Models and Inference with Network Structure

by

Brandon L. Oselio

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering - Systems)
in the University of Michigan
2019

Doctoral Committee:

Professor Alfred O. Hero III, Chair
Assistant Professor Walter Dempsey
Assistant Professor Daniel Romero
Associate Professor Vijay Subramanian

Brandon L. Oselio

boselio@umich.edu

ORCID iD: 0000-0001-5729-9929

To my family, who supports me regardless of the endeavor.

ACKNOWLEDGMENTS

This work, as with most research, is an effort in collaboration. I would first like to thank my advisor, Al Hero, for providing exceptional technical guidance and lending his expertise to all the problems faced while tackling these research problems. His ability to persevere through difficulties and ask the right questions are abilities that I will continue to aspire to throughout my career.

Many thanks go out to my colleagues, both past and present, in the Hero lab for their support throughout the years. I would particularly like to thank Salimeh Yasai Sekeh, as without our frequent discussions and collaborations, Chapters 3 and 4 of this thesis would not have been possible. Further, Kevin Xu provided data, code, and suggestions early in my research journey that proved invaluable.

I would like to thank Walter Dempsey, who is a collaborator and member of my committee. He was a great technical guide for the more difficult aspects of exchangeable models as we navigated the world of networks and interaction data together. I would also like to thank both Vijay Subramanian and Daniel Romero for being on my committee and sharing their insights on my research; your valuable comments significantly strengthened the finished product.

Finally, I would like to thank my friends who have helped me throughout my time at the University of Michigan, included but not limited to Kelsey, Adriana, Bryan, Sofia, and Elizabeth. Without you all, the time in Ann Arbor would've been much less enjoyable.

The support and understanding of my parents, both those in New Mexico and Wisconsin, was indenspensible in keeping me grounded, well fed, and just generally more content than what would've been otherwise possible.

Most importantly, I would like to thank my future husband, Marcus, for being there every step of the way. I can't wait for what's next.

PREFACE

This thesis concerns models and procedures for two types of data: network valued data, and data that has latent network structure of interest. It provides advancements in the modeling of data with these structures, taking into account both theoretical and practical considerations. Finally, it is bolstered with a number of empirical results on a variety of datasets.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

ABSTRACT

In this thesis, the focus is on data that has network structure and on problems that benefit from the application of network-based algorithms. We focus on four research problems of interest: scalable and realistic models for network valued data, graph-based estimation of information theoretic quantities, summarization of complex time-varying data using dynamic graphs, and finally community detection on large multi-layer networks. This work advances the state-of-the-art in several directions. First, it introduces a new framework for complex network interaction data using the concept of edge exchangability. Second, it obtains new tight bounds for the multi-class Bayes error rate based on a graph-based technique, specifically the minimal spanning tree. Third, it introduces a new estimation method for Henze-Penrose divergence, a quantity relevant for graph-based multi-class classification. Fourth, it introduces adaptive directed information for estimating directed interaction networks. Fifth, the thesis presents a comprehensive approach to multi-layer network community detection. Throughout, examples are provided using real datasets, such as the Enron email dataset, an arXiv dataset, and Twitter.

CHAPTER 1

Introduction

Complex, structured data is ubiquitous in both industrial and academic settings and has elicited a commensurate interest in utilizing structured data to inform inference and decisions. Often, this underlying structure can be exploited to make inference and summarization procedures more tractable. In the context of large datasets, it is imperative to consider the data in the context of this structure to build parsimonious models that represent the data well and provide theoretically grounded inference procedures. Similarly, searching for underlying structure can help to summarize the data more efficiently and find relevant attributes of the data of interest that might otherwise go undetected. In other datasets, the structure is explicit, and thus requires careful consideration when reasoning about modeling decisions.

In this thesis, the focus is on data that has network structure and on problems that benefit from the application of network based algorithms. In both cases, we are concerned with data that can be summarized with a relational structure of constituent nodes and edges, appropriately defined based on the context of the problem. Below, four research areas are introduced that utilize network structure that form the backbone of this thesis.

1.1 Models for network data

Models for network data are of increasing interest in the machine-learning, complex systems, and statistical literature [88], [106], [123], [165]. These models are concerned with

data that is in the form of sets of nodes and edges; we call this network-valued data. In most models, the network is reduced to an adjacency matrix representation, with theoretical results given when the number of nodes grows large. Here, the node is the sampling unit of interest.

For models to be useful, it is vital that they reflect observed properties of real-world networks, such as sparsity, clustering, and power-law or scale-free behavior. Many network models, such as the stochastic block model [107] and related variants, are node-exchangeable [6], meaning that the permutation of the labels of the nodes does not change the likelihood of the observed network. As a result of this property, it can be shown that data generated from these models cannot be sparse or capture scale-free behavior, both of which are often seen in real-world networks.

Recently, an alternative approach has emerged for modeling network data by treating the edge, or interaction, as the statistical sampling unit. This is particularly natural in the context of many datasets, such as email networks, social networks, and co-authorship networks. For such data, it is valuable to impose exchangeability on the edges as opposed to the nodes. These edge exchangeable models have generated recent attention [38], [62], [113] and are able to exhibit both sparsity and power-law behavior [62].

Previous edge exchangeable models cannot account for complex hierarchical structure in interactions. For instance, an email has a hierarchical structure induced by a sender and its many receivers, and a movie can be thought of as an interaction with multiple sets of individuals, such as directors, actors, and screenwriters. Chapter 2 introduces an interaction exchangeable framework for these more complicated structures.

1.2 Graph based estimation of information theoretic quantities

The estimation of information theoretic quantities is a problem of interest arising not only in information theory [145], but also feature selection [225], structure estimation for graphical models [13], and training [8], [104] and understanding [222] deep neural networks. Common examples of these quantities include Shannon mutual information and KL divergence. In general, these quantities can be difficult to compute for continuous random variables, and most methods rely on an intermediate density estimator for the underlying distribution. These "plug-in" estimators suffer from bias near the support boundaries of the marginal densities and can be computationally prohibitive. A recent relevant approach by [160] utilizes kernel density estimation along with a novel ensemble method to reduce the bias exhibited in non-ensemble approaches. However, it is computationally intensive, making its use untenable in large data situations.

Graph based estimation methods aim to skip the density estimation stage and directly estimate the quantity of interest by calculating a graph structure over the data sample, such as a minimal spanning tree [99] or k-NN graph [171]. One of the original algorithms in this area was for the estimation of Henze-Penrose (HP) divergence [81], [99], which is a member of the broad class of f divergences. This divergence measure has two important properties. First, it is possible to estimate the HP divergence directly from a minimal spanning tree across the data sample. As the minimal spanning tree can be computed in $O(n \log n)$, this approach to estimation is amenable to large datasets. Second, tight bounds have been proved that relate the HP divergence to the Bayes error rate of a binary classification problem. Thus, accurate estimation of the HP divergence allows for learning of the intrinsic hardness of the supervised learning task. Although tight bounds exist for the Bayes error rate for a binary classification task, this is not true for multi-class classification. In Chapter 3, we derive tight bounds for the multi-class case using a generalized

Henze-Penrose measure, and provide a graph based estimation procedure using a minimal spanning tree.

Like other estimation methods, graph based estimation methods are subject to the curse of dimensionality. Specifically, as the feature dimension of the data increases, the bias of the estimate also increases. In Chapter 4 the thesis introduces a new estimator of the HP divergence, based on a different computed graph, that reduces the bias for growing dimension.

1.3 Dynamic summarization of interacting agents

Networks have also been used to describe the influence and dependence between agents or features. A popular example of this is the partial correlation graph for Gaussian graphical models [126], and the corresponding algorithms used for estimation, such as graphical lasso type algorithms and thresholding approaches [76], [102]. These models seek to uncover a parsimonious dependence representation of the data with computational complexity that is scalable in high feature dimension. These models and quantities often assume, however, that each sample is i.i.d.; in many instances, we are interested in understanding the relationships between time series data that may contain complex causal dependencies across time. Examples of this type of model include spatio-temporal covariance modeling [90].

A quantity used to measure dependence across time is directed information [148]. Unlike partial correlation, directed information is assymmetric and time dependent. It was originally created as a generalization of Shannon mutual information for a channel with feedback [148]. Directed information allows for a more rich understanding of the influence and behavior among agents. Directed information is also closely related to Granger causality [11].

Work has been done to estimate directed information in the context of discrete Markov processes [115], and generalized linear models [192]. However, these graphs are generally

considered fixed, and the time series stationary. In Chapter 5, this thesis introduces adaptive directed information (ADI) to account for time varying signals, and demonstrates specific forms of ADI that elicit computationally efficient estimation procedures. In Chapter 6, we introduce an ensemble of ADI estimators that is more robust to the choice of parameters and to the type of time varying signal.

1.4 Information extraction from multi-layer networks

We often find heterogeneous relationships in data, reflected in more than one type of relationship between agents [120]. These types of relationships may impose different topological properties. For instance, in a social network context, people may be connected by more than one social platform. Alternatively, we may observe explicit links between agents but also infer implicit affinities based on agent features. Multi-layer networks can be used to account for this additional complexity.

A multi-layer network is a network where a set of nodes are connected by intra-layer and inter-layer edges. This structure is a generalization of single-layer networks, where there are only intra-layer relationships. These layers represent heterogeneity in the structure or labeling of the data; a layer might correspond to a type of connection, or a snapshot of the network at a specific time. The inter-layer structure represents ties among nodes in the different layers; this structure may be observed, assumed, or estimated depending on the application. The inter-layer structure in a social network often preserves the labels of the nodes, so that each node in a single layer is connected to its unique counterpart in the other layers. If the layers represent timesteps at any time instant, each entity might be connected to its counterpart in layers before and after the present layer, which represents the localization of that layer's characteristics in time.

As the multi-layer structure is more complicated than its single-layer counterpart, methods for single-layer analysis must be modified, and new methods can be developed specifi-

cally for multi-layer networks. In Chapter 7, this thesis provides a framework for modeling multi-layer networks, concentrating on centrality measures and community detection. In Chapter 8 the thesis provides a novel multi-layer community detection approach that selects approximately Pareto-optimal partitions between nodes.

Dynamic networks can be thought of as a special case of multi-layer networks. However, it is also possible to have a dynamic multi-layer network, where each layer is evolving concurrently over time. In Chapter 9, this thesis proposes a multi-layer summarization procedure for this case based on the dynamic stochastic blockmodel [234], which allows for an efficient representation of the dynamic network.

1.5 Outline and contributions of the thesis

This section lists the chapters and corresponding contributions in this thesis. Each chapter aims to be a self contained exposition on a specific topic; as a result, some introductory material for particular chapters are similar in scope.

Chapter 2 describes a new hierarchical model for edge exchangeable network data. Here, we consider incoming interactions that are a structured collection of nodes. The primary motivating example is an email dataset, where each interaction is a sent email that contains a sender and a set of recievers. We show that by allowing for heterogeneous behavior among senders and partially pooling global information, we are able to improve model fits for real-world data when compared to a non-hierarchical edge exchangeable model. We call our model the hierarchical vertex components model (HVCM), and test it on the Enron email dataset as well as an arXiv dataset.

Chapter 3 defines a novel generalization of the Henze-Penrose divergence [99] in the context of a multi-class classification problem. This measure is then used to obtain upper and lower bounds on the multi-class Bayes error, and these bounds are shown to be provably tighter than state-of-the-art bounds based on pairwise Henze-Penrose divergence

between classes. Further, it is possible to estimate the proposed measure with a single global minimal spanning tree, which makes the proposed bounds more computationally efficient to compute than the pairwise bounds.

Chapter 4 introduces a novel estimator for the Henze-Penrose divergence [99] using a graph other than the traditional minimal spanning tree (MST). Specifically, we find an optimal weighted matching between labeled data points. From this graph we create a statistic that is able to estimate the divergence, and further bound the Bayes error for a binary classification problem, similar to the method that Friedman and Rafsky used with the minimal spanning tree [81]. We demonstrate improvement in high dimensions over the MST statistic.

Chapter 5 develops a method for generating dynamic influence networks between agents with features that are nonstationary. Using adaptive directed information (ADI), dyadic influence is measured between entities. We demonstrate an approach to the estimation of ADI, and apply our method to a Twitter dataset centered around the 2016 US presidential election, and to a Twitter dataset based on hashtags regarding newly released movies.

Chapter 6 extends ADI to an ensemble method which allows for more flexibility in choosing appropriate windowing functions. Further, we describe a dynamic covariance model for Gaussian data, and apply this to a video tracking dataset. Using ADI, we are able to identify interactions of interest in the dataset and group these interactions according to their ADI profile.

Chapter 7 provides an overview of multi-layer networks, focusing on examples, centrality measures, and community detection methods. We also apply those methods to influence graphs based on the ADI measure that was introduced previously.

Chapter 8 discusses an approach for clustering over multi-layer networks. We propose a Pareto optimization approach for clustering with general fitness functions that allow the user to explore multiple different clusterings, which are all approximately Pareto optimal. This method is applied to both Twitter data and the Enron email dataset.

Chapter 9 discusses a denoising summarization approach to multi-layer networks, when it is assumed that the underlying mesoscopic community structure is the same in all layers. We further utilize this framework to propose a multi-layer extension of the dynamic stochastic block model (DSBM), and apply it to the Enron dataset.

Finally, Chapter 10 provides a summary and points to directions for further work.

1.6 List of publications

- S. Yasaei Sekeh, B. Oselio, and A. Hero, "Learning to bound the multi-class bayes error," *IEEE Transactions of Signal Processing (In review)*, 2019
- W. Dempsey, B. Oselio, and A. Hero, "Hierarchical network models for structured exchangeable interaction processes," *Journal of the American Statistical Association (In review)*, 2019
- B. Oselio, A. Hero, A. Sadeghian, *et al.*, "Time-varying interaction estimation using ensemble methods," in *2019 IEEE Data Science Workshop (DSW)*, Jun. 2019, pp. 69–75
- S. Yasaei Sekeh, B. Oselio, and A. Hero, "A dimension-independent discriminant between distributions," in *proc. IEEE Int. Conf. on Image Processing (ICASSP)*, 2018
- B. Oselio, S. Liu, and A. Hero, "Multi-layer relevance networks," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, IEEE, 2018, pp. 1–5
- B. Oselio and A. Hero, "Dynamic reconstruction of influence graphs with adaptive directed information," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2017, pp. 5935–5939
- B. Oselio and A. Hero, "Dynamic Directed Influence Networks: A Study of Campaigns on Twitter," in *Social, Cultural, and Behavioral Modeling, 9th International Conference*, 2016, pp. 152–161
- B. Oselio, A. Kulesza, and A. Hero, "Information extraction from large multi-layer social networks," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, 2015, pp. 5451–5455
- B. Oselio, A. Kulesza, and A. Hero, "Socio-spatial pareto frontiers of twitter networks," in *International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction*, Springer, 2015, pp. 388–393

- B. Oselio, A. Kulesza, and A. Hero, "Multi-objective optimization for multi-level networks," in *International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction*, Springer, 2014, pp. 129–136

- B. Oselio, A. Kulesza, and A. O. Hero, "Multi-layer graph analysis for dynamic social networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 4, pp. 514–523, 2014

- B. Oselio, A. Kulesza, and A. O. Hero, "Multi-layer graph analytics for social networks," in *2013 5th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, IEEE, 2013, pp. 284–287

CHAPTER 2

Hierarchical Network Models for Structured Exchangeable Interaction Processes

Network data often arises via a series of *structured interactions* among a population of constituent elements. E-mail exchanges, for example, have a single sender followed by potentially multiple receivers. Scientific articles, on the other hand, may have multiple subject areas and multiple authors. We introduce *hierarchical interaction exchangeable models* for the study of these structured interaction networks. In particular, we introduce the *hierarchical vertex components model* as a canonical example, which partially pools information via a latent, shared population-level distribution. Theoretical analysis and supporting simulations provide clear model interpretation, and establish global sparsity and power-law degree distribution. A computationally tractable Gibbs algorithm is derived for inferring sparsity and power law properties of complex networks. We demonstrate the model on both the Enron e-mail dataset and an ArXiv dataset, showing goodness of fit of the model via posterior predictive validation.

2.1 Introduction

Modern statistical network analysis focuses on the study of large, complex networks that can emerge in diverse fields, including social, biological, and physical systems [18], [74], [88], [127], [165]. The expanding scope of network analysis requires statistical models and

Symbol	Description
$\mathcal{P}$	Set of constituent elements
$(\bar{s}, \bar{r})$	Single observed interaction
$(\bar{S}, \bar{R})$	Single random interaction
$\mathcal{I}$	Structured interaction process
$\mathbf{y}, \mathbf{Y}$	Structured interaction-labeled network
$\#\mathcal{P}$	Cardinality of $\mathcal{P}$
$=_{\mathcal{D}}$	Equality in distribution
$\text{fin}(\mathcal{P})$	Set of all finite multisets of $\mathcal{P}$

Table 2.1: Glossary of commonly used symbols.

inferential tools that can handle the increasing complexity of network data structures. In this chapter, we focus on network data arising from sequences of interactions. Network data arising in this manner will benefit from a framework built upon the interaction as the statistical unit [151] rather than upon the constituent elements within each interaction as the statistical units. *Edge-exchangeable models* [61], [62] are well adapted to analysis of datasets containing these complex interactions.

While Crane and Dempsey (2017) [62] provide a framework for statistical analysis of interaction data, the proposed *Hollywood model* only captures basic global features. Specifically, the Hollywood model's asymptotic behavior reflects the empirical properties of sparsity and power law degree distributions observed in real-world network data, which are not as well reflected in classic statistical network models such as the ERGMs [228], graphon models [3], and stochastic blockmodels (SBMs) [107]. While edge exchangeability is attractive as a theoretical framework, the set of current edge exchangeable models is inadequate for modern network data with high structural complexity.

The edge exchangeable model proposed in this chapter is motivated by an important fact: most common complex networks constructed from interaction data are *structured*. A phone-call interaction, for example, takes the form of a sender and receiver pair. E-mail correspondence generalizes this type of interaction to one sender but potentially multiple receivers with different attributes like "To," "Cc," and "Bcc". This chapter makes a sub-

stantial push forward by constructing hierarchical models that reflect this common structure of *structured interaction data*. The model overlays local behavior (e.g., per sender) with global information by partial pooling through a shared global, latent distribution. Simulation and theoretical analysis confirm that the proposed hierarchical model can achieve simultaneously varying local power-law degree per sender and global power-law degree distribution. By explicitly modeling hierarchies of interactions, the proposed framework goes beyond previous work in modeling interaction data and network valued data.

2.1.1 Relevant prior work on interaction data

Interaction data often arises in settings where communications amongst a set of constituent elements over a specific time period are recorded [71], [223]. Examples are numerous and include: authorship and co-sponsorship of legislation [80], [208], sending and receiving e-mails [55], [223], posting and responding on a community forum [202], and traceroute [142]. In each case, the interaction (edge) is the statistical unit to be modeled, as contrasted with the subjects (nodes) of the interactions considered in other work [88]. See [61], [62] for further discussion of the advantages of defining interactions as the statistical units.

The literature contains several papers focused on statistical modeling of interaction data. Perry and Wolfe (2013) [188] construct a Cox proportional intensity model [57]. Butts (2008) [36] considered likelihood-based inference using a variant of the proportional intensity model to capture interaction behavior in social settings. Crane and Dempsey (2017) [62] consider non-hierarchical models for interaction data. They introduce the notion of edge exchangeable network models and explore its basic statistical properties. In particular, they show that edge exchangeable models allow for sparse structure and power law degree distributions, widely observed empirical behaviors that cannot be handled by conventional approaches.

An alternative approach emerges out of the recent work of Caron and Fox (2017) [42], who construct random graphs from point processes on $\mathbb{R}_+^2 = [0, \infty) \times [0, \infty)$. The ran-

dom graph is characterized by an object called a *graphex* [224]. Random graph models generated by this procedure can incorporate sparse, power law behavior into a well-defined population model. Finite random graphs can be obtained via a thresholding operation, termed *p-sampling* [224]. Such random graphs are vertex exchangeable in that they are built from exchangeable point processes. In this setting, exchangeability is a consequence of projectivity rather than the simple structured interaction data sampling scheme proposed in this chapter. See the contributed discussion to the paper by Caron and Fox (2017) [42], in particular contributions by Bharath [29] and Crane [59], for further discussion.

2.1.2 Relevant prior work on network-valued data

Interaction data is closely related to network-valued data. The most common approach to statistical network modeling in the literature [88] is to model the adjacency matrix of a graph - this includes the stochastic block model [107] and its many variants [1], [97], [128], latent space models [14], [105], and exponential random graph models [200]. In our setting, the graph is constructed from the observed interaction data rather than being directly observed itself. While the aforementioned models have generalizations to directed, non-binary valued graphs, none of them respect the fundamental structure of interaction data, where the sampling unit is the interaction itself. It has been observed that projection of interaction data with simple structure onto the space of adjacency matrices can fundamentally alter the characteristics of the network, such as sparsity; see [62, Theorem 4.4]. In contrast to [62], however, our proposed framework aims to respect the full structure of the data, including both the directed nature of the interaction data and its hierarchical set structure.

There has also been previous work on Bayesian modeling for network data [70], [129], [184]. The proposed model in this work follows this paradigm, utilizing the posterior predictive density for model behavior exploration and validation [85], which is an often overlooked aspect of network models.

2.1.3 Motivating statistical network question

The critical point is that while there are network models built from interaction data, none have the following three important properties: (1) a probabilistic symmetry faithful to the statistical units, (2) provable empirical properties of sparsity and power-law degree, and (3) hierarchical/multi-level structure to account for local variability in network connectivity. Our chapter addresses the following fundamental question of network modeling:

Is there a general framework for modeling probabilistic symmetry that provides better fit to sparse graphs with global network properties such as power-law degree distribution and also accounts for variations in local network behavior?

This expands upon a fundamental network modeling question raised first in [173], updated here to reflect the need for models that account for local variability in network connectivity (e.g., differences in receiver distributions per sender). This chapter answers in the affirmative. Indeed, we start from the observation made by [62] that the interaction is the fundamental statistical unit in many network settings; this simple observation leads to models with interaction exchangeability and global network properties. However, no prior work on interaction and network-valued data has focused on accounting simultaneously for these properties and local variation (e.g., across callers, senders, article topics).

An additional benefit of our approach is that the model class can be used to study network interdependency. Take the Enron e-mail network, presented as a case study in Section 2.7. Figure 2.8 shows how the model captures interdependency across distributions per e-mail sender, while Figures 2.6 and 2.7 show our model simultaneously captures the global power-law degree distribution and local behavior respectively. In our second case study of the ArXiv dataset, we show how the model output can be used to infer interdependency. Figure 2.11 visualizes a clustering that captures the interdisciplinary nature of science better than simple application of normalized spectral clustering to the raw data; see Figure A.2 in the appendix for a direct comparison.

2.1.4 Outline and main contributions

The main contributions of this chapter are as follows:

1. We start by formally defining structured interaction data in Definition 2.2.1. We then define exchangeable structured interaction processes in Definition 2.3.1.

2. We prove a representation theorem for these exchangeable processes in Theorem 2.3.2; we then define, in Section 2.3.1, *hierarchical vertex components models* (HVCM) – a subfamily of exchangeable processes that capture important interaction dynamics.

3. A particular computationally tractable HVCM is introduced in Section 2.4 and an efficient Gibbs sampling inferential algorithm is derived in Section 2.6.

4. We establish basic statistical properties in Section 2.5. In particular, we provide theoretical guarantees of sparsity and power law for the chosen HVCM – two important empirical properties of network data.

5. We demonstrate this HVCM on both the Enron e-mail dataset and ArXiv dataset in Section 2.7. In particular, we show how the HVCM can be used to perform goodness of fit checks for models of network data via posterior predictive checks, an often under-emphasized aspect of statistical network modeling.

2.2 Structured interaction data

We start by defining structured interaction data, illustrating with a sequence of concrete examples of increasing complexity.

Definition 2.2.1 (Structured interaction data)**.** Let $\mathcal{P}$ denote a set of constituent elements. Then for a set $\mathcal{P}$, we write $\text{fin}(\mathcal{P})$ to denote the set of all finite multisets of $\mathcal{P}$. A *structured interaction process* for an ordered sequence of sets $(\mathcal{P}_1, \ldots, \mathcal{P}_k)$ is a correspondence $\mathcal{I}$:

$I \to \text{fin}(\mathcal{P}_1) \times \ldots \times \text{fin}(\mathcal{P}_k)$ between a set I indexing interactions and the ordered sequence of finite multisets of $(\mathcal{P}_1, \ldots, \mathcal{P}_k)$.

Remark 1 (Difference from interaction data). In [62], an interaction process is defined as a correspondence $\mathcal{I} : I \to \text{fin}(\mathcal{P})$ where $\mathcal{P}$ is a single population. Structured interaction data, instead, consists of a series of finite multisets, and does not require each set of constituent elements to be equivalent. That is, each population $\mathcal{P}_k$ may contain different types of elements. This flexibility will allow us to introduce hierarchical structure into the exchangeable model.

Finally, let $\text{fin}_k(\mathcal{P})$ denote the multisets of size k, so that $\text{fin}(\mathcal{P})$ is the disjoint union $\cup_{k=1}^{\infty} \text{fin}_k(\mathcal{P})$.

Example 1 (Phone-calls)**.** Assume $\mathcal{P}_k$ are all equivalent and let $\mathcal{P}_k =: \mathbb{N}$ be a countably infinite population. A phone-call can be represented as an ordered pair of "sender" and "receiver" drawn from $\mathbb{N}$. Therefore, a phone-call interaction process is a correspondence $\mathcal{I} : I \to \text{fin}_1(\mathbb{N}) \times \text{fin}_1(\mathbb{N})$. For instance, $I(1) = (\{a\}, \{b\})$ is a phone-call from sender a to receiver b, both in population $\mathbb{N}$. This is distinct from $(\{b\}, \{a\})$ where sender and receiver roles are reversed.

Example 2 (E-mails)**.** Assume $\mathcal{P}_k$ are all equivalent and let $\mathcal{P}_k = \mathbb{N}$ be a countably infinite population. An e-mail can be represented as the ordered sequence of sets: sender, receivers. Then an e-mail interaction process is a correspondence $\mathcal{I} : I \to \text{fin}_1(\mathbb{N}) \times \text{fin}(\mathbb{N})$. For instance, $I(1) = (\{a\}, \{b, c\})$ is an e-mail from sender a to receivers b and c. This is distinct from $(\{b\}, \{a, c\})$ and $(\{c\}, \{a, b\})$. Figure 2.1 is a visualization of a similar structured interaction dataset formed from Facebook posts (i.e., poster followed by finite multiset of responders).

Example 3 (Scientific articles)**.** Consider summarizing a scientific article by its (1) list of subject areas and (2) list of authors. Then the scientific article process is a correspondence

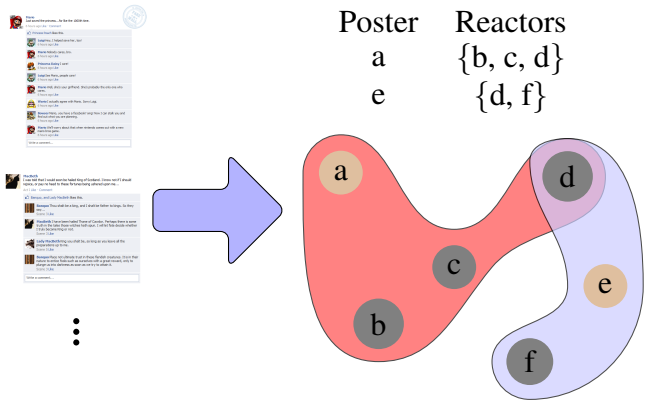


Figure 2.1: Example of network data from Facebook posts. The post process is a correspondence $\mathcal{I} : I \to \text{fin}_1(\mathbb{N}) \times \text{fin}(\mathbb{N})$. Here $I = \mathbb{N}$, and the first post $\mathcal{I}(1) = \{\{a\}, \{b, c, d\}\}$ represents user a posting to the forum and b, c, d reacting to the post. The second post $\mathcal{I}(2) = \{\{e\}, \{d, f\}\}$ represents user e posting to the forum and d, f reacting. User d reacts to both posts.

$\mathcal{I} : I \to \text{fin}(\mathcal{P}_1) \times \text{fin}(\mathcal{P}_2)$. For instance, $I(1) = (\{a, b\}, \{c, d\})$ is an article with subject areas a and b and authors c and d. Here, $\mathcal{P}_1$ and $\mathcal{P}_2$ are distinct populations.

Example 4 (Movies)**.** Consider summarizing a movie by its (1) genre, (2) list of producers, (3) director, and (4) list of actors. Of course, there is overlap in certain populations, as producers can be directors, directors can be actors, but none are a genre (unless Scorsese, Spielberg, or Tarantino are considered genres unto themselves). Then the movie process is a correspondence $\mathcal{I} : I \to \text{fin}_1(\mathbb{N}) \times \text{fin}(\mathbb{N}) \times \text{fin}_1(\mathbb{N}) \times \text{fin}(\mathbb{N})$. For instance, $I(1) = (\{a\}, \{b, c\}, \{d\}, \{d, e, f\})$ is a movie with genre a, producers b and c, director d, and actors d, e, and f. Note, in this example, the director is also one of the actors.

The above shows Definition 2.2.1 covers a wide variety of examples from network science. Next, we construct interaction-labeled networks and define exchangeable structured interaction processes.

Remark 2 (Covariates). In this chapter, we focus on the study of structured interaction processes in Definition 2.2.1 with no additional information, such as covariates. Incorporating such covariate information is quite difficult; see [2], [130], [147], [215], [217], [241] for examples of incorpating covariates into network analysis. Covariate information can come

in two forms: (1) covariate information on the interaction; and (2) covariate information on constituent elements. Examples of (1) include subject line or body text in an e-mail, or genre and gross movie sales for a movie. Examples of (2) include gender, age, job title, or university affiliation of authors of a scientific article. Certain interaction covariates can be incorporated into the models considered in this chapter. For example, in the ArXiv dataset, the article's subject can be viewed as covariate information on the interaction. We show how this can be incorporated as part of the structured interaction data structure, and therefore accounted for in the statistical models.

2.2.1 Interaction-labeled networks

For the remainder of this chapter, we focus on structured interaction processes of the form $\mathcal{I} : I \to \text{fin}(\mathcal{P}_1) \times \text{fin}(\mathcal{P}_2)$. This type of a structured interaction process captures the phone-call, e-mail, and scientific article examples. The arguments presented naturally extend to more general structured interaction processes as given in Definition 2.2.1. When two populations of constituent elements are equivalent, we write $\mathcal{P}_1 \equiv \mathcal{P}_2$. The interaction-labeled network is an equivalence clase constructed from the structured interaction process by quotienting out the labeling of the constituent elements. Let $\rho_j : \mathcal{P}_j \to \mathcal{P}'_j$ be a bijection for $j = 1, 2$. We write $\rho : \mathcal{P}_1 \times \mathcal{P}_2 \to \mathcal{P}'_1 \times \mathcal{P}'_2$ to be the composite bijection obtained by applying $\{\rho_j\}_{j=1,2}$ componentwise. If $\mathcal{P}_1 \equiv \mathcal{P}_2$, then $\rho_1 \equiv \rho_2$; that is, bijections among equivalent populations, e.g., the senders and receivers in an email network, denoted by $\bar{s}$ and $\bar{r}$, respectively, must agree. Then ρ induces an action on the product space $\text{fin}(\mathcal{P}_1) \times \text{fin}(\mathcal{P}_2)$ by the composite map

$$(\bar{s}, \bar{r}) = (\{s_1, \ldots, s_{k_1}\}, \{r_1, \ldots, r_{k_2}\}) \in \text{fin}(\mathcal{P}_1) \times \text{fin}(\mathcal{P}_2)$$
$$\to \rho\,(\bar{s}, \bar{r}) = (\{\rho_1 s_1, \ldots, \rho_1 s_{k_1}\}, \{\rho_2 r_1, \ldots, \rho_2 r_{k_2}\}) \in \text{fin}(\mathcal{P}'_1) \times \text{fin}(\mathcal{P}'_2)$$

Therefore, the bijection ρ acts on the structured interaction process via composition $(\rho\mathcal{I})(i) = \rho(\mathcal{I}(i)), i \in \mathbb{N}$. The structured interaction-labeled network is then the equivalence class constructed from the structured interaction network by quotienting out over bijections ρ:

$$\mathbf{y}_{\mathcal{I}} = \bigcup_{\substack{\#\mathcal{P}'_j = \#\mathcal{P}_j \\ j=1,2}} \{\mathcal{I}' : I \to \text{fin}(\mathcal{P}'_1) \times \text{fin}(P'_2) : \rho\mathcal{I} = \mathcal{I}' \text{ for some bijection } \rho : \mathcal{P}_1 \times \mathcal{P}_2 \to \mathcal{P}'_1 \times \mathcal{P}'_2\}\,, \tag{2.1}$$

where $\#\mathcal{P}_j$ is the cardinality of the population. Note we have only quotiented out labels for constituent elements, so the object $\mathbf{y}_{\mathcal{I}}$ still has uniquely labeled interactions. For simplicity, we write $\mathbf{y}$ and leave the subscript $\mathcal{I}$ implicit.

