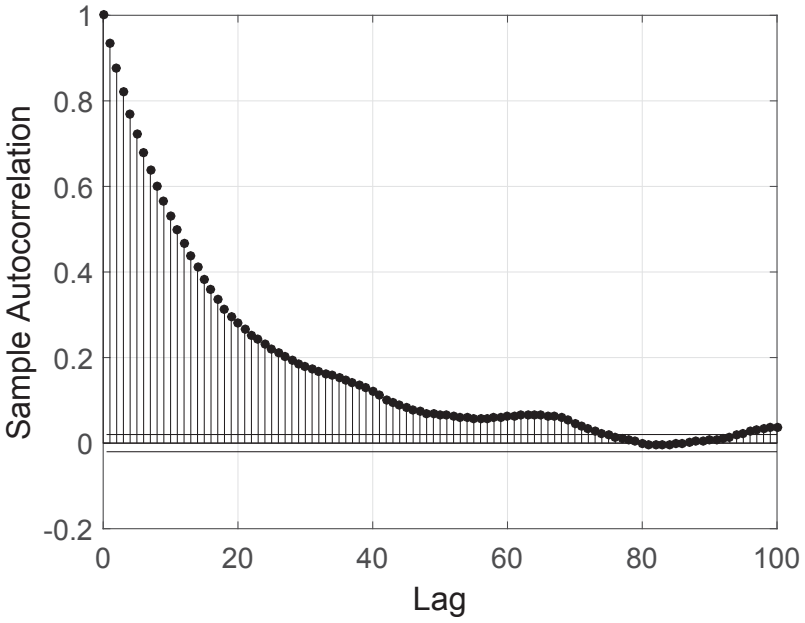
where W is a subset of the words found in the two documents. The TF-IDF is usually sparse, thus the distances amongst TF-IDFs are quite small and will lead to poor diversity. We re-feature each news article referred to as $\tilde{d}_i$ with binary vector where the j-th coordinate of d_i is 1 if news article j is amongst the N_{nei} nearest neighbors of news article i according to the cosine similarities, and 0 otherwise. The L-ensemble kernel matrix is computed by $L(i, j) = e^{-\alpha \cdot \text{cos-sim}(\tilde{d}_i, \tilde{d}_j)}$. We fix $N_{nei} = 700$ and $\alpha = 1$ throughout the experiment.

Sampling Analysis at $t = 1$

We apply the Markov Chain Monte Carlo (MCMC) method - a fast DPPs sampling algorithm - to initialize our particles. The mixing of likelihood of the sampled subsets is shown in Figure 3.3a. In our experiment, the burn-in period is used to wait for the mixing of the Markov chain. At the beginning, the starting subset is smaller (such as containing one single element) than the subset when it is mixed. In this case, the likelihood of the starting subset is usually high. In the following several iterations, it is highly possible that more elements will be added into the starting subset, which will lead to lower likelihood. This phenomenon leads to decreasing tendencies of sequential likelihoods, which matches the beginning curve in Figure 3.3a. Afterwards, when it approaches to mixing, the Markov chain will collect subsets by alternative operations of adding and deleting elements and the sequential likelihood is expected in oscillation tendency, as shown in Figure 3.3a. We set the burn-in period as $10,000$ in our experiment.



(a) Mixing of likelihood



(b) ACF

Figure 3.3: Analysis for fast DPPs sampling algorithm at $t = 1$

The consecutive samples are not independent due to the Markov property. We compute the AutoCorrelation Function (ACF) parameterized by lag k to determine the interval between two independent samples x_t, x_{t+k}. We apply indirect measure Q to define the autocorrelation coefficient (Sandvik, 2013) at lag k:

$$r_Q(k) = (< Q_t Q_{t+k} > - < Q_t >^2)/(< Q_t^2 > - < Q_t >^2). \tag{3.15}$$

And, in our experiment, we set the measurement Q_t for sample x_t (sampled at time stamp t) as its diversity likelihood, namely

$$Q_t = \det(x_t). \tag{3.16}$$

The plot of the autocorrelation function is shown in Figure 3.3b. As expected, the sample autocorrelation coefficient is decreasing with the increasing of lag k. When the $r_Q(k_{ACF})$ is below τ, we claim that the sampled subset x_t is independent with subset $x_{t+k_{ACF}}$. Here, we set $\tau = 0.02$ and $k_{ACF} = 80$. We collect $N = 100$ independent samples by extracting one sample subset every k_{ACF} iterations as initial particles for TV-DPPs sampling.

In the next section, we analyse the performance of our proposed fast TV-DPPs sampling with both qualitative and quantitative demonstrations.

Quantitative Analysis

We analyse the diversity and time complexity of our algorithm (SMC-DPPs) by comparing to the baseline algorithm, namely, separate fast DPPs (sep-DPPs) sampling algorithm at each time stamp.

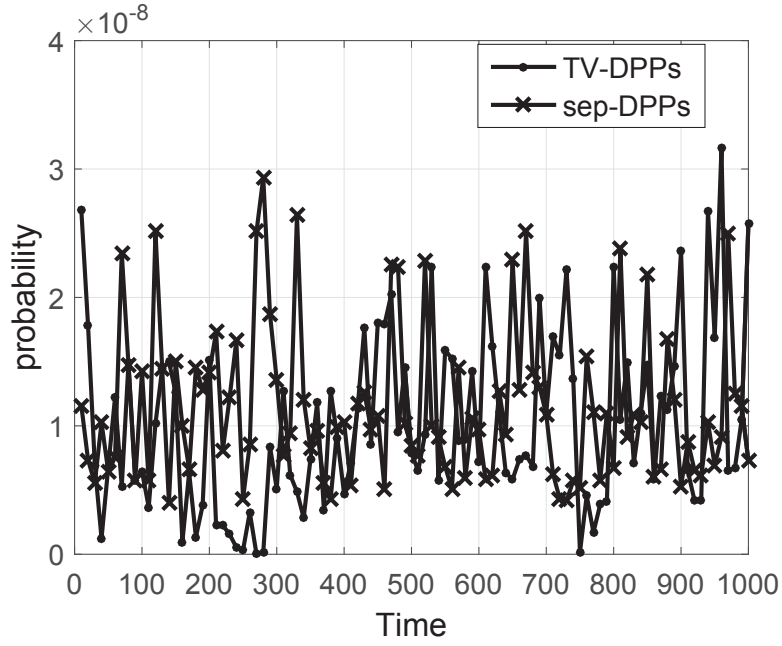
Accuracy Analysis: We compare TV-DPPs and the baseline sep-DPPs with regards to diverse probabilities and cosine similarities of subsets that ap-

proximate the maximal mode of DPP distributions. At each given time stamp, for TV-DPPs, we measure each particle with ground-truth DPP distribution, and find the particle with the maximal diverse probabilities. We do the same selection for sep-DPPs' samples. The comparison result is shown in Figure 3.4a. For clear visualization, here we illustrate 100 time stamps by taking one every 10 time stamps. The curves declare that our speedup TV-DPPs sampling algorithm is comparable to sep-DPPs in terms of diverse accuracy.

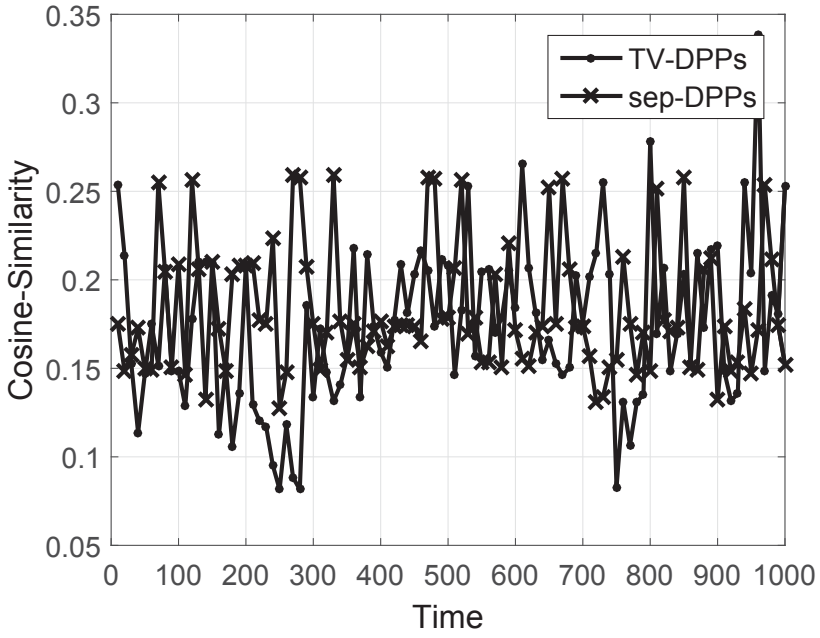
To further demonstrate the accuracy of our algorithm, we also compute the cosine similarities of news articles in the subsets owning the maximal diverse probabilities at each given time stamp. The cosine similarities of pairwise news articles are computed with TF-IDF feature vectors. The results for both TV-DPPs and sep-DPPs are plotted in Figure 3.4b. Clearly, our TV-DPPs algorithm performs comparably. These results illustrate the effectiveness of our algorithm to some extent.

Time Complexity Analysis: We also compute the time complexity of TV-DPPs and sep-DPPs. We directly apply the mixing time in (Kang, 2013) for burn-in time of every fast DPPs sampling. For sep-DPPs, N independent samples of diverse subsets are sampled by collecting one sample per k_{ACF} steps at one given time stamp. Then, the total time cost by the T-length sequence is: $T \times [|S| \log(|S|/\epsilon) + N \times k_{ACF}]$, where $S = 33$ in our case. For TV-DPPs, at the beginning $t = 1$, one sep-DPPs procedure is applied and N independent particles are prepared. After that, each particle costs constant time to move to the next time stamp. The total time complexity is summarized as $|S| \log(|S|/\epsilon) + N \times k_{ACF} + N \times (T - 1)$. Whether the algorithm is less time-consuming depends on $|S|$, N, k_{ACF}, as well as T. The time costs in log-space for comparison of sep-DPPs and TV-DPPs in our experiment are plotted in Figure 3.5.

At the beginning point, TV-DPPs (in blue) costs the same time as sep-



(a) Diverse probability



(b) Cosine similarity

Figure 3.4: Diversity comparison of news subsets selected by sep-DPPs and TV-DPPs with regard to both diverse probability and cosine similarity.

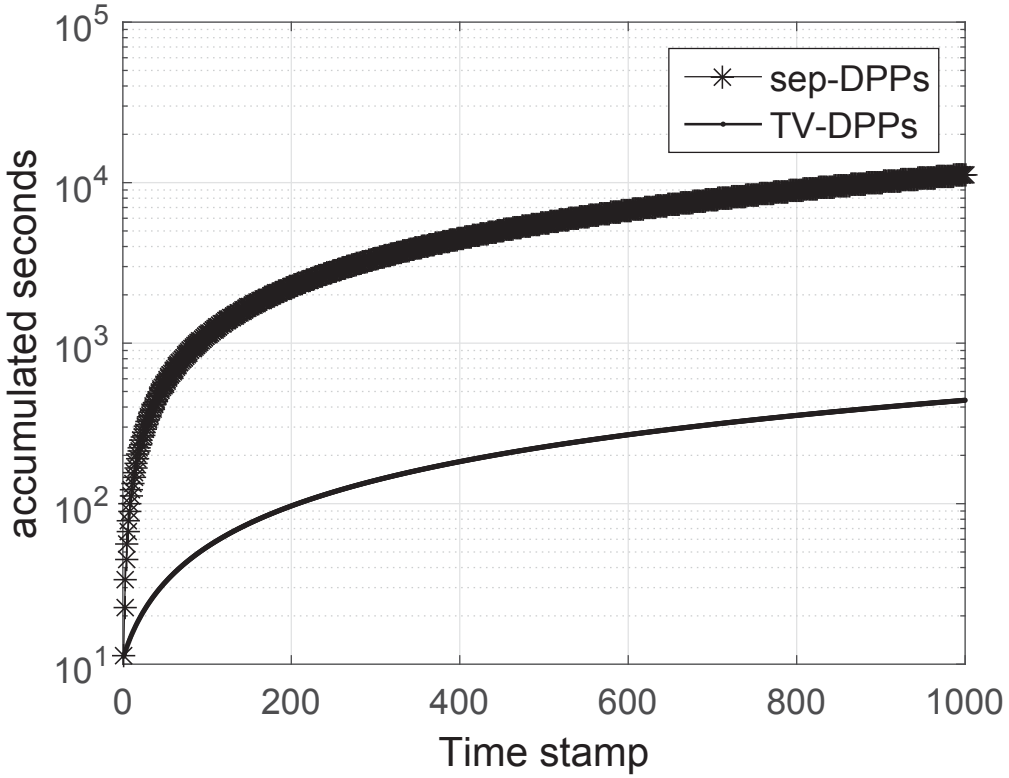


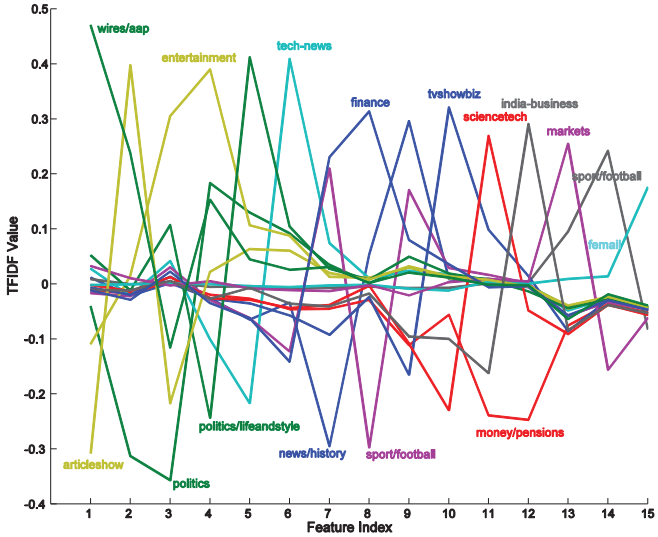
Figure 3.5: Time cost comparison between sep-DPPs and TV-DPPs for news recommendation. X-axis indicates the time stamps, while Y-axis shows the accumulated seconds over time.

DPPs (in red). With the t increasing, TV-DPPs algorithm performs much more efficiently than the sep-DPPs scheme.

Qualitative Analysis

To give an intuitive sense of diversity of sampled news article subset from the TV-DPPs samples, one news article subset randomly selected at $t = 12$ is visualized in Figure 3.6. Since news articles with high-dimensional TF-IDF features are difficult to visualize, here PCA was applied to reduce dimensions. There are 16 articles in total, thus the maximal dimension after PCA is 15. A parallel coordinates plot with 15 coordinates is displayed in Figure 3.6a. Different coloured curves represent different news articles, and their corresponding news titles are listed in Figure 3.6b. There are two strong information which strongly suggest the diversity property of the selected news article subset. First, any two different coloured curves are not overlapped, as clearly seen in Figure 3.6a. Second, there are no two news titles listed in Figure 3.6b textually the same. Obviously, this result qualitatively demonstrates the effectiveness of TV-DPPs.

We also plot the sequential sampling results from both TV-DPPs and sep-DPPs to give a holistic analysis in Figure 3.10. The X-axis represents time stamps, while the Y-axis denotes the indexes of news articles. As a particle filtering alike algorithm, TV-DPPs algorithm gives smoother samples of news articles along the timeline (shown in Figure 3.10a) than samples of the independent selection of sep-DPPs scheme (shown in Figure 3.10b). Thus, it can be concluded that the TV-DPPs algorithm gives more consistent subsets along time stamps.

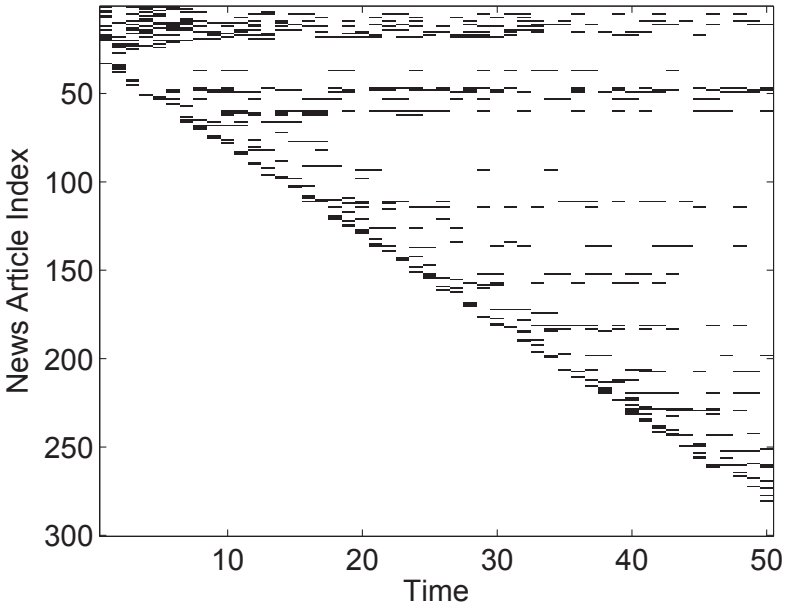


(a) A parallel coordinates plot of a set containing sixteen news articles is displayed, with X-axis representing the feature indices and Y-axis the coordinate values. Each curve represents one news article, and the non-overlapping phenomenon validates the diversity of the news article set.

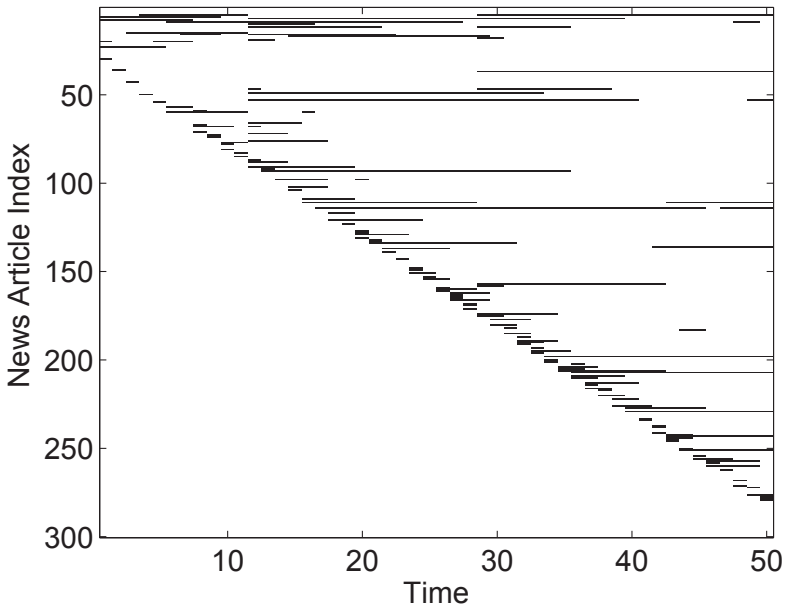
1. news/history: Tiananmen Square anniversary: Questions and Answers
2. politics: Senator plans bill to offer military veterans option on medical care
3. money/pensions: General Boss: Ban insurers as pensions advisers...
4. tech-news: Infosys investors lose faith in Narayana Murthy 2.0
5. sport/football: Rickie Lambert says his mum and dad cried at his return ...
6. entertainment: Shah Rukh's kids watch us: JusReign and Rupan Bal
7. sport/football: Graham Wallace to be grilled by North American-based ...
8. tvshowbiz: No wonder she's smiling! Michael Clarke's wife Kyly beams ...
9. politics/lifeandstyle: We should act against fast-food producers, not fat people
10. sciencetech: Look out gardeners! It's boom time for bugs that want to eat ...
11. femail: Girl who grew up with giants: Stalin, Chaplin, Lawrence of Arabia...
12. markets: JGBs slip as Japan capex data helps boost stocks
13. articleshow: Johnson's bouncer most pivotal of three turning points
14. india-business: Govt wants banks to fund local M&As
15. finance: Government should boost SMEs with NI breaks, says Lord Bilimoria
16. wires/aap: Crows expect injured Jacobs to face Freo

(b) Sixteen news titles with topic tags are listed. Each news title corresponds to one curve in the above plot with the same colour. This textual information further confirms the diversity of the news article set.

Figure 3.6: Demonstration of diverse subset of news articles sampled by TV-DPPs.



(a) Sequential diverse subsets of news dataset sampled by sep-DPPs.



(b) Sequential diverse subsets of news dataset sampled by the proposed TV-DPPs.

Figure 3.7: Demonstration of news topic drift: Comparing sequential subsets from TV-DPPs with the subsets from sep-DPPs, it extracts a more smooth news evolvement.

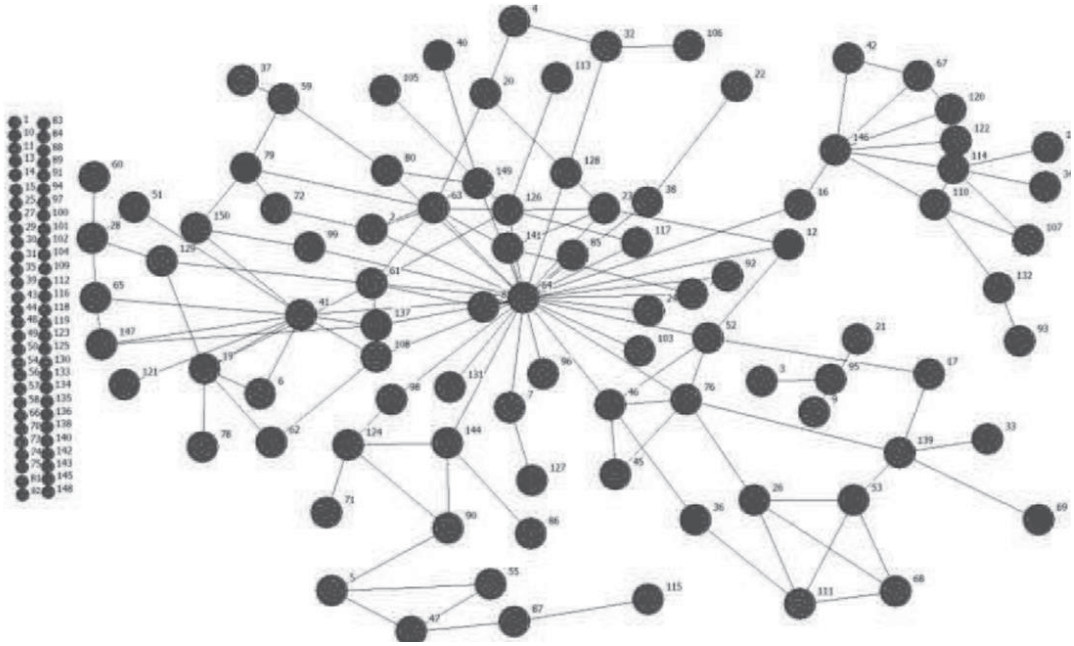


Figure 3.8: One example of Enron communication network.

3.5.2 Enron corpus

The Enron email corpus [4] contains a large set of email messages amongst the employees of the Enron corporation, and appeals to many researchers (The titles and indexes of its employees are also publicly accessible [5,6]). Various aspects of the Enron corpus have been investigated such as natural language processing [7] (Klimt and Yang, 2004), email classification [8], and quantitative analysis of social networks generated from the corpus (Zhong et al., 2014)(Mcauley and Leskovec, 2014) (Iwata et al., 2012)(Shetty and Adibi, 2004). Here we focus on discovering the dynamic event developments from its communication network.

The communication network is extracted as nodes and edges, where the nodes represent the employees in Enron and the edges represent undirected correspon-

[4] http://www.enron-mail.com/email/
[5] http://enrondata.org/assets/edo_enron-custodians-data.html
[6] http://enrondata.org/assets/edo_enron-custodians.txt
[7] http:www.ceas.cc/papers-2004/168.pdf
[8] http://www.cs.umass.edu/~ronb/

dences from senders to recipients. There are 150 nodes in total. Note that to avoid asymmetric distance between two correspondents, we simply drop the direction of each edge, and we also drop the communication frequencies between the correspondents. Email correspondences exclusively to or from addresses outside the Enron or Andersen domains are removed from our network. As introduced in (Diesner et al., 2005), the number of emails and people involved in email communication between May 1999 and March 2002 were relatively high, therefore we selectively focus on the period from August 1, 2001 to December 1, 2001. At each day, a snapshot of the communication network with fixed 150 nodes and edges is extracted from the correspondences between the 150 employees. Rather than from one single day's correspondence, here the edges are extracted from the past accumulating 30 days. It will not only artificially create overlaps between subsequent time stamps, but also stabilize the communication network at each time stamp. This will facilitate the detection of dynamic events happening in the Enron incorporation along the time line.

One example of the extracted communication network is shown in Figure 3.8. It is drawn by the NetDraw software provided by (Borgatti, 2002). The two columns of separate points demonstrate the employees who did not send or receive any email messages to or from other Enron employees during a period of around one month. The connected points show a hierarchical structure, i.e., the peripheral points tend to communicate with second-peripheral points, while the points in the centre of the communication network mutually connects each other and tend to have higher communication degrees.

We prepare the L-ensemble kernel matrix for DPP at each time stamp by a Gram matrix construction. First we calculate shortest path distances for every pairwise nodes based on the network structure. Then, the distance measurement is transformed into similarity through an exponential operator. Finally, we in-

troduce degree of each node, which is the number of edges incident to the node, as a qualification term. The degree exhibits how many employees this node is communicating with, which naturally qualifies the importance of the representing employee in Enron incorporation. Formally, the entry of L_{ij} in L-ensemble kernel matrix is computed by

$$L_{ij} = d_i \times \exp\{-\lambda \cdot dist(i,j)\} \times d_j, \tag{3.17}$$

where d_i represents the degree of node i, $dist(i,j)$ is the shortest path distance between the two nodes i and j, and λ is the parameter to adjust the relative similarity.

Quantitative Analysis

Because the sampling analysis at $t = 1$ is quite similar to the analysis for news recommendation and is not the main concern of this chapter, we simply skip this step and directly proceed to analyse the efficiency of the proposed TV-DPPs. Like the experimental setting for news recommendation, we compare the proposed model with baseline sep-DPPs, which samples subset at each time stamp by performing separate DPP sampling algorithms. We sequentially sample subsets for a period of one year starting from January 1, 2011 and the demonstrating chart is shown in Figure 3.9.

At the beginning $t = 1$, the two different schemes take exactly the same time to do diverse subset sampling, since sep-DPPs apply the fast MCMC sampling algorithm to do sampling for each times stamp while TV-DPPs apply the same fast MCMC sampling algorithm to initialize particles for the subsequent SMC sampling scheme. After that, sep-DPPs scheme repeats the same sampling algorithm which always cost around 57 seconds for a 150 nodes communication

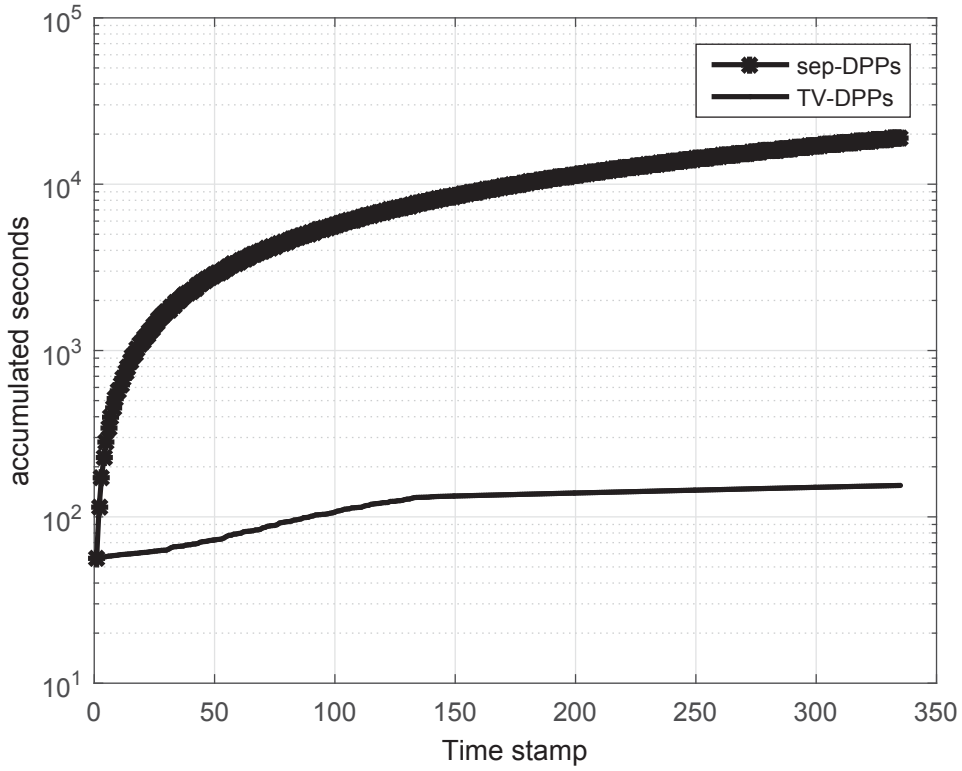


Figure 3.9: Time cost comparison between sep-DPPs and TV-DPPs for Enron communication network. X-axis represents the day stamps, while Y-axis shows the accumulated seconds.

network. But, for the proposed TV-DPPs scheme, it costs less than 0.2 seconds to get a subsequent diverse subset. In a long-time run, our proposed TV-DPPs scheme significantly improves the efficiency of time-varying diverse subset sampling task. The sampling results are explained in next section.

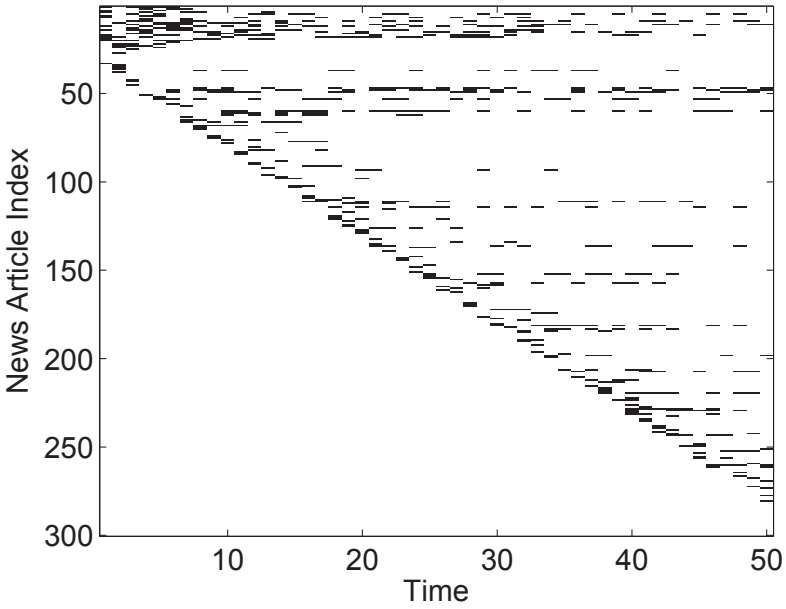
Qualitative Analysis

The sequential sampling results of both sep-DPPs and TV-DPPs along the timeline from August 1, 2001 to December 1, 2001 are plotted in Figure 3.11.

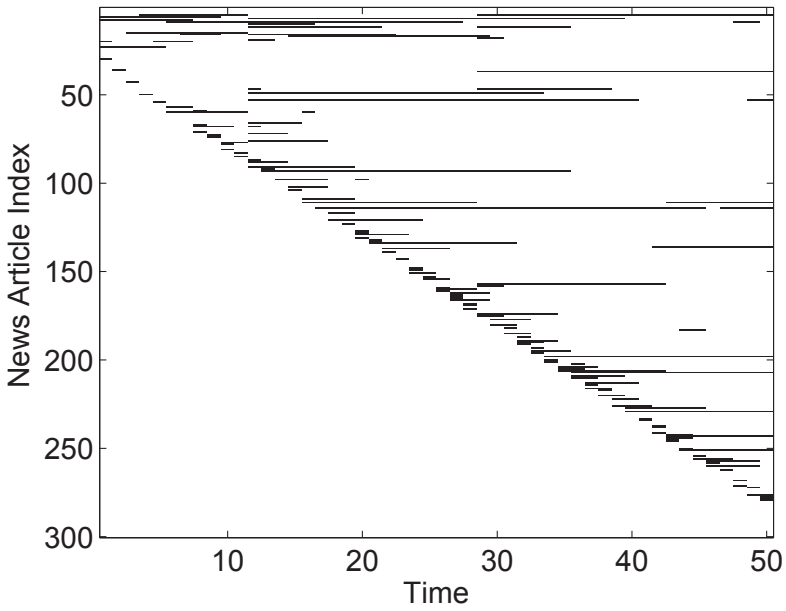
From Figure 3.11a, at each day, a diverse subset out of a 150 employees is coloured in purple. For the holistic time period, it can be seen that points are randomly scattered and it is hard to discover any events happening within the company. Comparatively, from Figure 3.11b, it can be easily seen that three clearly separated stages are obtained with smooth diverse subsets in each stage. Two separating points - one is around 2001-Oct-01 and the other around 2001-Nov-10 - are easily noticeable. Actually, these three stages coincide with different important events happening in Enron incorporation, and they are (Diesner et al., 2005): 1). Before October 2001, the communication topic is not indicated in related references. 2). In October 2001, Andersen criminally instructed Enron to destroy any documentation related to the circumstance. 3). In December 2001, Enron became insolvent and was forced to file for bankruptcy.

Although the TV-DPPs scheme is supposed to smoothly sample diverse subsets along time stamps, the resampling step in the time-varying DPP sampling procedure is designed to suddenly change the samples of diverse subset when these samples no longer correctly represent the present DPP distribution. This is the reason why three clearly different stages appear in the sampling results of the proposed TV-DPPs.

We detail each stage with a concrete time stamp of diverse sample shown

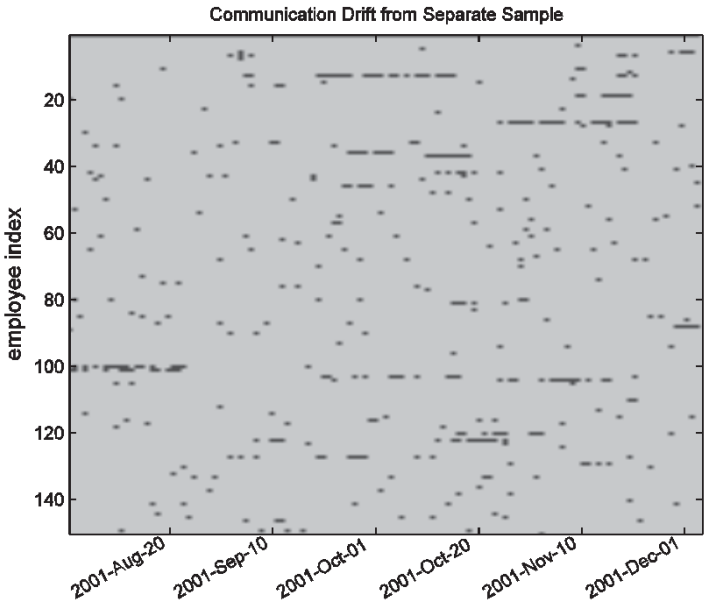


(a) Sequential diverse subsets of news dataset sampled by sep-DPPs.