In the remainder of the chapter, we assume the index set I is countably infinite and replace it by $\mathbb{N}$. For any $S \subset \mathbb{N}$, we define the *restriction* of $\mathcal{I} : \mathbb{N} \to \text{fin}(\mathcal{P}_1) \times \text{fin}(\mathcal{P}_2)$ to the subset $S \subset \mathbb{N}$ by $\mathcal{I}|_S$. This restricted interaction process induces a restriction to S of the interaction-labeled network. We write $\mathbf{y}_S$ to denote the interaction-labeled network associated with the restricted process $\mathcal{I}|_S$. For $S = [n] := \{1, \ldots, n\}$, we simply write $\mathcal{I}_n$ to denote the restricted structured interaction process and $\mathbf{y}_n$ to denote the corresponding structured interaction network.

2.3 Structured interaction exchangeable models

Let $\mathbf{y}$ denote the interaction-labeled network constructed from the structured interaction process $\mathcal{I} : \mathbb{N} \to \text{fin}(\mathcal{P}_1) \times \text{fin}(\mathcal{P}_2)$. Then for any finite permutation $\sigma : \mathbb{N} \to \mathbb{N}$, let $\mathcal{I}^\sigma$ denote the relabeled structured interaction process defined by $\mathcal{I}^\sigma(i) = \mathcal{I}(\sigma^{-1}(i)), i \in \mathbb{N}$. Then $\mathbf{y}^\sigma$ denotes the corresponding interaction labeled network constructed from $\mathcal{I}^\sigma$. Note that the choice of representative from the equivalence class does not matter. The above relabeling by permutation σ is not to be confused with the relabeling in the previous section by the bijection ρ. The bijection ρ relabels the constituent elements, and is used to construct the equivalence class defining the interaction-labeled network (i.e., the equivalence class).

The permutation σ reorders the interaction process, and therefore relabels the interaction-labeled network.

In the remainder of this chapter, we write $\mathbf{Y}$ to denote a random interaction-labeled network. We assume the interactions are labeled in the countably infinite set $\mathbb{N}$. Interaction exchangeability is characterized by the property that the labeling of the interactions (not the constituent elements) is arbitrary. We now define exchangeable structured interaction networks.

Definition 2.3.1 (Exchangeable structured interaction network process)**.** The structured interaction-labeled network $\mathbf{Y}$ is exchangeable if $\mathbf{Y}^{\sigma} =_{\mathcal{D}} \mathbf{Y}$ for all permutations $\sigma : \mathbb{N} \to \mathbb{N}$, where $=_{\mathcal{D}}$ denotes equality in distribution.

Next, we provide a representation theorem for structured interaction processes. We focus on the setting where each interaction $(\bar{s}, \bar{r})$ is either never observed or observed infinitely often. This is commonly referred to as the "blip-free" setting [60], where blips refer to interactions $(\bar{s}, \bar{r})$ that are observed once. We first define the $\text{fin}(\mathcal{P}_1) \times \text{fin}(\mathcal{P}_2)$-simplex

$$\mathcal{F} = \left\{ (f_{(\bar{s},\bar{r})})_{(\bar{s},\bar{r}) \in \text{fin}(\mathcal{P}_1) \times \text{fin}(\mathcal{P}_2)} \quad \text{and} \quad \sum_{(\bar{s},\bar{r}) \in \text{fin}(\mathcal{P}_1) \times \text{fin}(\mathcal{P}_2)} f_{(\bar{s},\bar{r})} = 1 \right\}$$

where $(\bar{s}, \bar{r}) := (\{s_1, \ldots, s_{k_1}\}, \{r_1, \ldots, r_{k_2}\})$ for $s_1, \ldots, s_{k_1} \in \mathcal{P}_1$ and $r_1, \ldots, r_{k_2} \in \mathcal{P}_2$. Let ϕ be a probability measure on the simplex and define $f \sim \phi$ to be a random variable drawn from this measure. Then, given $f \in \mathcal{F}$, let the sequence of interactions $\mathcal{I}(1), \mathcal{I}(2), \ldots$ be generated according to

$$\text{pr}\left(\mathcal{I}(i) = (\{s_1, \ldots, s_{k_1}\}, \{r_1, \ldots, r_{k_2}\}) \mid f\right) = f_{(\bar{s},\bar{r})}. \tag{2.2}$$

Then, given $\mathcal{I}$, set $\mathbf{Y} = \mathbf{y}_{\mathcal{I}}$. Theorem 2.3.2 states that all blip-free structured interaction exchangeable networks can be generated in this manner. The proof can be found in Section A.2 of the appendix.

Theorem 2.3.2 (Blip-free representation theorem)**.** Let $\mathbf{Y}$ be a structured interaction exchangeable network that is blip-free with probability 1. Then there exists a probability measure ϕ on $\mathcal{F}$ such that $\mathbf{Y} \sim \epsilon_\phi$, where

$$\epsilon_\phi(\cdot) = \int_{\mathcal{F}} \epsilon_f(\cdot)\phi(df).$$

2.3.1 Hierarchical vertex components model

Via Theorem 2.3.2, we can construct a particular family of interaction exchangeable random networks as follows. First, choose a distribution of senders, $f' = (f_s)_{s\in\mathcal{P}_1}$, in the simplex

$$\mathcal{F}_1 := \left\{ (f_s)_{s\in\mathcal{P}_1} : f_s \geq 0 \quad \text{and} \quad \sum_{s\in\mathcal{P}_1} f_s = 1 \right\}.$$

Next, choose a second element of $\mathcal{F}_1$, which we denote w. Finally, for each $s \in \mathcal{P}_1$, construct a conditional distribution over the receivers, i.e., the second component $\text{fin}(\mathcal{P}_2)$. That is, for every $s \in \mathcal{P}_1$, we choose $f''_s = (f_{r\,|\,s})_{r\in\mathcal{P}_2}$ in the simplex

$$\mathcal{F}_2 = \left\{ (f_r)_{r\in\mathcal{P}_2} : f_r \geq 0 \quad \text{and} \quad \sum_{r\in\mathcal{P}_2} f_r = 1 \right\}.$$

We combine these distributions to form $f \in \mathcal{F}_1 \times \mathcal{F}_1 \times (\otimes_{s\in\mathcal{P}_1}\mathcal{F}_2)$, which determines a distribution on the space $\text{fin}_1(\mathcal{P}_1) \times \text{fin}(\mathcal{P}_2)$ by

$$\text{pr}\left(E = (\bar{s}, \bar{r}) \mid f\right) = \nu_{k_1}\left[\prod_{i=1}^{k_1} f_{s_i}\right] \cdot \frac{\sum_{i=1}^{k_1} w_{s_i} \cdot \nu_{k_2}^{(s_i)}\left[\prod_{j=1}^{k_2} f_{r_j\,|\,s_i}\right]}{\sum_{i=1}^{k_1} w_{s_i}}, \tag{2.3}$$

where $\nu_l \geq 0$, $\nu_l^{(s)} \geq 0$, $\sum_{l=1}^{\infty}\nu_l = 1$, and $\sum_{l=1}^{\infty}\nu_l^{(s)} = 1$ for each $s \in \mathcal{P}_1$. This determines an interaction exchangeable network, which we call the *hierarchical vertex components model* (HVCM). Given f, $\mathcal{I}(1), \mathcal{I}(2), \ldots$ are independent, identically distributed (i.i.d.) random structured interactions drawn from (2.3). The associated random interaction

exchangeable network $\mathbf{Y} := \mathbf{y}_{\mathcal{I}}$ is obtained through (2.1), whose distribution we denote by ϵ_f.

In non-HVCMs [62], each constituent element had a single frequency of occurrence. By contrast, HVCMs allow the frequency of occurrence for elements in the second term of (2.3) (i.e., $r \in \mathcal{P}_2$) to depend on first component (i.e., $\bar{s} \in \text{fin}\,\mathcal{P}_1$) through $f_{r|s}$. This dependence is two-fold: (1) $w \in \mathcal{F}_1$ controls which f''_s is chosen across $s \in \bar{s}$; and (2) the local distributions can vary, leading to the size-biased ordering of $r \in \mathcal{P}_2$ varying as a function of s.

Remark (Vertex exchangeability versus interaction exchangeability). While HVCMs are expressed as a function of the vertices, they are interaction exchangeable and not vertex exchangeable. To see this, consider Theorem 2.3.2. A direct corollary is that vertices are sampled in size-biased order according to their relative frequency of occurrence. In hierarchical models, the size-biased sampling of the second component depends on the first component. Regardless, this implies the observed constituent elements are not exchangeable with the unobserved constituent elements. On the other hand, vertex exchangeability implicitly assumes the observed vertices and unobserved vertices are exchangeable.

2.4 Sequential description for a subfamily of HVCMs

Here we provide a sequential description of a particular subfamily of HVCMs. For ease of comprehension, we start with the setting of a single sender where the size of the first component is one (i.e., $\nu_{k_1} = 1[k_1 = 1]$). In this setting, the sequential description is presented in the context of e-mails. Let $(\tilde{\alpha}, \tilde{\theta})$ satisfy either (1) $0 \leq \tilde{\alpha} < 1$ and $\tilde{\theta} > 0$, or (2) $\tilde{\alpha} < 0$ and $\theta = -K\tilde{\alpha}$ for some $K \in \mathbb{N}$. In setting (1), the population $\mathcal{P}_1$ is infinite, while in setting (2) the population is finite and equal to K. In Section 2.4.3, we show how to extend this model to the general setting of multiple senders. For ease of comprehension, we let $\mathcal{S} = \mathcal{P}_1$ (senders) and $\mathcal{R} = \mathcal{P}_2$ (receivers) denote the two sets of constituent elements.

We introduce some additional notation. For each $n = 1, 2, \ldots$, the nth email E_n is given by the structured interaction $(\bar{S}_n, \bar{R}_n) = (\{S_{n,1}\}, \{R_{n,1}, \ldots, R_{n,k_{n,2}}\})$ where $S_{n,1} \in \mathcal{S}$ is the sender, and $R_{n,j} \in \mathcal{R}$ is the jth receiver of the nth article. Suppose n articles have been observed and define $\mathcal{H}_n = \{E_1, \ldots, E_n\}$ to be the observed history of the first n e-mails. For the $(n+1)$st e-mail, choose the sender according to

$$\text{pr}\left(S_{n+1,1} = s \,|\, \mathcal{H}_n\right) \propto \begin{cases} D_n^{out}(s) - \tilde{\alpha} & s \in \mathcal{S}_n \\ \tilde{\theta} + \tilde{\alpha}|\mathcal{S}_n| & s \notin \mathcal{S}_n. \end{cases} \tag{2.4}$$

where $D_n^{out}(s)$ is the outdegree of the subject s, and $\mathcal{S}_n$ are the set of unique senders in $(\bar{S}_1, \ldots, \bar{S}_n)$ and $|\mathcal{S}_n|$ is the set's cardinality.

Given $S_{n+1,1} = s \in \mathcal{S}$, we choose the number of recipients according to the discrete probability distribution function $\{\nu_k^{(s)}\}_{k=1}^{\infty}$. Finally, let $D_{n,j}(s, r)$ denote the indegree of receiver r when restricted to e-mails from sender s after the first n e-mails and the $j-1$ recipients of the nth e-mail; that is,

$$D_{n,j}(s, r) = \#\left\{(m, l) \,|\, R_{m,l} = r \text{ and } S_{m,1} = s \text{ for } m < n \text{ and } l \leq k_m, \text{ or } m = n \text{ and } l < j\right\}.$$

Finally, we define $m_{n,j}(s) = \sum_{r \in \mathcal{R}} D_{n,j}(s, r)$ to be the number of receivers (accounting for multiplicity) of e-mails from sender s. Each of these statistics is a measurable function of $\mathcal{H}_n$. Note, these statistics are *local* (i.e., specific to the particular subject). Here, we describe a procedure for sharing information across senders. To do this, we define a *partially observable* global set of information. First, define the observable variable $R_{n,j}$ to be the complete set of receivers; that is,

$$R_{n,j} = \left\{r \in \mathcal{R} \,|\, \exists\, R_{m,l} = r \text{ for } m < n \,\text{and}\, l \leq k_m, \text{ or } m = n \,\text{and}\, l < j\right\}.$$

Additionally, let $K_{n,j} = |R_{n,j}|$ be the cardinality of this set. For each $r \in R_{n,j}$ we posit

existence of a *latent degree* per sender $s \in \mathcal{S}_n$ and receiver r denoted by $V_{n,j}(s,r)$. We then define $V_{n,j}(\cdot,r) = \sum_{s\in\mathcal{S}_n} V_{n,j}(s,r)$ and $m_{n,j} = \sum_{r\in\mathcal{R}_{n,j}} V_{n,j}(\cdot,r)$. Next, define $R_{n,j}(s)$ to be the complete set of receivers when restricting to e-mails from sender $s \in \mathcal{S}_n$, and $\mathcal{H}_{n,j}$ to be the observable history $\mathcal{H}_{n-1}$ union $\{S_{n,1}, R_{n,1}, \ldots, R_{n,j}\}$. That is, $\mathcal{H}_{n,j}$ is the observed history up to the $j-1$th receiver on the nth e-mail, where $\mathcal{H}_{n,0}$ implies only sender information for the nth e-mail. Finally, for each $s \in \mathcal{S}$, let (α_s, θ_s) satisfy either (1) $0 \leq \alpha_s < 1$ and $\theta > 0$, or (2) $\alpha_s < 0$ and $\theta_s = -K'\alpha_s$ for some $K' \in \mathbb{N}$. In setting (1), the receiver population $\mathcal{P}_2$ is infinite, while in setting (2) the population is finite and equal to K'. For the remainder of this chapter, we assume setting (1).

Given the indegree distribution $\{D_{n,j}(s',r')\}_{r'\in R_{n,j}, s'\in\mathcal{S}_n}$, the latent degree distribution $\{V_{n,j}(s',r')\}_{r'\in R_{n,j}, s'\in\mathcal{S}_n}$, the current sender s, along with the observable history $\mathcal{H}_{n,j}$, the probability of choosing receiver r is proportional to

$$\frac{D_{n,j}(s,r) - \alpha_s V_{n,j}(s,r) + (\theta_s + \alpha_s V_{n,j}(s,r))\left(\frac{V_{n,j}(\cdot,r)-\alpha}{m_{n,j}+\theta}\right)}{m_{n,j}(s)+\theta_s}, r \in R_{n,j}(s) \tag{2.5}$$

and

$$\frac{\theta_s + \alpha_s V_{n,j}(s,r)}{m_{n,j}(s)+\theta_s} \cdot \frac{\theta + \alpha V_{n,j}(\cdot,r)}{m_{n,j}+\theta}, r \notin R_{n,j}(s). \tag{2.6}$$

Note the difference in the discount of indegree in (2.5) and outdegree in (2.4). For the sender distribution (2.4), the outdegree discount is $\tilde{\alpha}$; on the other hand, for (2.5), the indegree discount is $\alpha_s V_{n,j}(s,r)$. This reflects that in (2.4), sender s is chosen from a single distribution; however, in (2.5), receiver r can be chosen either locally or globally.

The remaining question is how to update the degree distributions. In (2.5) and (2.6), we can either observe r "locally", or we escape the local model and observe r due to the latent global information. Given $R_{n,j} = r$ we update both local and global degrees. If $r \notin R_{n,j}(s)$ then the global degree $V_{n,j}(s,r)$ increases from zero to one. If $r \in R_{n,j}(s)$ then the local degree $D_{n,j}(s,r)$ increases by one and the latent degree is increased by one with probability $\tau_{n,j}(s) = \frac{\theta_s + \alpha_s V_{n,j}(s,r)}{m_{n,j}(s)+\theta_s}$. The exact procedure for incrementing $V_{n,j}$ is

discussed in Section 2.6.

2.4.1 Partial pooling

The importance of the latent global degree distribution is that it allows information to be shared across the conditional receiver distributions. The above model formalizes the partial pooling of information. The degree of pooling is controlled by the escape probability $\tau_{n,j}(s)$, which in general decreases as the number of e-mails from sender s increases. Note that over time as more e-mails by sender s are seen, the escape probability $\tau_{n,j}(s)$ tends to zero whenever $\alpha_s < 1$. Therefore, the local impact of the latent global degree information becomes negligible once we have sufficient local information. However, the first time a sender-receiver pair is observed, it must occur via the shared global set of information. The global latent degrees $\{V_{n,j}(s,r)\}_{r\in R_{n,j}, s\in S_n}$ therefore contribute to the behavior of new and/or rarely seen senders.

2.4.2 Connection between sequential description and hierarchical vertex components models

The sequential description in Section 2.6 is equivalent to a particular HVCM. When $\alpha_s = 0,\ \forall s \in \mathcal{P}_1$, an analytic stick-breaking representation can be derived. This connects the sequential process directly to (2.3). To do so, we start by constructing the sender distribution. Here, we assume $\mathcal{P}_1 \equiv \mathcal{P}_2 \equiv \mathbb{N}$. For $s \in \mathbb{N}$, define independent random variables $\beta_s \sim \text{Beta}(1-\tilde{\alpha}, \tilde{\theta} + s\tilde{\alpha})$. Then, conditional on $\{\beta_s\}_{s=1}^{\infty}$, the probability of choosing sender $s \in \mathbb{N}$ is given by

$$f_s \mid \{\beta_{s'}\}_{s'=1}^{\infty} = \beta_s \prod_{i=1}^{s-1}(1-\beta_i),$$

where the product is set equal to one for $s = 1$, and $f' = \{f_s\}_{s=1}^{\infty}$. In our current setting, $\nu_{k_1} = 1[k_1 = 1]$ so the weights $w = \{w_s\}_{s=1}^{\infty}$ can be ignored for now. See Section 2.4.3 for a description of how these can be constructed in the more general case.

We now construct, for each $s \in \mathbb{N}$ the probabilities $\{f_{r\,|\,s}\}_{r=1}^{\infty}$ via a hierarchical model given $\alpha > 0$ and $\theta > -\alpha$, and we set $f'' = \{\{f_{r|s}\}_{r=1}^{\infty}\}_{s=1}^{\infty}$. To do this, we first define global independent random variables $\tilde{\beta}_r \sim \text{Beta}(1-\alpha, \theta + r\alpha)$ for $r \in \mathbb{N}$. Then, conditional on $\{\tilde{\beta}_r\}_{r=1}^{\infty}$, for $r \in \mathbb{N}$, we define associated stick-breaking probabilities $\tilde{\pi}_r = \tilde{\beta}_r \prod_{i=1}^{r-1}(1-\tilde{\beta}_i)$. These are probabilities of choosing receiver r based on the global random variables $\{\tilde{\beta}\}_{r=1}^{\infty}$. The local stick-breaking distributions are then defined via a perturbation of these global probabilities. That is, for $\theta_s > 0$, define independent random variables

$$\tilde{\beta}'_{r\,|\,s} \sim \text{Beta}\left(\theta_s \tilde{\pi}_r, \theta_s\left(1 - \sum_{l=1}^{r} \tilde{\pi}_l\right)\right)$$
$$f_{r\,|\,s}\,|\,\{\tilde{\beta}'_{j\,|\,s}\}_{j=1}^{\infty} = \tilde{\beta}'_{r\,|\,s}\prod_{i=1}^{r-1}\left(1-\tilde{\beta}'_{i\,|\,s}\right)$$

where the product is defined equal to one when $r = 1$. This yields a stick breaking representation for $f = \{f', f''\}$ for a particular hierarchical vertex components model. Partial pooling occurs via the shared global probabilities $\tilde{\pi}_r$. The local distributions satisfy $\mathbb{E}\left[f_{r\,|\,s}\,|\,\{\tilde{\pi}_{r'}\}_{r'=1}^{\infty}\right] = \pi_r$. Therefore, the distribution $\{f_{r\,|\,s}\}_{r=1}^{\infty}$ can be thought of as perturbation of the global distribution $\{\pi_r\}_{r=1}^{\infty}$ where θ_s controls the amount of perturbation. In particular, $f_{r\,|\,s} \to \pi_r$ with probability one as $\theta_s \to \infty$.

By construction, $f = (\{f_s\}_{s=1}^{\infty}, \{f_{r\,|\,s}\}_{s,r=1}^{\infty})$ is a random variable over the space $\mathcal{F}_1 \times \otimes_{s=1}^{\infty}\mathcal{F}_2$. Lemma 2.4.1 establishes the connection between these random variables and the canonical model for $\alpha_s = 0$. Although this model for $\alpha_s > 0$ does not admit a known closed-form stick breaking representation, Theorem 2.3.2 discussed in Section 2.3 ensures these asymptotic frequencies exist. Section A.3 of the appendix describes a specific probabilistic construction of these frequencies.

Lemma 2.4.1. *The sequential HVCM model for $\alpha_s = 0$ for $s \in \mathbb{N}$ is equivalent in distribution to (3), where f is distributed according to the stick-breaking construction described above.*

Proof can be found in Section A.3 of the appendix; see [112], [189] for further details on the stick-breaking representation.

Remark (Connections to CRFP). Note that the HVCM described above is closely related to the Chinese Restaurant Franchise Process, a well-known process in the machine learning literature [32], [189] that is almost exclusively used to model latent clusters in data. Here, we use these ideas in the construction of the interaction process. Thus, the objectives are quite different; for instance, there is almost no focus on the inference of the model parameters in the ML community; in our setting, these parameters are crucial to understanding the overall interaction process behavior. This model is most similar to [220], where it is used for language modeling. Similar to the CFRP, the above construction is related to the Pitman-Yor process and the GEM distributions [189]. More details can be found in Section 2.6 and Appendix A.

2.4.3 Accounting for multiple elements in first component

In the general setting, the first component, $\bar{S}_n$, is a random element of $\text{fin}(\mathcal{P}_1)$ (i.e., a random finite multiset of elements from $\mathcal{P}_1$). In the sequential description, we assumed the size of this multiset was one. We now consider $\bar{S}_n = \{S_{n,1}, \ldots, S_{n,k_{n,1}}\}$ for general $k_{n,1} \geq 1$. First, let $\mathcal{H}_{n,j}^{(s)} = \mathcal{H}_n \cup \{S_{n+1,1}, \ldots, S_{n+1,j}\}$ denote the history of the first n e-mails and the first j senders of the $n+1$st e-mail. Extension of 2.4 to handle multiple senders is straightforward by replacing $\mathcal{H}_n$ by $\mathcal{H}_{n,j}^{(s)}$ and defining all other terms similarly.

In the sequential description, the sender $S_{n,1}$ is used to specify which local statistics (i.e., $V_{n,j}(r,s)$, $D_{n,j}(r,s)$ and $m_{n,j}(s)$) to consider. However, when there are multiple senders, this choice is no longer straightforward. To address this, we introduce a random variable Z_n with domain $\bar{S}_n$. This variable indicates which local statistics will be used in receiver distributions (i.e., equations (2.5) and (2.6)). Define $\mathcal{S}_n^{(z)}$ to be the unique elements

in $\mathcal{H}_n^{(z)} := (Z_1, \ldots, Z_n)$. Then

$$\text{pr}\left(Z_n = s \,|\, \mathcal{H}_n^{(z)}, \bar{S}_n\right) \propto \mathbf{1} \begin{cases} D_n^{(z)}(s) - \tilde{\alpha}_z & s \in \mathcal{S}_n^{(z)} \cap \bar{S}_n \\ \tilde{\theta}_z + \tilde{\alpha}_z |\mathcal{S}_n^{(z)}| & s \notin \mathcal{S}_n^{(z)} \cap \bar{S}_n \\ 0 & s \notin \bar{S}_n \end{cases} \tag{2.7}$$

where (1) $0 < \tilde{\alpha}_z < 1$ and $\tilde{\theta}_z > 0$ if the population $\mathcal{S}$ is considered infinite, and (2) $\tilde{\alpha}_s < 0$ and $\tilde{\theta}_z = -K\tilde{\alpha}_z$ if population is finite and $|\mathcal{S}| = K$. This is equivalent to restricting (2.4) to be non-zero only on the domain $\bar{S}_n$. Moreover, it is conditional on the history $\mathcal{H}_n^{(z)}$ instead of $\mathcal{H}_n$. If $Z_n = s$ for $s \in \mathcal{S}_n^{(z)}$, then increase $D_n^{(z)}(s)$ by one. If $s \notin \mathcal{S}_n$, then set $D_n^{(z)}(s) = 1$.

2.5 Statistical properties

We now state several theoretical results for the proposed HVCM built from the sequential description in Section 2.4. For ease of comprehension, we refer to this model as the "canonical HVCM model".

Theorem 2.5.1. *The canonical HVCM with parameters $\Psi = (\tilde{\alpha}, \tilde{\theta}, \alpha, \theta, \{\alpha_s, \theta_s\}_{s \in \mathcal{P}_1})$ determines a structured exchangeable interaction probability distribution for all Ψ in the parameter space.*

Theorem 2.5.1 is not immediate from the sequential construction in Section 2.4, but is clear from the reparameterization of the model presented in Section 2.6, and its connection to the model previously discussed (this is formalized in Section A.3) of the appendix.

The remainder of this section focuses on the setting where the size of the first component is one (i.e., $\nu_{k_1} = 1[k_1 = 1]$). Moreover, we will make certain alternative assumptions concerning the sender distributions. These constraints allow sufficient complexity to be interesting, but assume sufficient regularity to push through the theoretical analysis. First, we

turn to the growth rates in the expected number of unique receivers. Unlike the Hollywood model, this rate depends on both the distribution over senders, the global parameter α, and the local parameters $\{\alpha_s\}_{s\in\mathcal{P}_1}$. Before stating the theorem, we require a formal definition of sparsity. For clarity, we define quantities in terms of receivers to distinguish vertices observed as senders and those observed as receivers (i.e., in $\mathcal{P}_1$ and $\mathcal{P}_2$ respectively).

For a structured interaction-labeled network $\mathbf{Y}$, let $v(\mathbf{Y})$ denote the number of non-isolated receivers; $e(\mathbf{Y})$ is the number of interactions; $M_k(\mathbf{Y})$ is the number of interactions with k receivers; $N_k(\mathbf{Y})$ is the number of receivers that appear exactly k times; and $d(\mathbf{Y}) = (d_k(\mathbf{Y}))_{k\geq 1}$ is the indegree distribution, where $d_k(\mathbf{Y}) = N_k(\mathbf{Y})/v(\mathbf{Y})$. Note that these are global statistics that do not depend on the interaction labels. We define local versions by superscripting each statistic by $s \in \mathcal{P}_1$. For instance, $v^{(s)}(\mathbf{Y})$ is the number of non-isolated receivers when restricting $\mathbf{Y}$ to only those interactions involving sender s. The statistics $e^{(s)}(\mathbf{Y}), M_k^{(s)}(\mathbf{Y}), N_k^{(s)}(\mathbf{Y}), d^{(s)}(\mathbf{Y})$ and $d_k^{(s)}(\mathbf{Y})$ are defined similarly.

Definition 2.5.2 (Global and local sparsity)**.** Let $(\mathbf{Y}_n)_{n\geq 1}$ be a sequence of interaction-labeled networks for which $e(\mathbf{Y}_n) \to \infty$ as $n \to \infty$. The sequence $(\mathbf{Y}_n)_{n\geq 1}$ is *sparse* if

$$\limsup_{n\to\infty} \frac{e(\mathbf{Y}_n)}{v(\mathbf{Y}_n)^{m_\bullet(\mathbf{Y}_n)}} = 0,$$

where $m_\bullet(\mathbf{Y}_n) = e(\mathbf{Y}_n)^{-1} \sum_{k\geq 1} kM_k(\mathbf{Y}_n)$ is the average arity (i.e., number of receivers) of the interactions in $\mathcal{E}_n$. A non-sparse network is *dense*. We say the sequence is $(\mathcal{E}_n)_{n\geq 1}$ is s*-locally sparse* if

$$\limsup_{n\to\infty} \frac{e^{(s)}(\mathbf{Y}_n)}{v^{(s)}(\mathbf{Y}_n)^{m_\bullet^{(s)}(\mathbf{Y}_n)}} = 0,$$

where $m_\bullet^{(s)}(\mathbf{Y}_n) = e^{(s)}(\mathbf{Y}_n)^{-1} \sum_{k\geq 1} kM_k^{(s)}(\mathbf{Y}_n)$ is the average arity (i.e., number of receivers) of the interactions in $\mathbf{Y}_n$ from sender $s \in \mathcal{P}_1$. A network that is not s-locally sparse is s*-locally dense*.

For $(X_n)_{n\geq 1}$ a sequence of positive random variables and $(y_n)_{n\geq 1}$ a sequence of positive non-random variables, let $X_n \simeq y_n$ indicate $\lim_{n\to\infty} X_n/y_n$ exists almost surely and equals

a finite and positive random variable. Theorem 2.5.3 shows the canonical model may be either globally sparse and/or dense. The theorem assumes a finite population of senders with number of e-mails per sender drawn from a multinomial distribution.

Theorem 2.5.3. *Suppose the sender population $\mathcal{P}_1$ is finite, consisting of d senders, i.e., $\mathcal{P}_1 = [d] := \{1, \ldots, d\}$. Assume, out of the n e-mails, the number of e-mails per sender s, denoted n_s, is drawn from a multinomial distribution with probabilities $(p_1, \ldots, p_d)$ such that $\sum_{s=1}^{d} p_s = 1$ and $p_s > 0$ for all $s \in [d]$. Let μ_s be the average size of emails for sender s and $\mu := \sum_{s=1}^{d} p_s \mu_s$ the average size of emails across all senders. Then $v(\mathbf{Y}_n) \simeq (\mu^{1/\alpha_\star} \mu_\star p_\star n)^{\alpha_0 \alpha_*}$ where $s^\star = \arg\max_{s \in [d]} \alpha_s$, $\mu_\star = \mu_{s^\star}$, and $\alpha_\star = \alpha_{s^\star}$. In particular, if $\mu^{-1} < \alpha \cdot \alpha_\star < 1$, then $(\mathbf{Y}_n)_{n \geq 1}$ is almost surely sparse.*

Theorem 2.5.3 establishes that the canonical HVCM for a special case of the sender distribution can capture degrees of sparsity. If $\mu_s = \mu$ for all $s \in \mathcal{P}_1$ and $\alpha\alpha_\star < \mu^{-1}$ then it must be the case that $\alpha\alpha_s < \mu^{-1}$ for all $s \in \mathcal{P}_1$. Therefore, a dense network must be s-locally dense for all $s \in \mathcal{P}_1$. However, a sparse network can be s-locally dense for some, but not all, $s \in \mathcal{S}$. We turn now to considerations of power-law degree distribution for interaction-labeled networks. We start with a definition.

Definition 2.5.4 (Global power-law degree distributions)**.** A sequence $(\mathbf{Y}_n)_{n \geq 1}$ exhibits *power-law degree distribution* [42], [62], [224] if for some $\gamma > 1$ the degree distributions $(d(\mathbf{Y}_n))_{n \geq 1}$ satisfy $d_k(\mathbf{Y}_n) \sim l(k)k^\gamma$ as $n \to \infty$ for all large k for some slowly varying function $l(x)$; that is, $\lim_{x \to \infty} l(tx)/l(x) = 1$ for all $t > 0$, where $a_n \sim b_n$ indicates that $a_n/b_n \to 1$ as $n \to \infty$. More precisely, $(\mathbf{Y}_n)_{n \geq 1}$ has power law degree distribution with index γ if

$$\lim_{k \to \infty} \lim_{n \to \infty} \frac{d_k(\mathbf{Y}_n)}{l(k)k^{-\gamma}} = 1. \tag{2.8}$$

Theorem 2.5.5 establishes the power-law degree distribution for the canonical HVCM for the case of $\alpha_s = 1, \forall s \in \mathcal{S}$.

Theorem 2.5.5. *Let $(\mathbf{Y}_n)_{n\in\mathbb{N}}$ obey the sequential description in Section 2.4 with parameters $(\tilde{\alpha}, \tilde{\theta})$ and let $\alpha_s = 1$ for all $s \in \mathcal{P}_1$. For each $n \geq 1$, let $p_n(k) = N_k(\mathcal{Y}_n)/v(\mathcal{E}_n)$ for $k \geq 1$ be the empirical receiver degree distribution where $N_k(\mathcal{E}_n)$ is the number of receivers of degree $k \geq 1$ and $v(\mathcal{E}_n)$ is the number of unique receivers in $\mathcal{E}_n$. Then, for every $k \geq 1$,*

$$p_n(k) \sim \alpha k^{-(\alpha+1)}/\Gamma(1-\alpha) \tag{2.9}$$

where $\Gamma(t) = \int_0^\infty x^{t-1}e^{-x}dx$ is the gamma function. That is, $(\mathcal{Y}_n)_{n\geq 1}$ has a power law degree distribution with exponent $\gamma = 1 + \alpha \in (1, 2)$.