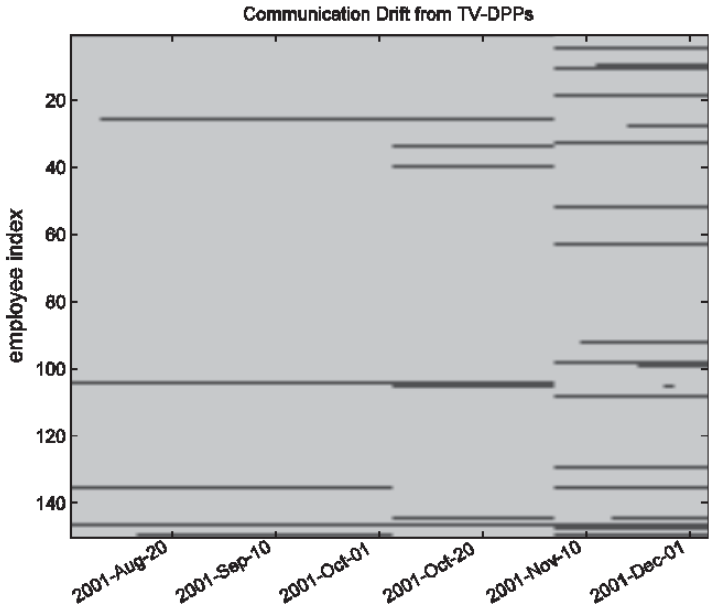


(b) Sequential diverse subsets of news dataset sampled by the proposed TV-DPPs.

Figure 3.10: Demonstration of news topic drift: Comparing sequential subsets from TV-DPPs with the subsets from sep-DPPs, it extracts a more smooth news evolvement.



(a) Sequential diverse subsets of Enron employees sampled by sep-DPPs.



(b) Sequential diverse subsets of Enron employees sampled by the proposed TV-DPPs.

Figure 3.11: Demonstration of Enron communication drift: No communication patterns or events can be detected from the above figure. However, three different stages are clearly obtained with smoothness at each stage. Two separating points - one is around 2001-Oct-01 and the other around 2001-Nov-10 - coincide with two important turn points for Enron incorporation.

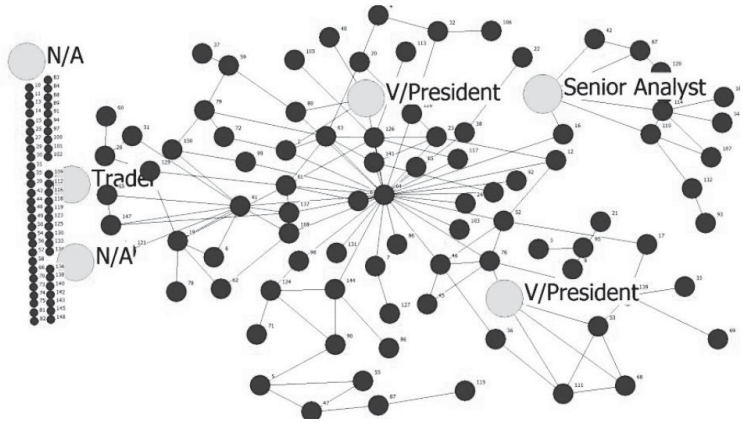
in Figure 3.12a-d, i.e., September 10, 2001 for the first stage, October 20, 2001 for the second stage and November 20, 2001 for the third stage respectively. In each sub-figure, there are 150 nodes in total, and the subset highlighted by bigger cyan circles is one diverse subset sample outputted from the proposed algorithm.

From the three examples of diverse subset, we claim that the DPP model at each time stamp prefers to select 'leader' points of peripheral branch, rather than points owning highest degrees. This is due to the combination of points' degree measurement and pairwise shortest distance measurement for the construction of the kernel matrix for sequential DPPs. In Figure 3.12a, the communication amongst Enron employees is not that active, and a diverse subset containing only 6 nodes is sampled and the related titles are N/A, N/A, Trader, Vice President, Vice President, Senior Analyst respectively. In Figure 3.12b, maybe due to Andersen's criminal command, the email communication amongst Enron employees was becoming more active during October in 2001, and a larger diverse subset is sampled by the proposed scheme. The position titles for the sampled nodes are N/A, Trader, Manager Director of UK, Vice President, Employee, Senior Analyst, and Senior Specialist respectively. It can be seen that more employees in important positions are involved. Finally, in Figure 3.12c, along with the event of filing for bankruptcy, more and more employees from different branches of Enron incorporation are involved and selected, and their titles are N/A, Manager, Employee, Director, CEO, Vice President, President and Senior Analyst respectively.

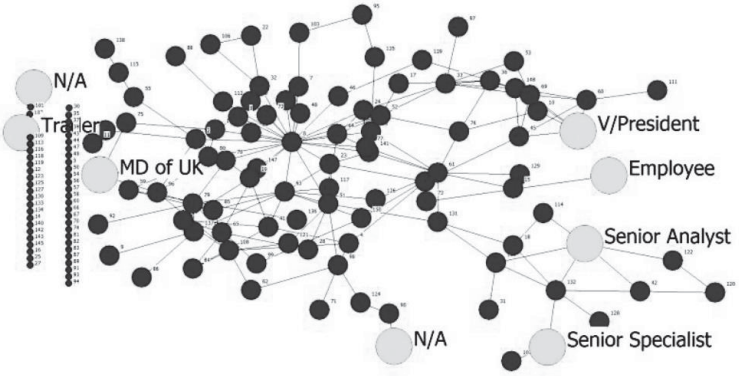
To conclude, using these reasonable explanation, we claim the effectiveness of the proposed scheme. And we infer that the proposed TV-DPPs can be broadly applied to real-world applications with similar settings to the Enron communication network.

3.6 Summary

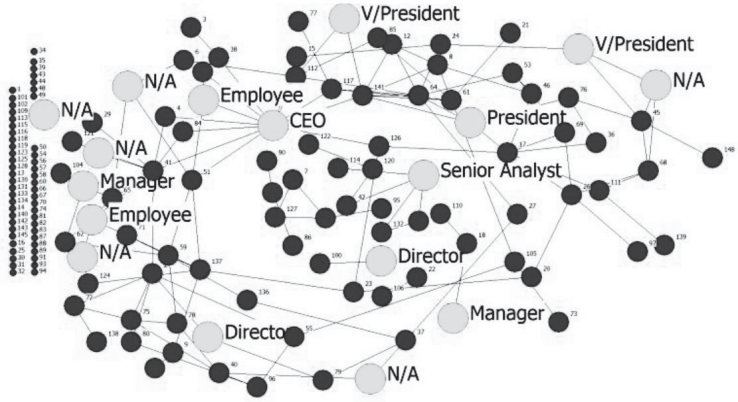
We proposed a fast sampling algorithm for DPPs for subset selection from a big dataset with time-varying structures. The algorithm uses the simplification of marginal density functions over successive time stamps, and also utilizes the Sequential Monte Carlo (SMC) sampling technique. The proposed algorithm provides us with a real-time diverse sampling scheme by utilizing the phenomenon in which typically only proportionally small changes occur at each time stamp with respect to the entire dataset. The most prominent application of our work is the online news service, in which news corpora are updated from the multiple sources continuously, before the most diverse subset is selected to be shown to the viewers. Another application mentioned in this chapter is Enron corpus which is based on online network structures. There are many other potential applications of our work to benefit the data mining community. One potential application is the modelling of latent attributes associated with a person-of-interest (POI) in a social network such as the POI's community membership over time. In this scenario, the POIâĂŹs friends in social network sites such as Facebook or Instagram can be treated as information sources (similarly to the online news setting) where their interests are constantly updated using text and/or images. At any given time stamp, we can select a diverse subset of all interests from his social circle as an additional important cue to infer the POIâĂŹs attributes.



(a) 2001-Sep-10



(b) 2001-Oct-20



(c) 2001-Nov-20

Figure 3.12: Three Enron communication networks from different stages detected by TV-DPPs.

Chapter 4

Diverse Learning for Mixtures of Exponential Family PCA Model

Exponential family principal component analysis (PCA) is one of the most ubiquitous techniques for general-typed data analysis, e.g., binary values and integers. A natural extension of it is a mixture model of local exponential family PCA in order to handle more data complexity, rather than only linearity in its essential forms. However, amongst mixing components of mixture models there may exist overlapping which may lead to model redundancy. To alleviate this problem, diversity is explicitly exploited in this paper to encourage repulsiveness amongst the mixing components. To the best of our knowledge, such a diversity prior has not been applied by existing works for mixture models of exponential family PCA. We encode this attractive attribute into a Bayesian scheme to obtain a new model, i.e., diversified exponential family PCA mixture model, wherein a determinantal point process (DPP) is exploited as a diversity prior distribution over joint local PCAs, where a similarity kernel between local PCAs is specified with a designed matrix-valued measure. Furthermore, ℓ_1 constraints are dedicatedly placed on transformation matrices of local PCAs to develop a systematic

way to address an important issue in traditional PCAs - selecting effective PCs of local hidden spaces. An iterative EM algorithm is derived to accomplish tasks of parameter learning and hidden variable inference. Experiments conducted on both synthetic and real-world datasets confirm the effectiveness of the proposed model.

4.1 Introduction

Principal component analysis (PCA) has been one of the most classic tool for linear data analysis, especially for dimensionality reduction (Jolliffe, 2002) and data distribution estimation (Anzai, 2012), and has been widely adopted in a variety of application scenarios, i.e., image analysis (Turaga and Chen, 2002; Geladi et al., 1989), image visualization (Kambhatla and Leen, 1997), data compression (Du and Fowler, 2007) and time series prediction (Ku et al., 1995). Previous works have developed many of its variants such as probabilistic PCA (Tipping and Bishop, 1999*b*), kernel PCA (Schölkopf et al., 1997), robust PCA (Candès et al., 2011), and sparse PCA (Zou et al., 2006). Amongst these variants, probabilistic PCA (PPCA) is a remarkable one. It formulates a deterministic PCA into a probabilistic framework with a Gaussian latent variable model, within which systematic statistical tools can be directly applied. As a result, several practical advantages are brought in, such as efficient parameter estimation.

Extensions based on a probabilistic version of PCA can straightforwardly address several tough concerns existing in traditional PCA. First, selecting a proper number of principal components is a crucial issue which is difficult to be systematically solved in traditional PCA. An important Bayesian extension to PPCA, i.e., Bayesian PCA (BPCA) (Nounou et al., 2002), is proposed to address this problem. Such a Bayesian version can also code with over-fitting problems. Sec-

ond, PCA and PPCA can only model continuous data in the observation space, and cannot handle other general data types such as discrete and binary, which are very common in real-world applications. An exponential family PCA (EPCA) (Collins et al., 2001) has been developed to address above issue by replacing the Gaussian distribution for observation likelihood with more general exponential families. Correspondingly, a Bayesian version of EPCA (Mohamed et al., 2009) has also been established to take advantage of Bayesian inference under the probabilistic framework. Furthermore, a simple version of above Bayesian EPCA has been proposed by J. Li and D. Tao (Li and Tao, 2013) to address the model selection problem with an automatic relevant determination (ARD) (MacKay, 1995) scheme. In this chapter, we focus on general data types, i.e., discrete and binary, and its related applications.

Due to the linear projection defined by PCA, it is limited to only simple data structure. This has naturally motivated various nonlinear developments of PCA to encode more complex data structure such as principal curve (Tibshirani, 1992) and GTM (Bishop et al., 1998). An alternative extension under probabilistic framework is mixing local exponential family PCA to increase model capacity (Li and Tao, 2013). However, there exist two significant limitations of mixture models. Firstly, traditional mixture models may exhibit model redundancy and severe model over-fitting problem. Intuitively, mixing components work together to cover an observation space. But the mixture models treat mixing components rather independently. Therefore, due to lack of explicit repulsive constraints amongst them, the learned components may overlap with each other. Thus, it may require more number of mixing components than it really needs to cover the observation space, which may lead to model redundancy and aggravate the over-fitting problem. Secondly, how to effectively determine both the number of mixing components and the number of effective principal components (PCs) are

still important but yet solved problems in practice for exponential family PCA mixture models. The work (Li and Tao, 2013) applies an ARD as a prior scheme over PCs to automatically set the number of PCs for each mixing component in an independent way, rather than in a holistic manner.

In this chapter, we address above mentioned issues with a diversified exponential family PCA mixture model. Diversity should be a solution to the above overlapping problem amongst mixing components. It encourages the mixing components to be as far with each other as possible in the model space to explicitly force each mixing component handle different part of the data space. As a result, overlapping problem can be alleviated. At the same time, it will also intuitively increase model generalization ability and reduce model overfitting risk. Recently DPP has been a popular statistical tool for diversity in machine learning area (Kulesza and Taskar, 2012). (Zou and Adams, 2012) provides a framework to encode DPP prior into generative latent variable model. Specifically, it applies a probability measure valued similarity kernel, i.e., to define DPP a concept of repulsion. Then DPP employs a determinant operator to assign probability values to each subset, the more repulsive, the higher values. Similarly, here to adopt DPP to exponential family PCA-MM, we have three challenges to be solved: 1) How to encode DPP into exponential family PCA-MM; 2) how to define a similarity kernel to measure repulsiveness amongst exponential family PCA mixing components; 3) how to do model selection, namely, determining effective numbers of PCs for each mixing component.

Regarding these challenges, we address them one by one and construct a holistic framework for extending the exponential family PCA mixture models with a diverse prior. The contribution of this chapter is summarized as below.

- We propose a framework to diversify the mixing components of a exponential family PCA-MM, As a result of which, the model redundancy and

over-fitting problems are expected to be alleviated.

- We design a reasonable matrix-valued similarity kernel as an input to a DPP prior to define the diversity amongst mixing components of exponential family PCA mixture model. In addition, with the decomposition of the transformation matrix of one mixing component, it not only keeps the orthogonality amongst PCs, but also generates an order of PCs similar to the variance order of these PCs.

- We dedicatedly incorporate a ℓ_1 term into the similarity kernel for two purposes. First, it contributes to the definition of repulsiveness amongst mixing components. Second, it provides clues for model selection, namely, selecting dimensions of PCs for each mixing component.

- We derive an iterative EM algorithm for parameter estimation and inference.

- Both verification and comparison experiments confirm the effectiveness of the proposed method.

This chapter is organized as follows. Section 4.2 review relevant works including both mixture models and diversity-related topics. Section 4.3 briefly introduce technical background. Our proposed model is developed in Section 4.4, and its parameter learning and inference is derived in Section 4.5. Finally, demonstrating experimental results and comparisons with related techniques are presented in Section 4.6, and Section 4.7 summarizes this chapter.

4.2 Related Work

4.2.1 PCA and its Mixture Extensions

We categorize existing PCA mixture extensions into two sets, one from a deterministic perspective, the other from a probabilistic perspective. One main commonality between these two categories is their motivation that they are trying to find better ways for data representations. PCA outputs a compact and decorrelated representation space, but is limited by its linearity. While mixture models is nonlinear, but may be negatively impacted by data noise in the original data feature space. Therefore, integrating PCA and mixture model together will mutually complement each other and retain both advantageous sides. The main difference between them is the strategies they adopt to integrate PCA and mixture model. (Turaga and Chen, 2002) is classified into the first category. It applies a mixture of eigenspaces to achieve a linear extension to the traditional PCA, and its objective is to minimize reconstruction errors. Since Tipping and Bishop has generalized PCA into a probabilistic framework to maximize likelihood. Most of the PCA mixture models are under the probabilistic framework. For example, (Tipping and Bishop, 1999*a*) naturally extends the traditional PCA to a well-defined mixture model, using the same way as Gaussian mixture models (GMM). And each mixing component represents one PCA subspace. Similar works are (Mahantesh et al., 2014; Zhang, 2004; Wang and Tang, 2005). A different work is proposed by (Watanabe et al., 2009), where it impose the mixture model on the subspace, rather than the observable PC mixing components.

One limitation of traditional PCA-MMs is that the likelihood is formulated with Gaussian distributions, which are only appropriate for continuous data. To handle integer or binary data type, Collins et al. (Collins et al., 2001) has generalized traditional PPCA to exponential family (EPCA). Subsequently, a

natural extension to EPCA is its corresponding mixture models. (Watanabe et al., 2009) assumes the mixing property comes from lower dimensional subspaces, and derives a variational Bayes approximation algorithm for learning. Comparatively, SePCA, proposed by Li and Tao (Li and Tao, 2013), is a simple version of Bayesian exponential family PCA. Unlike a Bayesian EPCA (Mohamed et al., 2009), it assigns prior distributions to its PCA's transformation matrix, and does not give prior over the hidden variables representing lower dimensional space. This strategy focuses on its PC loadings, rather than lower dimensional subspaces. Correspondingly, its mixture extension is formed with several PC loadings as mixing components, which is quite different from the above mentioned method. In this chapter, we adopt this strategy to focus on mixing transformation matrices. There also exist a branch of mixture models of exponential families, whereas no PCA is applied. Here we only list several works for interested readers, (Anaya-Izquierdo and Marriott, 2007; Akaho, 2008; McCULLAGH, 1994; Ardeshiri et al., 2013).

Finally, model selection has always been an important topic for PCA-MM, and has been explored by researchers. (Zhao, 2014) provides a hierarchical BIC to do efficient model selection, where each BIC is penalized by its own effective sample size, rather than the larger whole sample size. (Li and Tao, 2013) imposes an automatic relevance determination (ARD) over latent transformation matrix variables to determine the effective number of PCs. (Huang et al., 2004) explores a general notation of dimensionality in mixture models, and introduces a robust MED criterion to address the model selection. (Kim et al., 2001) proposes a fast and sub-optimal method of model order selection, and achieves model selection by pruning insignificant PCA bases.

4.3 Background

In this section, we briefly review related techniques as building blocks for our method, and they are simple exponential family PCA (SePCA), SePCA mixture models (SePCA-MM) and DPP.

4.3.1 Simple Exponential Family PCA (SePCA)

Simple exponential family PCA (SePCA, (Li and Tao, 2013)) has several advantages over the other PCA-related art-of-the-state methods. It can handle general observation data types, e.g., discrete and binary. As under a Bayesian framework, it inherits all of Bayesian inference advantages such as dealing with over-fitting problem and automatically determine the effective number of PCs. We adopt SePCA as one of the basic building block.

Given a set of observations $X = \{x_n\}_{n=1}^N$, where $x_n \in \mathcal{Z}^D$ or 2^D and D is the feature dimension, the SePCA distribution over joint variable set containing observable variables X, latent variables $Y = \{y_n\}_{n=1}^N$ and transformation matrix $W \in \mathcal{R}^{D\times d}$ is formulated as

$$p(X, Y, W; \alpha) = p(X|Y, W)p(Y)p(W; \alpha), \tag{4.1}$$

where $y_n \in \mathcal{R}^d$ represents low-dimentinal representations, with d representing lower space dimension and $d \ll D$.

We elaborate the three distributions on the right-hand of the above equation one by one. The first probability distribution on the right-hand of the above equation is the likelihood of observations. Given both latent representations and PCs, the distribution for each observation is independent with each other. As the name of SePCA indicated, the likelihood function employs an exponential

family, and its formulation for one observation is

$$p(x_n|y_n, W) = \exp\{x_n^T W y_n + g(W y_n) + h(x_n)\}. \tag{4.2}$$

Here, a canonical form of an exponential family is adopted, where $W y_n$ encodes natural parameters, $g(\cdot)$ and $h(\cdot)$ are known functions given a specific member of the exponential family. The second term of (4.1) represents a prior distribution over latent low-dimensional representations. It is assigned an isotropic unit Gaussian distribution, i.e., $y_n \sim \mathcal{N}(y_n|0, I)$. The third term represents a prior over transformation matrix W, which is composed by PCs. As required by PCA, this transformation matrix should be orthogonal in order to retain the uncorrelated property amongst PCs. Therefore, each column of the transformation matrix, i.e., each PC w_i, is independent with each other, and is assigned with an isotropic Gaussian prior controlled by a precision hyper-parameter. Formally,

$$p(W; \alpha) = \prod_{i=1}^{d} \mathcal{N}(0, \alpha_i^{-1}\mathbf{I}). \tag{4.3}$$

The model selection for effective number of PCs is motivated by ARD (MacKay, 1995), and is implemented by switching on or off each w_i with learned α_i according to the rule that smaller the α_i, less significant the w_i.

4.3.2 SePCA Mixture Models (SePCA-MM)

Proportionally mixing basic probability distributions is a simple and effective way to increase model complexity in order to handle with complex data structures. A relevant example is the mixture extension of SePCA (SePCA-MM). There are many complicated situations where a simple single PCA is incapable to handle but a mixture model of several PCAs is competent. For example,

groups of observations in a high-dimensional space might not be separable any more after being transformed into a low-dimensional space by a single SePCA. This is caused by the definition of a PCA whose projection axises are linear and dominated by variances of observations from all groups containing all observations. Unlikely, SePCA-MM fits groups of observations with different local SePCAs, and then integrates them with a mixing coefficient vector. Each mixing component of the mixture model, with different PCs from other mixing components, handles one group of observations independently. As a result, different groups of observations will be projected into different low-dimensional spaces, thus the separability exhibiting in the original space is preserved in several low-dimensional spaces. Therefore, a mixture model is able to solve the linearity of single component PCA to some extent, and also is a natural choice for above situation.

Formally, a layer of hidden discrete-valued variables $Z = \{z_n\}_{n=1}^{N}$ is introduced to SePCA to observations to indicate to which component they belong, where K is the number of mixing components and $z_n \in 2^K$ indexed by mixing components. A graphical representation for SePCA-MM is shown in Fig 4.1a. Comparing to traditional SePCA, K rather than 1 transformation matrices $\mathbf{W} = \{W^1, \ldots, W^K\}$ are introduced, and represented by a plate symbol. The joint distribution over both observable and hidden variables is formulated as below.

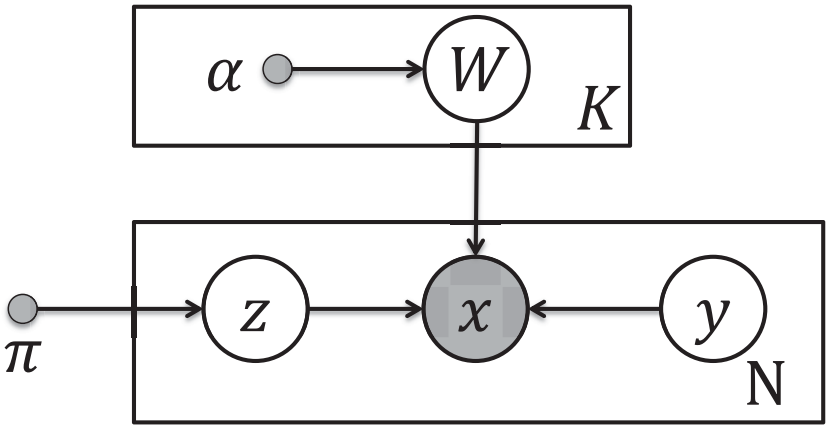
$$\begin{aligned} & p(X, Y, Z, \mathbf{W}; \pi, \lambda) \\ =& p(X|Y, \mathbf{W}, Z)p(Y)p(\mathbf{W}; \lambda)p(Z; \pi). \end{aligned} \tag{4.4}$$

As most of the probability distributions are the same with SePCA, we briefly summarize the changes with SePCA-MM. The first term defines the likelihood of

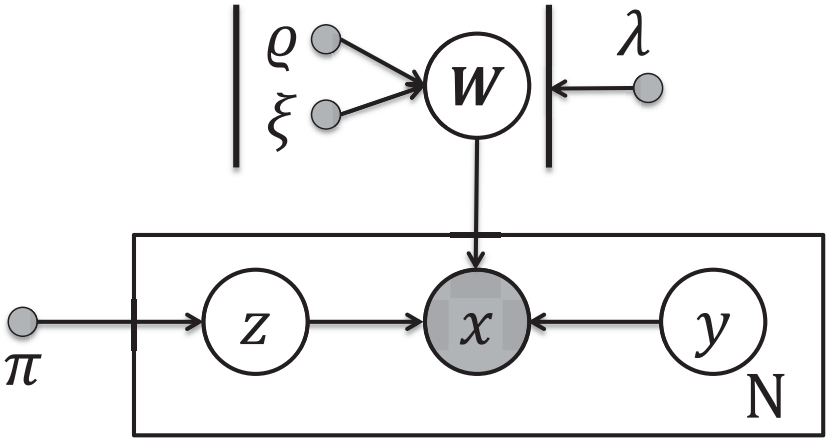
observations under one mixing component, specified by the values of Z and other hidden variables and parameters. It has the same form with (4.2) for an individual observation, but instituting W with a component specific W^{z_n}. The second term assigns isotropic unit Gaussian priors to each hidden low-dimensional representations, the same with the above. The third term gives prior over K mixing transformation matrices $\mathbf{W}$. It assumes each mixing component is independent from each other, and each PC within each mixing component is also independent to encode the orthogonality of a transformation matrix, and has the same prior distribution with (4.3). The fourth term presents a multinomial prior for binary indicator variables, and $p(Z;\pi) = \prod_{n=1}^{N} p(z_n;\pi)$ with $\sum_{k=1}^{K} \pi_k = 1$. Finally, the mixing concept is reflected by a proportional integration over probabilities of all mixing components, and is formally given as

$$p(x_n|y_n, \mathbf{W}) = \sum_{z_n} p(z_n) p(x_n|y_n, \mathbf{W}^{z_n}). \tag{4.5}$$

However, mixture models have limitations and cannot always do well for different application scenarios. For instance, the fitted mixing components may overlap with each other and can not be properly separated. As a result, this may not be tolerated by 'gaps' required applications, e.g. species delimitation (Yang and Rannala, 2010). Furthermore, it may need more components than it really needs to cover the whole space, which may easily leads to over-fitting. Under those circumstances, a diverse prior encouraging repulsiveness over mixing components is in demand to mitigate this problem. In this chapter, we focus on alleviating the above 'overlapping' problem, and show how to define a diversity prior for SePCA-MM as well as how to do model inference and parameter learning in next sections. Before that, we briefly introduce the DPP utilized to build a diversity prior in our method below.



(a) SePCA-MM



(b) Diversified SePCA-MM

Figure 4.1: Graphical representations of models: (a) SePCA mixture model; (b) Diversified SePCA mixture model. The only difference between these two models is obviously the priors over K mixture component parameters, i.e., Ws. Traditional SePCA assigns independent isotropic Gaussian distributions, represented by a plate. Comparatively, the proposed diversified SePCA-MM assigns a joint distribution over its component parameter, i.e., $\mathbf{W} = \{W^1, \ldots, W^K\}$. The distribution is a DPP, parameterized with ϱ, ξ, λ and represented by a double-struck.

4.4 The proposed model

In this section, we first introduce the core building blocks for our model, i.e., matrices-valued DPP. Then, based on it, the proposed diversified SePCA-MM is presented.

4.4.1 Motivation for Matrices-valued DPP

To reduce overlapping amongst mixing components of a SePCA-MM, parameterized with a set of matrix $\mathbf{W}$, it is natural to impose a diverse prior over matrices $\mathcal{W}$ to achieve diversity amongst them. However, it does not exist a proper L-ensemble diversity kernel defined over mixing components of the SePCA-MM in current literatures. Although a probability measure based kernel matrix can be considered applicable (Zou and Adams, 2012) when combining with a prior distribution for each transformation matrix applied in (Li and Tao, 2013), it is not ideal for our situation. On one hand, when the prior distribution employs an isotropic Gaussian distribution controlled by precision hyper-parameters, these parameters are used to automatically determine the effective number of PCs via an ARD scheme. On the other hand, as the means of the isotropic Gaussian priors are all fixed to 0, it is ambiguous to infer group labels for samples around 0. Furthermore, a diversity kernel matrix constructed with a probability measure over such prior distributions is only relevant to those precision hyper-parameters, and has no relevant terms for transformation matrices $\mathbf{W}$ themselves. Consequently, this kind of kernel cannot provide an intuitive geometric explanation for diversity amongst these mixing components. Overall, a similarity kernel over transformation matrices, i.e., mixing components of SePCA-MM, serving as a fundamental block of a diversity prior needs to be customised.

4.4.2 Matrices-valued DPP

In this subsection, we define a diversity prior over mixing components of SePCA-MM, i.e., $\mathbf{W} = \{W^1, \ldots, W^K\}$ with three steps - decomposing transformation matrix into quality and similarity parts, formulating quality and similarity terms, and defining diversity prior under DPP framework.

Decomposition

We decompose each transformation matrix W into two parts: An orthonormal matrix $\Upsilon \in \mathcal{R}^{D\times d}$ representing PCs and a diagonal matrix $\Phi \in \mathcal{R}^{d\times d}$, each of whose entries representing variances along each PC. Formally,

$$W = \Upsilon\Phi, \tag{4.6}$$

with $\Upsilon^T\Upsilon = I$ and $\Phi = \text{diag}(\phi_1, \ldots, \phi_d)$. For a set of transformation matrices $\mathbf{W}$, we symbolize the orthonormal matrix set and the variance matrix set with bold font, namely, $\mathbf{\Upsilon} = \{\Upsilon^1, \ldots, \Upsilon^K\}$ and $\mathbf{\Phi} = \{\Phi^1, \ldots, \Phi^K\}$ respectively.

Such a decomposition splits orthogonality and variance of a transformation matrix, and is motivated by three advantages. First, the orthonormal parts explicitly keep the orthogonality amongst PCs within PCAs. In other words, the features in low-dimensional space transformed by these PCs are retained uncorrelated, which is a fundamental characteristic of PCAs for dimensionality reduction. Second, the variance parts play the role of determining which PCs keep maximum empirical variances, whose value can be straightforwardly employed to retain important PCs and cut insignificant ones. In this view, it constructs an efficient way to automatically select effective number of PCs. Finally, it provides a convenient way to separate quality term and similarity term for defining a diversity kernel, as presented in subsequent sections.

Formulation for quality and similarity terms

Similar to (Kulesza and Taskar, 2012), a quality-similarity decomposition is employed to construct a similarity kernel for diversity prior. Such a decomposition explicates the tradeoff between similarity and diversity in a DPP prior. These two terms in our case are respectively defined over variance and orthonormality matrices from above decomposition.

Quality: The quality term of transformation matrix W is defined over a variance term Φ referred to as quality features as:

$$\mathcal{Q}(W) = \exp(-\frac{1}{2}\xi||\Phi||_1), \tag{4.7}$$

where ξ is a scale parameter to control two kinds of tradeoffs - between quality and diversity as well between diversity and likelihood. We will emphasize this point again after the whole model is established.

An ℓ_1-norm is applied to the diagonal matrix Φ. High quality scores can only be obtained with small ℓ_1-norm of variance matrix, which usually encourages sparsity for Φ. Under our case, the PCs corresponding to small-valued variance entries will be abandoned. In other words, this quality function highly values low-dimensional hidden space. In addition, as above mentioned, it automatizes the process of model selection.

Similarity: We refer to the orthonormal matrix Υ as similarity features for a transformation matrix W. One entry of a similarity function $\mathcal{S}$ over pairwise transformation matrices $\{W^1, W^2\}$ is constructed via their separable orthonormal parts $\{\Upsilon^1, \Upsilon^2\}$, and is defined as

$$\mathcal{S}(W^1, W^2) = \exp(-\frac{1}{2}\varrho \sum_{i,j=1}^{d} ||\Upsilon^1_{\cdot i} - \Upsilon^2_{\cdot j}||_2^2), \tag{4.8}$$

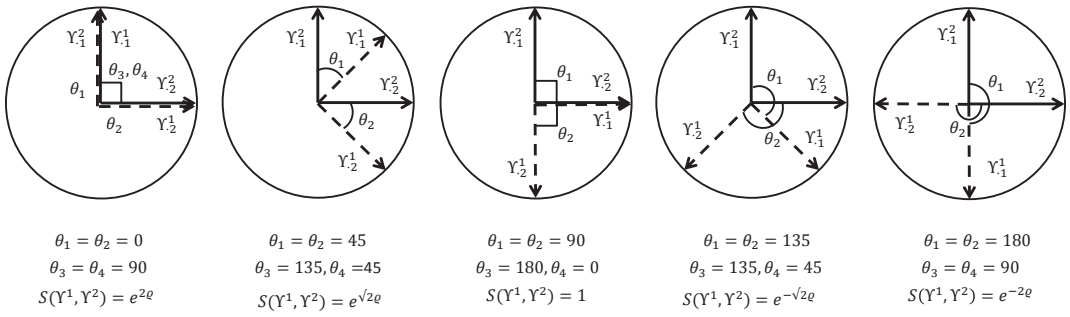


Figure 4.2: Illustration of proposed similarity formulations with two orthonormal matrices including two PCs. The dashed pairwise perpendicular lines represent PCs of orthonormal matrix Υ^1, while the solid pairwise lines represent PCs of Υ^2. θ_1 and θ_2 represent angles of direct-matching pairwise PCs as shown in all subfigures, while θ_3 and θ_4 represent angles of cross-matching pairwise PCs which are only demonstrated in the first subfigure and ignored by other subfigures to keep their symbols less crowded. From left to right, the angles between direct-matching pairwise PCs of the two orthonormal matrices starts from 0° and increases with 45°, while the similarities between these two matrices decreases from the largest value $e^{2\varrho}$ to the smallest value $e^{-2\varrho}$, which are calculated from (4.9)..

$$s.t.\ \Upsilon^{1T}\Upsilon^1 = I_d,$$
$$\Upsilon^{2T}\Upsilon^2 = I_d.$$

Here ϱ is a scale parameter, and $||\cdot||_2$ symbols the vector L2 norm. I_d is a $d \times d$-sized identity matrix with d the dimension of hidden spaces. The two constraints guarantee the orthonormality of Υs.

Expanding the vector L2 norm, the similarity function $\mathcal{S}$ is derived as:

$$\begin{aligned}\mathcal{S}(W^1, W^2) &= \exp(-\frac{1}{2}\varrho(\sum_{i,j=1}^{d}\sum_{k=1}^{D}(\Upsilon^1_{ki} - \Upsilon^2_{kj})^2)) \\ &\propto \exp(\varrho\sum_{i,j=1}^{d}\cos(<\Upsilon^1_{\cdot i}, \Upsilon^2_{\cdot j}>)).\end{aligned} \tag{4.9}$$

Here D is the number of dimensions in the observation space. Note that the only

difference between (4.9) and (4.8) is a constant term. This term is exactly the same for all entries of the similarity matrix, therefore does not make contributions to diversity modelling and is ignored in the following modelling procedure.

Geometrical explanation for similarity formulation: As it is easily seen from (4.9), the similarity between two transformation matrices is positively related to the sum of cosine values of angles formed by pairwise normalized PCs, which is, in other words, negatively correlated with the angle value between them. This similarity formulation is illustrated in Figure 4.2 with two orthonormal matrices, each with two columns. When the two matrices are the same, i.e., $\Upsilon^1 = \Upsilon^2$, the angle values for direct-matching pairwise PCs $\Upsilon^1_{\cdot 1}$-$\Upsilon^2_{\cdot 1}$ and $\Upsilon^1_{\cdot 2}$-$\Upsilon^2_{\cdot 2}$ are $0°$, and the angle values for cross-matching pairwise PCs $\Upsilon^1_{\cdot 1}$-$\Upsilon^2_{\cdot 2}$ and $\Upsilon^2_{\cdot 1}$-$\Upsilon^1_{\cdot 2}$ are $90°$ due to the orthonormality of Υs. Calculating via (4.9), the similarity between them is $\exp(2\varrho)$, which is the largest. The similarities of two transformation matrices with other angle values for direct-matching pairwise PCs such as $45°, 90°,$ and $135°$, are $e^{\sqrt{2}\varrho}, 1$, and $e^{-\sqrt{2}\varrho}$ respectively. Similarly, when each column $\Upsilon^1_{\cdot j}$ of a matrix Υ^1 is totally different with direct-matching column $\Upsilon^2_{\cdot j}$ of matrix Υ^2, the angle value formed by them is $180°$ and the corresponding cosine value is -1, and the resulting similarity between the two matrix is $e^{-2\varrho}$, which is the smallest.