2.6 Posterior inference

We now consider performing posterior inference for the canonical HVCM given an observed interaction network $\mathbf{Y}_n$. As in Section 2.4, we start with the setting where the size of the first component is one (i.e., $\nu_{k_1} = 1[k_1 = 0]$). The parameters $(v_{k_2}^{(s)})_{k_2\in\mathbb{N}}$, for all $s \in \mathcal{S}$ are estimated non-parametrically, and are not important for the remainder of the chapter; therefore, the details are omitted for these parameters.

We start by reparameterizing the HVCM in a more useful form for inference, and which gives an explicit structure for updating the latent degree $V_{n,j}$ - we call this the *extended canonical HVCM*, or extended model for short. In this representation, every "escape" from the local distribution and choice of receiver r leads to an auxiliary vertex v being introduced locally for a sender s - auxiliary vertices are not shared between senders. The *label* $l_s(v)$ of the auxiliary vertex is r; the auxiliary vertex accounts for the fact that the global distribution can select receiver r multiple times. Finally, the observed reciever is assigned to the auxiliary vertex, and we write that assignment $\phi_{n,j} = v$. The number of auxiliary vertices with label r and sender s is equal to the number of times the local distribution for sender s escapes and choose the global set of information (i.e., $V_{n,j}(r, s)$). The sum of the degrees across auxiliary vertices with label r and sender s is equal to the indegree for

receiver r (i.e., $D_{n,j}(r,s)$). Finally, we write d_{srv} to denote the degree of auxiliary vertex v in sender s that also has label r. Note that for $r' \neq l_s(v),\ d_{srv} = 0$.

Given $\mathcal{H}_n$ and $S_{n+1,1} = s$, the probability that $R_{n+1,j}$ is assigned to auxiliary vertex $\phi_{n+1,j} = v$ is:

$$\text{pr}(\phi_{n+1,j} = v \,|\, \mathcal{H}_n, S_{n+1,1} = s) \propto \begin{cases} d_{s\cdot v} - \alpha_s, & v \leq V_{n+1,j}(s,\cdot) \\ \alpha_s V_{n+1,j}(s,\cdot) + \theta_s, & v = V_{n+1,j}(s,\cdot) + 1, \end{cases}$$

Further, if $\phi_{n+1,j} = V_{n+1,j}(s,\cdot) + 1$, then we add an auxiliary vertex $V_{n+1,j}(s,\cdot) + 1$ with its label chosen with probability:

$$\text{pr}(l_s(V_{n+1,j}+1) \,|\, \phi_{n+1,j} = V_{n+1,j}(s,\cdot)+1, \mathcal{H}_{n,j}, S_{n+1} = s) \propto \begin{cases} V_{n,j}(\cdot,r) - \alpha, & r \in \mathcal{R}_{n+1,j} \\ \alpha V_{n+1,j}(\cdot,r) + \theta, & r \notin \mathcal{R}_{n+1,j}. \end{cases}$$

The likelihood of observing $\mathbf{Y}_N = \{\{S_{n,1}, k_n, \{R_{n,j}, \phi_{n,j}\}_{j=1}^{k_n}\}, \{l_s(\cdot)\}_{s\in\mathcal{S}_N}\}_{n=1}^N$ given the parameters $\Psi = (\tilde{\alpha}, \tilde{\theta}, \alpha, \theta, \{\alpha_s, \theta_s\}_{s\in\mathcal{P}_1})$ is given by

$$\text{pr}(\mathbf{Y}_N) = \text{pr}(\{\{R_{n,j}\}_{j=1}^{k_n}\}_{n=1}^N, l_s(\cdot)_{s\in\mathcal{S}_N} | \{S_{n,1}\}_{n=1}^N, \{k_n\}_{n=1}^N) \cdot \text{pr}(\{S_{n,1}\}_{n=1}^N)\,\text{pr}(\{k_n\}_{n=1}^N), \quad (2.10)$$

where

$$\text{pr}\left(\{\{R_{n,j}\}_{j=1}^{k_n}\}_{n=1}^N, l_s(\cdot)_{s\in\mathcal{S}_N} \,|\, \{S_{n,1}\}_{n=1}^N, \{k_n\}_{n=1}^N\right) = \frac{[\theta+\alpha]_\alpha^{K_N-1}}{[\theta+1]_1^{m_N-1}} \prod_r [1-\alpha]_1^{V_N(\cdot,r)-1} \prod_s \frac{[\theta_s+\alpha_s]_{\alpha_s}^{V_N(s,\cdot)-1}}{[\theta_s+1]_1^{m_N(s)-1}} \prod_{v=1}^{V_N(s,\cdot)} [1-\alpha_s]_1^{d_{srv}-1},$$

and

$$\text{pr}(\{S_{n,1}\}_{n=1}^{N}) = \frac{[\tilde{\theta}+\tilde{\alpha}]_{\alpha}^{\mathcal{S}_N}}{[\tilde{\theta}+1]_1^N}\prod_s [1-\tilde{\alpha}]_1^{D_N^{out}(s)-1},$$

$$\text{pr}(\{k_n\}_{n=1}^{N}) = \prod_{n=1}^{N} v_k^{(s)},$$

where $[a]_b^c = a(a+b)\dots(a+(c-1)b)$ for $c \in \mathbb{N}$ and $a, b \in \mathbb{R}_+$. The joint density as written in (2.10) is exchangeable with respect to re-ordering of the interactions.

Lemma 2.6.1 proves that the proposed canonical HVCM is recovered by marginalizing over configurations of auxiliary vertex labels and assignments, which leaves only the observed degrees $D_{n,j}$ and latent degrees $V_{n,j}$. The complete likelihood for the canonical model is given in Section A.3 of the appendix, and the likelihood is exchangeable, proving Theorem 2.5.1. Proof of Lemma 2.6.1 is also left to Section A.3 of the appendix.

Lemma 2.6.1. *Marginalizing the extended model over configurations of auxiliary vertex assignments and labels recovers the canonical model.*

2.6.1 Choice of priors

Here, we define two approaches to defining priors for the global parameters θ, α and local parameters $\theta_s, \alpha_s, s \in \mathcal{S}$.

2.6.1.1 Conjugate Bayesian Parameters

The first approach is to set the priors for θ parameters to a high-variance Gamma distribution, and the priors for the α parameters to the Beta distribution. In general, the global θ will be much larger than the local parameters, and the appropriate values will depend on the sparsity of the overall network - for instance, the global θ for the arXiv data is an order of magnitude greater than the global θ for the Enron dataset. we check the appropriateness of the prior distribution using posterior predictive checks on sparsity. See Section 2.7.3 for

details on posterior predictive checks and model comparison.

For datasets of reasonable size, we have found that the prior for the global parameters does not significantly affect the resulting posterior density. In the subsequent examples, the size of the datasets was more than sufficient to not be strongly affected by the choice of global priors. For the α parameter, this suggests using Beta$(1, 1)$ distribution, i.e., the Uniform distribution. With θ, different datasets can have a difference in posterior means that are 2 or 3 orders of magnitude. Although the posterior density is mostly unchanged, attempting inference with a mismatched θ prior will require more Gibbs samples before mixing occurs. We have found that $\theta \sim$ Gamma$(1, 10000)$ is an appropriate diffuse prior that allows for fast mixing. The lower-level parameter θ_s is generally much less than the global θ, so $\theta \sim$ Gamma$(1, 1000)$ is an appropriate prior that allows for variety in distribution but also has a prior mean that is lower than the global θ. For the local α_s, we again use the Uniform distribution.

2.6.1.2 Priors based on Hollywood model fits

The second approach, which is used in Section 2.8 for the arXiv dataset, is to fit the Hollywood model [62] to each of the local datasets, and then use a Gamma$(\hat{\theta}/100, 100)$ prior for each θ, where $\hat{\theta}$ is the estimate of θ for the Hollywood model. The priors for the α's are again set to Beta$(1, 1)$.

2.6.2 Gibbs sampling algorithm

Here we introduce a Gibbs sampling algorithm for sampling from the posterior distribution of Ψ given an observed interaction-labeled network $\mathbf{Y}_n$. To do this, we use auxiliary variable methods [73] to perform conjugate updates for all parameters. First, define the binary auxiliary variables $z_{r,j}$ for $r \in \mathcal{R}, j = 1, \ldots, v_{\bullet r} - 1$ and $z_{s,r,k,u}$ for $s \in \mathcal{S}, r \in \mathcal{R}, v = 1, \ldots, V_N(s, \cdot) - 1, u = 1, \ldots, d_{srv} - 1$. Next define auxiliary variables y_i for $i = 1, \ldots, v(\mathbf{Y}_n) - 1$ and y_{si} for $s \in \mathcal{S}$ and $i = 1, \ldots, d_{s\bullet\bullet} - 1$. Finally, define

auxiliary variables $x, \{x_s\}_{s\in\mathcal{S}} \in [0,1]$. We formally derive these updates in Section A.1 of the appendix; this algorithm is similar to the one described in [220], except for the modifications required for our model. While sampling each auxiliary vertex for the receivers, we also update the set of auxiliary vertices $[V_N(s,r)]$ and their degrees d_{srv}.

$$x \sim \text{Beta}(\theta + 1, m_N - 1) \tag{2.11}$$

$$y_i \sim \text{Bernoulli}\left(\frac{\theta}{\theta + \alpha \cdot i}\right), i = 1, \dots K_N - 1 \tag{2.12}$$

$$z_{rj} \sim \text{Bernoulli}\left(\frac{j-1}{j-\alpha}\right), r \in \mathcal{R}_n, j = 1, \dots, V_N(\cdot, r) - 1 \tag{2.13}$$

$$\theta \sim \text{Gamma}\left(\sum_{i=1}^{K_N-1} y_i + a, b - \log x\right) \tag{2.14}$$

$$\alpha \sim \text{Beta}\left(c + \sum_{i=1}^{K_N-1}(1-y_i), d + \sum_{r}\sum_{j=1}^{V_N(\cdot,r)-1}(1-z_{r,j})\right) \tag{2.15}$$

$$x_s \sim \text{Beta}\left(\theta_s + 1, V_N(s,\cdot) - 1\right), s \in \mathcal{S}_N \tag{2.16}$$

$$y_{si} \sim \text{Bernoulli}\left(\frac{\theta}{\theta + \alpha \cdot i}\right), s \in \mathcal{S}_N, i = 1, \dots d_{s\bullet\bullet} - 1 \tag{2.17}$$

$$z_{srvu} \sim \text{Bernoulli}\left(\frac{j-1}{j-\alpha}\right), s \in \mathcal{S}_N, r \in \mathcal{R}_N, v = 1, \dots, V_N(s,\cdot) - 1, u = 1, \dots, d_{srv} - 1 \tag{2.18}$$

$$\theta_s \sim \text{Gamma}\left(\sum_{i=1}^{d_{s\bullet\bullet}-1} y_{si} + a_s, b_s - \log x_s\right), s \in \mathcal{S}_N \tag{2.19}$$

$$\alpha_s \sim \text{Beta}\left(\phi\alpha + \sum_{i=1}^{d_{s\bullet\bullet}-1}(1-y_{si}), \phi(1-\alpha) + \sum_{r}\sum_{v=1}^{V_N(s,\cdot)-1}\sum_{u}^{d_{srv}-1}(1-z_{srvu})\right), s \in \mathcal{S}_n \tag{2.20}$$

There are two important differences between this algorithm and [220]. First, in the case of multiple elements in the first component, we perform an approximate sampling procedure found in Section 2.6.3 to find the latent Z_i. Second,the language model in [220] has multiple levels of hierarchical parameters, where we have only two levels of components. Convergence can be checked via traceplots and, in our experiments, occurs within the first

hundred or so iterations; see Figure 2.3 for traceplots in the email network example.

2.6.3 Approximate sampling in the case of multiple elements in the first component

In the case $\bar{S}_n$ may contain multiple elements, one can sample from the posterior

$$\text{pr}(Z_i = s|\mathcal{H}_n^{(z)}, \bar{S}_i) \propto \text{pr}(Z_i = s|\bar{S}_i)\,\text{pr}(\bar{R}_i = \bar{r}_i|Z_i = s).$$

Note that, in general, the joint likelihood $\text{pr}(\bar{R}_i = \bar{r}_i|Z_i = s)$ is difficult to calculate due to the marginalization over all possible vertex label configurations for $\bar{R}_i$. Instead, we propose a sampling procedure to approximate this quantity, by sequentially sampling the vertex labels $\bar{V}_i$ using the given counts, where $\bar{V}_i$ denotes the multiset $V_{i,1}, \ldots, V_{i,k_{i,2}}$:

$$\begin{aligned}&\text{pr}(\bar{R}_i = \{R_{i,1}, \ldots, R_{i,k_{i,2}}\}, \bar{V}_i|\mathcal{H}_n^{(z)}, Z_i) = \\ &\quad \prod_{j=1}^{k_{i,2}} \text{pr}(R_{i,j} = r_{i,j}, V_{i,j} = v_{i,j}|\mathcal{H}_n^{(z)}, R_{i,j-1} = r_{i,j-1}, R_{i,j-1} = r_{i,j-1}, \ldots, Z_i = s).\end{aligned}$$

After sampling $\bar{V}_i$ for a number of runs, we average the likelihoods to get an estimate of $\text{pr}(\bar{R}_i = \bar{r}_i|Z_i)$.

2.7 Application to Enron email network

In this section the proposed HVCM model and inference procedure is applied to the Enron email dataset. Further, techniques to demonstrate the goodness of fit of the model are discussed, and are applied in comparison with with the previously published "Hollywood" model [62] and the generalized gamma process (GGP) model [42]; in particular, the HVCM model is shown to have better model fit at the local level compared to others.

2.7.1 Dataset overview

The Enron email dataset consists of approximately 500,000 emails collected from 1998 to 2002 and was originally collected by the Federal Energy Regulatory Commission during its investigation into the company [55]. The dataset originates from an email dump of 150 users. In total, there are 19,752 unique senders, 70,572 unique receivers, for a total of 79,735 unique entities. The dataset has been used as a testbed for classification [122], topic modeling [150], and graph-based anomaly detection [190], [209], among other tasks.

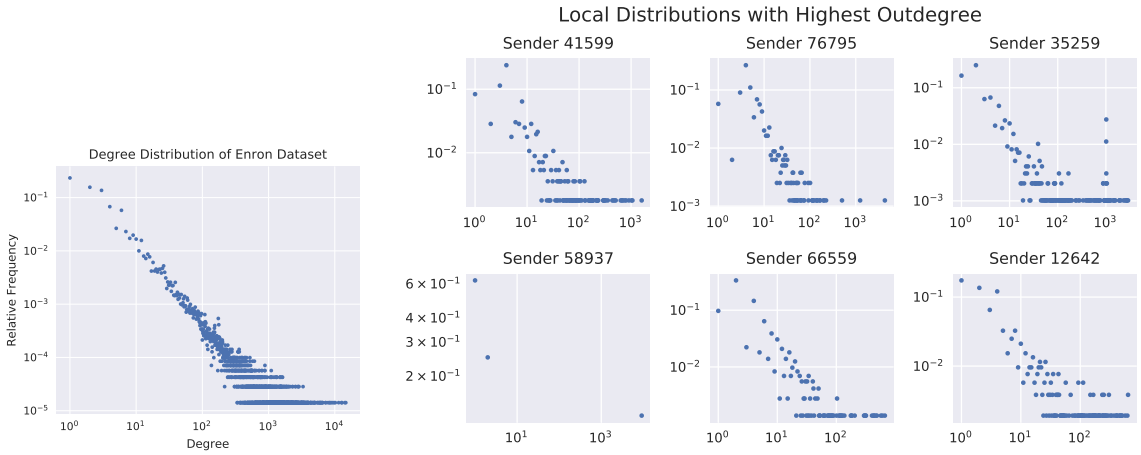


Figure 2.2: Global receiver distribution (left) and some examples of local degree distributions (right). There is variation in the shape of these distributions; the HVCM accounts for and parameterizes this difference in behavior when compared with the global degree distribution.

Figure 2.2 shows the global receiver degree distribution, as well as the local receiver distributions for the six senders with the largest number of emails. There is significant variation in behavior of the local degree distributions, both in comparison to themselves and to the behavior of the global degree distribution. This suggests that a modeling approach that allows for these differences is critical to accurately capturing the behavior of the entities, and thereby allowing for superior data summarization, sound inferences and strong prediction performance. While the Hollywood and GGP model would be unable to account for this variation, the proposed HVCM is equipped to capture this behavior.

2.7.2 Fit to the data

Figure 2.3: Trace plots and histograms for global parameters of the Enron data. Mixing occurs after 50 iterations or less. For the posterior predictive checks, the last 500 posterior samples were used.

The Gibbs sampling algorithm introduced Section 2.6 is applied to the dataset for 1000 iterations, discarding the first 500 as burn-in. For this dataset, the following priors were used:

$$\text{pr}(\theta) \sim \text{Gamma}(2, 1000), \qquad \text{pr}(\alpha) \sim \text{Beta}(1, 1)$$

$$\text{pr}(\theta_s) \sim \text{Gamma}(1, 20), \qquad \text{pr}(\alpha_s) \sim \text{Beta}(1, 0.9)$$

Trace plots and histograms of the posterior samples of the global parameters α and θ are displayed in Figure 2.3. Note that discarding 500 posterior samples as burn-in is rather conservative, as the Gibbs sampler sampled chain mixes in less than 100 iterations.

We show the histogram of posterior means of the local parameters θ_s and α_s in Figure 2.4, along with their priors. The θ_s parameters are shown on a log scale. These local histograms show significant diversity among the posterior parameter estimates, as we are

fitting local variations in behavior. For α_s, the choice of prior has very little effect on the posterior samples, except in the case of a small amount of local data for that particular sender s. The choice of prior for θ_s has more influence on the posterior distribution; our prior of Gamma$(1, 20)$ is set to bias the local θ_s towards 0; this will allow for a better fit on the local data than a prior with larger variance or mean; this result is borne out when posterior predictive checks are applied to the local sender distributions, i.e., Figures 2.5 and 2.7.

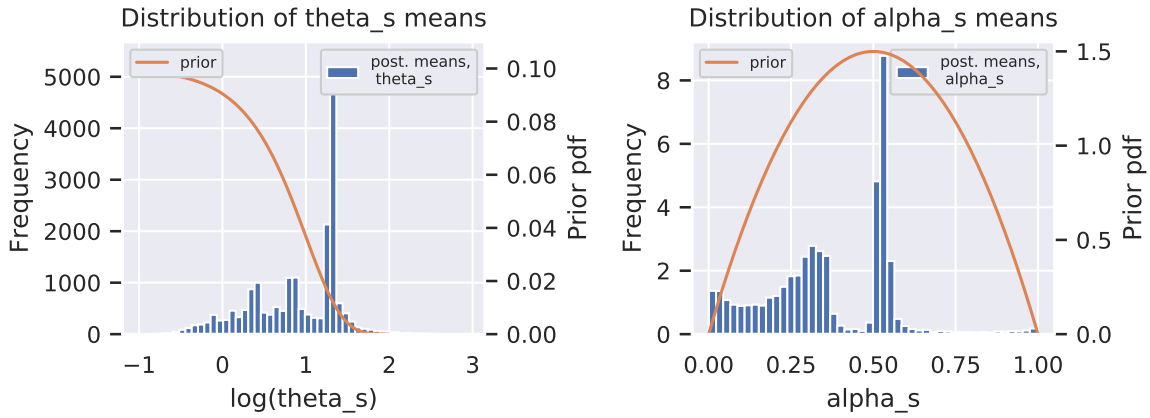


Figure 2.4: Histograms of local α_s and θ_s. Prior pdfs are shown in orange. The θ_s prior is set to fit the local distributions; the α_s posterior means are robust to the prior distribution chosen.

2.7.3 Posterior predictive checks (PPC) and model comparison

In this section, examples of posterior predictive model checks are shown in order to demonstrate the goodness of fit of the proposed HVCM. Posterior predictive checks are often used in order to verify that the proposed fitted model generates reasonable values on statistics of interest; these checks can also be used to diagnose where the model fails to perform well [85].

Multiple synthetic datasets are generated according to the posterior predictive distribution as prescribed in [85], and statistics of interest are calculated and compared with the

statistics of the real data. The synthetic data is generated from the model with the parameters set to a posterior sample generated from the inference procedure. Since we are interested in the ability of the model to account for variation in local behavior, we take the sender sequence and number of receivers for each email as given, in order to directly compare the local receiver distributions of the posterior predictive data with the real data.

In addition to generating posterior predictive checks for the fitted HVCM, they are also generated for the Hollywood [62] and GGP [42] models for comparison. In the following subsections, a variety of posterior predictive statistics are described, both for the global dataset and for the local data per sender. These checks show that the proposed HVCM both provides a good global fit of the data, in addition to significantly improving the fit to local distributions compared to the Hollywood model. Table 2.2 details the 95% posterior predictive intervals for the global statistics, and Table 2.3 summarizes the posterior predictive coverage rate for local distributions for the proposed model and the Hollywood model. The statistics compared are number of unique receivers in the dataset and number of receivers with degree 1, 10, and 100.

2.7.3.1 Number of unique receivers

The first statistic we consider is the number of unique receivers, both in the global dataset as well as each local sender datasets. The number of unique receivers can be thought of as a surrogate for sparsity, and thus an important statistic for a candidate model to replicate. Figure 2.5 displays the results.

On the left plot, the PPC statistics are shown for the number of unique receivers in the global dataset. Both the Hollywood model and the proposed HVCM perform well on the global statistic. On the left are four examples of the PPC statistics for the number of unique receivers on the local sender datasets with the most emails. Only the results from the proposed HVCM is shown, because neither the GGP model nor the Hollywood model is able to take into account variation among the local distributions; if the sender labels

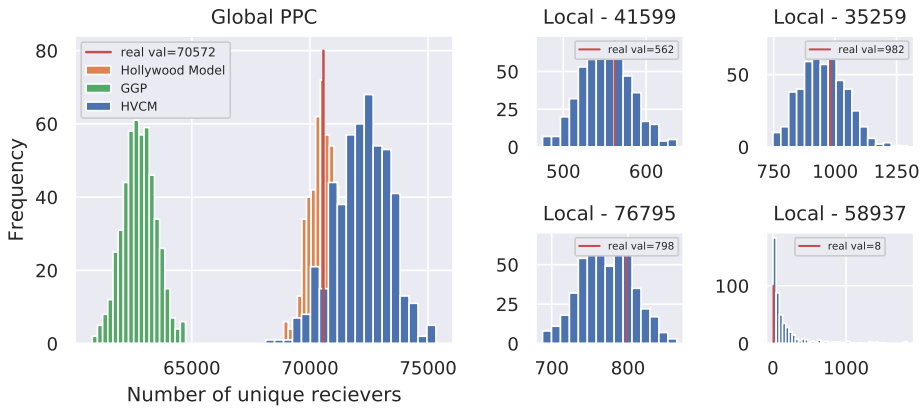


Figure 2.5: PPC Statistics for number of unique receivers, global (left) and examples of local (right).

are attached post-hoc to the synthetic data generated from the GGP or Hollywood model, they are completely unable to replicate any local behavior statistics. The HVCM clearly accounts for the varying local behavior, even when that local behavior is unusual (in the case of sender 58937). The superiority of the model compared to the Hollywood model is clearly shown in Table 2.3, as the proposed model's local posterior predictive intervals in the local distributions covers the real values 99% of the time, as opposed to the Hollywood model's coverage rate of 39%.

2.7.3.2 Degree distribution

An important global behavior to capture is the global degree distribution. In order to evaluate this fit, posterior predictive intervals of the number of nodes with degree 1, 10, and 100 are shown in Table 2.2. Note that the HVCM performs the best, where the real number of receivers with degree 10 are within the PP interval. Figure 2.6 shows this result. When comparing the degree distributions, it is also clear that the Enron data does not perfectly align with the posterior predictive example, as the synthetic data overestimates the number of receivers with degree 1 and underestimates the number of receivers with degree 100. However, it is also clear that this model fit is still superior to both comparators, via Table 2.2. Further, Table 2.3 demonstrates that the coverage for the posterior predictive intervals is

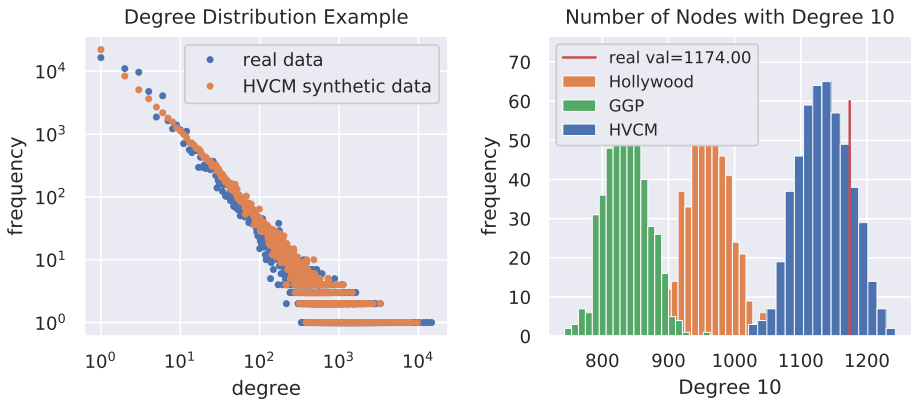


Figure 2.6: Comparison of degree distribution between a posterior predictive sample from the proposed model and the real data (left) and PPC of the number of receivers with degree 10.

much more robust in the proposed model for each of the degree statistics. Figure 2.7 also compares local degree distributions between the HVCM and the real data. In the both the global and local case, the HVCM is able to better replicate the degree distribution.

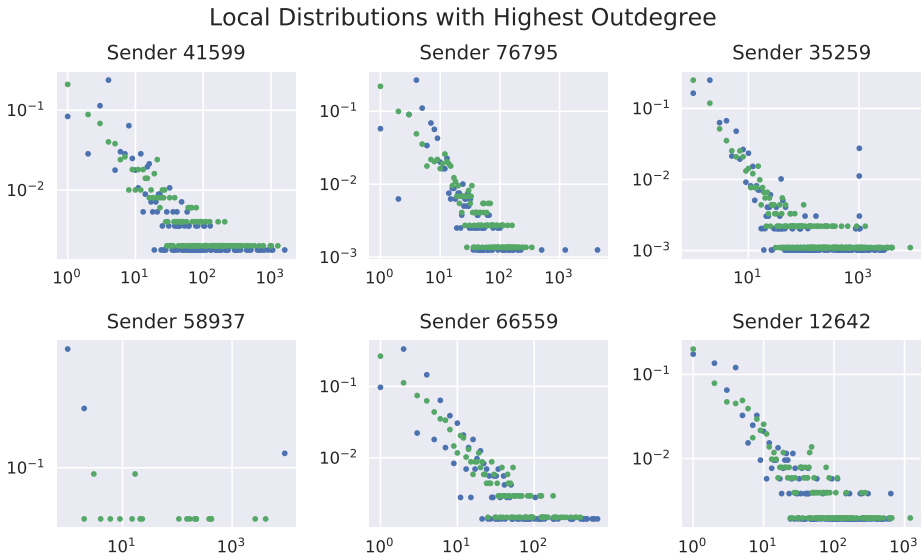


Figure 2.7: Comparison of HVCM and Hollywood model for local distributions.

	Unique Receivers	Receivers with degree 1	Receivers with degree 10	Receivers with degree 100
HVCM	**(69881, 74299)**	(21504, 23022)	**(1057, 1207)**	(30, 56)
Hollywood Model	**(69382, 71671)**	(23031, 23997)	(893, 1022)	(31, 59)
GGP Model	(61309, 64175)	(20653, 22006)	(778, 898)	(26, 51)
Actual Value	70572	16495	1174	15

Table 2.2: Posterior predictive confidence intervals (95%) for global statistics.

	Unique Receivers	Receivers with degree 1	Receivers with degree 10	Receivers with degree 100
HVCM	19725 / 19752	18233 / 19752	808 / 960	14 / 22
Hollywood Model	7652 / 19752	7652 / 19752	48 / 960	1 / 22

Table 2.3: Posterior predictive coverage rates of the local distributions when using the 95% posterior predictive interval.

2.7.3.3 Node sharing across local distributions

In order to visualize how effectively the proposed HVCM is capturing the varying dependencies between the local and global distributions, we count the number of receivers that are seen in a particular number of local sender distributions. This allows for direct comparison of the effectiveness of the models to capture the interdependency and interaction among the local datasets. Figure 2.8 shows the results.

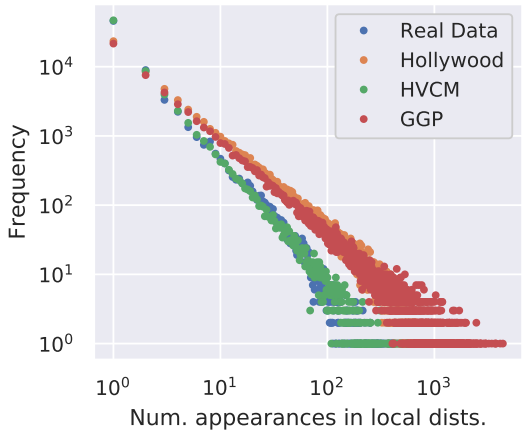


Figure 2.8: Distribution of nodes that have been in x number of local sender distributions.

It is quite clear that the proposed HVCM replicates the observed behavior in the real data, while both the GGP and Hollywood models fail to capture the degree of pooling across the local datasets. Specifically, the other models seem to overestimate the rate at which receivers are shared across the local distributions.

2.7.3.4 L1 distance from degree distribution

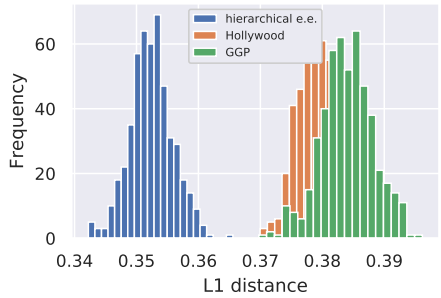


Figure 2.9: Histograms of L1 distance between degree distributions of synthetic PPC datasets and Enron global degree distribution. The proposed model better captures the distribution than the Hollywood model and GGP model.

With our posterior predictive samples, we can also directly examine the difference in distribution between synthetic data and the real data. Figure 2.9 shows histograms of the TV distance between the global degree distributions of the synthetic data generated from the posterior predictive distribution and the real dataset. Again, the proposed HVCM leads to an improvement over the Hollywood model and GGP model.

2.8 ArXiv dataset

In this section, a larger and more complex dataset is used to demonstrate the flexibility of the proposed HVCM. The hierarchical exchangeable model is applied to the arXiv dataset https://archive.org/details/arxiv-bulk-metadata, which contains nearly all arXiv articles from 1986 to 2017. Like the Enron dataset, the arXiv data has a hierarchical structure — each article is required to have at least one associated subject.

However, unlike the email dataset, which had only one sender per email, each article may have more than one subject. Our proposed model is well suited to this case of multiple entities and the data can still be appropriately represented by Equation (2.3). Further, our model allows for the direct study of interdisciplinarity among authors and overlap among the subject classes on arXiv.

The arXiv subjects have been divided into 11 main classes; the full list can be found on https://arxiv.org/help/prep. In order to reduce the effect of author name ambiguity, we restricted ourselves to articles which have at least one subject from the `math`, `cs`, `stat`, and `physics` subject classes. A full description of the subjects of interest is found in Section A.4 of the appendix. Figure 2.10 shows a degree distribution for the subjects, along with a histogram of the number of subjects per article. In total, there are 510812 scientific articles with 413029 unique authors and 130 unique subjects, There is also a broad range of subject frequencies, with the most popular subject being `math-ph` (mathematical physics) with 47942 articles, and the least popular subject `cs.GL` (general literature) with 130 articles.

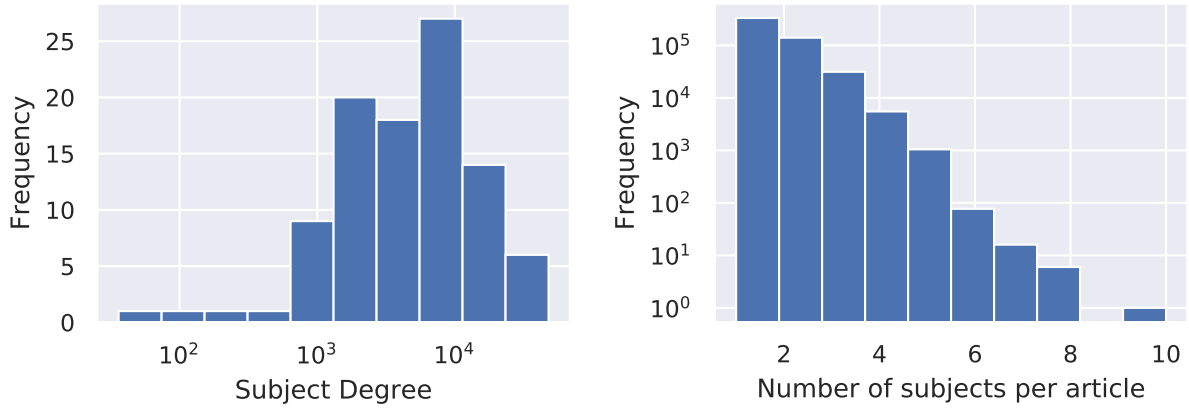


Figure 2.10: Degree distribution of subjects, log scale. This degree distribution does not exhibit a power-law. The proposed HVCM accounts for this extra complexity. (α_s are not constrained to be equal to 1.)