Diversity prior

Each entry of an L-ensemble kernel $\mathcal{K}$ required by a DPP distribution represents the similarity of its indexed two transformation matrices, and is defined by multiplying the quality term with similarity term, just like (Kulesza and Taskar, 2012). Formally,

$$\mathcal{K}(W^1, W^2) = \mathcal{Q}(W^1)\mathcal{S}(W^1, W^2)\mathcal{Q}(W^2). \tag{4.10}$$

A DPP distribution over transformation matrices is defined via a determinant operator over the L-ensemble kernel $\mathcal{K}$. We take a two-element subset $\{W^1, W^2\}$ as an example. Its DPP probability is computed as

$$\mathcal{P}_{\mathcal{K}}(\{W^1, W^2\}) \propto \mathcal{Q}^2(W^1)\mathcal{Q}^2(W^2)\det(\mathcal{S}(\{W^1, W^2\})) \tag{4.11}$$

$$= \exp(-\xi(||\Phi^1||_1 + ||\Phi^2||_1)) \times \left\{1 - \exp(2\varrho \sum_{i,j=1}^{d} \cos(< \Upsilon^1_{\cdot i}, \Upsilon^2_{\cdot j} >))\right\}, \tag{4.12}$$

where $\mathcal{S}(\{W^1, W^2\})$ represents the similarity submatrix indexed by W^1 and W^2. Note that the normalization term is ignored here. since it is a constant for various subsets once the parameters ξ and ϱ are given.

We draw two characteristics of above definition: 1). The computations for quality term and similarity term are independent due to a nice property of the determinant operator. 2). From (4.12), the diverse probability over a transformation matrix subset increases along with the increment of the quality of each element, as well as with the decrement of the joint similarities amongst them.

In summary, the DPP distribution prefers subsets of transformation matrices that each element within it has small number of PCs while these elements are diverse with each other.

4.4.3 Diversified SePCA-MM

Until now, combining the established diverse prior over mixing components into a SePCA-MM framework, a diversified SePCA-MM is composed. The corresponding graphical representation is shown in Figure 4.1b, where two bold vertical lines drawing over $\mathbf{W}$ represent a DPP prior over the joint mixing components. Finally, the joint distribution over both observable and hidden variables is for-

mulated as:

$$
\begin{aligned}
& p(X, Y, Z, \mathbf{W}; \pi, \lambda, \varrho, \xi) \\
& \quad \propto \prod_{n=1}^{N} p(z_n; \pi) p(y_n) p(x_n | y_n, W^{z_n}) \mathcal{P}_{\mathcal{K}}^{\lambda}(\mathbf{W}; \varrho, \xi) \qquad (4.13) \\
& \quad s.t. \qquad W^k = \Upsilon^k \Phi^k, \quad k = 1, \ldots, K, \\
& \qquad\qquad \Upsilon^{kT} \Upsilon^k = I_d, \qquad (4.14) \\
& \qquad\qquad \Phi^k = \mathrm{diag}(\phi_1^k, \ldots, \phi_d^k), \\
& \qquad\qquad \sum_{k=1}^{K} \pi_k = 1,
\end{aligned}
$$

where $\mathbf{W}$ is a K-sized transformation matrix set, each element W^k of which is decomposed into an orthonormal matrix Υ^k and a diagonal matrix Φ^k, representing diverse features and quality features respectively. K is the number of mixture components. The constraint for π is to satisfy a discrete categorical distribution.

The parameters and their functionalities are summarized as follows. ξ and ϱ are introduced in the diversity prior (4.12), wherein ξ is for quality measure of individual transformation matrix and ϱ is for similarity measure of joint transformation matrix subset. When combined into the diversity prior, they can be adjusted to balance between quality and similarity. Furthermore, these two parameters play a trade-off role when they are as a part of the whole model in (4.13), where they are utilized to balance between diversity of $\mathbf{W}$ and the model fitness of datasets in different application scenarios. We separate the trade-off role of these two parameters from their balancing role by introducing a new parameter λ as shown in (4.13). It makes no substantial change to the original model, since it is actually extracted from the expression of ξ and ϱ. Changing

the value of λ gives a clear way to demonstrate the efficacy of the proposed diversity prior in the experimental section.

Note that the parameters ξ and ϱ are fixed in the model and for parameter learning and inference , but can be chose to be adapted to real-world dataset. Actually, this is a simplification for our model to avoid intractable learning for these two parameters, since they exist in the normalization term of the DPP prior. One drawback of this simplification is that it limits us to a point estimation for $\mathbf{W}$, whose learning and inference procedure are shown in next section.

4.5 Learning and inference

We address parameter learning and inference of the proposed model within an expectation-maximization (EM) framework.

4.5.1 Learning Parameters: M-step

Learning π

The objective of log-likelihood maximization in terms of π is

$$\max_{\pi} \log p(X;\pi) = \sum_{n=1}^{N} \log \sum_{z_n} p(x_n|z_n)p(z_n;\pi), \tag{4.15}$$

$$s.t. \sum_{k} \pi_k = 1. \tag{4.16}$$

Directly applying Lagrange multiplier method is intractable due to the integration over z_n inside the log operator. Therefore, we approximate the above log-likelihood with its lower bound function obtained by Jensen's inequality, which

is

$$\mathcal{L}_{qZ}(\pi) = \sum_{n=1}^{N} \sum_{z_n} q(z_n) \log p(x_n|z_n) \tag{4.17}$$
$$+ \sum_{n=1}^{N} \sum_{z_n} q(z_n) \log p(z_n; \pi) + \mathcal{H}_{q_{zn}},$$

where $q(z_n)$ is a distribution over z_n, and it makes the above lower bound equal to the (4.15) when it is the posterior distribution over z_n. $\mathcal{H}_{q_{zn}}$ is the entropy for $q(z_n)$. In above equation, the first term expressing an expectation over conditional log-likelihood in terms of $q(z_n)$ and the third term symbolized the entropy of $q(z_n)$ are irrelevant to the parameters π. Therefore, they are simply ignored when learning for π. Only the second term, which is an expectation over log-z_n prior in terms of $q(z_n)$, combined with the normalization constraint (4.16) is used to learn π. Its analytic solution is obtained with Lagrange multiplier method and is:

$$\pi_k = \frac{\sum_n q(z_n = k)}{\sum_k \sum_n q(z_n = k)} = \frac{N_k}{N}. \tag{4.18}$$

Here N refers to as the number of all samples, while $N_k = \sum_n q(z_n = k)$, summarizing over probabilities of all samples being labeled as group k. It can be seen that π_k updates the ratio of samples in each component in the mixture model with posterior inference of hidden variables Z.

4.5.2 Inference: E-step

Posterior inference on Z

The posterior distributions over Z are computed through the Bayes' rule, namely,

$$q(Z) = p(Z|X;\pi) = \frac{p(Z;\pi)p(X|Z)}{\sum_Z p(Z;\pi)p(X|Z)}. \tag{4.19}$$

Here $p(Z;\pi)$ is Z's prior distribution. $p(X|Z)$ is a Z-conditional likelihood, which is also needed in (4.17). It can be computed by integrating over two other hidden variables Y and $\mathbf{W}$ from their known joint distribution. Formally

$$p(X|Z) = \int_Y \int_{\mathbf{W}} p(Y)p_{\mathcal{K}}(\mathbf{W})p(X|Y,\mathbf{W},Z). \tag{4.20}$$

However, either of the two integrations has closed-form. As for the integration over Y, there is a generalised exponential family term in the $\{Y,\mathbf{W},Z\}$-conditional likelihood $p(X|Y,\mathbf{W},Z)$, and is not conjugated with its prior distribution. Similarly, for the integration over $\mathbf{W}$, its DPP-based prior distribution has no conjugate property with exponential family terms. One intuitive approximation to the two integrations is adopting a Monte Carlo method. It firstly draws two sets of samples from Y and $\mathbf{W}$'s prior distributions respectively, and secondly substitutes the samples for variables to compute the conditional likelihood, and finally replaces the integration with a tractable average operator over those samples' likelihoods. However, although the prior distribution of $p(Y)$ can be easily sampled, the joint transformation matrix set $\mathbf{W}$ cannot be easily obtained from its DPP-based prior distribution.

Therefore, we again approximate the above $\{Y,\mathbf{W}\}$-marginalized conditional likelihood with its lower bound approximation in log space. With Jensen's in-

equality, one lower bound to (4.20) in log space is obtained as

$$\mathcal{L}_q(\mathbf{W}, Y) = \{\mathcal{H}_{q_\mathbf{W}} + \mathcal{H}_{q_Y} + \int_{\mathbf{W}} \int_{Y} q(\mathbf{W}) q(Y) \log p(X, Y, \mathbf{W}|Z) \Big\}. \tag{4.21}$$

Here $q(\mathbf{W})$ and $q(Y)$ are $\mathbf{W}$ and Y's posterior distributions respectively. The first two terms $\mathcal{H}_{q_\mathbf{W}}$ and $\mathcal{H}_{q_Y}$ are entropies of posterior distributions of $\mathbf{W}$ and Y respectively. The third term computes the expectation over log Z-conditional joint distribution in terms of posterior distributions of $\mathbf{W}$ and Y. Instead of integrating over $\mathbf{W}$ and Y's posterior distribution, which is intractable, we simplify it by substituting the expectations with MP estimations from their posterior distributions, namely

$$\mathcal{L}_q(\mathbf{W}, Y) \approx \log p(X, Y^{\mathrm{MP}}, \mathbf{W}^{\mathrm{MP}}|Z) + \text{const.} \tag{4.22}$$

Here the const summarizes over entropy terms. In the next two subsections, we detail the alternative mode estimations for posterior distributions of $\mathbf{W}$ and Y.

Posterior mode estimation for W

The posterior distribution over $\mathbf{W}$ is formulated by its prior distribution and likelihood via Bayes' rule, namely,

$$q(\mathbf{W}) = p(\mathbf{W}|X) \propto p_{\mathcal{K}}(\mathbf{W}) p(X|\mathbf{W}).$$

Given Y^{MP}, the objective for mode estimation of posterior distribution of $\mathbf{W}$ in log-space is

$$\max_{\mathbf{W}} \mathcal{L}_q(\mathbf{W}) = \log p_{\mathcal{K}}(\mathbf{W}) + \sum_{Z} q(Z) \log p(X, Y^{MP}, Z|\mathbf{W}), \tag{4.23}$$

where the first term is related to $\mathbf{W}$'s diverse prior distribution, and the second term is an expectation over log-likelihood of $\mathbf{W}$ in terms of $q(Z)$.

We apply a coordinate ascend method to iteratively search the optimal solution for the transformation matrix set $\{W^k\}_{k=1}^K$. Within each coordinate, since each W^k composes of two components, i.e., similarity orthonormal matrix Υ^k and quality diagonal matrix Φ^k, we alternatively update them.

For Υ: Due to the orthonormal constraints (4.14), we search its optimal along the Grassmann manifold. The derivation in Euclidean space is

$$\begin{aligned}\frac{\partial \mathcal{L}_q(\mathbf{W})}{\partial \Upsilon_{ij}^k} &= \lambda \text{trace}\{\mathcal{K}^{-1}(\mathbf{W})\mathcal{Q}(\mathbf{W})\frac{\partial \mathcal{S}(\mathbf{W})}{\partial \Upsilon_{ij}^k}\mathcal{Q}(\mathbf{W}) + \\ &(q(Z=k))^T \text{diag}\left(g'^T(\Upsilon^k\Phi^k Y^{MP}) \cdot \left(\frac{\partial \Upsilon^k}{\partial \Upsilon_{ij}^k}\Phi^k Y^{MP}\right)\right) \\ &+ \text{trace}(X^T\frac{\partial \Upsilon^k}{\partial \Upsilon_{ij}^k}\Phi^k Y^{MP}\text{diag}(q(Z=k))), \end{aligned} \tag{4.24}$$

with each entry of $\frac{\partial \mathcal{S}(W)}{\partial \Upsilon_{ij}^k}$ from

$$\frac{\partial \mathcal{S}(W^k, W^{k'})}{\partial \Upsilon_{ij}^k} = \mathcal{S}(W^k, W^{k'}) \cdot \varrho \cdot \sum_m (\Upsilon_{im}^{k'} - \Upsilon_{ij}^k),$$

Let $G(\Upsilon^k) = \frac{\partial -\mathcal{L}_q(\mathbf{W})}{\partial \Upsilon^k}$, the derivation on the Grassmann manifold is defined as:

$$GG(\Upsilon^k) = G(\Upsilon^k) - \Upsilon^k\Upsilon^{kT}G(\Upsilon^k).$$

At point Υ^k with direction $GG(\Upsilon^k)$, the corresponding geodesic equation is

$$\Upsilon^k(t) = \Upsilon^k V\cos(\Sigma t)V^T + U\sin(\Sigma t)V^T, \tag{4.25}$$

where matrices U, Σ, and V are from the compact SVD of $GG(\Upsilon^k)$, namely,

$GG(\Upsilon^k) = U\Sigma V^T$. The optimal searching is along the geodesic defined by (4.25). This is actually a one-dimensional searching with respect to variable t, namely, $\min_t -\mathcal{L}_q(\Upsilon^k(t)\phi^k)$. Suppose it reaches the minimum value at t', then the corresponding point on the Grassmann manifold is $\Upsilon^{k'} = \Upsilon^k(t')$.

For Φ: We iteratively update them with gradient ascend method. Each entry of their derivative over Φ is listed as below, and the details are derived in Appendix.

$$\begin{aligned}\frac{\partial \mathcal{L}_q(\mathbf{W})}{\partial \Phi_{ii}^k} &= \text{trace}\{\mathcal{K}^{-1}(\mathbf{W})\frac{\partial \mathcal{Q}(\mathbf{W})\mathcal{S}(\mathbf{W})\mathcal{Q}(\mathbf{W})}{\partial \Phi_{ii}^k}\} \\ &+ \text{trace}(X^T\mathbf{\Upsilon}^k\frac{\partial \mathbf{\Phi}^k}{\partial \Phi_{ii}^k}Y^{MP}\text{diag}(q(Z=k)))+ \\ &q^T(Z=k)\text{diag}\left(g'^T(\Upsilon^k\Phi^kY^{MP})\cdot\Upsilon^k\frac{\partial \Phi^k}{\partial \Phi_{ii}^k}Y^{MP}\right). \end{aligned} \tag{4.26}$$

For term $\frac{\partial \mathcal{Q}(\mathbf{W}^k)\mathcal{S}(\{\mathbf{W}^k,\mathbf{W}^{k'}\})\mathcal{Q}(\mathbf{W}^{k'})}{\partial \Phi_{ii}^k}$, when $k = k'$, it is

$$\frac{\partial \exp(-\xi(||\Phi^k||_1))}{\partial \Phi_{ii}^k} = \exp(-\xi(||\Phi^k||_1))\cdot -\xi\cdot\frac{\partial|\Phi_{ii}^k|}{\partial \Phi_{ii}^k};$$

otherwise, it is computed as

$$\mathcal{S}(\{\mathbf{W}^k,\mathbf{W}^{k'}\})\mathcal{Q}(\mathbf{W}^{k'})\exp(-\frac{1}{2}\xi(||\Phi^k||_1))\cdot -\frac{1}{2}\xi\cdot\frac{\partial|\Phi_{ii}^k|}{\partial \Phi_{ii}^k}.$$

Due to the term $||\Phi^k||_1 = \text{trace}(|\Phi^k|)$, the derivation for Φ_{ii}^k involves all other elements $\{\Phi_{jj}^k\}_{j\neq i}$. Therefore, we iteratively update the diagonal elements of Φ^k one by one. A sub-gradient method at point 0 is applied, where the objective is nondifferientiable, and that the term $\frac{\partial|\Phi_{ii}^k|}{\partial \Phi_{ii}^k}$ is -1 or 1 or 0 depends on which increases the objective most.

Posterior mode estimation for Y

Similarly to $\mathbf{W}$, the posterior distribution on Y is computed as

$$q(Y) \propto p(Y)p(X|Y).$$

Given $\mathbf{W}^{MP}$, the objective for mode estimation of posterior distribution on Y in log-space is

$$\max_{Y} \mathcal{L}_q(Y) = \log p(Y) + \sum_{Z} q(Z) \log p(X, \mathbf{W}^{MP}, Z|Y).$$

To achieve the optimal of each independent y_n, the gradient ascend method is applied and its gradient is computed as below.

$$\frac{\mathcal{L}_q(y_n)}{\partial y_n} = -y_n + \sum_{z_n} q(z_n) \left((x_n^T W^{MP,z_n})^T + (g'^T (W^{MP,z_n} y_n) \cdot W^{MP,z_n})^T \right).$$

4.5.3 Algorithm Summary and Bernoulli Distributions

The whole procedure of parameter learning and inference for the proposed method under an EM framework is summarized in Algorithm 4.1.

One simple example from exponential family is Bernoulli distribution, and is designated to handle binary-typed data sets. We apply this kind of distribution throughout our experimental section. To make our chapter self-contained, we briefly instantiate the general terms in our model, e.g., $g(WY)$ and $g'(WY)$, with Bernoulli distribution.