We apply our posterior sampling methods found in Section 2.6, and in particular use the approximate method of calculating the posteriors of the indicator variables Z_i using the

methods described in Section 2.6.3. Trace plots of posterior estimates of certain parameters, posterior predictive checks for the data and other details of the inference can be found in Section A.4 of the appendix

2.8.1 Subject overlap

The fitted model allows us to explore the amount of overlap between arXiv subjects. Two subjects are considered overlapping if the model has difficulty distinguishing between them when they are used as labels for the same article. This difficulty can be measured using the Shannon entropy, which is defined over discrete probability distributions $p = [p_1, p_2, \ldots, p_k]$ as:

$$H(p_1, p_2, \ldots, p_k) = -\sum_k p_k \log_2 p_k.$$

Entropy is at its maximum when the distribution p is the uniform distribution, i.e., when all outcomes are equally likely. In order to estimate subject overlap for subjects s_1 and s_2, every article which lists s_1 and s_2 among its subjects is found, and the entropy of the posterior mean of the Z_i distribution given that the subject is either s_1 or s_2 is calculated, and the entropy is averaged over the articles. This score, $\text{SO}(s_1, s_2)$ is computed as:

$$\text{SO}(s_1, s_2) = \frac{1}{|\{\bar{S}_i : s_1, s_2 \in \bar{S}_i\}|} \sum_{i: s_1, s_2 \in \bar{S}_i} H(\text{pr}(Z_i = s | \{\bar{S}_i, \bar{R}_i\}, Z_i \in \{s_1, s_2\})) \quad (2.21)$$

Figure 2.11 shows a heatmap of the subject overlap scores for subjects that are seen in the same article at least 100 times. The subjects are ordered according to a normalized spectral clustering [166], using the subject overlap matrix SO as the affinity matrix, and setting the number of clusters to 6.

From this analysis, we conclude the following. Cluster 1, which includes cs.AI (Artificial Intelligence) and cs.IR (Information Retrieval), is a group of subjects that pertain to

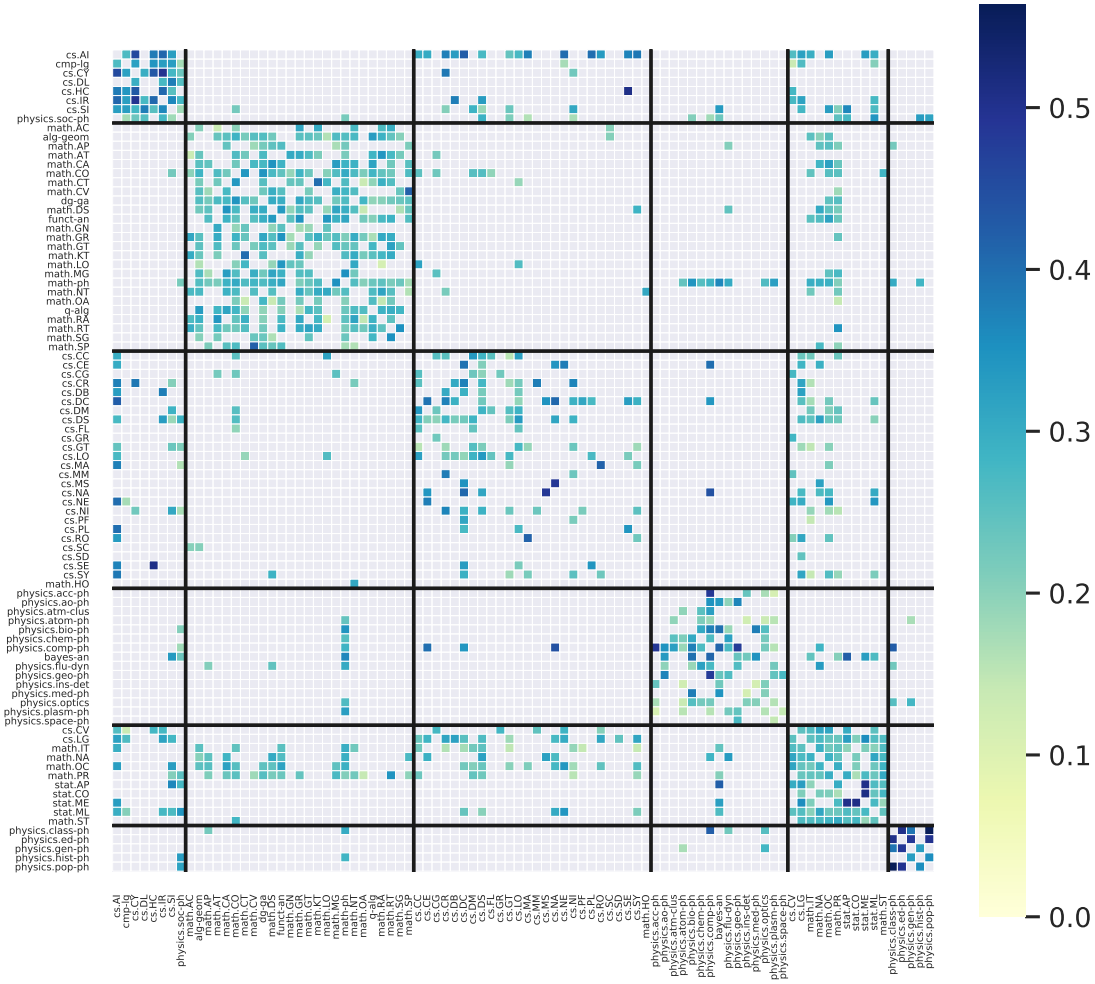


Figure 2.11: Heatmap of two-way entropy per article. For each pair of subjects s_1, s_2, and every article that contains both s_1 and s_2, the entropy of $\text{pr}(Z_i = z | Z_i \in \{s_1, s_2\})$ is calculated and summed. Finally, each entry is normalized by the total number of occurences of s_1 and s_2 appearing together in the same article.

algorithmic approaches to artificial intelligence. Note that this cluster is differentiated from cluster 5, which tends to represent more theoretical papers that rely heavily on statistical techniques; this cluster includes math.ST (Statistical Theory), stat.ML (Machine Learning), and stat.ME (Methods). Cluster 2 can be considered the core math cluster, which encapsulates many pure and applied math subjects. Similarly, cluster 3 is the core computer science

cluster, which are the computer science subjects that generally don't use statistics such as cs.SE (Software Engineering) and cs.CE (Computer Engineering). Cluster 4 is the core physics cluster, with the subjects of physics that tend not to be interdisciplinary outside of physics as other physics subjects. Finally, cluster 6 consists of subjects that involve the philosophy, teaching or history of physics. Table 2.4 lists the pairs of subjects with the most overlap according to the entropy score 2.21. Note that these pairs correspond with the general intuition of subjects that would have a large degree of interdisciplinarity.

Table 2.4: Pairs of subjects with highest subject overlap score.

s_1	s_2	SO(s_1, s_2)
stat.ME (Methods)	stat.CO (Computation)	0.509
cs.SE (Software Engineering)	cs.HC (High Perf. Comp.)	0.507
physics.class-ph (Classical Physics)	physics.ed-ph (Education)	0.504

We compare these results with results of a direct application of a spectral clustering algorithm to the co-authorship network in Section A.4 of the appendix. This direct application of spectral clustering to the data is unable to recover the meaningful groupings that the proposed HVCM produces.

2.9 Concluding remarks

This chapter has presented the class of exchangeable structured interaction models. By exploiting the common hierarchical nature of structured network data, complex models with both appropriate invariance and empirical properties are introduced. The canonical HVCM captures partial pooling of information, and can model complex local-behavior with global power-law degree behavior. A Gibbs sampling algorithm is proposed and applied to the Enron e-mail and arXiv datasets. While the focus of this chapter has been on e-mail and similarly structured interaction datasets, extensions to more complex examples will be in considered future work. This chapter lays the foundation for how the interaction exchangeability framework can account for complex behavior. Of course, many interaction net-

works occur with time-stamps; therefore, extensions to account for temporal dependence is required and will be an important next step.

CHAPTER 3

Learning to Bound the Multi-class Bayes Error with Graph Based Methods

In the context of supervised learning, meta learning uses features, metadata and other information to learn about the difficulty, behavior, or composition of the problem. Using this knowledge can be useful to contextualize classifier results or allow for targeted decisions about future data sampling. In this chapter, we are specifically interested in learning the Bayes error rate (BER) based on a labeled data sample. Providing a tight bound on the BER that is also feasible to estimate has been a challenge. Previous work [230] has shown that a pairwise bound based on the sum of Henze-Penrose (HP) divergence over label pairs can be directly estimated using a sum of Friedman-Rafsky (FR) multivariate run test statistics. However, in situations in which the dataset and number of classes are large, this bound is computationally infeasible to calculate and may not be tight. Other multi-class bounds also suffer from computationally complex estimation procedures. In this chapter, we present a generalized HP divergence measure that allows us to estimate the Bayes error rate with log-linear computation. We prove that the proposed bound is tighter than both the pairwise method and a bound proposed by Lin [135]. We also empirically show that these bounds are close to the BER. We illustrate the proposed method on the MNIST dataset, and show its utility for the evaluation of feature reduction strategies.

Symbol	Description
$(\mathbf{x}, y)$	Observed feature-label pair
$(\mathbf{X}, Y)$	Random feature-label pair
$p_1, p_2, \ldots, p_m$	Prior label probabilities
$f_1, f_2, \ldots, f_m$	Class conditional feature pdfs
ϵ^m	Bayes error rate

Table 3.1: Glossary of commonly used symbols.

3.1 Introduction

Meta learning is a method for learning the intrinsic quality of data directly from a sample of the data, metadata, or other information [44], [191]. The purpose of meta learning is to collect knowledge that might be helpful at other levels of processing and decision-making. Examples where meta learning is applied include sequential design of experiments [41], reinforcement learning [95], and sensor management [233] in the fields of statistics, machine learning, and systems engineering, respectively. In supervised learning, and particularly for multi-class classification, one form of meta learning is to learn bounds or estimates on the Bayes error rate (BER). The BER is the minimal achievable error probability of any classifier for the particular learning problem, and knowledge of it can be used at other stages of meta learning, such as in the selection of the classifiers, model selection, and feature reduction. Hence, finding computable bounds and approximations to the BER is of interest, and is the problem we consider in this chapter.

Consider the problem where a feature vector $\mathbf{X}$ is labeled over m classes $C_1, \ldots, C_m$. Available are i.i.d. pairs $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$, called training data, where $\mathbf{x}_i$ is a realization of the random vector (feature) $\mathbf{X} \in \mathbb{R}^d$ and y_i is a realization of the random variable (label) $Y \in \{1, 2, \ldots, m\}$. Assume the prior label probabilities $p_k = P(Y = k)$, with $\sum_{k=1}^m p_k = 1$ and the conditional feature densities $f_k(\mathbf{x}) = f(\mathbf{x}|Y = k)$, for $k = 1, \ldots, m$. Then the Bayes error rate is given by

$$\epsilon^m = 1 - \int \max\{p_1 f_1(\mathbf{x}), p_2 f_2(\mathbf{x}), \ldots, p_m f_m(\mathbf{x})\} \, \mathrm{d}\mathbf{x}. \tag{3.1}$$

This represents the error achieved by the Bayes classifier, g_{Bayes} that minimizes the average $0-1$ loss. The Bayes classifer assigns an estimated class label $\hat{y}$ to an observation $\mathbf{x}$ according to the maximum a posteriori (MAP) rule

$$\hat{y} = g_{Bayes}(\mathbf{x}) = \underset{k \in \{1,2,\ldots,m\}}{\arg\max} \; P(Y = k | \mathbf{X} = \mathbf{x}).$$

Many different upper and lower bounds on the BER (3.1) exist for the case of $m = 2$ classes, and many of these are related to the family of f-divergences. A bound based on Chernoff α-divergence has been proposed in [53], but in general it is not very tight in the finite sample regime. In [26], a tighter bound for the 2-class BER using Henze-Penrose [99] divergence was proposed. The HP bound has the advantage that it can be directly estimated from the training data using a minimal spanning tree. The same framework can be extended in a pairwise fashion to the m-class multi-class classification problem. However, when m is relatively large, the derived pairwise bounds are loose and often times trivial [236]. The method proposed in this chapter alleviates this problem by introducing new bounds based on a generalized Henze-Penrose measure adapted to m-class problem, and whose tightness does not diminish as m increases. Additionally, the new bounds improve upon other bounds that were designed specifically for the multi-class problem, such as the generalized Jensen-Shannon (JS) divergence bound [135].

Most approaches to estimation of bounds on Bayes error use plug-in, also called substitution, estimators. These approaches require estimation of the multivariate feature densities followed by evaluation of the BER bounds using these estimated densities in place of the true densities. Recently, approaches to estimating BER bounds using direct estimators have been proposed. For example, graph-based BER bound estimation approaches bypass density estimation entirely, producing an estimator of the information divergence using geometric functions of the data. These procedures scale linearly in dimension, resulting in faster computation than plug-in methods for high dimensional features. In the original

2-class setting, as shown in [26], bounds based on Henze-Penrose divergence can be estimated directly from data by employing the Friedman-Rafsky (FR) run test statistic [81], [99], which computes a minimal spanning tree (MST) over the data, and counts the number of edges that connect dichotomous data points. A brute force extension of the FR approach to the m-class classification problem would require an MST computation for each pair of classes, or $O(m^2)$ MSTs, which significantly reduces its computational tractability for large m. The extension proposed in this chapter also uses a graph-based estimation procedure, but only requires a single MST calculation on the entire dataset. Thus, the proposed approach is more computationally efficient when m and n are large.

3.1.1 Related work

Broadly defined, meta learning is a set of methods of learning from knowledge that can be used to improve performance or understanding of the problem. Estimating the Bayes multi-class classification error is a meta learning problem. The principles behind the frameworks proposed in [136] and [83] have been utilized to estimate the multi-class BER by bounding the BER by a sum of pairwise BERs for each pair of class labels. There exist many useful bounds on the pairwise BER that are based on information divergence measures, i.e., measures of dissimilarity between two distributions. Several bounds for the pairwise BER have been proposed, including: Chernoff bound [53]; Bhattacharyya bound [117]; and HP-divergence [26]. The Henze-Penrose divergence yields tighter bounds on the BER than those based on the Bhattacharya distance for equal label priors. For the multi-class BER, the sum of pairwise bounds given in [26] was proposed.

Another approach to bounding the BER of multi-class classifiers uses the Jensen-Shannon (JS) divergence. The JS-divergence assigns a different weight to each conditional probability distribution and this inspired Lin to propose a bound for the binary BER where the weights depend on the priors. The generalized multi-class Jensen-Shannon divergence is related to the Jensen difference proposed by Rao [196], [197]. In [135], the author pro-

posed a generalized JS-divergence that was used to derive a bound on the Bayes error rate for multi-class classification.

In the nonparametric setting, the most popular approach for estimating bounds has been plug-in estimators, which require estimation of probability densities that are subsequently "plugged into" the expression for the divergence function, [154]–[156]. These approaches are generally multi-step procedures and computationally expensive, especially in high dimensions [155]. In [168], Nguyen et al. proposed a divergence estimation method based on estimating the likelihood ratio of two densities that achieves the parametric mean squared error (MSE) convergence rate when the densities are sufficiently smooth.

Direct estimation approaches bypass density estimation, producing an estimator of the information divergence using geometric functions of the data. As previously mentioned, the MST-based Friedman-Rafsky two sample test statistic [81], [99] is an asymptotically consistent estimator of the HP-divergence, which can then be used to estimate upper and lower bounds for the 2-class classification problem [26]. There are other graph-based estimators that have been proposed in the literature. In [98], Henze proposed a graph-based estimator for HP-divergence that employs the K-nearest neighbor (K-NN) graph instead of the MST. The authors of [158] developed an approach for estimating general f-divergences called the Nearest Neighbor Ratio (NNR), also using K-NN graphs. In [172] the authors developed a general divergence estimator based on Locality Sensitive Hashing (LSH). In [237], the authors showed that a cross match statistic based on optimal weighted matching can also be used to directly estimate the HP divergence. None of these papers on geometric methods proposed extensions to multi-class classification, which is the main contribution of this chapter.

3.1.2 Contribution

We introduce a computationally scalable and statistically consistent method for learning to bound the multi-class BER. First, we propose a novel measure, the generalized Henze-

Penrose (GHP) integral, for bounding multi-class BER. We show how this generalized integral can be used to calculate bounds on the Bayes error rate, and prove that they are tighter than both the pairwise and JS multi-class BER bounds. Further, we empirically show that the bounds' performance is consistent and is robust to sample size and the number of classes.

Our second contribution is a scalable method for estimating the GHP integral, and subsequent estimation of the BER bounds. The proposed algorithm uses a single global minimal spanning tree (MST) constructed over the entire dataset. We show that this is more computationally efficient than the pairwise approach, which must compute $O(m^2)$ pairwise MSTs.

3.1.3 Organization of the chapter

The chapter is organized as follows. In Section 3.2.1 we briefly review the HP divergence and propose the generalized HP-integral (GHP) measure. The motivation and theory for the various bounds such as the pairwise HP divergence and generalized JS divergence for the multi-class Bayes error is reviewed in Section 3.3, and a new bound based on our GHP measure is given. We numerically illustrate the theory in Section 3.5. In Section 3.6 we apply the proposed method to a real dataset, the MNIST image dataset. Finally, Section 3.7 concludes the chapter. The main proofs of the theorems are found in Appendix B.

3.2 The divergence measure and generalizations

In this section we recall the Henze-Penrose (HP) divergence between pairs of densities and define a generalization for multiple densities (≥ 2) that will be the basis for bounding the multi-class BER called the Generalized HP (GHP) integral.

3.2.1 Henze-Penrose divergence

For parameters $p \in (0,1)$ and $q = 1-p$ consider two density functions f and g with common domain $\mathbb{R}^d$. The Henze-Penrose divergence $D(f_1, f_2)$ is given by

$$D(f,g) = \frac{1}{4pq}\left[\int \frac{\big(pf(\mathbf{x}) - qg(\mathbf{x})\big)^2}{pf(\mathbf{x}) + qg(\mathbf{x})}\,\mathrm{d}\mathbf{x} - (p-q)^2\right]. \tag{3.2}$$

The HP-divergence (3.2), first introduced in [25], has the following properties: (1) $0 \leq D \leq 1$, (2) $D = 0$ iff $f(\mathbf{x}) = g(\mathbf{x})$. Note that the HP-divergence belongs to the f-divergence family [10], [63], [161].

In the multi-class classification setting, as defined in the introduction, consider a sample of i.i.d. pairs $(\mathbf{x}_i, y_i)$, $i = 1, \ldots, n$, where $y_i \in \{1, \ldots, m\}$ are class labels with prior probabilities $\{p_k\}_{k=1}^m$ and, given $y_i = k$, $\mathbf{x}_i$ has conditional density f_k. Define $\widetilde{p}_{ij} = p_i/(p_i+p_j)$. Note that $\widetilde{p}_{ij} \neq \widetilde{p}_{ji}$ and $\widetilde{p}_{ij} + \widetilde{p}_{ji} = 1$. Let $\mathbb{S}^{(i)}$ be the support set of the conditional distribution f_i. The Henze-Penrose (HP) divergence measure between distributions f_i and f_j with union domain $\mathbb{S}^{(ij)} = \mathbb{S}^{(i)} \cup \mathbb{S}^{(j)}$ is defined as follows (see [25], [26], [99]):

$$D(f_i, f_j) = \frac{1}{4\widetilde{p}_{ij}\widetilde{p}_{ji}}\Big[\int_{\mathbb{S}^{(ij)}} \frac{\big(\widetilde{p}_{ij}f_i(\mathbf{x}) - \widetilde{p}_{ji}f_j(\mathbf{x})\big)^2}{\widetilde{p}_{ij}f_i(\mathbf{x}) + \widetilde{p}_{ji}f_j(\mathbf{x})}\,\mathrm{d}\mathbf{x} - (\widetilde{p}_{ij} - \widetilde{p}_{ji})^2\Big]. \tag{3.3}$$

An alternative form for $D(f_i, f_j)$ is given in terms of the HP-integral:

$$\mathrm{HP}_{ij} := \mathrm{HP}\big(f_i, f_j\big) = \int_{\mathbb{S}^{(ij)}} \frac{f_i(\mathbf{x})f_j(\mathbf{x})}{p_i f_i(\mathbf{x}) + p_j f_j(\mathbf{x})}\mathrm{d}\mathbf{x}, \tag{3.4}$$

yielding the equivalent form to (3.3)

$$D(f_i, f_j) = 1 - (p_i + p_j)\mathrm{HP}_{ij}.$$

In [26] it was shown that $0 \leq D(f_i, f_j) \leq 1$, and that the HP-integral is upper bounded by

$$\mathrm{HP}\big(f_i, f_j\big) \leq (p_i + p_j)^{-1}.$$

3.2.2 Generalized HP-integral

Define the union of all support sets as $\mathbb{S} = \bigcup_{k=1}^{m} \mathbb{S}^{(k)}$ and the difference between the m-fold support set and the 2-fold support set $\mathbb{S}^{(ij)}$ as $\overline{\mathbb{S}}^{(ij)} = \mathbb{S} \big/ \mathbb{S}^{(ij)}$. We denote $f^{(m)}(\mathbf{x})$ the marginal distribution of $\mathbf{X}$,

$$f^{(m)}(\mathbf{x}) := \sum_{k=1}^{m} p_k f_k(\mathbf{x}) = \sum_{k=1}^{m} p_k f(\mathbf{x}|y=k).$$

Define the generalized HP-integral (GHP-integral) by

$$\mathrm{GHP}_{ij}^{m} := \mathrm{GHP}^{m}\big(f_i, f_j\big) = \int_{\mathbb{S}} f_i(\mathbf{x}) f_j(\mathbf{x}) \big/ f^{(m)}(\mathbf{x}) \, \mathrm{d}\mathbf{x}. \tag{3.5}$$

The following theorem establishes a relation between the HP-integral and the GHP-integral:

Theorem 3.2.1. *Consider conditional probability densities $f_1, \ldots, f_m$ with priors $p_1, \ldots, p_m$ such that $p_1 + p_2 + \cdots + p_m = 1$. The HP-integral and the GHP-integral are related as follows:*

(a) If $\big(\mathbb{S}^{(i)} \cup \mathbb{S}^{(j)}\big) \cap \bigcup_{k \neq i,j} \mathbb{S}^{(k)} = \emptyset$, then

$$\mathrm{HP}(f_i, f_j) = \mathrm{GHP}^{m}\big(f_i, f_j\big), \tag{3.6}$$

(b) If $\big(\mathbb{S}^{(i)} \cup \mathbb{S}^{(j)}\big) \cap \bigcup_{k \neq i,j} \mathbb{S}^{(k)} \neq \emptyset$, then there exists a constant C depending only on priors $p_1, p_2, \ldots, p_m$ such that

$$\mathrm{HP}(f_i, f_j) \leq \mathrm{GHP}^{m}\big(f_i, f_j\big) + C\Big(1 - D\Big(\widetilde{p}_{ij} f_i + \widetilde{p}_{ji} f_j, \sum_{k \neq i,j} \widetilde{p}_k^{ij} f_k\Big)\Big), \tag{3.7}$$

where $\widetilde{p}_{ij} = p_i/(p_i + p_j)$ *and* $\widetilde{p}_k^{ij} = p_k \big/ \sum_{r \neq i,j} p_r$.

The full proof of Theorem 3.2.1 is given in Appendix B. Part (a) can be easily derived. The proof of part (b) depends on the fact that there exists a constant C_1 depending on the p_i and p_j such that for every f_i and f_j

$$f_i(\mathbf{x}) f_j(\mathbf{x}) \leq C_1 \left(p_i f_i(\mathbf{x}) + p_j f_j(\mathbf{x})\right)^2, \tag{3.8}$$

and

$$D\Big(\widetilde{p}_{ij} f_i + \widetilde{p}_{ji} f_j, \sum_{k \neq i,j} \widetilde{p}_k^{ij} f_k\Big) = 1 - \frac{1}{(p_i + p_j) \sum_{r \neq i,j} p_r} \int g_{ij}(\mathbf{x}) \, \mathrm{d}\mathbf{x}, \tag{3.9}$$

where $\widetilde{p}_{ij}$ and $\widetilde{p}_{ji}$ are as before, $\widetilde{p}_k^{ij} = p_k \big/ \sum_{r \neq i,j} p_r$, and

$$g_{ij}(\mathbf{x}) := \left(p_i f_i(\mathbf{x}) + p_j f_j(\mathbf{x})\right) \sum_{k \neq i,j} p_k f_k(\mathbf{x}) \big/ f^{(m)}(\mathbf{x}). \tag{3.10}$$

Theorem 3.2.1 implies that the HP divergence (3.7) increases when the support set of samples with labels i and j are nearly disjoint from the support sets of the the other labels $k \neq i, j$ $k = 1, \ldots, m$. In this case the HP-integral becomes closer to the GHP-integral. Specifically, (3.7) approaches (3.6) as the intersection between support sets $\mathbb{S}^{(ij)}$ and $\bigcup_{k \neq i,j} \mathbb{S}^{(k)}$ decreases, i.e. the conditional distributions become less overlapping.

3.3 Bounds on the Bayes error rate

Before introducing the new bound on multi-class BER, we first review the pairwise bounds on the multi-class Bayes error rate given by Berisha et. al. [230] and by Lin [135].

3.3.1 Pairwise HP bound

For the case of m classes the authors in [230] have shown that the multi-class BER ϵ^m in (3.1) can be bounded by

$$\frac{2}{m}\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}(p_i+p_j)\epsilon_{ij} \leq \epsilon^m \leq \sum_{i=1}^{m-1}\sum_{j=i+1}^{m}(p_i+p_j)\epsilon_{ij}, \tag{3.11}$$

where ϵ_{ij} represents the pairwise Bayes risk of the two class sub-problem of classifying between classes i and j:

$$\epsilon_{ij} = \int \min\left\{\widetilde{p}_{ij}f_i(\mathbf{x}), \widetilde{p}_{ji}f_j(\mathbf{x})\right\}\,\mathrm{d}\mathbf{x}. \tag{3.12}$$

In [26], it has been shown that

$$\frac{1}{2} - \frac{1}{2}\sqrt{u_{\widetilde{p}_{ij}}(f_i, f_j)} \leq \epsilon_{ij} \leq \frac{1}{2} - \frac{1}{2}u_{\widetilde{p}_{ij}}(f_i, f_j), \tag{3.13}$$

where

$$u_{\widetilde{p}_{ij}}(f_i, f_j) = 4\widetilde{p}_{ij}\widetilde{p}_{ji}\; D(f_i, f_j) + (\widetilde{p}_{ij} - \widetilde{p}_{ji})^2, \tag{3.14}$$

and $D(f_i, f_j)$ is defined in (3.3). Using both (3.11) and (3.13), we obtain bounds for the multi-class Bayes error rate. While these bounds have been successfully applied [81], [26], it has the disadvantage of high computational complexity due to the presence of $\binom{m}{2}$ summands in (3.11).

3.3.2 JS bound

The generalized Jensen-Shannon divergence is defined as

$$JS(f_1, f_2, \ldots, f_m) = \bar{H}\left(\sum_{k=1}^{m} p_k f_k\right) - \sum_{k=1}^{m} p_i \bar{H}(f_i),$$

where $\bar{H}$ is the Shannon entropy function

$$\bar{H}(f) = -\int f(\mathbf{x}) \log f(\mathbf{x}) \, \mathrm{d}\mathbf{x}.$$

In [135] this divergence measure was used to obtain a bound on the multi-class Bayes error rate. The Bayes error rate ϵ^m is upper bounded by

$$\epsilon^m \leq \frac{1}{2}\big(H(p) - JS(f_1, f_2, \ldots, f_m)\big), \tag{3.15}$$

and is lower bounded by

$$\epsilon^m \geq \frac{1}{4(m-1)}\big(H(p) - JS(f_1, f_2, \ldots, f_m)\big)^2. \tag{3.16}$$

Here $H(p) = -\sum_{k=1}^{m} p_k \log p_k$ is Shannon entropy and JS is generalized Jensen-Shannon divergence.

The bounds in (3.15) and (3.16) can be approximated by plug-in estimation or by direct methods, such as the NNR method [171] or other graph methods [101]. We will show that the JS bound suffers from lack of tightness.

3.3.3 Proposed multi-class Bayes error probability bound

To simplify notation, denote

$$\delta_{ij} := \int \frac{p_i p_j f_i(\mathbf{x}) f_j(\mathbf{x})}{p_i f_i(\mathbf{x}) + p_j f_j(\mathbf{x})} \mathrm{d}\mathbf{x}, \quad \delta_{ij}^m := \int \frac{p_i p_j f_i(\mathbf{x}) f_j(\mathbf{x})}{f^{(m)}(\mathbf{x})} \mathrm{d}\mathbf{x},$$

and note that $\delta_{ij} = \frac{(p_i+p_j)}{4}\big(1 - u_{\widetilde{p}_{ij}}(f_i, f_j)\big)$, where $u_{\widetilde{p}_{ij}}$ is defined in (3.14) and $\delta_{ij} \geq \delta_{ij}^m$.

Theorem 3.3.1. *For given priors $p_1, p_2, \ldots, p_m$ and conditional distributions $f_1, f_2, \ldots, f_m$,*

the multi-class BER ϵ^m satisfies

$$\epsilon^m \leq 2 \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \delta_{ij}^m. \tag{3.17}$$

And is lower bounded by δ_{ij}^m as

$$\epsilon^m \geq \frac{m-1}{m} \left[1 - \left(1 - 2 \frac{m}{m-1} \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \delta_{ij}^m \right)^{1/2} \right]. \tag{3.18}$$

In the following theorem we show that the proposed upper and lower bounds are tighter than the JS upper (3.15) and lower (3.16) bounds.

Theorem 3.3.2. *For given priors $p_1, p_2, \ldots, p_m$ and conditional distributions $f_1, f_2, \ldots, f_m$, for $m \geq 3$*

$$\epsilon^m \leq 2 \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \delta_{ij}^m \leq \frac{1}{2} \big(H(p) - JS(f_1, f_2, \ldots, f_m) \big). \tag{3.19}$$

And

$$\begin{aligned} \epsilon^m &\geq \frac{m-1}{m} \left[1 - \left(1 - 2 \frac{m}{m-1} \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \delta_{ij}^m \right)^{1/2} \right] \\ &\geq \frac{1}{4(m-1)} \big(H(p) - JS(f_1, f_2, \ldots, f_m) \big)^2. \end{aligned} \tag{3.20}$$

Theorem 3.3.3 shows that proposed upper and lower bounds are tighter than bounds in (3.11), i.e., the pairwise (PW) bounds.

Theorem 3.3.3. *For given priors $p_1, p_2, \ldots, p_m$ and conditional distributions $f_1, f_2, \ldots, f_m$, the multi-class classification BER ϵ^m is upper bounded*

$$\epsilon^m \leq 2 \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \delta_{ij}^m \leq 2 \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \delta_{ij}. \tag{3.21}$$

and is lower bounded by δ_{ij}^m as

$$\epsilon^m \geq \frac{m-1}{m}\left[1-\left(1-2\frac{m}{m-1}\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}\delta_{ij}^m\right)^{1/2}\right] \tag{3.22}$$
$$\geq \frac{2}{m}\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}(p_i+p_j)\left[\frac{1}{2}-\frac{1}{2}\sqrt{u_{\widetilde{p}_{ij}}(f_i,f_j)}\right].$$

where $u_{\widetilde{p}_{ij}}$ is given in (3.14).

The full proofs of Theorems 3.3.1, 3.3.2, and 3.3.3 are given in Appendix B. To derive the inequalities (3.17)-(3.22) a set of general inequalities for posterior probabilities $a_i := p(i|\mathbf{x})$ are established.

The proof of the tightness results (3.19)-(3.22) for JS and pairwise upper and lower bounds requires a different approach that involves deriving upper and lower bounds on the summed posterior probabilities,

$$\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}p(i|\mathbf{x})p(j|\mathbf{x}).$$

This approach provides tighter upper and lower bound that those given in [135] and [230].

3.4 Learning the bounds from data

Here we review the pairwise Friedman-Rafsky (FR) statistic and introduce a generalized FR statistic. Given a labeled sample $\mathbf{X} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ define the subset of samples having label k as: $\mathbf{X}^{(k)} = \{(\mathbf{x}_i, y_i)\}_{i=1,y_i=k}^n$, $k = 1, \ldots, m$. The cardinality of the subset $\mathbf{X}^{(k)}$ is $n_k = \sum_{i=1}^n I(y_i = k)$ where $I(B)$ denotes the indicator function of event B. We denote the pairwise FR statistic by $\mathfrak{R}_{n_i,n_j}$ and the generalized FR statistic by $\mathfrak{R}_{n_1,n_j}^{(ij)}$ that are computed as follows:

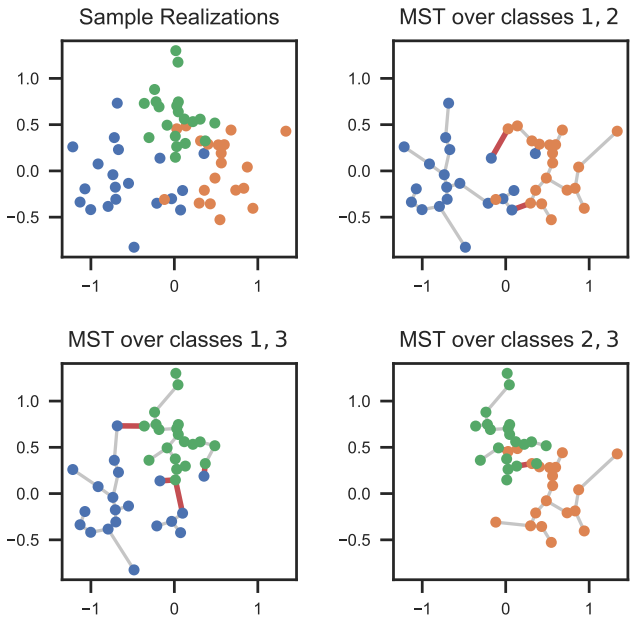


Figure 3.1: Estimating $\mathfrak{R}_{n_i,n_j}$ for three classes. A set of $m(m-1)/2$ Euclidean MSTs are computed for each unordered pair of classes $\{(i,j)\}_{i>j}$, and then the dichotomous edges (in red) are counted to find $\mathfrak{R}_{n_i,n_j}$.

1. $\mathfrak{R}_{n_i n_j} := \mathfrak{R}_{n_i n_j}(\mathbf{X})$ is the number of dichotomous edges in a Euclidean minimal spanning tree (MST) spanning the samples in the pairwise union of samples with labels i and j , $\mathbf{X}^{(i)} \cup \mathbf{X}^{(j)} \in \mathbb{S}^{(i)} \cup \mathbb{S}^{(j)}$, where $\mathbf{X}^{(k)} = \{(\mathbf{x}_i, y_i)\}_{i=1,Y_i=k}^{n}$. A dichotomous edge is an edge that connects a sample from class i to a sample from class j. The pairwise FR statistic $\mathfrak{R}_{n_i n_j}$ for three classes is illustrated in Figure 3.1.

2. $\mathfrak{R}_{n_i,n_j}^{(ij)} := \mathfrak{R}_{n_i,n_j}^{(ij)}(\mathbf{X})$ is the number of dichotomous edges connecting a sample from class i to a sample from class j in the global MST constructed on all samples with all classes $1, 2 \ldots, m$ i.e. $\bigcup_{k=1}^{m} \{(\mathbf{x}_i, y_i)\}_{i=1,y_i=k}^{n}$ or $\mathbf{X}^{(1)} \cup \mathbf{X}^{(2)} \cup \cdots \cup \mathbf{X}^{(m)} \in \mathbb{S}^{(1)} \cup \mathbb{S}^{(2)} \cup \ldots \mathbb{S}^{(m)}$ where $\mathbf{X}^{(k)} = \{(\mathbf{x}_i, y_i)\}_{i=1,y_i=k}^{n}$, $k = 1, \ldots, m$. Figure 3.2 represents the generalized FR statistic for three classes.

Using the theory in [99] and [26], the estimator $\mathfrak{R}_{n_i,n_j}$ is a statistically consistent estimator of the pairwise HP-bound for classifying class i vs. class j. This yields an estimate of the

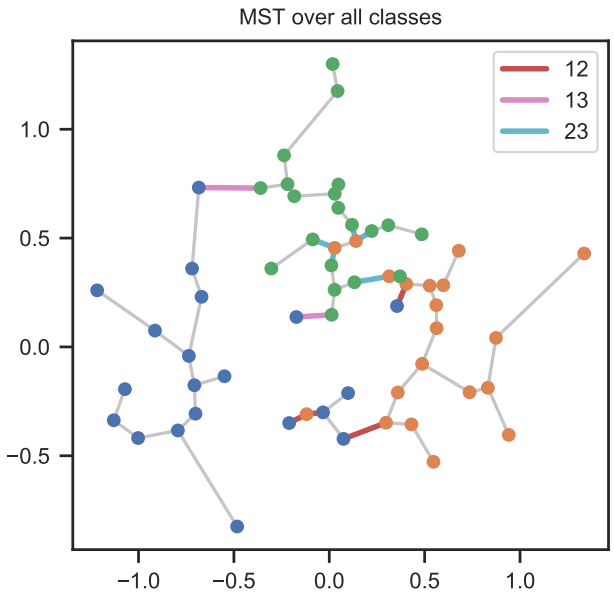


Figure 3.2: Estimating $\mathfrak{R}^{(ij)}_{n_i,n_j}$ for three classes. A single MST is constructed over all classes $i = 1, 2, 3$ of points. For each (i, j), count the edges connecting points from classes i and j. These edges are shown in 3 different colors each corresponding to the three types of pairs $(i, j) = (1, 2), (2, 3), (1, 3)$.

bounds (3.11) on multi-class BER, requiring the construction of $\binom{m}{2}$ MSTs spanning all distinct pairs of label classes. The next theorem implies that $\mathfrak{R}^{(ij)}_{n_i,n_j}$ can be used to estimate the tighter bound on multi-class BER given in Theorem 3.3.1 using only a single global MST.

Theorem 3.4.1. *Let* $\mathbf{X} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ *be an i.i.d.* m*-class labeled sample and* $n_k = \sum_{i=1}^n I(y_i = k)$*, be the cardinality of samples with label* k*. For distinct classes* i, j *let* $i, j = 1, \ldots, m$*,* $n_i, n_j \to \infty$*,* $n \to \infty$ *such that* $n_i/n \to p_i$ *and* $n_j/n \to p_j$*. Then*

$$\frac{\mathfrak{R}^{(ij)}_{n_i,n_j}(\mathbf{X})}{2n} \longrightarrow \delta^m_{ij} \quad \textit{(a.s.)} \tag{3.23}$$

The proof for Theorem 3.4.1 uses arguments similar to those used in [99] to establish the convergence of the Friedman Rafsky two sample statistic, but generalized to more than two labeled populations. Furthermore, using arguments similar to those in [237], it is

also possible to establish convergence of the cross-match statistic by running the optimal weighted matching graph over pairs of label classes. Details on the proof are given in the Appendix B.

3.5 Simulation study

Here we illustrate the proposed method for learning bounds on Bayes error rate (BER). Section 3.5.1 focuses on numerical comparison of the upper bounds in (3.11), (3.15), (3.17) and lower bounds in (3.11), (3.16), (3.18). Section 3.5.2 focuses on the empirical estimation of these bounds, including a comparison of runtime.

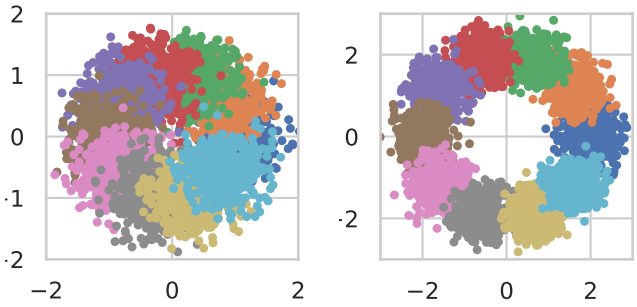


Figure 3.3: Example of generated data for experiments. The data on the left has 10 classes whose means are arranged around a circle with mean parameter $\mu = 1$. The data on the right has 10 classes, with mean parameter $\mu = 2$. In both cases, $\sigma^2 = 0.1$, and both plots show a sample of 5000 data points.

For each of the following simulations, data is generated in the following way: given m classes with priors $p_1, p_2, \ldots, p_m$, the class conditional distributions are mean shifted bivariate normal: $f_i \sim \mathcal{N}(\mu_i, \sigma^2 I)$. The means μ_i are arranged uniformly around the circumference of a circle of radius μ:

$$\mu_i = \left[\mu \cos\left(2\pi \frac{i}{m}\right), \mu \sin\left(2\pi \frac{i}{m}\right)\right].$$

Figure 3.3 shows two examples for 5000 points and 10 classes and $\sigma^2 = 0.1$, with the left plot having mean parameter $\mu = 0.7$, and the right plot setting $\mu = 2$. Unless stated

otherwise, the feature dimension is $d = 2$.

3.5.1 Comparison of bounds

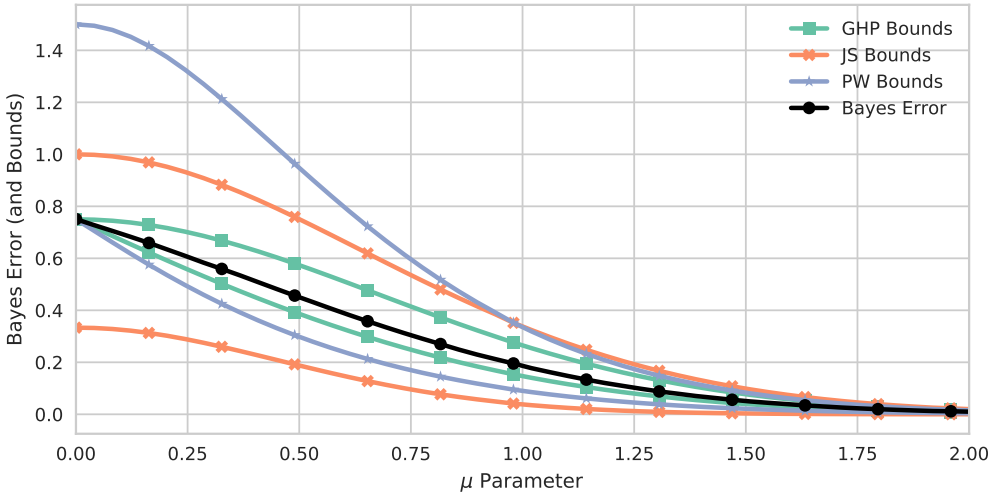
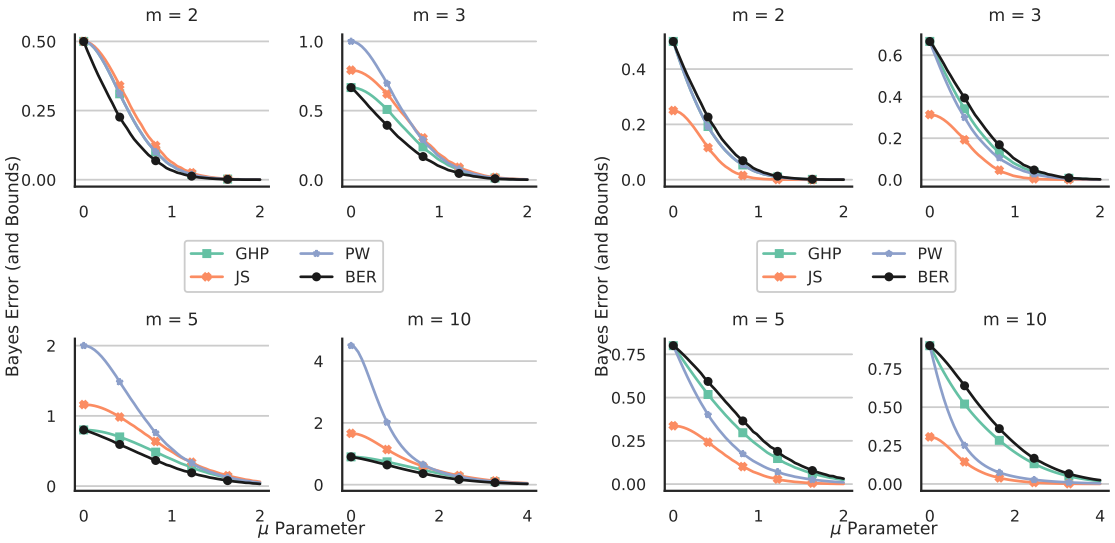


Figure 3.4: Bounds on the Bayes error for $m = 4$ and uniform priors. We note that even for a relatively small number of classes, the proposed new GHP upper and lower bound are much tighter than the competitors. For this experiment, $\sigma^2 = 0.3$.

This section uses the synthetic data described in Section 3.5.1 and Monte Carlo approximation to estimate the bounds. We first explore how the difficulty of the classification problem affects the bounds. Figure 3.4 shows upper and lower bounds of the Bayes error rate for each type of bound as a function of the mean parameter μ. Here, the number of classes m is 4. Note that when μ is smaller and the classes are poorly separated (creating a harder classification problem), both the Jensen-Shannon (JS) and pairwise (PW) upper bounds perform poorly and become trivial, exceeding one. However, for relatively small m, the pairwise lower bound remains fairly tight. The proposed GHP bounds are uniformly better than either the JS or PW bounds, as predicted by the theory. Further, note that the proposed bound is tight around the actual Bayes error rate (BER). Finally, as μ grows and the classes become well separated, the JS and PW bounds become tighter to the Bayes error. In light of Theorem 1, this makes sense for the pairwise bounds, as well separated classes



(a) Upper bounds.

(b) Lower bounds.

Figure 3.5: Comparison of upper and lower bounds of BER for different number of classes for the distribution illustrated in Figure 3.3. Shown is the exact Bayes misclassification error rate (BER) and three upper bounds including the Jensen-Shannon bound (JS), pairwise HP bound (PW), and the proposed generalized HP bound (GHP). As m increases, the two bounds, JS and PW, are not tight to the BER, unlike the proposed GHP, as m increases.

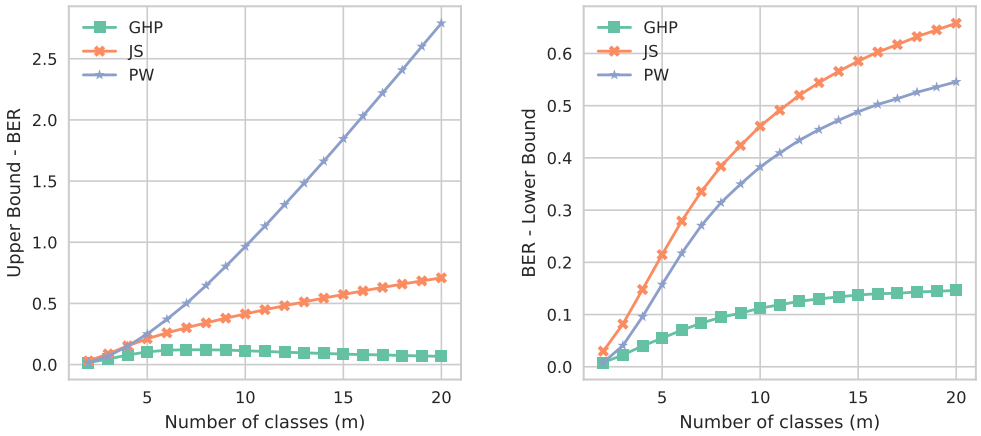
cause the pairwise Henze-Penrose divergence and the GHP integral to become equivalent.

Figure 3.5a and Figure 3.5b show the behavior of the bounds as a function of m. For the upper bounds, we note that the JS and PW bounds become much looser as m increases. On the other hand, the proposed GHP upper bound remains tight.

In Figure 3.5b, we note that the PW lower bound does perform better than the JS lower bound, but the proposed GHP lower bound uniformly outperforms both. Note that, as mentioned previously, all bounds become tighter as the classification problem becomes easier (i.e., the classes become well separated).

The difference between the bounds and the BER, called the tightness of the bound as a function of m is shown in Figure 3.6a and Figure 3.6b for upper bounds and lower bounds, respectively. Figure 3.6a highlights our proposed GHP bound's ability to stay close to the BER, even as the class size continues to increase. In comparison, both the JS and pairwise upper bounds continue to drift farther away from the Bayes error. Figure 3.6b shows a sim-

ilar effectiveness in the proposed lower bound, although both the JS and pairwise bounds have better behavior than in Figure 3.6a, due to the lower bounds being guaranteed to be greater than or equal to 0.



(a) Upper bounds.

(b) Lower bounds.

Figure 3.6: Tightness of upper and lower bounds vs. m, where tightness is quantified as the absolute value of the difference between the bound and the true BER. This experiment was performed for $\mu = 1$. The pairwise upper bound quickly becomes useless as m increases. The JS bound performs slightly better, but only our proposed GHP upper bound stays close to the Bayes error. In the proposed GHP lower bound, there is a slight decrease in tightness as m increases. However, it is much smaller in comparison with the pairwise and JS lower bounds.

3.5.2 Statistical consistency and runtime

This section illustrates the improvement in both statistical accuracy and runtime performance of the proposed generalized HP (GHP) approach as compared to the JS and pairwise HP (PW) methods of [135] and [230].

Figure 3.7 Shows the MSE between the estimated and true upper bound as a function of the number of samples n, for different feature dimensions d. The behavior of the lower bound convergence has analagous behavior and is not shown. Note that as d increases, the MSE grows, illustrating the well known curse of dimensionality for high dimensional datasets.

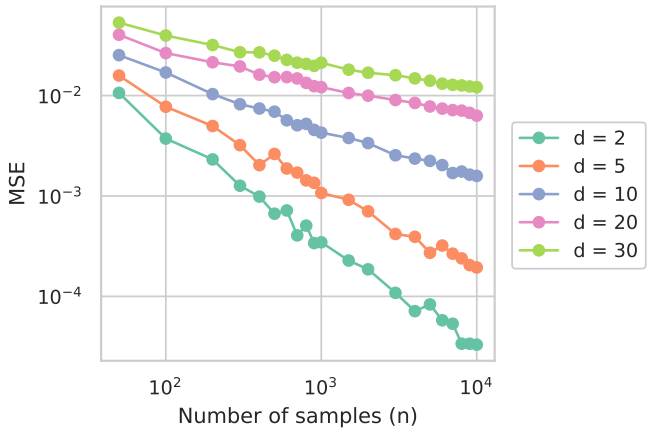
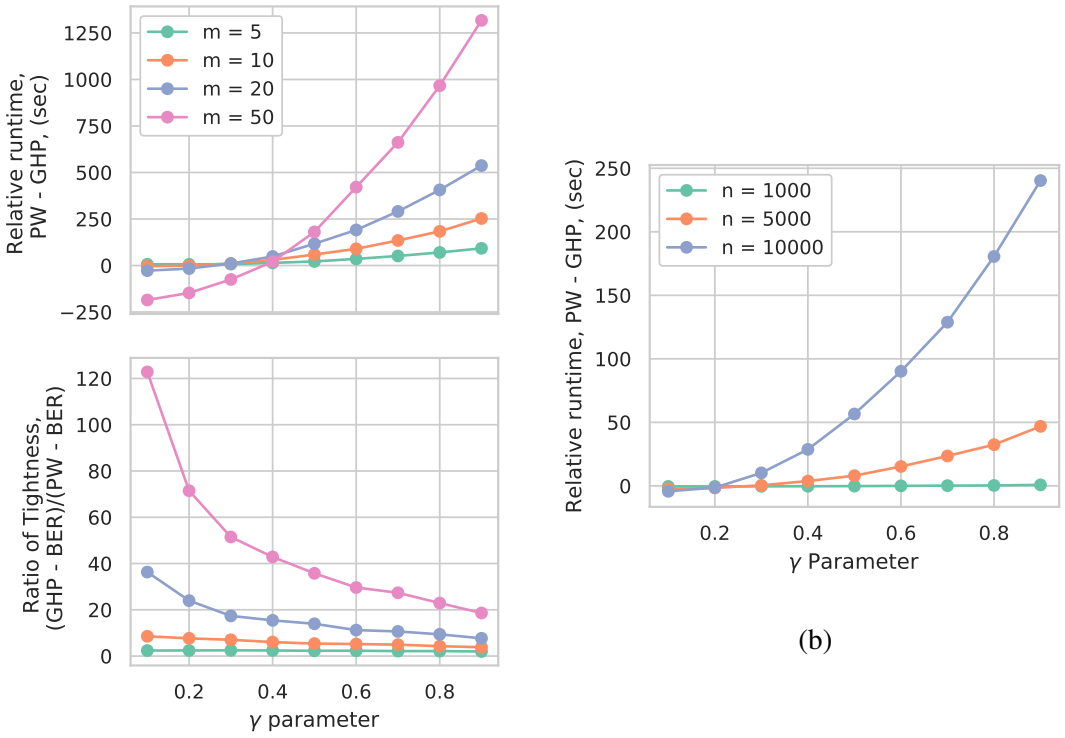


Figure 3.7: Convergence in MSE of MST estimate of the proposed GHP upper bound to the true upper bound on BER. The simulation parameters were as in Figure 3.3 for $\mu = 0.7$ and $\sigma^2 = 0.1$, and the results were averaged over 100 trials. For $d > 2$, $d - 2$ dimensions had zero mean with Gaussian noise having variance 0.1.

Figures 3.8a and 3.8b show the relative runtime of the proposed method in comparison with the pairwise HP method. For each of these figures, we introduce a parameter γ, which controls the prior class imbalance in the problem. For a particular γ and number of classes m, we create priors $p_1 = \gamma, p_2 = p_3 = \ldots = p_m = (1-\gamma)/(m-1)$. For $\gamma = 1/m$, all class probabilities are the same; there is no imbalance. A larger class imbalance will cause the pairwise estimation procedure to perform many large MST calculations, which significantly increases the runtime of the algorithm.

Figure 3.8a shows the relative runtime (PW method minus GHP method) as a function of γ, for different m, along with the ratio of tightness of GHP compared with PW for the upper bound of the BER. Here, we set $n = 10000, \mu = 1, \sigma^2 = 0.3$. Observe that for large number of classes and small class imbalance γ, the pairwise method is slightly faster but, in this regime PW yields a useless bound that is overly loose - the proposed GHP bound is over 120 times tighter than the pairwise bound in this case. As γ grows, we see significant relative speedup of the proposed GHP method relative to the other methods. From Figure 11 it is evident that, while the PW bound has faster computation time in the regime of small γ (graph in bottom panel), it is very loose in this small γ regime (graph in top panel).



(a)

(b)

Figure 3.8: (a) Relative runtime of pairwise and proposed GHP algorithm vs. class imbalance parameter γ, and ratio of tightness of GHP compared with PW, where tightness is defined by the upper bound minus the BER. For large class imbalance (large γ), and large m, the proposed GHP method achieves significant speedup, while for small class imbalance, the PW bound becomes overly loose. (b) Relative runtime of pairwise and proposed GHP algorithm vs. class imbalance parameter γ. For large γ, and large sample size n, the proposed method achieves significant speedup.

Figure 3.8b shows the relative runtime as a function of γ, for different sample sizes n, with $m = 10, \mu = 1$, and $\sigma^2 = 0.3$. Similarly to Figure 3.8a, the greatest speedup occurs when n and γ are large.

3.6 Real data experiments

We utilize our proposed bounds to explore feature generation for the MNIST dataset. The MNIST dataset consists of grey-scale thumbnails, 28 x 28 pixels, of hand-written digits 0 - 9. It consists of a training set of 60,000 examples, and a test set of 10,000 examples. The

digits have been size-normalized and centered in a fixed-size image. MNIST has been well studied in the literature, and is known to have a low error-rate. To illustrate the utility of the proposed BER bound learning approach, we estimate the Bayes error rate bounds as a function of feature dimension. Specifically, we focus on PCA and a simple autoencoder. The validity of the proposed lower bound is demonstrated by comparison to the accuracy of three types of classifiers: the K-NN classifier, linear SVM, and a random forest classifier.

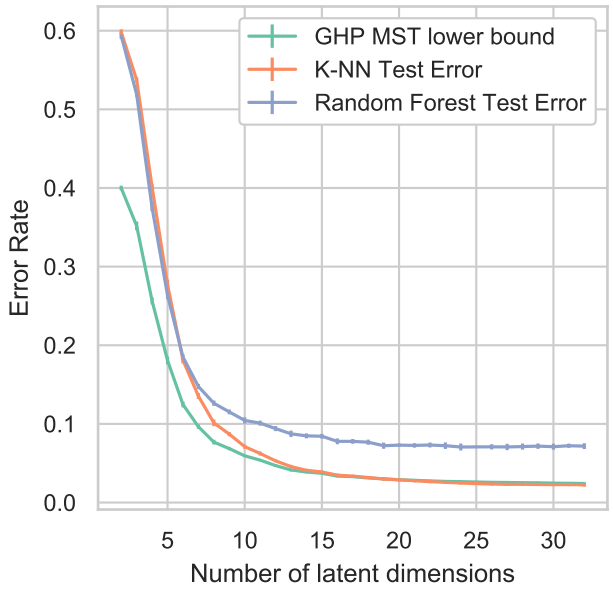


Figure 3.9: Number of latent dimension for PCA vs. error rates and estimated lower bound. The k-NN classifier test error (orange curve) approaching the proposed lower bound (green curve) as the number of latent dimensions increases beyond 15, establishing that the k-NN comes close to achieving optimal performance.

The PCA results are shown in Fig. 3.9. Plotted are the estimated lower bound for the BER and the test error rates of the 3-NN and Random Forest classifier versus the number of principal components used. As expected, the test errors of both classifiers are greater than the lower bound for Bayes error. Further, it is evident that no more than 20 latent dimensions are needed in order to minimize the lower bound, which is confirmed by the behavior of the test errors of the classifiers implemented.

Table 3.2 shows Bayes error bounds and error rates for the MNIST feature sets. Autoencoder-

Table 3.2: Bounds on the BER and classifier test error rates for different feature sets.

Bayes Error Bounds and Error Rates for MNIST Feature Sets					
Features	lower bound	upper bound	Linear SVM	K-NN, K=3	Rand. For.
PCA-4	0.247	0.427	0.449	0.392	0.370
PCA-8	0.070	0.135	0.241	0.107	0.129
PCA-16	0.029	0.058	0.166	0.037	0.077
PCA-32	0.020	0.040	0.113	0.026	0.073
Autoencoder-4	0.290	0.486	0.662	0.442	0.412
Autoencoder-8	0.097	0.184	0.317	0.144	0.155
Autoencoder-16	0.041	0.082	0.213	0.058	0.099
Autoencoder-32	0.026	0.052	0.144	0.032	0.086

X or PCA-X are feature sets that use X latent dimensions or X principal components, respectively. The autoencoder is a 1-layer autoencoder, and trained using Adagrad. Interestingly, we see that PCA-32 feature set outperforms Autoencoder-32. More layers or a convolutional model could assist the autoencoder in achieving lower bounds and test error rates.

3.7 Conclusion

In this chapter, a new bound on the Bayes error rate of multiclass classification was introduced. It was established by theory and simulation that the proposed bound is tighter than both the pairwise Henze-Penrose bound and the generalized Jenson-Shannon bound. Furthermore, a fast and efficient empirical estimator was presented that allows one to learn the bound from training data without the need for density estimation. The estimation method is based on the global minimal spanning tree that spans all labeled features over feature space, allowing for a more computationally tractable approach than the standard practice of summing estimates of pairwise BERs. The proposed bound learning method was illustrated on the MNIST dataset.

CHAPTER 4

Mutual Information Estimation using Dimension-Independent Graph Based Methods

Henze-Penrose divergence is a non-parametric divergence measure that can be used to estimate a bound on the Bayes error in a binary classification problem. In this chapter, we show that a cross-match statistic based on optimal weighted matching can be used to directly estimate Henze-Penrose divergence. Unlike an earlier approach based on the Friedman-Rafsky minimal spanning tree statistic, the proposed method is dimension-independent. The new approach is evaluated using simulation and applied to real datasets to obtain Bayes error estimates.

Symbol	Description
$(\mathbf{x}, y)$	Observed feature-label pair
$(\mathbf{X}, Y)$	Random feature-label pair
p_0, p_1	Prior label probabilities
f_0, f_1	Class conditional feature pdfs
ϵ	Bayes error rate
$G = (V, E, D)$	Weighted graph
V	Vertex set
E	Edge set
D	Edge weights

Table 4.1: Glossary of commonly used symbols.

4.1 Introduction

Many information theoretic measures have been applied to measure the discrimination between probability density functions. They have been used in various applications in signal processing, classification, image registration, clustering and structure learning, see [94], [96], [158], [226]. A special class of divergence measures, called f-divergences have the property that the divergence functional f is convex and $f(1) = 0$. Among the different divergence functions belonging to the f-divergence family, [10], [63] the Henze-Penrose (HP) divergence has been of great interest due to its application to binary classification, in particular to bound the Bayes error rate.

Let $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N \in \mathbb{R}^d$ be realizations of random vector $\mathbf{X}$ and class labels $y \in \{0, 1\}$ which are reaalizations of the random binary variable Y, with prior probabilities $p_0 = P(Y = 0)$ and $p_1 = P(Y = 1)$, and such that $p_0 + p_1 = 1$. Given conditional distributions $f_0(\mathbf{x})$ and $f_1(\mathbf{x})$, the Bayes error rate is given by

$$\epsilon = \int_{\mathbb{R}^d} \min \left\{ p_0 f_0(\mathbf{x}), p_1 f_1(\mathbf{x}) \right\} \mathrm{d}\mathbf{x}. \tag{4.1}$$

The Bayes error rate is the expected risk for the Bayes classifier, which assigns a given feature vector $\mathbf{x}$ to the class with the highest posterior probability, and is the lowest possible error rate of any classifier for a particular joint distribution. It is thus a reasonable measure for assessing the intrinsic difficulty of a particular classification problem. By estimating and bounding this value, we can then have a better understanding of the problem difficulty, which allows the user to make more informed decisions.

We define the HP-divergence between f_0 and f_1, $D_c(f_0, f_1)$ by

$$\frac{1}{4c_0c_1} \left[\int_{\mathbb{R}^d} \frac{\left(c_0 f_0(\mathbf{x}) - c_1 f_1(\mathbf{x}) \right)^2}{c_0 f_0(\mathbf{x}) + c_1 f_1(\mathbf{x})} \, \mathrm{d}\mathbf{x} - (c_0 - c_1)^2 \right], \tag{4.2}$$

where $c_0 \in (0, 1)$ and $c_1 = 1 - c_0$. Note that for all c_0 and c_1, $0 \leq D_c(f_0, f_1) \leq 1$ and

when $f_0 = f_1$ the HP-divergence becomes zero.

The authors of [26] showed that HP-divergence yields tighter bounds on the Bayes error rate ϵ, given in (4.1), than those based on the Bhattacharya distance, [30]. In particular, the following bound on the Bayes error rate holds:

$$\frac{1}{2} - \frac{1}{2}\sqrt{u_p(f_0, f_1)} \leq \epsilon \leq \frac{1}{2} - \frac{1}{2}u_p(f_0, f_1), \tag{4.3}$$

where $u_p(f_0, f_1) = 4p_0p_1D_p(f_0, f_1) + (p_0 - p_1)^2$.

In this chapter we propose a new direct estimator for HP-divergence using a statistic based on optimal weighted matching [201]. Matching for general graphs is a combinatorial optimization problem that can be solved in polynomial time. In [201], the optimal weighted matching was used to find a statistical test for equal posterior distributions using the cross match statistic. We demonstrate that the same statistic described in that series of papers can be utilized to estimate HP-divergence. We emphasize that the proposed weighted matching estimator is completely different from weighted K-NN graph estimators.

The rest of the chapter is organized as follows. Section 4.2 briefly describes related work on HP-divergence and optimal weighted matching. Section 4.3 defines the cross-match statistic, and in Section 4.4 we prove that the cross-match statistic approximately tends to the HP-divergence when samples sizes of two classes increases simultaneously in a specific regime. Section 4.5 shows sets of simulations for our proposed method and compares the Friedman-Rafsky (FR) and cross-match estimators experimentally, and we estimate the Bayes error rate on a few real datasets. Finally, Section 4.6 concludes the chapter.

4.2 Related work

Several estimators for HP-divergence have been proposed in the literature: Plug-in estimates were introduced in [205] and later have been studied more in [155], [157], [159].