For Bernoulli distribution with parameter α, the distribution is $p(x|\alpha) = \alpha^x(1-\alpha)^{1-x}$ and its corresponding natural parameters are $\theta = \log \frac{\alpha}{1-\alpha}, g(\theta) =$

Algorithm 4.1: Parameter Learning and Inference

Data: Observations X, fixed parameters ξ, ϱ, stopping criterion ϵ.
Result: $\mathbf{W}^{MP}$, $Y^{MP}, \mathcal{L}^{new}$, q(Z), π.
Randomly initialization π, $\mathbf{\Phi}^{MP}$, $\mathbf{\Upsilon}^{MP}$, Y^{MP};
$W^{k,MP} = \Upsilon^{k,MP}\Phi^{k,MP}$, $k = 1, \ldots, K$;
$q(Z) \leftarrow$ (4.19) and $p(X, Y^{MP}, \mathbf{W}^{MP}|Z)$;
$\mathcal{L}^{new} = \mathbb{E}_{q(Z)}(\log p(Z;\pi) + \log p(X, Y^{MP}, \mathbf{W}^{MP}|Z))$;
repeat
  $\mathcal{L}^{old} = \mathcal{L}^{new}$;
  M-step:
  $\pi \leftarrow$ (4.18) and $q(Z)$;
  E-step:
  $\mathcal{L}_0^{new} = \mathbb{E}_{q(Z)}(\log p(Z;\pi) + \log p(X, Y^{MP}, \mathbf{W}^{MP}|Z))$;
  repeat
    $\mathcal{L}_0^{old} = \mathcal{L}_0^{new}$;
    $q(Z) \leftarrow$ (4.19) and $p(X, Y^{MP}, \mathbf{W}^{MP}|Z)$;
    Alternatively update Υ^k and Φ^k with (4.24) and (4.26);
    $W^{k,MP} = \Upsilon^{k,MP}\Phi^{k,MP}$, $k = 1, \ldots, K$;
    $Y^{MP} \leftarrow$ (4.27);
    $\mathcal{L}_0^{new} = \mathbb{E}_{q(Z)}(\log p(Z;\pi) \ + \ \log p(X, Y^{MP}, \mathbf{W}^{MP}|Z))$;
  until $|\mathcal{L}_0^{new} - \mathcal{L}_0^{old}| < \epsilon$;
  $\mathcal{L}^{new} = \mathbb{E}_{q(Z)}(\log p(Z;\pi) \ + \ \log p(X, Y^{MP}, \mathbf{W}^{MP}|Z))$;
until $|\mathcal{L}^{new} - \mathcal{L}^{old}| < \epsilon$;

$-\log(1+e^{\theta})$, $h(x) = 0$, and $\alpha = \frac{e^{\theta}}{e^{\theta}+1}$. Therefore,

$$g(W^{z_n} y_n) = \sum_{i=1}^{D} g([W^{z_n} y_n]_i), \tag{4.27}$$

$$g'(W^{z_n} y_n) = \begin{bmatrix} \frac{1}{1+e^{[W^{z_n} y_n]_1}} - 1 \\ \vdots \\ \frac{1}{1+e^{[W^{z_n} y_n]_D}} - 1 \end{bmatrix}. \tag{4.28}$$

4.5.4 Algorithm Complexity Analysis

We analysis the proposed model's scaling ability by analyzing its time complexity of its learning and inference algorithm. Inside each loop of Algorithm 4.1 the

most time-consuming steps are updating $\boldsymbol{\Upsilon}$ (4.24), $\boldsymbol{\Phi}$ (4.26), and Y^{MP} (4.27). For $\boldsymbol{\Upsilon}$, its complexity is $\bigcirc(KDd \times (N^2D + K^3 + Dd^2))$, where the first term just before the symbol $\times$ counts the matrix elements of K orthonormal matrices, and the second term is the complexity for each matrix element update with one term for matrix multiplications, the second term for diverse kernel inversion and the third term for derivation computation on the Grassmann manifold with single value decomposition (SVD). For $\boldsymbol{\Phi}$, its complexity is $\bigcirc(Kd \times (N^2d + K^3))$, where again the first term counts the matrix elements of K diagonal matrices decomposed from the mixing transformation matrices, and the second term is the complexity for each diagonal element update. For Y^{MP}, its complexity is $\bigcirc(N \times D \times d)$, where N is for each data point and $D \times d$ for matrix multiplication. Consequently, when it comes to scale analysis, either large-scale datasets (big N) or high-dimensional datasets (big D) has squared time complexity since the largest term of all these time complexities is $\bigcirc(KN^2D^2d)$.

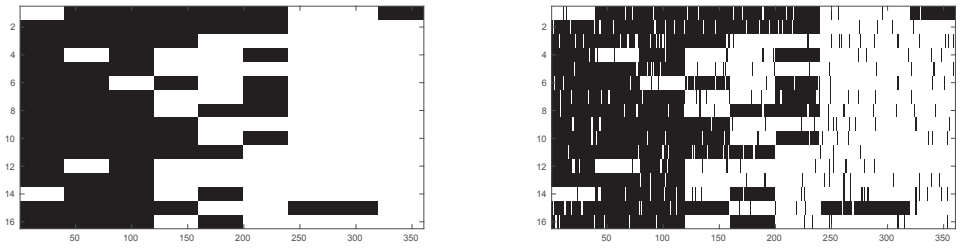
4.6 Experimental Results

In this section, we verify the proposed diversified exponential family PCA mixture models on both a synthetic dataset and a real-world dataset, i.e., USPS hand-written digit dataset.

4.6.1 Synthetic Experiments

In this subsection, we mainly focus on verifying the effectiveness of the ℓ_1 constraints for selecting dominant PCs and of the diversity prior over mixing components for model redundancy reduction via a synthetic dataset. We first introduce the generating process of our synthetic dataset and experimental settings, and then present the experimental results.

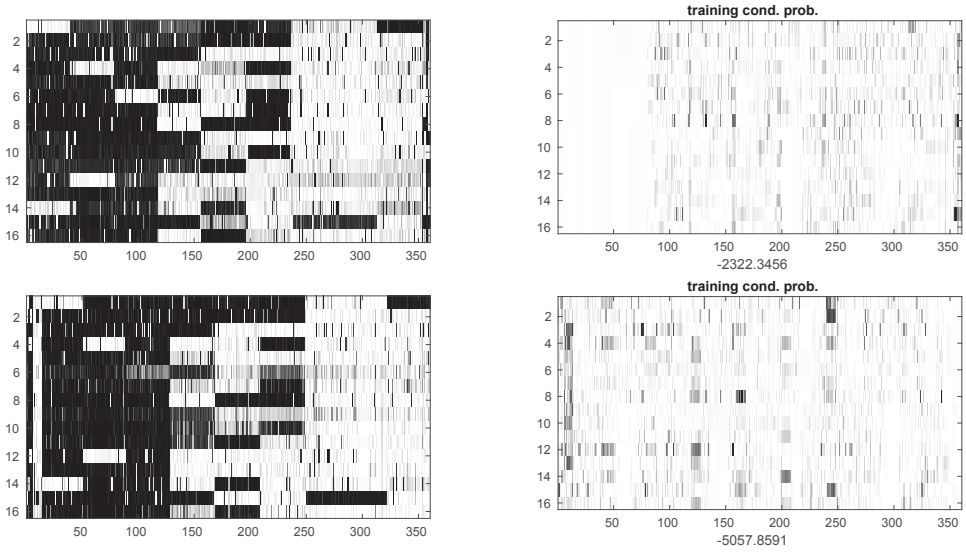
We generated our synthetic dataset in a similar way with Mohamed et al. (2009). The reason for that we did not directly use their datasets is that they are used to verify a single PCA component, rather than a mixture models of PCAs. Therefore, we developed our own complex dataset in three steps. First, three Bernoulli distributions with mean parameters 0.9, 0.5 and 0.1 are used to generate three categories of binary data, each of which contains three 16-D binary prototype vectors with each bit independently drawn from $\{0, 1\}$ with the same Bernoulli distribution. Then, 40 duplicate copies of each prototype are made, composing a dataset with 360 samples in total. One data sample generated from these steps is illustrated in Figure 4.3a. Each bit is flipped with probability 0.1 to generate a noisy sample as training dataset. One such sample is shown in Figure 4.3b.



(a) True data (b) Noisy data

Figure 4.3: Black indicates 0 and white indicate 1.

Our experimental settings are listed below. We fixed $\xi = 0.1$, and $\rho = 0.1$. Because parameters K, d, λ are associated with model redundancy, we postpone their specifications in each verifying experiments later. Regarding initialization, we set π with a normal distribution, $\{\Phi^k\}_{k=1}^K$ with a random diagonal matrix, and $\{\Upsilon^k\}_{k=1}^K$ with a orthonormal matrix whose entries are randomly generated and processed with a Gram-Schmidt process. Since these initializations may have impact on model optimas, to make fair comparisons we set these initializations

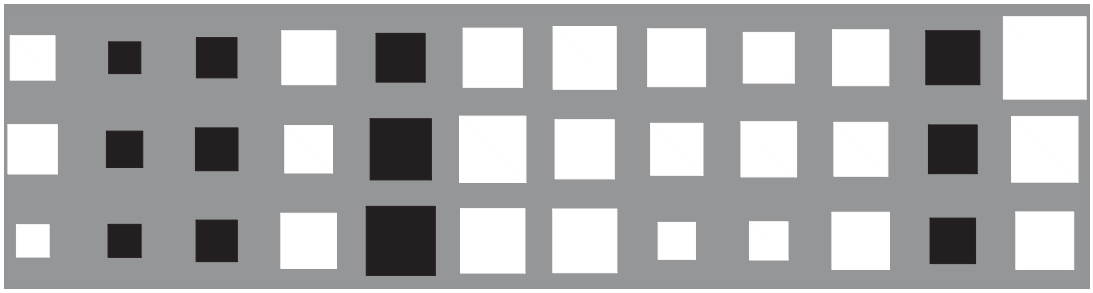


(a) Reconstructed mean parameters (b) Training conditional probabilities

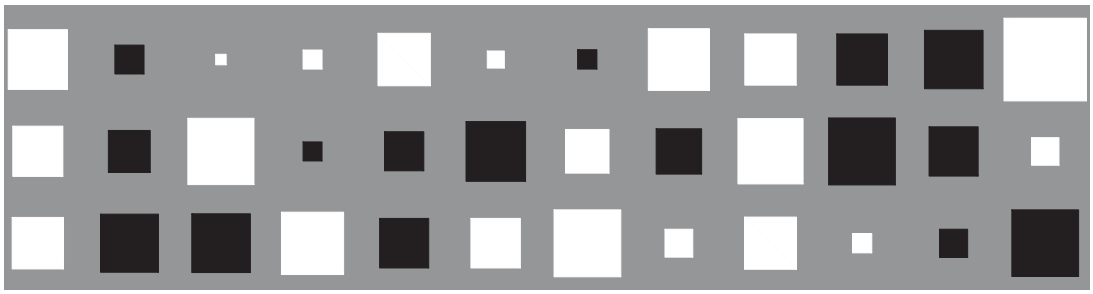
Figure 4.4: The first row is the result from traditional PCA-MM, while the second row is the result from the proposed diversifed PCA-MM.$d = 12, K = 3$

for the proposed diversified PCA-MM exactly the same as set in the baseline algorithm - PCA-MM.

Automatic determine dominating PCs: We fixed $K = 3$ which is supposed to be the true number of mixing components, and $d = 12$ which may be higher than what the synthetic dataset is actually required to be reconstructed. The reconstructed mean parameters from baseline algorithm PCA-MM and the proposed diversified PCA-MM are shown in Figure 4.4. Comparing to the true noisy data sample, both of them achieve adequate results as shown in the first column. However, their conditional likelihoods shown in the second column indicate that the baseline algorithm PCA-MM accomplishes slightly better result than the proposed diversified PCA-MM. We attributes its better performance to its full employment of high-dimensional hidden spaces, whose dimensionalities are determined by transformation matrix cardinalities, which are indirectly controlled by magnitudes of diagonal quality features $\{\Phi\}$. The diagonal values



(a) Traditional PCA-MM

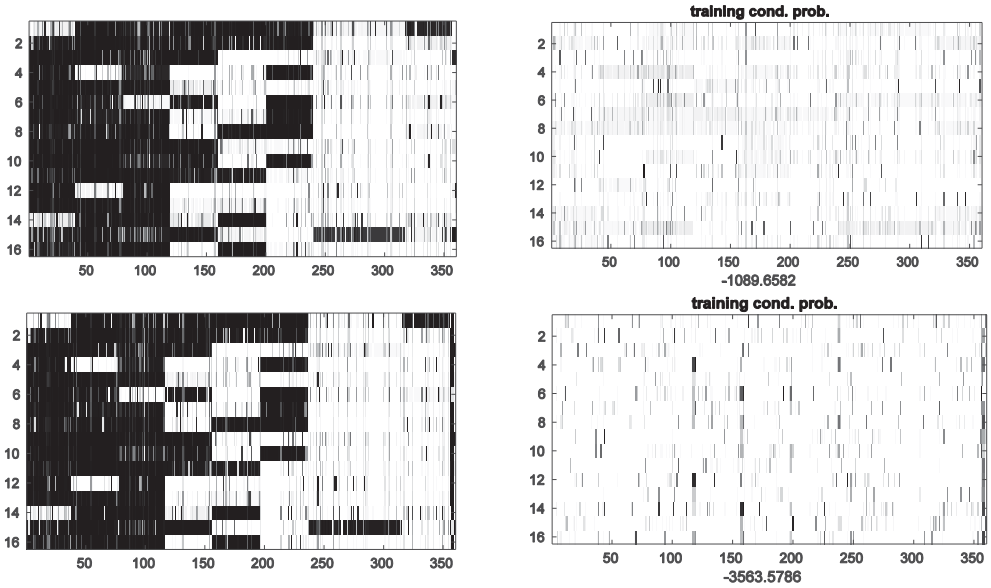


(b) Diversified PCA-MM

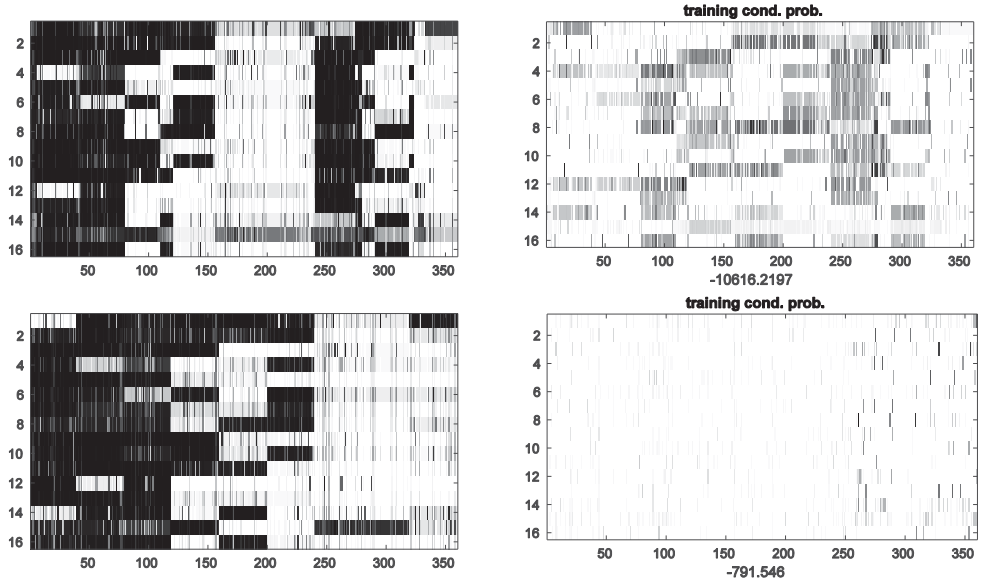
Figure 4.5: Hinton diagram of diagonal of $\{\Phi^k\}_{k=1}^K$, where white boxes indicate positive values, and black ones indicate negative values. The magnitudes of Φ^k's are symbolized by the sizes of boxes.

of quality features for both traditional and diversified PCA-MM are visualized with Hinton diagram and shown in Figure 4.5. It can be easily seen that the Φ values for the traditional PCA-MM are quite the same indicated with the sizes of the boxes, while the Φs values for the diversified PCA-MM are varies from extremely small sizes to quite large sizes. In other words, the former model simply employs all available PCs to reconstruct the training data, while the latter model, the proposed one, automatically chooses only dominating PCs to achieve reconstructing tasks in despite of slight performance sacrifice. In conclusion, the proposed model with ℓ_1 constraints over Φs arms itself with the power of automatically determining the dominant PCs in the mixture models.

Diversity verification: We fixed $d = 4$ and varied the number of mixing component K to verify the impact of diversity prior on model redundancy reduction.



(a) K=3



(b) K=2

Figure 4.6: Illustrate effectiveness of diversity over model redundancy reduction with fixed $d = 4$ and $K = 3$ and $K = 2$ for both traditional PCA-MM (the first rows) and diversified PCA-MM (the second rows).

We compare reconstructing results of the proposed diversified PCA-MM with $\lambda = 1000$ to the results of traditional PCA-MM with $\lambda = 0$ in situations when $K = 2$ and $K = 3$. The reconstructed mean parameters and their training conditional likelihoods are shown in Figure 4.6. From this figure, when $K = 3$, both models (the baseline model results shown in the first row in Figure 4.6a, and the proposed diversified PCA-MM results in its second row) achieve promising and comparable results, with the traditional PCA-MM slightly better than ours in terms of conditional likelihood (again we attribute the inferior of our model to less dominating PCs); however, when $K = 2$, the traditional PCA-MM can not reconstruct the training noisy, but our proposed diversified version achieves the task perfectly, as supported by both qualitative result (shown in left-bottom of Figure 4.6b) and quantitative result (shown in right-bottom of Figure 4.6b). This can be explained by that due to potential overlap amongst learned mixing components of traditional PCA-MM, it needs more mixing components to cover the entire data space. In contrast, by introducing diversity prior, our model has effectively produced more compact model to reconstruct the noisy data. We conclude that when fixed $d = 4$, the best number of mixing component required to reconstruct the training data is 2, namely $K = 2$.