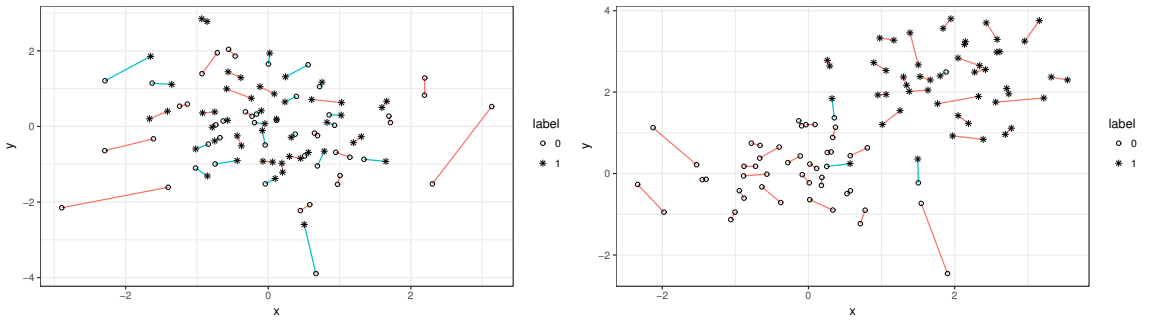


Figure 4.1: An example of the cross-match statistics for two cases $f_0 = f_1$ (left-generated from standard Gaussian distributions) and $f_0 \neq f_1$ (right-Generated from Gaussian distributions with means $[0, 0]$, $[2, 2]$). The total number of blue edges is the cross match statistics.

Plug-in approaches estimate the underlying distribution function and then plug this value into the divergence function. The drawback with the plug-in estimates is that these methods are not accurate near support boundaries and are also more computationally complex. There have been a number of attempts to non-parametrically approximate divergence measures using graph-based algorithms such as minimal spanning tree (MST), [5], [240] and k-nearest neighbors graphs (k-NNG), [23].

One of the most common direct estimators is based on Friedman-Rafsky (FR) multivariate test statistic [81]. This approach is constructed from the MST on the concatenated data set drawn from sufficiently smooth probability densities. Henze and Penrose [99] showed that the FR test is consistent against all alternatives. Therefore, the HP-divergence has the appealing property that there exists an asymptotically consistent direct estimator in terms of the FR test statistic, see [25], [26], [99]. The variance of the FR test statistic under the assumption of equal distributions depends on the dimension of the data d, which may be unknown, especially when the support of the densities is a common but unknown lower dimensional manifold.

Optimal weighted matching is a well studied combinatorial optimization problem [183]. It has been used extensively in operations engineering. Previous statistical work using weighted matching have derived useful applications of the cross-match test statistic in fields

like biological networks [141], [201].

4.3 The cross-match test statistic

Consider N i.i.d. samples $\mathcal{X}_N = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\},\ \mathbf{x}_i \in \mathbb{R}^d$ and corresponding labels $y_i \in \{0, 1\}$. Define $\mathbf{y} = (y_1, \ldots, y_N)$, and further $m = \sum_{l=1}^{N} y_l$, and $n = N - m$, so that m is the number of samples in $\mathbf{x}$ with class 1, and n is the number with class label 0. Further, we create D, a $N \times N$ Euclidean distance matrix, with $D_{ij} = ||\mathbf{x}_i - \mathbf{x}_j||$. Without loss of generality, we assume N is even, as we can always add a 'ghost point' $\mathbf{x}_{N+1}$, where $D_{iN+1} = 0, \forall i$. In the following, we consider a complete weighted graph $G = (V, E, D)$, with the vertices $V = 1, \ldots, N$ representing the sample points $\mathbf{x}_1, \ldots, \mathbf{x}_N$, edges $E = \{\{i, j\},\ i, j \in V\}$, and weights for each edge $\{i, j\}$ as D_{ij}.

A complete matching $M \subset E$ on a weighted graph is a set of edges such that no two edges in M share a common vertex, and every vertex is used in the matching. The complete minimum weighted matching M^* is defined as the matching on G such that $M^* = \arg\min_M \sum_{i,j \in M} D_{ij}$. We note that this is similar to the FR test [81], which uses the same matrix D to find the minimal spanning tree. The FR test statistic is the total number of edges in the D-based MST connecting different labeled nodes.

Using this matching, we find the *cross-match statistic*, $\mathcal{A}(\mathcal{X}_N)$ which is the number of edges that match dichotomous samples, i.e. samples with different class labels, that is

$$\mathcal{A}(\mathcal{X}_N) = \sum_{\{i,j\} \in M^*} \Big(y_i(1 - y_j) + (1 - y_i) y_j \Big). \tag{4.4}$$

In Figure 4.1 we show two numerical examples. The left plot shows samples from two equal distributions, and the right plot shows samples from differing distributions. Qualitatively, we see that $\mathcal{A}$ is much greater for the equivalent distributions than for the differing distributions because the optimal matching tries to reduce long distances. This in turn will reduce the number of edges between points with different labels.

In Proposition 1 in [201], under the assumption of equal distributions, the expectation and variance of $\mathcal{A}(\mathbf{x})$ are derived:

$$E[\mathcal{A}] = \frac{mn}{N-1}, \quad Var[\mathcal{A}] = \frac{2n(n-1)m(m-1)}{(N-3)(N-1)^2}. \tag{4.5}$$

We note that the mean and variance of the cross-match statistic under equal distributions are dimension-independent, but this is not true for the FR statistic, whose variance is dependent on the degrees of the MST. The maximal degrees of the MST is in fact dependent on the dimension d of the underlying samples, e.g., the MST has maximal degree 4 in $d = 2$ dimensions while its maximal degree is known to be between 13 or 14 in 3 dimensions [199]. This dependence causes the FR statistic to perform poorly in higher dimensions. In Section 4.5 we perform a set of experiments where dimension varies to demonstrate the advantage of the cross-match statistic over the FR statistic.

4.4 HP-divergence estimation

Here we introduce the cross-match statistic as an estimate of the HP-divergence given in (4.2). Assume that we have two sets of samples $\mathcal{X}_m = \{\mathbf{X}_1, \ldots, \mathbf{X}_m\}$ and $\mathcal{U}_n = \{\mathbf{U}_1, \ldots, \mathbf{U}_n\}$ with two different labels. In order to show asymptotic convergence to HP-divergence, we make the following assumption regarding the cross-match statistic (similar to Lemma 1 in [99]).

Assumption 1: For disjoint sets $\mathcal{X}_m, \mathcal{U}_n$ and $\{s, t\}$ we have

$$\Big|\mathcal{A}(\mathcal{X}_m \cup \{s,t\} \cup \mathcal{U}_n) - \mathcal{A}(\mathcal{X}_m \cup \mathcal{U}_n)\Big| \leq k_d. \tag{4.6}$$

where k_d is a constant that may depend on d. This means that even if the optimal matching changes a great deal, the number of edges that are between the two samples is still approximately the same.

We empirically check this assumption in Figure 4.2. We generate two sets of d-dimensional samples from standard Gaussian with mean $\mu_0 = [0]_d$, $\mu_1 = [1]_d$ and $\Sigma_0 = \Sigma_1 = I_d$ for $d = 2, 4, 6, 8$. We plot the difference in cross-match statistic when adding two points (labeled by $\mathcal{A}_{\text{diff}}$), and perform this test over varying sample size. We see that $\mathcal{A}$ does not vary significantly when adding a new sample in the tested cases.

Lemma 4.4.1. *Let $g\colon \mathcal{R}^d \times \mathcal{R}^d \to [0,1]$ be a symmetric and measurable function, such that for almost every $\mathbf{x} \in \mathcal{R}^d$, $g(\mathbf{x}, .)$ is measurable with $\mathbf{x}$ a Lebesgue point of the functions $f(.)g(\mathbf{x}, .)$ and $f(.)$. For each N, let $\mathbf{Z}_1^N, \mathbf{Z}_2^N, \ldots, \mathbf{Z}_N^N$ be independent d-dimensional variables with common density function f_N convergent to f as $N \to \infty$ and set $\mathcal{Z}_N = \{\mathbf{Z}_1^N, \ldots, \mathbf{Z}_N^N\}$. Consider the complete minimum weighted matching M^* on $\mathcal{Z}_N$. Then*

$$\lim_{N\to\infty} N^{-1} E \sum_{1\le i<j\le N} \sum g(\mathbf{Z}_i^N, \mathbf{Z}_j^N) \mathbf{1}\big\{(\mathbf{Z}_i^N, \mathbf{Z}_j^N) \in M^*(\mathcal{Z}_N)\big\} = \frac{1}{2} \int_{\mathcal{R}^d} g(\mathbf{x}, \mathbf{x})\ f(\mathbf{x}). \quad (4.7)$$

Proof: For given $\mathbf{x}$ in a subset $\mathcal{S} \in \mathcal{R}^d$, the degree of vertex $\mathbf{x}$ in $M^*(\mathcal{S})$ is one. Let $\mathbf{x}$ be a Lebesgue point of $f(.)$ and $f(.)g(\mathbf{x}, .)$ and $\mathcal{Z}_N^{\mathbf{x}}$ be the point process $\{\mathbf{x}, \mathbf{Z}_2^N, \mathbf{Z}_3^N, \ldots, \mathbf{Z}_N^N\}$. Let $\mathcal{B}(\mathbf{x}, r) = \big\{\mathbf{y} : \|\mathbf{y} - \mathbf{x}\| \le r\big\}$. Therefore, we can write

$$\begin{aligned} & E \sum_{j=2}^{N} \big|g(\mathbf{x}, \mathbf{Z}_j^N) - g(\mathbf{x}, \mathbf{x})\big| \mathbf{1}\big\{\mathbf{Z}_j^N \in \mathcal{B}(\mathbf{x}, N^{-1/d})\big\} \\ & = (N-1) \textstyle\int_{\mathcal{B}(\mathbf{x}, N^{-1/d})} \big|g(\mathbf{x}, \mathbf{y}) - g(\mathbf{x}, \mathbf{x})\big| f_N(\mathbf{y})\ \mathrm{d}\mathbf{y} \\ & = (N-1) \textstyle\int_{\mathcal{B}(\mathbf{x}, N^{-1/d})} \big|g(\mathbf{x}, \mathbf{y}) f_N(\mathbf{y}) - h(\mathbf{x}, \mathbf{x}) f_N(\mathbf{x}) \\ & \qquad\qquad + g(\mathbf{x}, \mathbf{x})(f_N(\mathbf{x}) - f_N(\mathbf{y})\big|\ \mathrm{d}\mathbf{y}, \end{aligned} \quad (4.8)$$

Since $\mathbf{x}$ is a Lebesgue point of f_N and $g(\mathbf{x}, .)f_N(.)$ then (4.7) tends to zero. Note that the degree of vertex in $M^*(\mathcal{Z}_N^{\mathbf{x}})$ is one. For almost all $\mathbf{x}$,

$$E \sum_{j=2}^{N} g(\mathbf{x}, \mathbf{Z}_j^N) \mathbf{1}\big\{(\mathbf{x}, \mathbf{Z}_j^N) \in M^*(\mathcal{Z}_N^x)\big\} = g(\mathbf{x}, \mathbf{x}) + o(1). \quad (4.9)$$

The function g has range $[0,1]$ so the left hand side of (4.9) is bounded by one. By the dominated convergence theorem

$$
\begin{aligned}
& N^{-1} E \sum \sum_{1 \le i < j \le N} g(\mathbf{Z}_i^N, \mathbf{Z}_j^N) \mathbf{1}\big\{(\mathbf{Z}_i^N, \mathbf{Z}_j^N) \in M^*(\mathbf{Z}_N)\big\} \\
& = \tfrac{1}{2} E \sum_{j=2}^{N} g(\mathbf{Z}_1^N, \mathbf{Z}_j^N) \mathbf{1}\big\{(\mathbf{Z}_1^N, \mathbf{Z}_j^N) \in M^*(\mathbf{Z}_N)\big\} \\
& = \tfrac{1}{2} \int_{\mathbf{x}} f_N(\mathbf{x}) E \sum_{j=2}^{N} g(\mathbf{x}, \mathbf{Z}_i^N) \mathbf{1}\big\{(\mathbf{x}, \mathbf{Z}_j^N) \in M^*(\mathbf{Z}_N)\big\}.
\end{aligned}
\tag{4.10}
$$

The last line in (4.8) tends to right hand side of (4.7). □

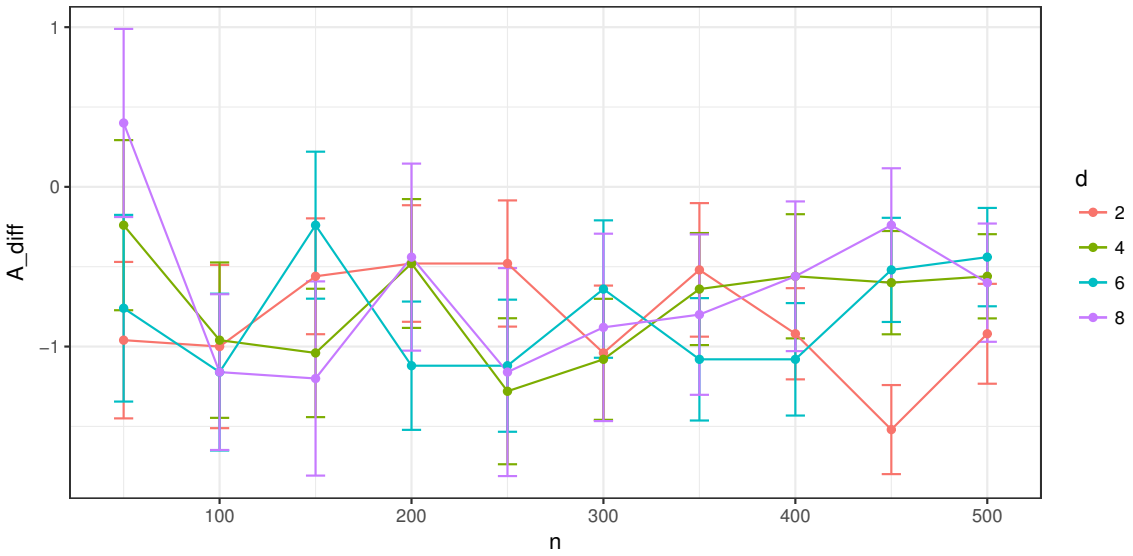


Figure 4.2: The cross-match statistics difference with error bars at the standard deviation from 50 trials for the Gaussian samples by adding two points.

The following theorem proves the direct estimate of HP-divergence based on $\mathcal{A}(\mathcal{X}_N)$. Due to space limitations only an outline of the proof is given.

Theorem 4.4.2. *As $m \to \infty$ and $n \to \infty$ such that $m/N \to p_1$ and $n/N \to p_0$, where $N = m + n$. Denote $\mathcal{A}_{m,n} := \mathcal{A}(\mathcal{X}_m \cup \mathcal{U}_n)$ the cross-match statistic given by the optimal*

weighted matching over $\mathcal{X}_m$ and $\mathcal{U}_n$. Then under Assumption 1 we have

$$1 - \left(\frac{N}{m\ n}\right) \mathcal{A}_{m,n} \to D_p(f_0, f_1), \quad a.s. \tag{4.11}$$

Proof: The proof shares some similarity with the FR convergence proof of the HP-divergence in [99]. The primary difference lies in handling the difference between the cross-match statistic when nodes are added, i.e. (4.6). We use Lemma 4.4.1 and Poissonization to prove (4.11).

Let M_m and N_n be Poisson variables with mean m and n such that $m+n$ is even and independent of one another and of $\mathbf{X}_i$ and $\mathbf{U}_j$. Let $\mathcal{X}'_m$ and $\mathcal{U}'_n$ be the Poisson processes $\{\mathbf{X}_1, \ldots, \mathbf{X}_{M_m}\}$ and $\{\mathbf{U}_1, \ldots, \mathbf{U}_{N_n}\}$, respectively. Set $\mathcal{A}'_{m.n} = \mathcal{A}(\mathcal{X}'_m \cup \mathcal{U}'_n)$, the cross-match statistic. By (4.6), we have

$$\left|\mathcal{A}'_{m,n} - \mathcal{A}_{m,n}\right| \leq k_d \left(|M_m - m| + |N_n - n|\right). \tag{4.12}$$

Note that $(m+n)^{-1}E\left|\mathcal{A}'_{m,n} - \mathcal{A}_{m,n}\right| \to 0$. Poissonization makes the identities of the points of $\mathcal{X}'_m \cup \mathcal{U}'_n$ conditionally independent, given their positions. For each m and n let $\mathbf{Z}_1^{m,n}, \mathbf{Z}_2^{m,n}, \ldots$ be independent discrete variables with common density $f_{m,n}(\mathbf{x}) = (mf_0(\mathbf{x}) + nf_1(\mathbf{x}))/(m+n)$. Let $W_{m,n}$ be an independent Poisson variable with even valued mean $(m+n)$. Let $\mathcal{Z}'_{m,n} = \{\mathbf{Z}_1^{m,n}, \ldots, \mathbf{Z}_{W_{m,n}}^{m,n}\}$ be a non-homogeneous Poisson process of rate $mf_0 + nf_1$. Following the same arguments in [99], assign a mark from the set $\{1, 2\}$ to each point of $\mathcal{Z}'_{m,n}$. Specifically, a point $\mathbf{x}$ is assigned mark 1 with probability $mf_0(\mathbf{x})/\big(mf_0(\mathbf{x}) + nf_1(\mathbf{x})\big)$ and mark 2 otherwise. Let $\widetilde{\mathcal{X}}_m$ and $\widetilde{\mathcal{U}}_n$ be the set of points of $\mathcal{Z}'_{m,n}$ marked 1 and 2 respectively. Also denote $\widetilde{\mathcal{A}}_{m,n}$ the cross match statistic given from optimal weighted matching over $\widetilde{\mathcal{X}}_m \cup \widetilde{\mathcal{U}}_n$. Define the probability of two points in $\mathcal{Z}'_{m,n}$ having different marks by $g_{m,n}(\mathbf{x}, \mathbf{y})$:

$$g_{m,n}(\mathbf{x}, \mathbf{y}) = \frac{mf_0(\mathbf{x})nf_1(\mathbf{y}) + nf_1(\mathbf{x})mf_0(\mathbf{y})}{(mf_0(\mathbf{x}) + nf_1(\mathbf{x}))(mf_0(\mathbf{y}) + nf_1(\mathbf{y}))}. \tag{4.13}$$

We know that $m/N \to p_0$ and $n/N \to p_1$, hence $g_{m,n}(\mathbf{x}, \mathbf{y}) \to g(\mathbf{x}, \mathbf{y})$ where

$$g(\mathbf{x}, \mathbf{y}) = \frac{p_0 p_1 \big(f_0(\mathbf{x}) f_1(\mathbf{y}) + f_1(\mathbf{x}) f_0(\mathbf{y})\big)}{\big(p_0 f_0(\mathbf{x}) + p_1 f_1(\mathbf{x})\big)\big(p_0 f_1(\mathbf{y}) + p_1 f_1(\mathbf{y})\big)}. \tag{4.14}$$

So, the conditional expectation $E\big[\widetilde{\mathcal{A}}_{m,n} | \mathcal{Z}'_{m,n}\big]$ becomes:

$$\sum_{1 \le i < j \le W_{m,n}} \sum g_{m,n}(\mathbf{Z}_i^{m,n}, \mathbf{Z}_j^{m,n}) \mathbf{1}\big\{(\mathbf{Z}_i^{m,n}, \mathbf{Z}_j^{m,n}) \in M^*(\mathcal{Z}'_{m,n})\big\}. \tag{4.15}$$

By taking expectations in (4.15), one yields $E\big[\widetilde{\mathcal{A}}_{m,n}\big]$.

Let $\mathcal{Z}_{m,n} := \big\{\mathbf{Z}_1^{m,n}, \mathbf{Z}_2^{m,n}, \ldots, \mathbf{Z}_{m,n}^{(m+n)}\big\}$ be the original non-Poissonized set of points. By the fact that

$$E\big[|M_m + N_n - (m+n)|\big] = o(m+n),$$

the Poissonized limit of $E\big[\tilde{\mathcal{A}}_{m,n}\big]$. Set $f(\mathbf{x}) = p_0 f_0(\mathbf{x}) + p_1 f_1(\mathbf{x})$, then $f_{m,n}(\mathbf{x}) \to f(\mathbf{x})$. Using Lemma 4.4.1, we get

$$\frac{E\big[\widetilde{\mathcal{A}}_{m,n}\big]}{(m+n)} \to \frac{1}{2} \int_{\mathcal{R}^d} g(\mathbf{x}, \mathbf{x}) f(\mathbf{x}) = p_0\, p_1 \int_{\mathcal{R}^d} \frac{f_0(\mathbf{x}) f_1(\mathbf{x})}{p_0 f_0(\mathbf{x}) + p_1 f_1(\mathbf{x})}. \tag{4.16}$$

This completes the proof of Theorem 4.4.2. □

4.5 Experiments

We perform multiple experiments to demonstrate the utility of the proposed direct estimator of HP-divergence in terms of dimension and sample size. We subsequently apply our estimator to determine empirical bounds on the Bayes error rate for various datasets.

For the following simulations, the sample sizes for each class were equal ($m = n$). Each simulation used a multivariate Normal distribution for each class.

We first analyze the estimator's performance as the sample size $N = m + n$ increases. For each value of N, the simulation was run 50 times, and the results were averaged.

Samples from each class were i.i.d. 2-dimensional Normal random variables, with $\mu_0 = [0, 0]$ and $\mu_1 = [1, 1]$, $\Sigma_0 = \Sigma_1 = I_2$.

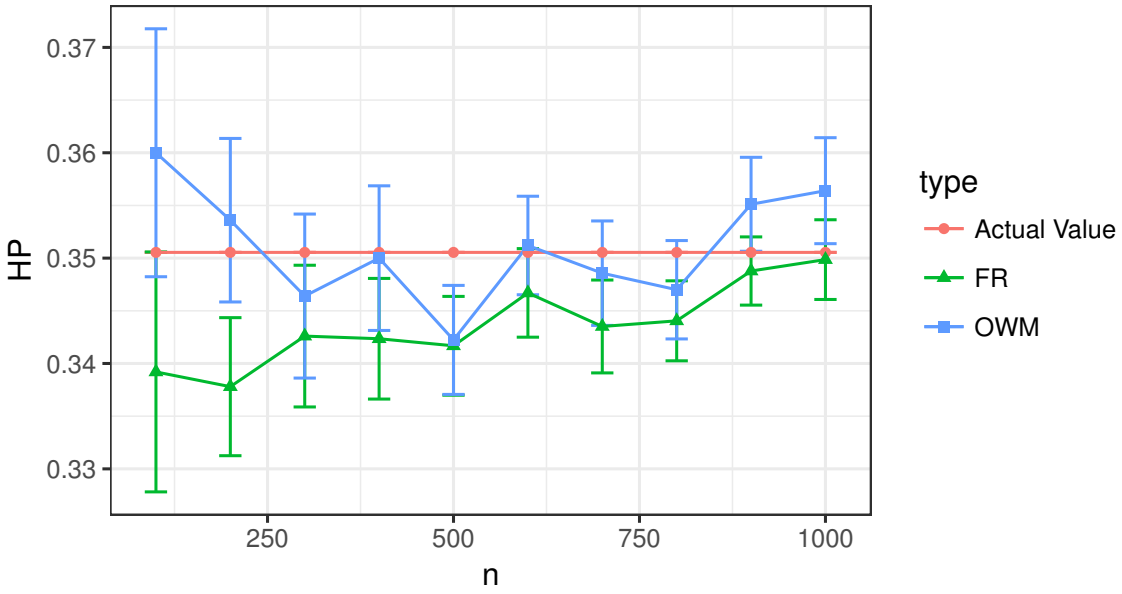


Figure 4.3: HP-divergence estimation vs. sample size n. Error bars denote the standard deviation over 50 trials. The proposed estimator and the FR estimator perform approximately equivalently over this range of sample sizes.

We see that as N increases the performance of the FR estimator and our proposed estimator (labeled OWM) are comparable for N up to 1000. The observed variance of our estimators are slightly higher than the FR estimator. For dimension $d = 2$ this is not surprising as we would expect the FR estimator to perform the best in this case.

Figure 4.4 (top) shows the averaged estimates of the HP-divergences over increasing dimension. Here we see that the proposed cross-matching estimator shows improvement with respect to the FR estimator, as expected. For each dimension evaluated in Figure 4.4, $N = 1000$, and $\mu_0 = [0]_d$ and $\mu_1 = [0.5]_d$, $\Sigma_0 = \Sigma_1 = I_d$. The proposed cross-matching estimator is slightly less biased as dimension increases, and as shown in Figure 4.4 (bottom) we improve in empirical MSE.

Next we show the results of applying the HP-divergence estimator to 4 different real data sets. Table 1 shows the cross match statistics and estimated upper bounds for Bayes

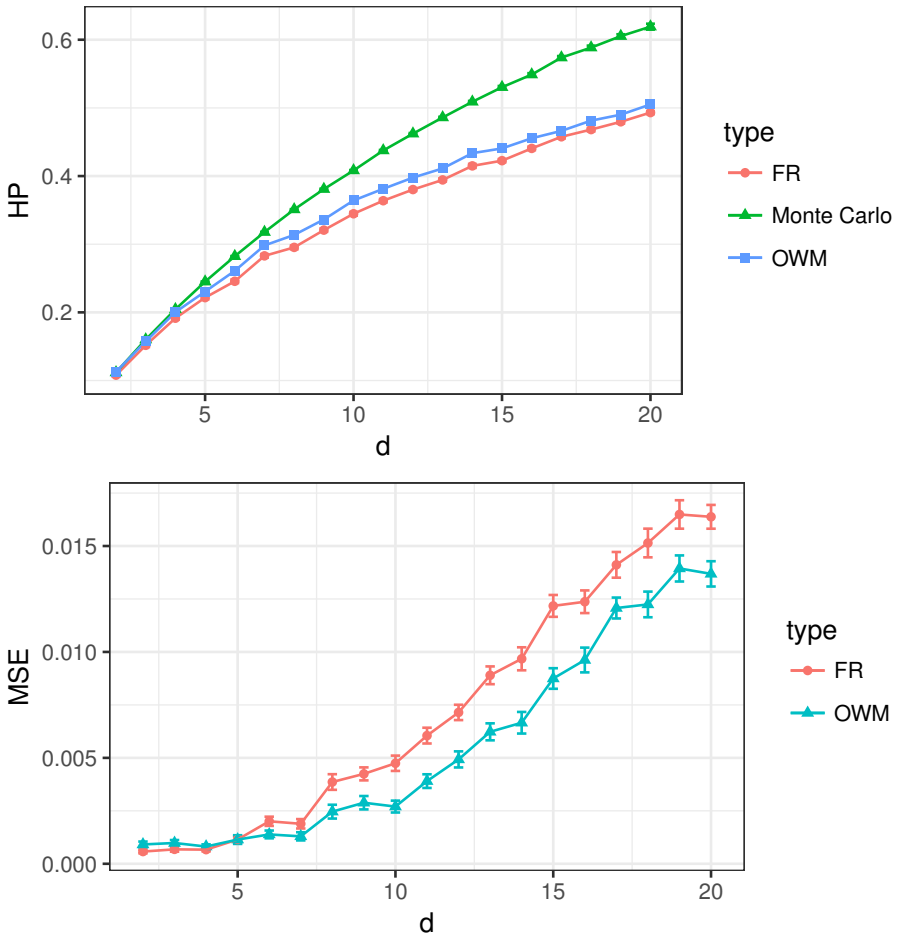


Figure 4.4: HP-divergence (top) and empirical MSE (bottom) vs. dimension. The empirical MSE of both estimators increases for larger dimensional data sets. The MSE is better for the proposed (OWM) estimator.

Error (denoted by the column labeled ϵ).

Bayes Error Bounds					
Data set	$\mathcal{A}(\mathcal{X}_N)$	$\widehat{D}_c$	n	m	ϵ
Breast cancer [231]	33	0.791	488	241	0.093
Mines vs. Rocks [133]	7	0.864	97	111	0.067
Pima diabetes [133]	67	0.641	549	283	0.161
Hyper thyroid [133]	37	0.743	3012	151	0.023

Table 4.2: $\mathcal{A}(\mathcal{X}_N)$, $\widehat{D}_c$, n, m and ϵ are the cross-match statistics, HP-divergence estimates using $\mathcal{A}(\mathcal{X}_N)$, sample sizes and upper bounds for Bayes error respectively.

4.6 Conclusion

We proposed a new dimension-independent direct estimator of HP-divergence using a statistic derived from optimal weighted matching. The estimator is more accurate than the FR approach and its variance is independent of the dimension of the support of the distributions. This translates to improved MSE performance as compared to other HP-divergence estimation methods, especially for high dimension. We validated our proposed estimator using simulations, and illustrated the approach for the meta-learning problem of estimating Bayes classification error for four real-world data sets.

CHAPTER 5

Dynamic Estimation of Influence Graphs with Adaptive Directed Information

In this chapter, we introduce an adaptive version of directed information to estimate an influence graph over nodes with time-varying features. Originally developed as a generalization of the Shannon mutual information for quantifying the effect of feedback in a simple communication channel, directed information (DI) measures the amount of causal, time-varying influence that one node's actions have on another node. By estimating these quantities, we can infer a directed graph that captures the flow of influence between nodes. We introduce an online time-averaged version of DI called adaptive directed information (ADI) to study the difference in graphical structure over time. This method is applied to two Twitter US political datasets to track changes in the graphical structure between candidates' Twitter feeds.

Symbol	Description
$N_1, N_2, \ldots, N_n$	Agents
$X_1, X_2, \ldots X_t$	Input time series
$Y_1, Y_2, \ldots Y_t$	Output time series
$\mathbf{X}^T$	$[X_1, X_2, \ldots, X_T]$
$H(\cdot)$	Shannon entropy

Table 5.1: Glossary of commonly used symbols.

5.1 Introduction

Estimating structure and interaction among targets of interest is a common problem investigated by the signal processing community. Here we are interested in estimating graphical structure that captures directed interactions from observational data generated by multiple agents. Often, it is possible to capture information on the joint behavior of these agents over time. For instance, we may want to infer the interaction of equities in the stock market over time from reported trading activity, or infer social interaction of moving objects in a scene from video. We introduce an adaptive version of the information theoretic measure directed information to quantify these interactions in an on-line recursive fashion.

Directed information (DI) was introduced in [148] to address the problem of feedback in a simple channel. DI can be thought of as an extension of mutual information (MI), and it has extensions to both infinite alphabet channels and continuous time [229]. Graphs created from DI, often called influence graphs, have been explored in the literature previously [11], [75], [193]. The authors of [11] considered influence graph estimation using the well known Granger causality measure that is equivalent to DI under a Gaussian assumption. The difficulty with DI is that its high computational and sample complexity do not allow for easy and scalable estimation methods, especially when the data is high dimensional, non-Gaussian and discrete. We describe a method that allows a time-varying DI graph to be estimated under a Markov model.

Standard DI, while able to take into account time-varying properties of the agents over time, is insensitive to abrupt changes in interaction, dependence, and influence, due to its heavy weight on past observations. We introduce adaptive directed information (ADI) that modifies DI so that it is more capable of picking up subtle shifts in the nodes' interactions. We show in this chapter that, under a Markov assumption, the ADI can be computed efficiently in an online fashion using a recursive updating scheme over time. To our knowledge, with the exception of our preliminary work [176] and the fuller treatment given in this chapter, the recursive update form of the ADI has not appeared elsewhere in the literature.

Under a simplifying instantaneous conditional independence assumption the ADI updates depend only on the joint distributions of third order. To illustrate the ADI we apply it to two Twitter datasets to estimate the influence graph among Twitter users.

This chapter is organized as follows: Section 5.2 discusses related work on influence estimation, DI, and DI graphs. Section 5.3 will introduce the problem and some notation conventions. Section 5.4 will introduce the concept of DI and ADI. Section 5.5 will demonstrate the chosen model for text information, and some assumptions made to make ADI estimation tractable. Section 5.6 will explain the process of generating DI and ADI graphs. Section 5.7 will introduce the two Twitter datasets, and discusses results from the described methods. Finally, Section 5.8 summarizes the contributions of the chapter.

5.2 Related work

Influence among actors has been studied in many settings [16], [89], [216]. DI has been studied extensively both theoretically and in the context of applications. The estimation of the directed information rate for stationary ergodic processes has been studied in [114]. Some applications of DI are covered in [186], [187] regarding gambling and portfolio theory. In addition, [195] uses DI to infer biological regulatory networks.

DI graphs have also been studied, most recently in [193], which focuses on the estimation of the causal DI graph, as well as DI estimation. The authors of [193] identify sample complexity for both non-parametric and parametric estimators for DI. The focus of [193] is on cases where the processes are stationary. [11] also discusses DI graphs, with their focus on the relationship to Granger causality. To our knowledge, no one has introduced an adaptive version of DI.

5.3 Setup and notation

Consider a set of n agents $(N_1, N_2, \ldots N_n)$, represented as nodes in a graph, that generate P-dimensional features that evolve over T time samples. We assume that the features are binary. We denote a random vector evolving over a time period t as a capital letter with a subscript, e.g., X_t. A capital letter with a superscript T represents the random vectors up to and including T, $\mathbf{X}^T = X_1, X_2, \ldots, X_T$. Finally, a lowercase letter with a superscript and a subscript, x_t^i, represents the scalar random feature i at time t.

5.4 Directed information

5.4.1 Definition and properties

Directed information is an information theoretic measure originally introduced by [148] to study the effect of feedback on channel capacity. Given a discrete communications channel $P(Y_t|\mathbf{X}^t, \mathbf{Y}^{t-1})$, with input time series $X_1, X_2, \ldots, X_t$ and outputs $Y_1, Y_2, \ldots, Y_t$, the directed information (DI) is defined as:

$$\text{DI}(\mathbf{X}^T \to \mathbf{Y}^T) = \sum_{t=1}^{T} \text{I}(\mathbf{X}^t; Y_t|\mathbf{Y}^{t-1}). \tag{5.1}$$

The DI is asymmetric, $\text{DI}(\mathbf{X}^T \to \mathbf{Y}^T) \neq \text{DI}(\mathbf{Y}^T \to \mathbf{X}^T)$. Furthermore, when the channel exhibits no feedback, e.g.,

$$P(X_t|\mathbf{X}^{t-1}, Y^{t-1}) = P(X_t|\mathbf{X}^{t-1}), \tag{5.2}$$

DI is equivalent to the standard Shannon mutual information [56].

5.4.2 Adaptive directed information

DI can account for the time-varying nature of interaction among targets (i.e. changing $P(Y_t|\mathbf{X}^t, \mathbf{Y}^{t-1})$), but does not vary over time and places equal weight on each time point in the time series. We introduce the adaptive directed information (ADI) as a time varying modification of DI defined as a discrete time filter $g(t,i)$ applied to the sequence $\mathrm{I}(\mathbf{X}^i; Y_i|\mathbf{Y}^{i-1})$, $i = 1, \ldots, \infty$:

$$(\mathrm{ADI}_{N_x \to N_y})_t = \sum_{i=1}^{t} g(t,i)\mathrm{I}(\mathbf{X}^i; Y_i|\mathbf{Y}^{i-1}) \tag{5.3}$$

The filter, $g(t,i)$ can be chosen in various ways, including the windowed exponential $g(t,i) = e^{-(t-i)\lambda}c_t, i \leq t, \lambda > 0$, where $c_t = (1-e^{-\lambda})/(1-e^{-(t+1)\lambda})$, or the uniform window of length T, $g(t,i) = 1/T, |t-i| \leq T$.

5.5 Empirical estimation of DI and ADI

Empirical estimation of the DI and ADI from data poses challenges, especially in high feature dimension P. The complexity of estimation can be reduced by imposing Markov assumptions, performing dimension reduction on the feature space, and making simplifying approximations to the joint distributions. Under a jointly Markov assumption on the pair of time series $\{(X_i, Y_i)\}_i$ we obtain a simplification of the following conditional probabilities:

$$P(X_t, Y_t|\mathbf{X}^{t-1}, \mathbf{Y}^{t-1}) = P(X_t, Y_t|X_{t-1}, Y_{t-1}), \tag{5.4a}$$

$$P(X_t|\mathbf{X}^{t-1}) = P(X_t|X_{t-1}), \tag{5.4b}$$

$$P(Y_t|\mathbf{Y}^{t-1}) = P(Y_t|Y_{t-1}). \tag{5.4c}$$

One can simplify further by imposing the additional "instantaneous conditional inde-

pendence" property that X_t and Y_t are independent given past information:

$$P(X_t, Y_t|X_{t-1}, Y_{t-1}) = P(X_t|X_{t-1}, Y_{t-1})P(Y_t|X_{t-1}, Y_{t-1}), \tag{5.5}$$

which only involves "third order distributions", i.e., distributions involving three random variables. In order to exploit this factorization to estimate DI and ADI, we write DI in terms of conditional Shannon entropies:

$$\text{DI}(\mathbf{X}^T \to \mathbf{Y}^T) = \sum_{t=1}^{T} H(Y_t|\mathbf{Y}^{t-1}) - H(Y_t|\mathbf{Y}^{t-1}, \mathbf{X}^t) \tag{5.6}$$

Using (5.4), we obtain:

$$\text{DI}(\mathbf{X}^T \to \mathbf{Y}^T) = \sum_{t=1}^{T} H(Y_t|Y_{t-1}) - H(Y_t|Y_{t-1}, X_t, X_{t-1}) \tag{5.7}$$

$$= \text{DI}(\mathbf{X}^{T-1} \to \mathbf{Y}^{T-1}) + H(Y_T|Y_{T-1}) - H(Y_T|Y_{T-1}, X_T, X_{T-1}). \tag{5.8}$$

Using standard properties of conditional entropy and (5.8), the DI expands to

$$\begin{aligned}\text{DI}(\mathbf{X}^T \to \mathbf{Y}^T) = \text{DI}(\mathbf{X}^{T-1} \to \mathbf{Y}^{T-1}) &- H(Y_{T-1}) - H(Y_T, Y_{T-1}, X_T, X_{T-1}) \\ &+ H(Y_{T-1}, X_T, X_{T-1}) + H(Y_T, Y_{T-1})\end{aligned} \tag{5.9}$$

$$\begin{aligned}= \text{DI}(\mathbf{X}^{T-1} \to \mathbf{Y}^{T-1}) &+ H(Y_T, Y_{T-1}) - H(Y_{T-1}) - H(Y_T|X_{T-1}, Y_{T-1}) \\ &- H(X_T|X_{T-1}, Y_{T-1}) - H(X_{T-1}, Y_{T-1}) + H(Y_{T-1}, X_T, X_{T-1}).\end{aligned} \tag{5.10}$$

Hence, the DI can be computed from third order distributions in recursive form where only third order entropy is required for updating the DI at time $T-1$ to time T.

We can calculate ADI directly from DI, but if we choose to use an windowed exponen-

tial filter $g(t,i)$, we obtain the recursion:

$$\begin{aligned}(\text{ADI}_{N_x\to N_y})_t &= \alpha(\text{ADI}_{N_x\to N_y})_{t-1} + (1-\alpha)[H(Y_T, Y_{T-1}) \\ &- H(Y_{T-1}) - H(Y_T|X_{T-1}, Y_{T-1}) - H(X_T|X_{T-1}, Y_{T-1}) \\ &- H(X_{T-1}, Y_{T-1}) + H(Y_{T-1}, X_T, X_{T-1})],\end{aligned} \tag{5.11}$$

where $\alpha = (e^{-\lambda} - e^{-(t+1)\lambda})/(1 - e^{-(t+1)\lambda})$.

5.5.1 Estimating joint distributions of binary vectors

Under the instantaneous conditional independence assumption the third order distributions of the form $P(X_T, Y_T, Y_{T-1})$ must be estimated in order to calculate ADI. We implement this estimator by binning together groups of time samples in order to estimate the distributions.

For concreteness we specialize to feature vectors $X = [x^1, \ldots, x^P]$ and $Y = [y^1, \ldots, y^P]$ with binary elements, i.e., $x^i, y^i \in \{0, 1\}$. While any feature dependency model could be accommodated, for simplicity we will assume elementwise independence of the feature vectors — namely, that the j-th scalar feature x_j^t is jointly independent of the other scalar features x_i^t and y_i^t, for $i \neq j, t = 1, \ldots, T$. This allows us to factorize the joint distributions of three feature vectors into third order distributions of scalar variables. Hence, for example,

$$P(X_n, X_{n-1}, Y_{n-1}) = \prod_{i=1}^{P} P_i(x_n^i, x_{n-1}^i, y_{n-1}^i) \tag{5.12}$$

$$= \prod_{p=1}^{P} \theta_{p_1}^{(1-t_1)(1-t_2)(1-t_3)} \theta_{p_2}^{(1-t_1)(1-t_2)(t_3)} \ldots \theta_{p_8}^{t_1 t_2 t_3}. \tag{5.13}$$

$\{\theta_{p_i}\}$ are parameters that must be estimated. We propose using maximum likelihood esti-

mators with Stein regularization [49]:

$$\hat{\theta}_{p_i} = (1 - \lambda_S)\hat{\theta}^{ML}_{p_i} + \lambda_S, \tag{5.14}$$

where $\hat{\theta}^{ML}_{p_i}$ is the maximum likelihood estimate of $\theta^{ML}_{p_i}$, and λ_S can be chosen to optimize bias-variance tradeoff as in [49].

The factorization (17) allows the entropy to be computed from individual feature entries:

$$H(Y_{T-1}, X_T, X_{T-1}) = \sum_{i=1}^{P} H(y^i_{T-1}, x^i_T, x^i_{T-1}). \tag{5.15}$$

We will apply the proposed ADI estimator to text data, specifically corresponding to the content of tweets from Twitter. From this data, we bin the tweets, forming documents of collected tweets over time, and model each word as a binary random variable indicating its presence or absence. These vectors are then used to estimate the $\{\theta_{p_i}\}$ parameters.

5.5.2 Computational and model complexity

Each probability estimate for a third order distribution takes $\mathcal{O}(t)$ computations, where t is the number of samples used to calculate the estimate. There are $\mathcal{O}(P)$ entropies to calculate for each estimate of directed information, and each entropy can be calculated in $\mathcal{O}(1)$. We must calculate the DI T/t times for each pair, and there are $n(n-1)/2 = \mathcal{O}(n^2)$ pairs. In total, calculation of every pairwise DI in the graph requires $\mathcal{O}(TPn^2)$ computations. ADI has an identical complexity analysis. For each DI calculation, we estimate $16P$ parameters, and these parameters can be used for both orderings of the pair. Therefore, our method must estimate $(16PTn(n-1))/(2t)$ parameters. This compares favorably with other methods that attempt to estimate higher order distributions; for general vectors of binary features and pairwise DI, one must estimate $\mathcal{O}(2^P)$ parameters for each pair.

5.6 Creating influence networks

Once pairwise DI and ADI have been calculated for all n nodes, we are able to infer graphical structure. The most naïve way to do this is to simply use each non-zero DI entry as a directed weighted edge between targets; this approach can be quite noisy. A more reasonable approach is to create a hypothesis test for each edge, and only keep the edges that have a statistically significant influence.

For DI, there are two possible ways to do this. One method, the approach of [195], uses a functional transformation leading to approximation of p-values for existence of an edge. Another method, proposed in [49], invokes a central limit theorem for DI. In this chapter, the latter approach is used.

5.7 Application to Twitter datasets

The methods described above are applied to two datasets. The first, which is a dataset regarding the United States Presidential primary candidates, are all the tweets from the campaign Twitter accounts of each candidate from Oct. 1st, 2015 to Jan. 13th, 2016. The second dataset is of the members of the United States Senate, over the same time period.

5.7.1 2015 US presidential candidates dataset

This dataset consists of 15 primary candidates. In total, there are 8918 tweets in the dataset. After cleaning, stemming, and binning the tweets into 12-hour time periods, the features (words) are further filtered as follows: if the word is used in less than 10 of the bins or greater than 50% of them, it is discarded. In total, 1554 features remain.

Figure 5.1 shows the relative DI for the entire time period, after hypothesis testing at a 5% family-wise error rate probability, where the magnitude of relative DI is $|\mathrm{DI}_{\mathbf{X}^T \to \mathbf{Y}^T} - \mathrm{DI}_{\mathbf{Y}^T \to \mathbf{X}^T}|$, and the direction of the arrow represents the sign (arrow points towards N_y if $\mathrm{DI}_{\mathbf{X}^T \to \mathbf{Y}^T}$ is larger). The width and shade of the directed edge is related to the magnitude

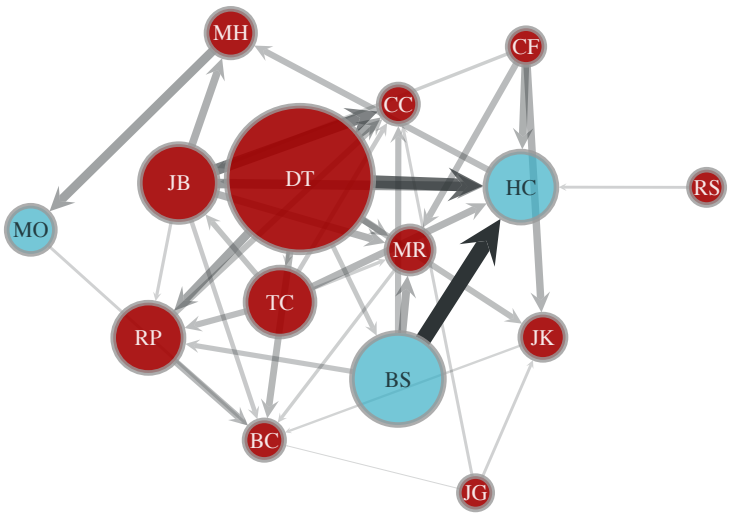


Figure 5.1: Relative DI network of US Presidential primary candidates. The width of the directed edge as well as the shade is related to the magnitude of the DI, and the size of each node represents the volume of tweets.

of the relative DI. Further, the size of each node represents the volume of tweets. The network in Figure 5.1 has some interesting properties. First, we see that nodes such as Hillary Clinton and Rand Paul are sinks of influence, that is they have high indegree and are influenced by many others. Conversely, there are nodes with high outdegree, such as Jeb Bush and Bernie Sanders that are less influenced by others.

2:47 - ...criminal justice system, but we cannot do that as long as corporations are allowed to profit from mass **incarceration**. - BS
20:00 - Tonight is the first **#DemDebate**. Attend a **debate** ...- BS
20:28 - One third ...prospect of **incarceration** ...We can do better. - HC
3:00 - In this **debate**, we tried to ... **#DemDebate**- HC

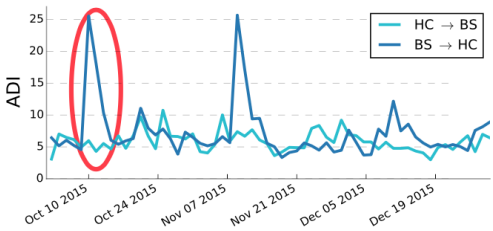


Figure 5.2: ADI for Bernie Sanders and Hillary Clinton. Above the graph are representative tweets related to the circled spike.

Figure 5.2 demonstrates the utility of ADI. ADI was calculated using an windowed exponential filter with $\lambda = 0.7$. Using ADI, we are able to see the time-varying nature

of influence, this time specifically between Bernie Sanders and Hillary Clinton. We see two large spikes in the ADI over time. The tweets above the graph partially contribute to the circled spike. Specifically, we see that Bernie Sanders discusses incarceration and the upcoming Democratic debate before Hillary Clinton does, which results in a spike of ADI from Bernie Sanders to Hillary Clinton.

5.7.2 2015 US Senate dataset

This dataset is of the members of the United States Senate, from October 1st, 2015 to January 13th, 2016. In total, the dataset consists of 96090 tweets. Figure 5.3 displays updated versions of ADI graphs at consecutive timesteps. Some senators are not displayed as they have no significant edges. We notice that there are nodes of high activity such as RB (Rob Bishop) and MK (Marcy Kaptur). Further, we see significant evolution in the network, with nodes adapting their behavior; this shows the method's ability to estimate changes in influence.

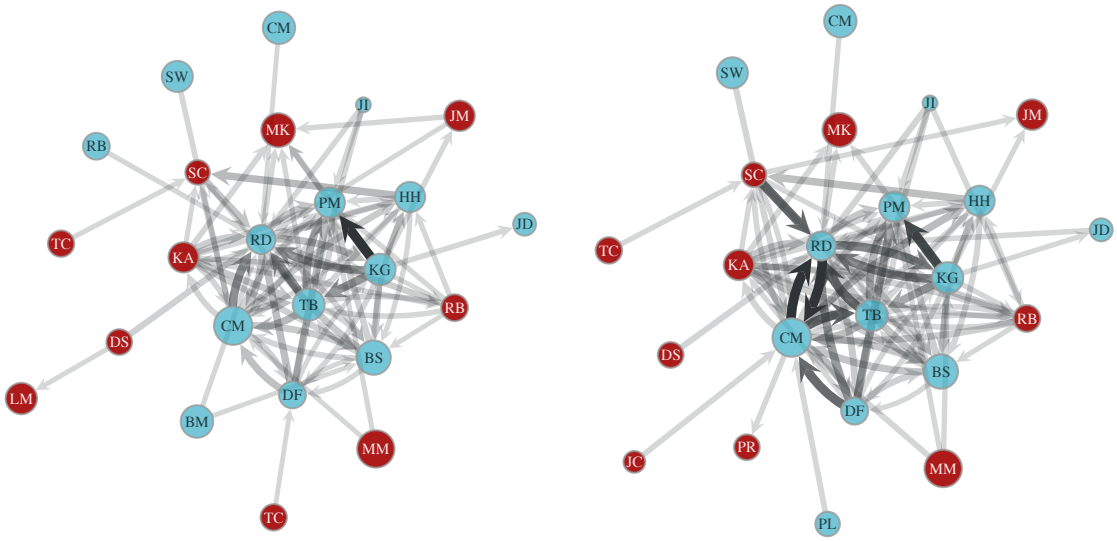


Figure 5.3: The width of the directed edge as well as the shade is related to the magnitude of the DI, and the size of each node represents the volume of tweets. We see a large connected component exhibiting mutual interaction, and significant evolution in the network, with nodes adapting their behavior.

In Figure 5.4 we plot the total degrees over time of ADI for a subset of senators. Total degree for a particular node is defined as the outdegree (sum of outgoing ADI) minus the indegree (sum of incoming ADI). These senators were chosen to show examples of nodes that have high average influence (large positive total degree), senators which receive influence approximately equal to the amount they influence (small total degree), and senators that are recipients of influence on average (large negative total degree). We note that in all cases ADI captures variation in degree over time. This is compared to total degree computed using DI, which is not sensitive to these temporal effects.

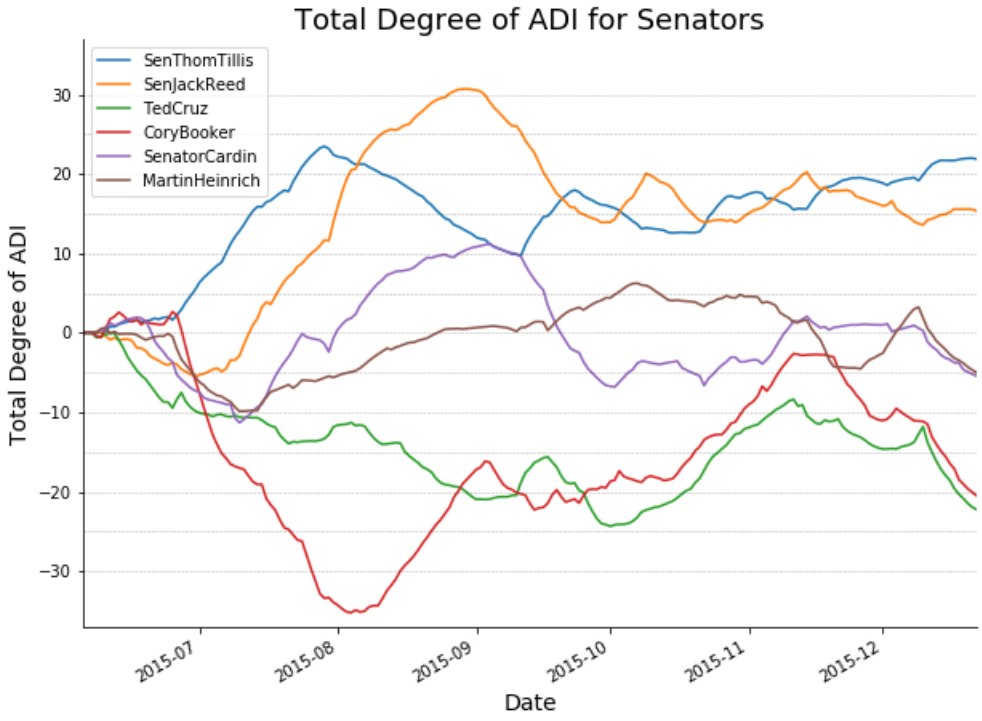


Figure 5.4: These senators were chosen as representative of senators that are high influencers (SenThomTillis, SenJackReed), average senators (SenatorCardin, MartinHeinrich), and senators that are high receivers (TedCruz, CoryBooker). We note that there is large variation over time of the total degree for each of these senators.

5.8 Conclusion

We presented an adaptive version of directed information, called ADI. ADI better captures time-varying interactions between agents in a network by representing the time evolution of DI as the output of a discrete filter with instantaneous DI as input. We further presented efficient, recursive methods to compute DI and ADI under Markovian and conditional independence assumptions. Finally, we illustrated these methods on two political Twitter datasets from the 2015 US Presidential campaign.

CHAPTER 6

Ensemble Estimation of ADI with an Application to Tracking in Video

Directed information (DI) is a useful tool to explore time-directed interactions in multivariate data. However, as originally formulated DI is not well suited to interactions that change over time. In previous work, adaptive directed information was introduced to accommodate non-stationarity, while still preserving the utility of DI to discover complex dependencies between entities. There are many design decisions and parameters that are crucial to the effectiveness of ADI. Here, we apply ideas from ensemble learning in order to alleviate this issue, allowing for a more robust estimator for exploratory data analysis. We apply these techniques to interaction estimation in a crowded scene, utilizing the Stanford drone dataset as an example.

Symbol	Description
X_t^i	Feature for agent i at time t
$\mathbf{X}_T^i$	$[X_1^i, X_2^i, \ldots, X_T^i]$
$\mathrm{I}(\cdot)$	Shannon mutual information

Table 6.1: Glossary of commonly used symbols.

6.1 Introduction

The study of interactions among entities of interest encompasses a broad array of applications and is crucial to understanding complex processes. Often times, we are interested in the directionality over time of these relationships. Examples include social influence estimation [175], [182], [193], entity interaction in video [48], and biological recording analysis, such as EEG [50], [192]. These interactions can also be used to summarize highly complex data topology, allow analysts to obtain a qualitative snapshot of the temporal interactions of the data, and make better informed decisions based on these simplified representations.

One tool that allows for the extraction of interactions is called directed information (DI). Originally created to analyze an information-theoretic channel with feedback, DI has been used in many contexts to estimate directed relationships between entities, including genetic data and social data. One deficiency of directed information is its inflexibility with respect to time-varying distributions [175], [176]. Adaptive directed information (ADI) was developed as an extension of directed information to better track changes in relationships over time.

In this chapter, we address some of the issues associated with using ADI. Specifically, ADI requires a choice of filter and corresponding filter parameters, and the quality of the resulting interaction estimate is not generally robust to these choices. In addition, simple filters may have difficulty adapting to both abrupt changes in interaction, as well as slowly time-varying systems. An estimate that is able to accomplish both smoothing over time, as well as the ability to adapt to abrupt changes in interactivity quickly is desired.

In this chapter, a form of ensemble learning is used to improve interaction estimation with ADI. Specifically, following [100], [206], we generate a filter that is a convex combination of simpler filters with different parameter specifications and whose weights are dependent on the data. In order to address the possibility of abrupt changes in the system, a growing ensemble of estimators is used to account for these changes in interactivity.

The proposed ADI estimator is applied to interaction estimation in a crowded scene, utilizing video from the Stanford drone dataset [198]. Utilizing a dynamic covariance model, the ADI is estimated and used to uncover interesting phenomena in specific scenes across the Stanford campus.

The chapter is organized as follows: Section 6.2 discusses related work. Section 6.3 introduces the mathematical concepts of DI and ADI, and introduces our ensemble estimator. Section 6.4 introduces the dynamic covariance model used to estimate ADI. Section 6.5 discusses the results on the Stanford Video Dataset. Finally, Section 6.6 concludes the chapter.

6.2 Related work

Directed information has been studied in the context of theory and applications. Estimators for DI have been proposed for the case of a finite or countably infinite feature space [115], [140], [192]. Most, if not all, estimators use the stationary Markov assumption, including plugin estimators [175], [176]. Directed information has been used in many contexts, including EEG analysis [50], neural spike trains [192], and social influence analysis [175], [176]. Changepoint detection methods [12] is one approach to track time-varying data, and parametric as well as non-parametric methods exist. However, with few exceptions, e.g., [17] these methods are mostly univariate and often require a parametric model or use simple moment-based statistics that do not capture dependency.

Other methods of influence estimation have been studied, particularly in the context of i.i.d. observations; examples include glasso [82] and hub discovery-type methods [103]. In addition, semi-parametric extensions of these models have been created for non-Gaussian data [138]. The family of directed information measures and in particular ADI is concerned with directionality in time and with more complicated time-varying signals. In this chapter, we assume a parametric multivariate Gaussian model, which is appropriate for the

particular dataset.

The ensemble method used stems from the prediction with multiple experts, a popular problem in machine learning [43], [100], [206]. Here, we use these techniques for smoothing.

6.3 ADI and ensemble estimation

6.3.1 Definition of DI and ADI

We begin with some notation. We assume that we have $1, 2, \ldots, N$ entities each with features $\mathbf{X}^i_{1:T} = [X^i_1, X^i_2, \ldots, X^i_T]$. In this chapter, $X^i_t \in \mathbb{R}^d$. Directed information between X^i and X^j is defined as follows:

$$\mathrm{DI}(\mathbf{X}^i_{1:T} \to \mathbf{X}^j_{1:T}) = \sum_{t=1}^{T} \mathrm{I}(\mathbf{X}^i_{1:t};\ X^j_t | \mathbf{X}^j_{1:t-1}), \tag{6.1}$$

where $\mathrm{I}(X;Y|Z)$ is the Shannon conditional mutual information. Many interesting conservation properties have been derived for directed information, including a close connection to the standard Shannon mutual information; these will not be repeated here, but the reader is referred to papers [11], [148], [149]. When considering the asymptotic behavior of DI for stationary processes, one defines the directed information rate:

$$\overline{\mathrm{DI}}(\mathbf{X}^i \to \mathbf{X}^j) = \lim_{T\to\infty} \frac{1}{T} \mathrm{DI}(\mathbf{X}^i_{1:T} \to \mathbf{X}^j_{1:T}).$$

If we assume that the entities form a k-Markov process, then

$$\mathrm{I}(\mathbf{X}^i_{1:t}; X^j_t | \mathbf{X}^j_{1:t-1}) = \mathrm{I}(\mathbf{X}^i_{1:t}; X^j_t | \mathbf{X}^j_{t-k:t-1}).$$

When stationarity cannot be assumed, then the traditional definition of $\overline{\mathrm{DI}}$ is inapplicable. However, the instantaneous DI summand of (6.1) retains valuable information about

temporal interactivity of the entities i and j. In [175], we proposed to adaptively estimate this quantity using adaptive directed information (ADI), which is defined as follows:

$$\text{ADI}(\mathbf{X}^i_{1:T} \to \mathbf{X}^j_{1:T}) = \sum_{t=1}^{T} g(t,T)\mathbf{I}(\mathbf{X}^i_{1:t}; X^j_t|\mathbf{X}^j_{1:t-1}),$$

where $g(t,T)$ is a user-defined taper function. In past work [175], the focus has been on the exponential filter $g(t,T) = \alpha(1-\alpha)^{t-T}$, so that ADI obeys the recursive update:

$$\text{ADI}^{i\to j}_{1:t} = \alpha\mathbf{I}(\mathbf{X}^i;\ X^j_t|\mathbf{X}^j_{1:t-1}) + (1-\alpha)\text{ADI}^{i\to j}_{1:t-1},$$

where $\text{ADI}^{i\to j}_{1:t} = \text{ADI}(\mathbf{X}^i_{1:t} \to \mathbf{X}^j_{1:t})$. However, the parameter α of the exponential filter must be tuned according to the specific application. The goal of the this chapter is to improve the robustness of ADI when the underlying state is unknown and rapidly changing. In order to accomplish this, an ensemble filter is defined:

$$g^*(t,T) = \frac{\sum_{i=1}^{n_t} w_{i,t} g_i(t,T;t_0)}{\sum_{i=1}^{n_t} w_i}, \tag{6.2}$$

where $g_i(t,T)$ are "base filters" with different parameter specifications. Implicitly, the weights w_i are allowed to depend on past data. Further, the number of base filters included in the ensemble (n_t) is allowed to grow with t, and filter functions will be causal, i.e., $g(t,T;t_0) = 0$ for $t < t_0$.

6.3.2 Expanding fixed shares of estimation

We apply an ensemble method based on the simple fixed shares algorithm [100], which was originally introduced in [206].

A set of base filter functions is defined, $G = \{g_1, \ldots, g_k\}$ along with a parameter τ which defines the rate at which new filters are introduced into 6.2

At each time t, an estimate $\hat{\mathbf{I}}(\mathbf{X}^i_{1:t};\ X^j_t|\mathbf{X}^j_{1:t-1})$ is obtained and used to both update the

weights w_i and to update the ADI estimate.

The weights w_i are updated in a similar manner to [206]:

$$v_{i,t} = w_{i,t-1} e^{-\gamma(y_{i,t}-i_t)^2}, w_{i,t} = (1-\beta)v_{i,t} + \frac{\beta}{n_t}\sum_{i=1}^{n_t} v_{i,t},$$

where $\beta \in [0,1]$ and $\gamma > 0$ are user-defined hyperparameters. Theorem 6.3.1 provides a bound for the MSE, assuming that $\mathrm{I}(\mathbf{X}^i_{1:t};\ X^j_t|\mathbf{X}^j_{1:t-1})$ is piecewise constant, and the estimate has i.i.d. noise with bounded variance. We use the abbreviation $i_t = \mathrm{I}(\mathbf{X}^i_{1:t};\ X^j_t|\mathbf{X}^j_{1:t-1})$, and similarly $\hat{i_t} = \hat{\mathrm{I}}(\mathbf{X}^i_{1:t};\ X^j_t|\mathbf{X}^j_{1:t-1})$ for convenience.

Theorem 6.3.1. *Let $\hat{i}_t = i_t + \epsilon_t$, where ϵ_t is independent with mean 0 and variance σ_t^2, and i_t is piecewise constant with m transitions. Then the MSE of the ADI ensemble estimator is bounded by:*

$$\mathbb{E}\left[\sum_{t=1}^{T}(\overline{\mathrm{ADI}}(t) - i_t)^2\right] \leq \frac{m}{\gamma}\ln n_t - \frac{1}{\gamma}\ln \beta^m(1-\beta)^{T-m} + \frac{\gamma}{8}T + m\sigma_*^2 \ln\left(\frac{T}{e}\right), \quad (6.3)$$

where $\sigma_*^2 = \max_t \sigma_t^2$.

The proof of Theorem 6.3.1 is given in Appendix C.1.

6.4 Spatial interaction estimation in a scene

We illustrate ADI by applying it to discover salient time-varying interactions among actors in a scene. Here, the components $n = 1, \ldots, N$ are actors moving around in space. For each sampled frame t and actor i, define the position vector $X^i_t = [x^i_t, y^i_t]$ on the plane.

6.4.1 Dynamic covariance model

We propose a dynamic Gaussian model, following the model in [51]. Assume that the combined feature matrix is distributed as:

$$\mathbf{X} \sim \mathcal{N}(m_t, \Sigma_t), \tag{6.4}$$

where m_t is a mean vector and Σ_t is a covariance matrix. We assume that m_t and Σ_t are slowly varying, and further use a kernel estimate of these quantities:

$$\hat{m}_t = \frac{1}{\sum_{i=1}^{T} K_h(i-t)} \sum_{i=1}^{T} K_h(i-t) X_i. \tag{6.5}$$

$$\hat{\Sigma}_t = \frac{1}{\sum_{i=1}^{T} K_h(i-t)} \sum_{i=1}^{T} K_h(i-t)(X_i - \hat{m}_i)(X_i - \hat{m}_i)^T, \tag{6.6}$$

where $K_h(t)$ is a kernel function. The conditional mutual information is a function of the covariance matrices under a Markovian Gaussian random process.

$$\hat{\mathbf{I}}(\mathbf{X}^i_{1:t}; X^j_t | X^j_{t-1}, X^{[N]/\{i,j\}}_{t-1}) = \frac{1}{2} \log \frac{\left| \hat{\Sigma}_{X^j_t | X^j_{t-1}, X^{[N]/\{i,j\}}_{t-1}} \right|}{\left| \hat{\Sigma}_{X^j_t | X^j_{t-1}, X^i_{t-1}, X^{[N]/\{i,j\}}_{t-1}} \right|}.$$

6.5 Application to Stanford drone dataset

In this section, the proposed ensemble ADI estimator is applied to the Stanford Drone Dataset [198], which is a collection of 60 annotated videos across 8 scenes shot on the Stanford campus. These annotations allow for tracking the movement of pedestrians, cars, bicyclists and other moving actors in the scene. These estimated locations of actors are smoothed by a moving mean estimator in order to reduce artifacts introduced by the discretization of the annotations. These smoothed locations for each actor in the scene are then used to calculate the ADI.

For the analysis, an rbf kernel was used in (6.5) with parameter $h = 5$, and the ADI en-

semble parameters were set to $\tau = 10, \beta = 0.01, \gamma = 1$, and $G = \{\exp(0.1), \exp(0.2), \text{unif}\}$. After calculating ADI, only interactions where the actors were within a certain distance (in pixels) from each other were considered - in this case, 100.