4.6.2 Real-world Dataset experiment

USPS digits:

We tested our model by conducting experiments on a real-world dataset, i.e., the USPS hand-written digits. We choose to use part of this dataset to train the diversified mixture models, as the same with Li and Tao (2013): Images from three digit categories of $2, 3$ and 4 compose our training dataset. Two hundred images of each category are randomly selected, and the size of each binary image

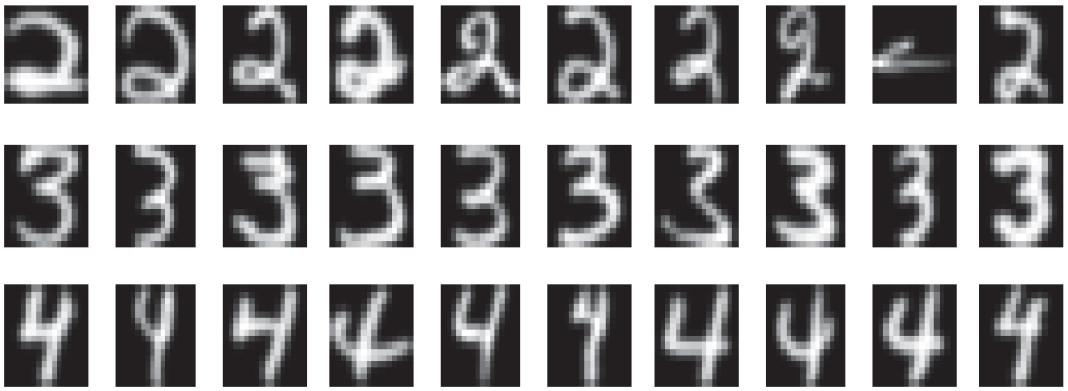


Figure 4.7: Samples from USPS digits $2, 3, 4$.

is 16×16. Sample images from this dataset are shown in Figure 4.7.

We fixed $\xi = 5e-3, \rho = 0.1$. These two parameters are roughly set according to Eq. (4.12), where we need to keep the value inside an exponential operator neither too big nor too small to implement algorithms properly. In addition, we fixed $K = 3$, $d = 2, \lambda = 100$. Regarding initalization, again we set π with a normal distribution, and Φ and Υ randomly. The fitted three local PCA transformation matrices are shown in Figure 4.8. It can be easily seen that with extremely small hidden space $d = 2$, our diversified scheme is still able to capture the main shape for each digit represented by each mixing component, as plotted in each column.

Cora:

We also test our model in a text clustering scenario. The dataset we applied is the Cora dataset [1]. It consists of machine learning papers and two categories out of seven, i.e., Case-Based and Genetic-Algorithms, are chosen to train the mixture model. 200 papers are randomly selected from each category as training samples, and 50 papers from the rest are treated as testing samples. Each

[1]http://linqs.umiacs.umd.edu/projects/projects/lbc/

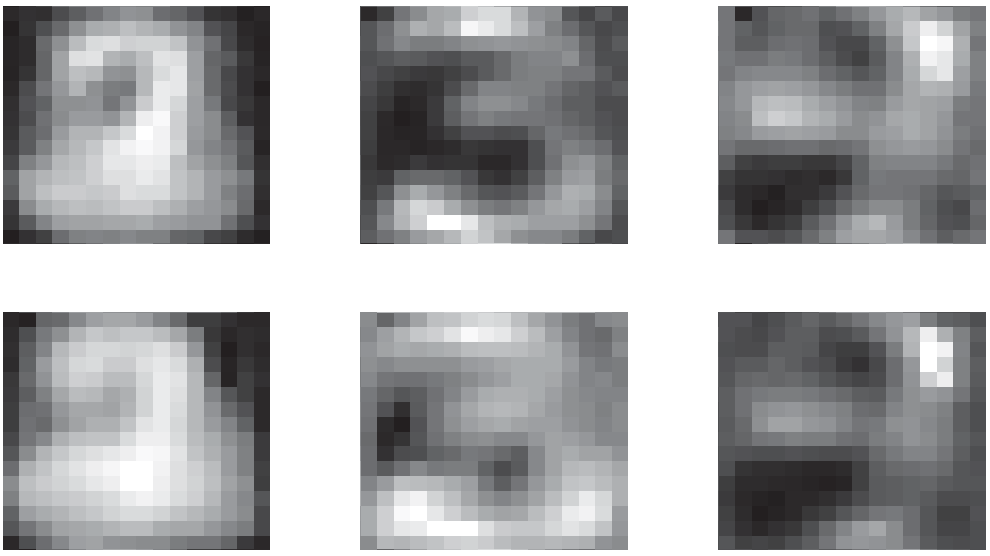
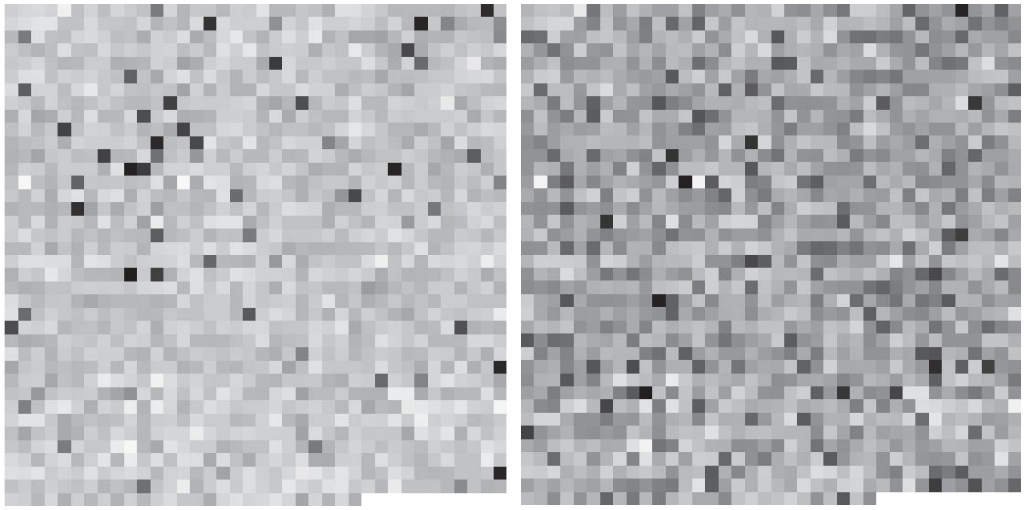


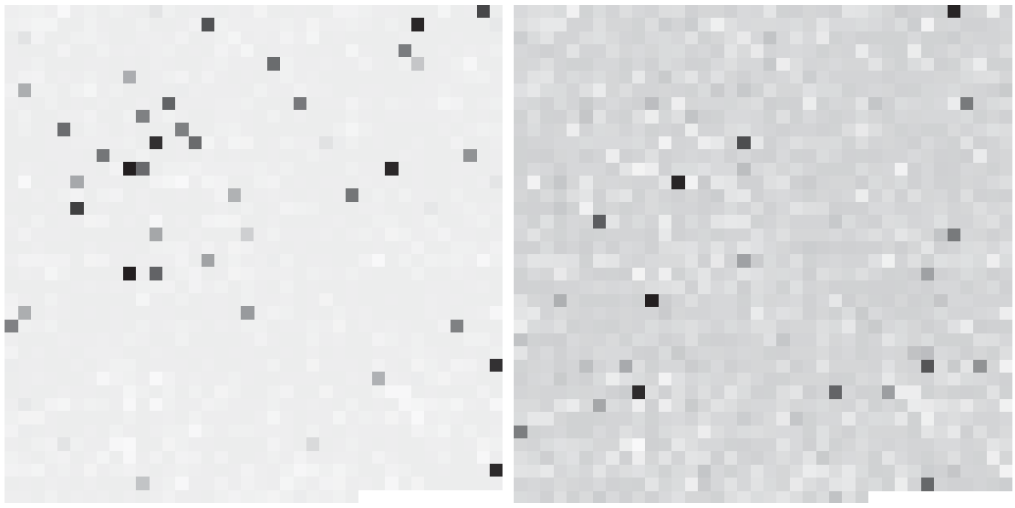
Figure 4.8: The fitted three local PCA transformation matrices of our diversified mixture models with $K = 3, d = 2$. Each row represents one PC.

paper is represented by a high-dimensional vector - 1433 binary vector, where each dimension stands for a unique word with value 1 showing existence of its corresponding unique word while value 0 representing its non-existence.

The experimental setting for our model is: $\xi = 1e-3, \rho = 0.1, K = 2, d = 2, \lambda = 1e5$, and for the traditional mixture model, $\lambda = 0$. The dominant PCs of the fitted transformation matrices for both the traditional setting and the diversified setting are shown in Figure 4.9. Comparatively, the PCs of two mixing components of the diversified model are not only diverse with each other, but also as succinct as possible, as most of their values are much smaller than the dominant ones (shown as dark black pixels). Further, the effectiveness of the diversity is verified by clustering accuracy of test papers where the diversified version improves the baseline by 4.25%. We attribute this superiority to the fact that the true representations of text clusters repel with each other.



(a) Learned from the traditional mixture model with test clustering accuracy 54.5%.



(b) Dominant PCs of two mixing components learned from the diversified PCA-MM, and its test clustering accuracy is 58.75%.

Figure 4.9: Dominant PCs of two mixing components with dark black pixels indicating larger absolute values than light grey pixels.

4.7 Summary

In this chapter, we have proposed a diversified exponential family PCA mixture models. The proposed model is considered powerful due to three advantages. First, it is capable of handling with general data types such as binary and integer, rather than limited to continuous data type. Second, the mixing development makes it competent to deal with complex general data-typed structures. Third, the diversified extension effectively alleviates one of the potential drawbacks mixture models exhibit, namely, inferred overlapping mixing components. The third advantage is achieved by this chapter. How to explicitly encode diverse prior over mixing components of SePCA-MM and how to solve the diversified model are our main contributions. In addition, due to the proposed diversifying strategy, the proposed model provides a straightforward way to do model selection, namely, determining the effective number of PCs of each mixing PCA component. Empirical results verify the effectiveness of the proposed model. Our future work will mainly focus on solving the proposed model using more accurate approximations such as Monte Carlo methods and variational inference methods.

Chapter 5

Conclusion and Further Study

This chapter first concludes the whole thesis, and then proposes several further research directions of the topic.

5.1 Conclusions

This thesis proposes a diversified framework to extend traditional probabilistic graphical models to alleviate potential overlapping problems. Different formats of diverse priors for different forms of probabilistic graphical models (PGMs) have been designed in this thesis. By explicitly encoding these diverse priors into traditional PGMs, three different diversified PGMs have been elaborated and applied to various application scenarios.

In Chapter 2, diversified hidden Markov models have been proposed based on the assumption that the state-transition probabilities should be diversified to reinforce the discriminability amongst hidden states. Therefore, the diversity is imposed over transition matrix of each HMM. In addition, to satisfy the normalization constraint of each row of transition matrix, a probability-valued DPP measure is applied to construct the diverse prior. The resulted diversified

HMMs are solved in an EM-framework and its performance is verified by competitive results when comparing to the state-of-the-arts in sequential labelling application scenarios, i.e., PoS tagging and OCR.

In Chapter 3, a time-varying determinantal point processes (TV-DPP) has been proposed to serve the purpose of diverse sequential subset recommendation. Diversity is naturally required by news readers or other information extractors not only at one specific time stamp but also along a timeline. Therefore, a diversity prior should be imposed over both individual data subsets as well as neighbouring data subsets along the timeline. By taking advantage of one observation that the data changes between time stamps are relatively small, this chapter proposes a fast sampling algorithm based on sequential Monte Carlo techniques for diverse subset selection. Empirical results on diverse news recommendation and Enron Corpus communication network evolving confirm the efficiency of the proposed diverse scheme.

In Chapter 4, diversified exponential family principal component analysis mixture models have been proposed with the intention of reducing overlapping amongst mixing components. Straightforwardly, a diverse prior is imposed over mixing components of the mixture models to encourage repulsiveness amongst them. As each mixing component is actually a probabilistic PCA (PPCA), a transformation-matrix-valued diverse prior has been designed. An approximation EM-framework based on Jensen's inequality has been derived for parameter learning and inference. Experimental results on both a synthetic dataset and a real handwritten digits image dataset confirm the effectiveness of the proposed model.

5.2 Further study

Although the presented diversified models in this thesis have achieved promising results to some extent, some issues still remain open and need to be explored further as future studies:

- The parameter learning for the proposed diversified PGMs is actually point estimation. Bayesian estimation will be an important extension for current version, which will inherit all advantages from Bayesian inference, such as avoiding overfitting. However, one tough nut to crack is how to efficiently and accurately calculate the non-conjugate prior-likelihood objective of the diversified PGMs. One can resort to approximation algorithms, such as efficient variational inference, accurate sampling approximation algorithms.

- The experimental results demonstrate that the proposed diversified framework has achieved initial success in three different specific PGMs, namely, HMMs, TV-DPPs and PPCA. In the light of this, expanding this framework to more other forms of PGMs which exhibit similar problems as existing in these three will expect to effectively alleviate the drawback of these models and improve their performance.

- The proposed diversified framework has already been successfully applied to various application scenarios, from optical character recognition (OCR), part-of-speech tagging (PoS tagging), news recommendation to hand-written digits reconstruction. However, as mentioned in Introduction chapter, diversity is favoured by a extremely large amount of applications. Therefore, it is believed that the proposed diversified framework will definitely benefit more application scenarios to be explored.

REFERENCES

Abel, F., Gao, Q., Houben, G.-J. and Tao, K. (2013), Twitter-based user modeling for news recommendations, *in* 'Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence (IJCAI)', pp. 2962–2966.

Affandi, R. H., Fox, E. B., Adams, R. P. and Taskar, B. (2013), Nyström approximation for large-scale determinantal processes, *in* 'Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics (AISTATS)'.

Affandi, R. H., Fox, E. B. and Taskar, B. (2013), Approximation inference in continuous determinantal point processes, *in* 'Proceedings of Advances in Neural Information Processing Systems (NIPS)'.

Akaho, S. (2008), Dimension reduction for mixtures of exponential families, *in* 'International Conference on Artificial Neural Networks (ICANN)', Springer, pp. 1–10.

Amer-Yahia, S., Bonchi, F., Castillo, C., Feuerstein, E., Mendez-Diaz, I. and Zabala, P. (2014), 'Composite retrieval of diverse and complementary bundles', *IEEE Transactions on Knowledge and Data Engineering (T-KDE)* .

Anaya-Izquierdo, K. and Marriott, P. (2007), 'Local mixture models of exponential families', *Bernoulli* pp. 623–640.

Anzai, Y. (2012), *Pattern recognition & machine learning*, Elsevier.

Ardeshiri, T., Özkan, E. and Orguner, U. (2013), 'On reduction of mixtures of the exponential family distributions'.

Asai, K., Hayamizu, S. and Handa, K. (1992), 'Prediction of protein secondary structure by the hidden markov model'.

Batmanghelich, N. K., Quon, G., Kulesza, A., Kellis, M., Golland, P. and Bornn, L. (2014), 'Diversifying sparsity using variational determinantal point processes', *arXiv:1411.6307* .

Bicego, M., Cristani, M. and Murino, V. (2007), Sparseness achievement in hidden markov models, *in* 'Proceedings of fourteenth International Conference on Image Analysis and Processing', pp. 67–72.

Bishop, C. M., Svensén, M. and Williams, C. K. (1998), 'Gtm: The generative topographic mapping', *Neural Computation* **10**(1), 215–234.

Bishop, C. M. et al. (2006), *Pattern recognition and machine learning*, Vol. 4, Springer New York.

Borgatti, S. P. (2002), 'Netdraw software for network visualization', *Lexington, KY: Analytic Technologies* p. 95.

Borodin, A. and Rains, E. M. (2005), 'Eynard–mehta theorem, schur process, and their pfaffian analogs', *Journal of Statistical Physics* **121**(3-4), 291–317.

Candès, E. J., Li, X., Ma, Y. and Wright, J. (2011), 'Robust principal component analysis', *Journal of the ACM (JACM)* **58**(3), 11.

Chen, P. and Hsu, L. (1991), 'On a sequential subset selection procedure for the least probable multinomial cell', *Stochastic Analysis and Applications* **20**(9), 2845–2862.

Collins, M., Dasgupta, S. and Schapire, R. E. (2001), A generalization of principal components analysis to the exponential family, *in* 'Proceedings of Advances in Neural Information Processing Systems (NIPS)', pp. 617–624.

Del Moral, P., Doucet, A. and Jasra, A. (2006), 'Sequential monte carlo samplers', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **68**(3), 411–436.

Diesner, J. and Carley, K. M. (2005), Exploration of communication networks from the enron email corpus, *in* 'SIAM International Conference on Data Mining: Workshop on Link Analysis, Counterterrorism and Security', Citeseer, Newport Beach, CA.

Diesner, J., Frantz, T. L. and Carley, K. M. (2005), 'Communication networks from the enron email corpus: "it's always about the people. enron is no difference."', *Computational & Mathematical Organization Theory* **11**(3), 201–228.

Du, Q. and Fowler, J. E. (2007), 'Hyperspectral image compression using jpeg2000 and principal component analysis', *Geoscience and Remote Sensing Letters* **4**(2), 201–205.

Fang, M., Yin, J. and Tao, D. (2014), Active learning for crowdsourcing using knowledge transfer, *in* 'Proceedings of the Twenty Eighth AAAI Conference on Artificial Intelligence (AAAI)'.

Fang, Y., Wang, R., Dai, B. and Wu, X. (2015), 'Graph-based learning via auto-grouped sparse regularization and kernelized extension', *IEEE Transactions on Knowledge and Data Engineering* **27**.

Feng, S. and Manmatha, R. (2006), A hierarchical, hmm-based automatic evaluation of ocr accuracy for a digital library of books, *in* 'Proceedings of the sixth ACM/IEEE-CS joint Conference on Digital Libraries', pp. 109–118.

Figueiredo, M. A. and Jain, A. K. (2000), 'Unsupervised learning of finite mixture models', *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)* **24**, 381–396.

Gael, J. V., Vlachos, A. and Ghahramani, Z. (2009), The infinite hmm for unsupervised pos tagging, *in* 'Proceedings of Conference on Empirical Methods in Natural Language Processing', pp. 678–687.

Geladi, P., Isaksson, H., Lindqvist, L., Wold, S. and Esbensen, K. (1989), 'Principal component analysis of multivariate images', *Chemometrics and Intelligent Laboratory Systems* **5**(3), 209–220.

Gilks, W. R. and Berzuini, C. (2001), 'Following a moving target - monte carlo inference for dynamic bayesian models', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **63**(1), 127–146.

Gillenwater, J., Kulesza, A., Fox, E. and Taskar, B. (2014), Expectation-maximization for learning determinantal point processes, *in* 'Proceedings of Advances in Neural Information Processing Systems (NIPS)', pp. 3149–3157.

Gillenwater, J., Kulesza, A. and Taskar, B. (2012*a*), Discovering diverse and salient threads in document collections, *in* 'Proceedings of Conference on Empirical Methods on Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)'.

Gillenwater, J., Kulesza, A. and Taskar, B. (2012*b*), Near-optimal map inference for determinantal point processes, *in* 'Proceedings of Advances in Neural Information Processing Systems (NIPS)', pp. 2735–2743.

Ginibre, J. (1965), 'Statistical ensembles of complex, quaternion, and real matrices', *Journal of Mathematical Physics* **6**(3), 440–449.

Goldwater, S. and Griffiths, T. L. (2007), 'A fully bayesian approach to unsupervised part-of-speech tagging', *Association of Computational Linguistics* pp. 744–751.

Golub, G. H. and Van Loan, C. F. (2012), *Matrix computations*, Vol. 3, JHU Press.

Gong, B., Chao, W.-L., Grauman, K. and Sha, F. (2014), Diverse sequential subset selection for supervised video summarization, *in* 'Proceedings of Advances in Neural Information Processing Systems (NIPS)', pp. 2069–2077.

Guestrin, C., Krause, A. and Singh, A. P. (2005), Near-optimal sensor placements in gaussian processes, *in* 'Proceedings of the Twenty Second International Conference on Machine Learning (ICML)', ACM, pp. 265–272.

Hartline, J., Mirrokni, V. and Sundararajan, M. (2008), Optimal marketing strategies over social networks, *in* 'Proceedings of the Seventeenth International Conference on World Wide Web', ACM, pp. 189–198.

Hough, J. B., Krishnapur, M., Peres, Y. and Virag, B. (2006), 'Determinantal processes and independence', *Probability Surveys* **3**, 206–229.

Hu, J., Brown, M. K. and Turin, W. (1992), 'Hmm based on-line handwriting recognition'.

Huang, K., Ma, Y. and Vidal, R. (2004), Minimum effective dimension for mixtures of subspaces: A robust gpca algorithm and its applications, *in* 'Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)', Vol. 2, IEEE, pp. II–631.

Iwata, T., Yamada, T., Sakurai, Y. and Ueda, N. (2012), 'Sequential modeling of topic dynamics with multiple timescales', *ACM Transaction on Knowledge Discovery from Data (TKDD)* **5**(4), 19:1–19:27.

Jebara, T., Kondor, R. and Howard, A. (2004), 'Probability product kernels', *Journal of Machine Learning Research (JMLR)* **5**, 819–844.

Johnson, M. (2007), Why doesn't em find good hmm pos-taggers?, *in* 'Proceedings of Joint Conference on Empirical Methods in Natural Language Proceeding and Computational Natural Language Learning', pp. 296–305.

Jolliffe, I. (2002), *Principal component analysis*, Wiley Online Library.

Kambhatla, N. and Leen, T. K. (1997), 'Dimension reduction by local principal component analysis', *Neural Computation* **9**(7), 1493–1516.

Kang, B. (2013), Fast determinantal point process sampling with application to clustering, *in* 'Proceedings of Advances in Neural Information Processing Systems (NIPS)', pp. 2319–2327.

Kantas, N., Doucet, A., Singh, S. S. and J.M.Maciejowski. (2009), An overview of sequential monte carlo methods for parameter estimation in general state-space models, *in* 'Proceedings of the 15th IFAC Symposium on System Identification (SYSID)'.

Kim, G., Xing, E. P., Fei-Fei, L. and Kanade, T. (2011), Distributed cosegmentation via submodular optimization on anisotropic diffusion, *in* 'Proceedings of IEEE International Conference on Computer Vision (ICCV)', IEEE, pp. 169–176.

Kim, H.-C., Kim, D. and Bang, S.-Y. (2001), A pca mixture model with an

efficient model selection method, *in* 'Proceedings of International Joint Conference on Neural Networks (IJCNN)', Vol. 1, IEEE, pp. 430–435.

Klimt, B. and Yang, Y. (2004), The enron corpus: A new dataset for email classification research, *in* 'Proceedings of European Conference on Machine Learning (ECML)', Springer, pp. 217–226.

Kojima, M. and Komaki, F. (2014), Determinantal point process priors for bayesian variable selection in linear regression, *in* 'ArXiv:1406.2100'.

Krevat, E. and Cuzzillo, E. (2006), 'Improving off-line handwritten character recognition with hidden markov models', *IEEE Transactions on Pattern Analysis & Machine Intelligence* **33**.

Krogh, A., Larsson, B., von Heijne, G. and Sonnhammer, E. L. L. (2012), 'Predicting transmembrane protein topology with a hidden markov model: Application to complete genomes'.

Ku, W., Storer, R. H. and Georgakis, C. (1995), 'Disturbance detection and isolation by dynamic principal component analysis', *Chemometrics and Intelligent Laboratory Systems* **30**(1), 179–196.

Kulesza, A. and Taskar, B. (2010), Structured determinantal point processes, *in* 'Proceedings of Advances in Neural Information Processing Systems (NIPS)', pp. 1171–1179.

Kulesza, A. and Taskar, B. (2011*a*), k-dpps: Fixed-size determinantal point processes, *in* 'Proceedings of International Conference on Machine Learning (ICML)', pp. 1193–1200.

Kulesza, A. and Taskar, B. (2011*b*), 'Learning determinantal point processes', *Proceedings of Conference on Uncertainty in Artificial Intelligence (UAI)* .

Kulesza, A. and Taskar, B. (2012), 'Determinantal point processes for machine learning', *Foundations and Trends in Machine Learning* **5**.

Li, J. and Tao, D. (2013), 'Simple exponential family pca', *IEEE Transactions on Neural Networks and Learning Systems (T-NNLS)* **24**(3), 485–497.

Li, Z., Liu, J., Yang, Y., Zhou, X. and Lu, H. (2014), 'Clustering-guided sparse structural learning for unsupervised feature selection', *IEEE Transactions on Knowledge and Data Engineering (T-KDE)* **26**.

Lin, H. and Bilmes, J. A. (2012), 'Learning mixtures of submodular shells with application to document summarization', *arXiv:1210.4871* .

Liu, T. and Tao, D. (2016), 'Classification with noisy labels by importance reweighting', *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)* .

Liu, T., Tao, D., Song, M. and Maybank, S. (2016), 'Algorithm-dependent generalization bounds for multi-task learning', *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)* .

Liu, X. and Chen, T. (2003), Video-based face recognition using adaptive hidden markov models, *in* 'Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)', pp. I–340.

Long, M., Wang, J., Ding, G., Shen, D. and Yang, Q. (2014), 'Transfer learning with graph co-regularization', *IEEE Transactions on Knowledge and Data Engineering (T-KDE)* **26**.

Macchi, O. (1975), 'The coincidence approach to stochastic point processes', *Advances in Applied Probability* pp. 83–122.

MacKay, D. J. (1995), 'Probable networks and plausible predictions - a review of practical bayesian methods for supervised neural networks', *Network: Computation in Neural Systems* **6**(3), 469–505.

Mahantesh, K., Manjunath Aradhya, V. and Niranjan, S. (2014), A study of subspace mixture models with different classifiers for very large object classification, *in* 'International Conference on Advances in Computing, Communications and Informatics (ICACCI)', IEEE, pp. 540–544.

Marcus, M., Santorini, B. and Marcinkiewicz, M. (1993), 'Building a large annotated corpus of english: The penn treebank', *Computational Linguistics* .

Mcauley, J. and Leskovec, J. (2014), 'Discovering social circles in ego networks', *ACM Transaction on Knowledge Discovery from Data (TKDD)* **8**(1), 4:1–4:28.

McCULLAGH, P. (1994), 'Exponential mixtures and quadratic exponential families', *Biometrika* **81**(4), 721–729.

McSherry, D. (2002), Diversity-conscious retrieval, *in* 'Advances in Case-Based Reasoning', Springer, pp. 219–233.

Mitkov, R. (2003), *The oxford handbook of computational linguistics*, Oxford University Press.

Mohamed, S., Ghahramani, Z. and Heller, K. A. (2009), Bayesian exponential family pca, *in* 'Proceedings of Advances in Neural Information Processing Systems (NIPS)', pp. 1089–1096.

Morwal, S., Jahan, N. and Chopra, D. (2012), 'Named entity recognition using hidden markov model (hmm)', pp. 15–23.

Murveit, H. and Moore, R. (1990), Integrating natural language constraints into hmm-based speech recognition, *in* 'Proceedings of International Conference on Acoustics, Speech, and Signal Processing', pp. 573–576.

Niu, F. and Abdel-Mottaleb, M. (2005), Hmm-based segmentation and recognition of human activities from video sequences, *in* 'Proceedings of IEEE International Conference on Multimedia and Expo', pp. 804–807.

Nounou, M. N., Bakshi, B. R., Goel, P. K. and Shen, X. (2002), 'Bayesian principal component analysis', *Journal of Chemometrics* **16**(11), 576–595.

Ohsaka, N., Akiba, T., Yoshida, Y. and Kawarabayashi, K.-i. (2014), Fast and accurate influence maximization on large networks with pruned monte-carlo simulations, *in* 'Proceedings of the Twenty Eighth AAAI Conference on Artificial Intelligence (AAAI)'.

Qi, G.-J., Aggarwal, C. C. and Huang, T. S. (2013), Online community detection in social sensing, *in* 'Proceedings of the Sixth ACM International Conference on Web Search and Data Mining', ACM, pp. 617–626.

Rabiner, L. R. (1989), A tutorial on hidden markov models and selected applications in speech recognition, *in* 'Proceedings of the IEEE', Vol. 77, pp. 257–286.

Rakthanmanon, T., Campana, B., Mueen, A., Batista, G., Westover, B., Zhu, Q., Zakaria, J. and Keogh, E. (2013), Data mining a trillion time series subsequences under dynamic time warping, *in* 'Proceedings of the Twenty Third International Joint Conference on Artificial Intelligence (IJCAI)', AAAI Press, pp. 3047–3051.

Rao, B. D., Engan, K., Cotter, S. F., Palmer, J. and Kreutz-Delgado, K. (2003), 'Subset selection in noise based on diversity measure minimization', *IEEE Transactions on Signal Processing (T-SP)* **51**(3), 760–770.

Reichart, R. and Korhonen, A. (2013), Improved lexical acquisition through dpp-based verb clustering, *in* 'Association for Computational Linguistics (ACL) (1)', pp. 862–872.

Rocková, V. and George, E. I. (n.d.), 'Determinantal priors for variable selection'.

Sahlin, K. (2011), Estimating convergence of markov chain monte carlo simulations, Master's thesis, Stockholm University.

Sandvik, A. W. (2013), Monte carlo simulations in classical statistical physics, Technical report, Department of Physics, Boston University.

Schölkopf, B., Smola, A. and Müller, K.-R. (1997), Kernel principal component analysis, *in* 'International Conference on Artificial Neural Networks (ICANN)', Springer, pp. 583–588.

Sha, F. and Saul, L. K. (2006), Large margin hidden markov models for automatic speech recognition, *in* 'Proceedings of Advances in Neural Information Processing Systems (NIPS)', pp. 1249–1256.

Shah, A. and Ghahramani, Z. (2013), Determinantal clustering process - a nonparametric bayesian approach to kernel based semi-supervised clustering, *in* 'Proceedings of Conference on Uncertainty in Artificial Intelligence (UAI)', Citeseer, p. 566.

Shetty, J. and Adibi, J. (2004), 'The enron email dataset database schema and brief statistical report', *Information Sciences Institute Technical Report, University of Southern California* **4**.

Snoek, J. and Adams, R. P. (2013), A determinantal point process latent variable model for inhibition in neural spiking data, *in* 'Proceedings of Advances in Neural Information Processing Systems (NIPS)', pp. 1932–1940.

Soleymani, F. (2012), 'A rapid numerical algorithm to compute matrix inversion', *International Journal of Mathematics and Mathematical Sciences* **2012**.

Tao, L., Elhamifar, E., Khudanpur, S., Hager, G. D. and Vidal, R. (2012), Sparse hidden markov models for surgical gesture classification and skill evaluation, *in* 'Proceedings of International Conference on Natural Language Processing and Knowledge Engineering', pp. 167–177.

Taskar, B., Guestrin, C. and Koller, D. (2003), Max-margin markov networks, *in* 'Proceedings of Advances in Neural Information Processing Systems (NIPS)', p. 25.

Teh, Y. W., Jordan, M. I., Beal, M. J. and Blei, D. M. (2006), 'Hierarchical dirichlet processes', *Journal of the American Statistical Association* **101**(476).

Tibshirani, R. (1992), 'Principal curves revisited', *Statistics and Computing* **2**(4), 183–190.

Tipping, M. E. and Bishop, C. M. (1999*a*), 'Mixtures of probabilistic principal component analyzers', *Neural Computation* **11**(2), 443–482.

Tipping, M. E. and Bishop, C. M. (1999*b*), 'Probabilistic principal component analysis', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **61**(3), 611–622.

Tollefson, E., Goldsman, D., Kleywegt, A. and Tovey, C. (2014), 'Optimal selection of the most probable multinomial alternative', *Sequential Analysis* **33**(4), 491–508.

Turaga, D. S. and Chen, T. (2002), Face recognition using mixtures of principal components, *in* 'Proceedings of International Conference on Image Processing (ICIP)', Citeseer, pp. 101–104.

Wang, Q. I. and Schuurmans, D. (2005), Improved estimation for unsupervised part-of-speech tagging, *in* 'Proceedings of International Conference on Natural Language Processing and Knowledge Engineering', pp. 219–224.

Wang, S., Zhang, C., Qian, H. and Zhang, Z. (2014), Using the matrix ridge approximation to speedup determinantal point processes sampling algorithms, *in* 'Proceedings of the Twenty Eighth AAAI Conference on Artificial Intelligence (AAAI)'.

Wang, W. and Carreira-Perpinán, M. A. (2013), Projection onto the probability simplex: An efficient algorithm with a simple proof and an application, *in* 'arXiv:1309.1541'.

Wang, X. and Tang, X. (2005), Subspace analysis using random mixture models, *in* 'Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)', Vol. 1, IEEE, pp. 574–580.

Watanabe, K., Akaho, S., Omachi, S. and Okada, M. (2009), 'Variational bayesian mixture model on a subspace of exponential family distributions', *IEEE Transactions on Neural Networks (T-NN)* **20**(11), 1783–1796.

Wu, F., Zilberstein, S. and Jennings, N. R. (2013), Monte-carlo expectation maximization for decentralized pomdps, *in* 'Proceedings of the Twenty Third International Joint Conference on Artificial Intelligence (IJCAI)', AAAI Press, pp. 397–403.

Wu, H. C. and Luk., R. W. P. (2008), 'Interpreting tf-idf term weights as making relevance decisions', *ACM Transactions on Information Systems (T-IS)* **26**(3), 13:1–13:37.

Xie, P., Zhu, J. and Xing, E. (2016), Diversity-promoting bayesian learning of

latent variable models, *in* 'Proceedings of International Conference on Machine Learning (ICML)'.

Xiong, H., Liu, T. and Tao, D. (2016), Diversified dynamical gaussian process latent variable model for video repair, *in* 'Proceedings of the Thirties AAAI Conference on Artificial Intelligence (AAAI)', pp. XX–XX.

Xu, C., Tao, D. and Xu, C. (2015), 'Multi-view intact space learning', *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)* .

Xuan, J., Lu, J., Zhang, G. and Luo, X. (2015), 'Topic model for graph mining', *IEEE Transactions on Cybernetics* **45**(12), 2792–2803.

Yang, S., Yi, Z., Ye, M. and He, X. (2014), 'Convergence analysis of graph regularized non-negative matrix factorization', *IEEE Transactions on Knowledge and Data Engineering (T-KDE)* **26**.

Yang, Z. and Rannala, B. (2010), 'Bayesian species delimitation using multilocus sequence data', *Proceedings of the National Academy of Sciences* **107**(20), 9264–9269.

Zhang, F. (2004), A mixture probabilistic pca model for multivariate processes monitoring, *in* 'Proceedings of the American Control Conference', Vol. 4, IEEE, pp. 3111–3115.

Zhao, J. (2014), 'Efficient model selection for mixtures of probabilistic pca via hierarchical bic', *IEEE Transactions on Cybernetics* **44**(10), 1871–1883.

Zhong, E., Fan, W. and Yang, Q. (2014), 'User behavior learning and rransfer in composite social networks', *ACM Transaction on Knowledge Discovery from Data (T-KDD)* **8**(1), 6:1–6:32.

Zou, H., Hastie, T. and Tibshirani, R. (2006), 'Sparse principal component analysis', *Journal of Computational and Graphical Statistics* **15**(2), 265–286.

Zou, J. Y. and Adams, R. P. (2012), Priors for diversity in generative latent variable models, *in* 'Proceedings of Advances in Neural Information Processing Systems (NIPS)', pp. 2996–3004.