6.5.1 Interaction example between pedestrians

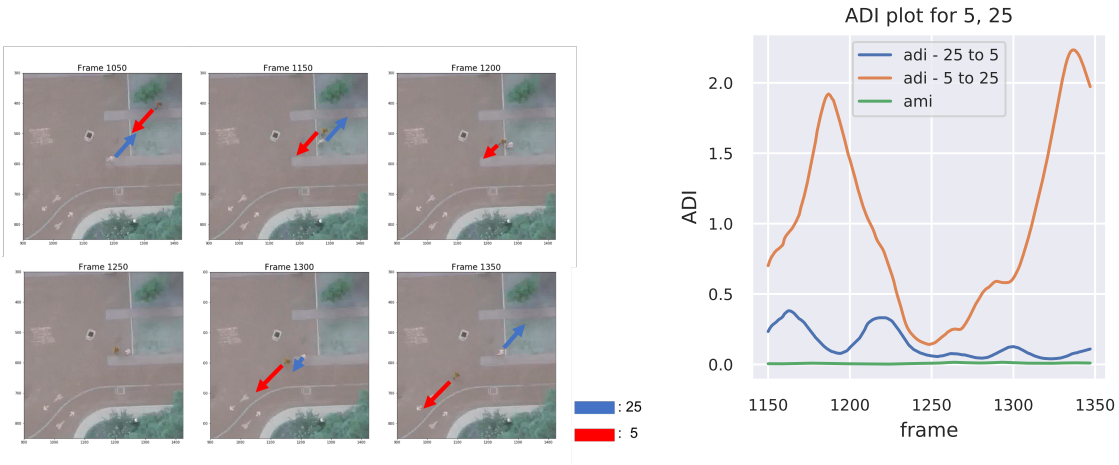


Figure 6.1: Stanford video dataset example. Here, we capture an interaction of two people meeting in the scene. The shown video frames and corresponding ADI are demonstrating them coming towards each other, interacting briefly, as actor 25 even walks in the other direction to continue the conversation, and then resuming their original path. The title on the plots pair of labeled actors in "video0" of the bookstore scene, and the line labeled i to j represents $\text{ADI}^{i \to j}$.

Figure 6.1 shows one example of ADI and the corresponding interaction between two pedestrians. The pedestrians labeled 5 and 25 stop to chat briefly, with 25 actually reversing course for a small time to continue the conversation at frame 1280 to 1300 to continue the conversation. The estimated ADI is able to identify this interaction, and to identify that there is more influence from 5 to 25 than vice versa over this small window. This is compared with an adaptive version of mutual information:

$$\text{AMI}(\mathbf{X}^i_{1:T}, \mathbf{X}^j_{1:T}) = \sum_{t=1}^{T} g^*(t, T) \hat{\text{I}}(X^i_t; X^j_t | \mathbf{X}^i_{1:t-1}, \mathbf{X}^j_{1:t-1}),$$

where the ensemble method outlined for ADI is applied to the estimated summand

$\hat{\mathbf{I}}(X_t^i; X_t^j | \mathbf{X}_{1:t-1}^i, \mathbf{X}_{1:t-1}^j)$.

6.5.2 Visualization of interactions based on ADI

We can use ADI as a tool to cluster and visualize many interactions in the dataset. First, the ADI for all interactions between actors in the bookstore scene from the Stanford Drone dataset across 5 different videos are collected, totaling $m = 539$ interactions. Using symmetrized ADI, $\text{ADI}^{i,j} = \text{ADI}^{i \to j} + \text{ADI}^{j \to i}$, the maximal cross correlation between each interaction is found, and this correlation is used as an affinity measure $a_{k,l}$, with the corresponding affinity matrix $A = [a_{k,l}]_{k,l=1,\ldots m}$. Note that $a_{k,l} = a_{l,k}$, and so A is symmetric. A can then be used to apply a number of visualization and clustering techniques. Here, we use t-SNE dimension reduction and visualization method [144], by transforming A to a distance matrix $D = [d_{i,j}]_{i,j=1,\ldots m}$, where $d_{i,j} = \sqrt{2(1 - a_{i,j})}$ and applying the method to this matrix. Figure 6.2 shows the results. The colors correspond to different types of interactions, such as between pedestrians, or between a pedestrian and a bike, etc.

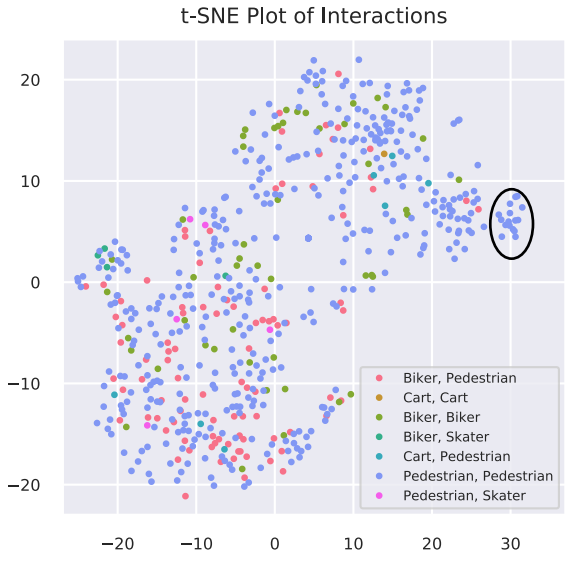


Figure 6.2: t-SNE plot of interactions based on ADI. The highlighted cluster of pedestrian interactions is characterized by low levels of interaction over a long period of time combined with spikes of activity.

The visualization shows small clusterings of interactions. An example is circled in black, with representative traces shown in Figure 6.3. More generally, we see that the pedestrian-biker interactions mostly cluster in the bottom-left portion of the plot, while the biker-biker and pedestrian-pedestrian interactions are less cohesive as a group, implying heterogeneity among these types of interactions. The small highlighted cluster of pedestrian interactions, for example, are characterized by long periods of low ADI combined with abrupt spikes. These are observed to correlate to pedestrians walking slowly in the same direction or standing still along with occasional changes in velocity or direction.

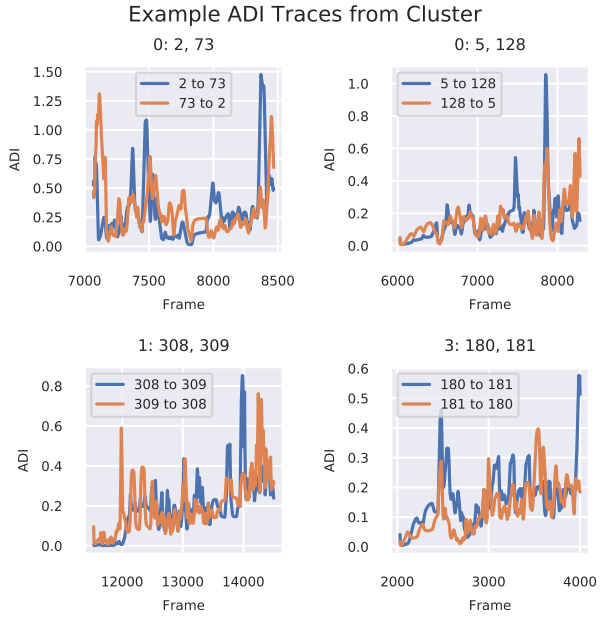


Figure 6.3: Representative ADI traces from highlighted cluster. The majority of these interactions are pedestrians that are moving slowly together or standing still in close proximity, with abrupt direction and velocity changes. The titles on the plots represent the origin video and pair of labeled actors in the dataset, and the line labeled i to j represents $\text{ADI}^{i \to j}$.

6.5.3 Relationship between ADI and velocity

In this section we study the relationship between the velocity profile and ADI profile of particular types of interactions. For each interaction and each actor i the instantaneous velocity vector $\mathbf{v}_t^i = [v_{t,x}^i, v_{t,y}^i]$ is calculated, along with the corresponding instantaneous

magnitude $v_t^i = \|\mathbf{v}_t^i\|$. Further, the instantaneous velocity angle between two actors i and j is calculated:

$$\theta_t^{i,j} = \arccos\left(\frac{\mathbf{v}_t^i \cdot \mathbf{v}_t^j}{v_t^i v_t^j}\right).$$

Using the relative velocity angle, we can look for two specific types of interactions, and how their ADI profiles differ; those with high angle, so that the two actors are approaching from opposite directions, and low angle, where the two actors are moving in the same direction. Figure 6.4 shows four representative interactions, two with low velocity angles and two with high velocity angles.

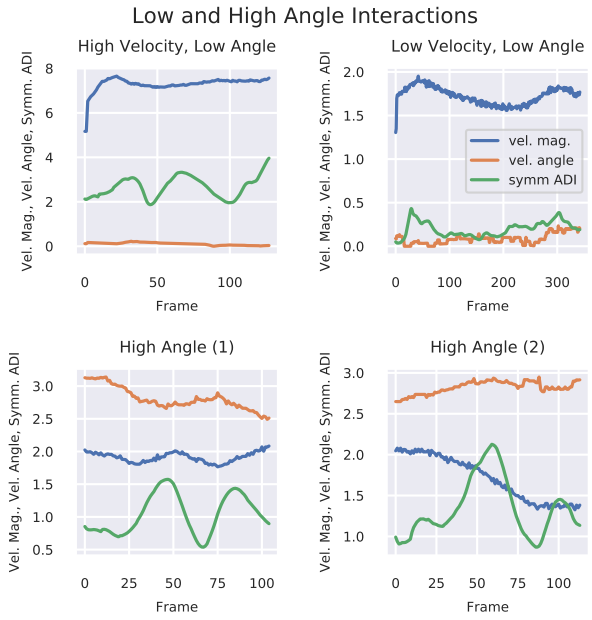


Figure 6.4: Representative profiles of low and high velocity angle interactions. The top row shows two low-angle interactions, one with high total velocity. The high total velocity interaction has a relatively constant symmetrized ADI profile, while the low total velocity interaction has an ADI profile close to 0. The high angle interactions have more variable ADI profiles relative to their magnitude, and tend to be sensitive to changes in total velocity.

In general, interactions with high total velocity, defined as $v_t^i + v_t^j$, and low velocity angle see a stable and non-zero symmetrized ADI. In the low total velocity setting, the ADI is normally much smaller than its high velocity counterpart. Two examples of low-angle interactions are shown in the top row of Figure 6.4. In the high angle case, ADI is

less constant, and in many cases responds more to changes in total velocity, as shown in the bottom row of Figure 6.4.

6.5.4 Average ADI between different types of actors

Figure 6.5 shows a graph of the average ADI between types of actors in the bookstore scene from the Stanford drone dataset across 5 different videos.

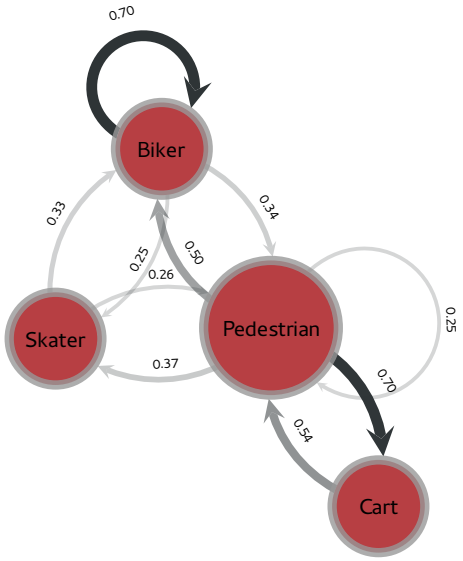


Figure 6.5: Average ADI between types of actors in the bookstore scene for the Stanford drone dataset. Bikers have the largest levels of interaction, while skaters have the least.

Skaters tend to have the lowest average ADI with other groups, followed by pedestrians, with bikers and carts having the largest interaction magnitudes. Interestingly, pedestrians influence bikers and carts more than the two groups influence pedestrians on average, possibly signifying that bikers and carts are more cautious and thus are more affected by pedestrians in the vicinity. As seen in Figure 6.4, the velocity magnitudes in interactions can play a role, specifically that the magnitudes of velocity and ADI are positively correlated. With bikers being among the fastest moving actors in this graph, it makes sense that they have some of the largest interaction magnitudes.

6.6 Conclusion

In this chapter, we introduced an ADI estimator that utilizes an ensemble technique in order to make ADI more robust to user-specified parameters. The estimator is applicable to real-world scenarios where directed information evolves as a function of time. We illustrated the power of the ensemble ADI estimator to detect latent interactions in a video using the Stanford drone dataset. In the future, ADI can be used as a data summarization and exploration tool or as a component in a larger system.

CHAPTER 7

Multi-layer Networks

Many real-world complex systems can be described by multi-layer networks, where a set of elementary units (e.g, human, gene, sensor, or other types of 'nodes') are connected by intra-layer and inter-layer relationships ('edges'). Social network data are one of the best known examples of multi-layer networks, where social entities are linked due to a social tie, and each layer represents a different type of relationship. Networks changing over time (dynamic networks) can also be placed in a multi-layer framework. In this chapter, we begin by introducing various types and formulations of multi-layer networks; centrality measures for multi-layer networks are discussed, including how they differ from single-layer counterparts. We also review community detection methods for multi-layer networks. We then discuss topology estimation of multi-layer networks, particularly in the more specific case of dynamic networks, and discuss some recent techniques. Lastly, two empirical studies of multi-layer networks are explored, based on social media and biological data, respectively. We demonstrate some of the ideas in the chapter to explore and exploit the multi-layer nature of these examples.

7.1 Introduction

In this chapter, we are interested in describing a framework that allows for heterogeneous structured data. We often find heterogeneous structure in social media data - there may exist more than one type of relationship between agents; these relationships may impose

Symbol	Description
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$	a graph with vertex set $\mathcal{V}$ and edge set $\mathcal{E}$
$\mathcal{M}$, $\mathcal{G}_M$, $\mathbf{M}$	a multi-layer network $\mathcal{M}$ with supra-graph $\mathcal{G}_M$ and tensor form $\mathbf{M}$
$\mathbf{A}_M$, $\mathbf{L}_M$	supra-adjacency matrix $\mathbf{A}_M$ and supra-Laplacian matrix $\mathbf{L}_M$
$\mathbf{A}^{(\alpha)}$, $\mathbf{L}^{(\alpha)}$	adjacency matrix and Laplacian matrix for network at layer α
$[L]$	an integer set $\{1, 2, \ldots, L\}$
$\circ$, $\otimes$	outer/tensor product, and Kronecker product
$\mathbf{X}^{(t)}$, $\mathbf{Y}^{(t)}$	features from time $1, \ldots, t$

Table 7.1: Glossary of commonly used symbols.

different topological characteristics. For instance, people may be connected by more than one social platform. Alternatively, we may observe explicit links between agents but also infer implicit affinities based on agent features.

Another example of this heterogeneous structure arises when relationships between agents appear and disappear over time; agents begin talking to each other at one time, and end at another time, possibly signifying a change in relation. Both of the above examples can be explained by a multi-layer network framework.

A multi-layer network is a network where a set of elementary units are connected by intra-layer and inter-layer relationships ('edges'). This structure is a generalization of single-layer networks, where there are only intra-layer relationships. These layers represent heterogeneity in the structure or labeling of the data; a layer might correspond to a type of connection, or a discrete timestep. The inter-layer structure represents ties among nodes in the different layers; this structure may be observed, assumed, or estimated depending on the application. The inter-layer structure in a social network often corresponds to the labels of each node, so that each node in a single layer is connected to its counterparts in the other layers. If the layers represent timesteps, each entity might be connected to its counterpart in layers before and after the present layer, which represents the localization of that layer's characteristics in time.

As the multi-layer structure is more complicated than its single-layer counterpart, methods for single-layer analysis must be modified to accommodate accordingly, and new meth-

ods are developed specifically for the multi-layer case. This chapter will review some of the approaches for modeling multi-layer networks, and some of the methods that are specific to this structure.

The rest of this chapter will proceed as follows: Section 7.2.1 will discuss the mathematical formulation of multi-layer networks. Section 7.3 will cover some examples of multi-layer node centralities. Section 7.4 will review some types of multi-layer community detection methods. Section 7.5 will utilize some of the techniques discussed in the chapter on two application datasets. Finally, Section 7.6 will provide some concluding remarks.

7.2 Mathematical formulation of multi-layer networks

In this section, we focus on the mathematical formulation of multi-layer networks. Different from single-layer networks, they allow multiple types of interactions between each pair of nodes. In what follows, we introduce two network representations: supra-adjacency representation and tensor representation, each of which generalizes the notation of a single-layer network. We next show some real-life examples that involve the multi-layer network structure.

7.2.1 Modeling and representation

A single-layer network (also called a monoplex network) can be represented by a graph [54]. A graph is a tuple $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges that connects pairs of nodes. A multi-layer network generalizes the notion of single-layer network by incorporating the inter-layer connections; see Figure 7.1A for an illustrative example. More formally, a multi-layer network is a pair $\mathcal{M} = (\mathcal{T}, \mathcal{C})$ [227], where $\mathcal{T} = \{\mathcal{G}_\alpha,\, \alpha \in [L]\}$ is a family of graphs $\mathcal{G}_\alpha = (\mathcal{V}_\alpha, \mathcal{E}_\alpha)$ with $\mathcal{V}_\alpha \subseteq \mathcal{V}$, $[L] := \{1, 2, \ldots, L\}$, and $\mathcal{C} = \{\mathcal{E}_{\alpha\beta} \subseteq \mathcal{V}_\alpha \times \mathcal{V}_\beta,\, \alpha, \beta \in [L]\}$ denotes the set of inter-layer connections ($\alpha \neq \beta$). Here α is the layer index, and by convention, $\mathcal{E}_{\alpha\alpha} = \mathcal{E}_\alpha$. When

$L = 1$, the multi-layer network $\mathcal{M}$ simplifies to a single-layer network. In the rest of the chapter, unless specified otherwise, we assume that each layer contains the same set of nodes with $|\mathcal{V}_\alpha| = |\mathcal{V}| = N$ for $\alpha \in [L]$, where $|\mathcal{V}|$ denotes the cardinality of the node set $\mathcal{V}$.

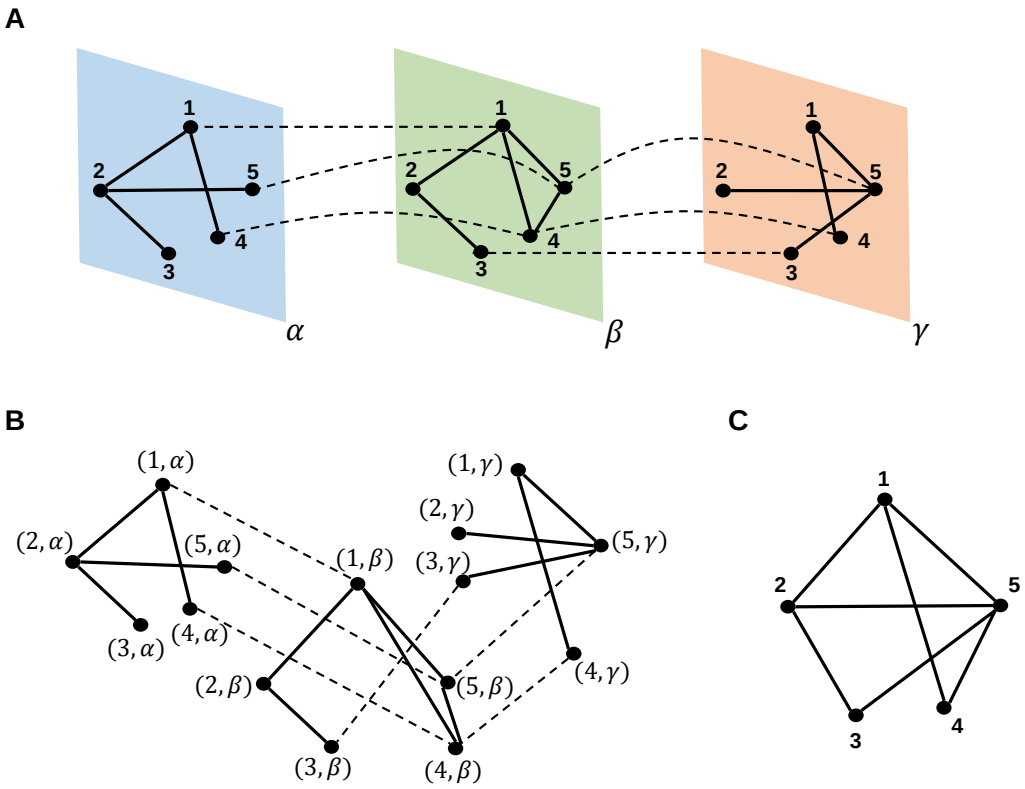


Figure 7.1: Example with 5 nodes and 3 layers labeled α, β and γ. (A) Multi-layer network where solid line represents intra-layer connection, and dash line represents inter-layer connection. (B) Supra-graph representation. (C) Aggregated network.

7.2.1.1 Supra-graph representation

Let $\mathcal{V}_M \subseteq \mathcal{V} \times [L]$ denote a set of node-layer combinations corresponding to $\mathcal{M}$, where $(v, \alpha) \in \mathcal{V}_M$ signifies that the node $v \in \mathcal{V}$ is present in layer $\alpha \in [L]$. Let $\mathcal{E}_M \subseteq \mathcal{V}_M \times \mathcal{V}_M$ be the set of edges between node-layer tuples. The multi-layer network $\mathcal{M}$ can then be described by a graph $\mathcal{G}_M = (\mathcal{V}_M, \mathcal{E}_M)$, known as a supra-graph, leading to a supra-adjacency matrix $\mathbf{A}_M$ and/or a supra-Laplacian matrix $\mathbf{L}_M$ [120]. Figure 7.1B shows the supra-graph

representation of the multi-layer network in Figure 7.1A. Based on such a representation, many methods for single-layer networks, e.g., centrality-based network diagnostics and community detection methods, can be extended to multi-layer networks [58], [212].

In contrast to the supra-graph, network aggregation provides the simplest representation for a multi-layer network, where connections between nodes are aggregated in all layers to a single layer. The resulting graph is given by $\mathcal{G}_a = (\mathcal{V}_a, \mathcal{E}_a)$, where $\mathcal{V}_a = \cup_{\alpha=1}^{L} \mathcal{V}_\alpha$ and $\mathcal{E}_a = \cup_{\alpha=1}^{L} \mathcal{E}_\alpha$. Often the aggregated network can be cast as a convex combination (e.g., linear combination) of graph adjacency matrices across all layers [22], [47]. Although such an aggregation may cause loss of information about the inter-layer network structure [120], it becomes useful when modeling across networks that have very similar inter-layer connectivity. Figure 7.1C shows the aggregated representation of the multi-layer network in Figure 7.1A.

7.2.1.2 Tensor representation

A multi-layer network can be represented in a tensor form [67], [124], [227]. Let $\mathbf{M} \in \mathbb{R}^{N \times L \times N \times L}$ denote the fourth-order adjacency tensor of the L-layer network $\mathcal{M}$. Each element of $\mathbf{M}$ is defined by

$$M_{i\alpha j\beta} = \begin{cases} w_{i\alpha j\beta} & \text{if } (v_i^{(\alpha)}, v_j^{(\beta)}) \in \mathcal{E}_{\alpha\beta} \\ 0 & \text{otherwise,} \end{cases} \tag{7.1}$$

for $i, j \in [N]$ and $\alpha, \beta \in [L]$, where $v_i^{(\alpha)} \in \mathcal{V}_\alpha$ denotes node i at layer α, and $w_{i\alpha j\beta}$ is the weight corresponding to the edge $(v_i^{(\alpha)}, v_j^{(\beta)})$. We refer readers to [124] for a detailed background on tensors.

We can express the multi-layer adjacency tensor (7.1) as a linear combination of tensors

in the canonical basis

$$\mathbf{M} = \sum_{i=1}^{N}\sum_{\alpha=1}^{L}\sum_{j=1}^{N}\sum_{\beta=1}^{L} w_{i\alpha j\beta}(\mathbf{e}_i \circ \mathbf{e}_\alpha \circ \mathbf{e}_j \circ \mathbf{e}_\beta), \tag{7.2}$$

where $\circ$ represents the vector outer product (tensor product)[1], $\mathbf{e}_i$ is a basis vector in $\mathbb{R}^N$ with 1 at the ith coordinate and 0s elsewhere, and $\mathbf{e}_\alpha$ is a basis vector in $\mathbb{R}^L$. The tensor representation (7.2) can be viewed as a generalization of the graph adjacency matrix $\mathbf{A} \in \mathbb{R}^{N\times N}$ for the single-layer network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$,

$$\mathbf{A} = \sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}\mathbf{e}_i\mathbf{e}_j^T = \sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}(\mathbf{e}_i \circ \mathbf{e}_j), \tag{7.3}$$

where w_{ij} is the weight associated with edge $(v_i, v_j) \in \mathcal{E}$.

In addition to the fourth-order tensor representation (7.2), a multi-layer network is also modelled by a third-order tensor in [84], where each slice corresponds to the network at one layer, e.g., a dynamic network at one snapshot. In contrast with the third-order tensor, the forth-order tensor encodes detailed information on the inter-layer connection between any two nodes at different layers, namely, (7.1). Also, the fourth-order tensor $\mathbf{M}$ can be flattened out to the supra-adjacency matrix $\mathbf{A}_M$ of dimension $NL \times NL$. Therefore, the fourth-order tensor is a natural representation of multi-layer networks, and many techniques on tensor algebra [67] can be used for network analysis.

7.2.2 Examples of multi-layer networks

We next introduce three important classes of multi-layer networks: node-colored networks, edge-colored networks, and temporal networks [120].

Node-colored networks are graphs in which each node is labelled by one color. Considering each color as designating a layer, node-colored graphs can be represented as multi-

[1] If $\mathbf{X} = \mathbf{a}_1 \circ \mathbf{a}_2 \circ \ldots \circ \mathbf{a}_n$, then each element of the tensor $\mathbf{X}$ is given by $X_{i_1 i_2 \ldots i_n} = [\mathbf{a}_1]_{i_1}[\mathbf{a}_2]_{i_2}\cdots[\mathbf{a}_n]_{i_n}$, where $[\mathbf{x}]_i$ denotes the ith entry of $\mathbf{x}$.

layer networks. They are often used to model heterogeneous networks that contain nodes of different types.

Example 1 Bibliographic information networks contain information about researchers (authors) and publications they produce (documents). Links exist between papers and/or authors by the authorship, colleagueship, published venues, or topics [214].

Example 2 Internet of things (IoT) denotes the inter-networking of smart phones, computers, vehicles, buildings, and other devices embedded with electronics, sensors and actuators [91]. IoT allows autonomous exchange of useful information between 'heterogeneous nodes'.

Edge-colored networks are graphs with multiple types of edges, where similar to node-colored networks, color distinguishes between layers. Edge-colored graphs can be represented by multi-layer networks, where nodes in each layer are fixed and linked by edges with a unique color. They can be used to model multirelational networks where nodes have relations of different types [37].

Example 3 Public social networks link social entities by several types of relationships, including friendship, vicinity, kinship, and membership in the same cultural society [239].

Example 4 Urban transportation networks describe the urban ecosystem, where nodes represent spatial locations (e.g., restaurants, shopping malls, schools, parks and other places of interest), and edges represent vehicles of different types, e.g., taxis, buses, and subways, that are used to travel between two locations [9].

A temporal network is given by an ordered sequence of graphs. It can be interpreted as a special case of an edge-colored multigraph, where the set of time instants provides the set of edge colors, and the inter-layer edges are between nodes and their counterparts across all time steps. The chromatin contact map over a time course of cell growth/development is an example of a temporal network in biology [139].

7.3 Diagnostics for multi-layer networks: centrality analysis

The study of centrality, i.e., evaluating the degree of nodal importance to the network structure, is often used to identify and rank essential nodes in complex networks. A number of centrality measures are commonly used, such as degree, eigenvector, clustering coefficient, closeness, betweenness, hubness and authority, differing in what type of influence is to be emphasized [165]. For example, degree centrality measures the total number of connections a node has, while eigenvector centrality measures the importance of a node by the importance of its neighbors [28]. Most centrality methods are only directly applicable to single-layer networks. Here we generalize some important single-layer centrality methods to multi-layer networks.

7.3.1 Overlapping degree and multiplex participation coefficient

Nodal degree is the simplest feature in network diagnostics. There exist several ways to define multi-layered degree centrality. The simplest way is to use network aggregation, where two nodes are considered to be adjacent if and only if the number of edges that connect them in a multi-layer network is larger than a threshold [34], [35]. However, this measure does not fully consider the inter-layer effect.

In a multi-layer network, it is essential to study how the nodal degree is distributed across different layers. We recall from Section 7.2.1 that $\mathcal{M}$ denotes a multi-layer network with N nodes and L layers, the degree of node i on layer α becomes

$$k_i^{(\alpha)} = \sum_{j=1}^{N} A_{ij}^{(\alpha)}, \tag{7.4}$$

where $A_{ij}^{(\alpha)}$ is the (i,j)th entry of the adjacency matrix associated with graph $\mathcal{G}_\alpha$ on layer

α. The degree of node i in a multi-layer network is a vector quantity

$$\mathbf{k}_i = [k_i^{(1)}, k_i^{(2)}, \ldots, k_i^{(L)}],\ i \in [N]. \tag{7.5}$$

The overlapping degree of node i across all layers is defined as [22]

$$o_i = \sum_{\alpha=1}^{L} k_i^{(\alpha)} = \mathbf{1}^T \mathbf{k}_i,\ i \in [N], \tag{7.6}$$

where $\mathbf{1}$ is the $L \times 1$ vector of all ones. The overlapping degree (7.6) can be used to identify hubs, nodes with high degree in the network. However, a node that is a hub in one layer may only have a few connections in another layer. Thus a more suitable multi-layer hub definition is the multiplex participation coefficient [22], [93],

$$P_i = \frac{L}{L-1}\left[1 - \sum_{\alpha=1}^{L}\left(\frac{k_i^{(\alpha)}}{o_i}\right)^2\right]. \tag{7.7}$$

Here P_i takes values in $[0, 1]$ and measures the degree to which the degree of node i is uniformly distributed among the L layers. If $P_i = 1$, then node i has exactly the same number of edges on each layer, namely, $k_i^{(\alpha)} = o_i/L$. If $P_i = 0$, all the edges of node i are concentrated in just one layer. The multiplex participation coefficient thus captures heterogeneity of nodal degrees across layers in multi-layer networks.

7.3.2 Eigenvector centrality in supra-graph

Eigenvector centrality describes the impact of a node on the network's global structure, and is defined by the dominant eigenvector of the graph adjacency matrix. Eigenvector centrality is widely used in many applications. For example, it is closely related to hubness and authority centrality used in the hyperlink-induced topic search (HITS) algorithm [121]. Since computing the dominant eigenvalue and eigenvector can be computed in a distributed

setting, eigenvector centrality is often preferable to other types of global centralities such as betweenness [87], [218].

The simplest way to generalize the concept of eigenvector centrality for multi-layer networks is to use network aggregation and apply single-layer based methods [211]. However, as shown in Figure 7.1, network aggregation oversimplifies the multi-layer network. Therefore, we consider the supra-graph representation $\mathcal{G}_M$ of a multi-layer network with L layers and N nodes. The supra-adjacency matrix $\mathbf{A}_M \in \mathbb{R}^{NL\times NL}$ of $\mathcal{G}_M$ can be separated into two parts: the intra-layer component $\mathbf{A}_M^L$ and the inter-layer component $\mathbf{A}_M^I$. That is,

$$\mathbf{A}_M = \mathbf{A}_M^L + \mathbf{A}_M^I,\ \mathbf{A}_M^L = \mathrm{diag}(\{\mathbf{A}^{(\alpha)}\}_{\alpha=1}^L) \tag{7.8}$$

where $\mathrm{diag}(\{\mathbf{A}^{(\alpha)}\}_{\alpha=1}^L)$ denotes a block-diagonal matrix with diagonal elements $\mathbf{A}^{(\alpha)}$ for $\alpha \in [L]$, and recall that $\mathbf{A}^{(\alpha)}$ is the graph adjacency matrix on layer α. The inter-layer supra-adjacency matrix $\mathbf{A}_M^I$ defines the inter-layer connectivity between every two layers. If the inter-layer connectivity is identical for all nodes [212], then $\mathbf{A}_M^I = \mathbf{A}^I \otimes \mathbf{I}_N$, where $\mathbf{A}^I \in \mathbb{R}^{L\times L}$ is an inter-layer adjacency matrix whose elements represent the strength of the connection between every pair of layers. For example, in a temporal network, if layers are connected at consecutive time steps, then the inter-layer supra-adjacency matrix becomes

$$\mathbf{A}_M^I = \mathbf{A}^I \otimes \mathbf{I}_N,\ \mathbf{A}^I = \begin{bmatrix} 0 & 1 & 0 & \cdots \\ 1 & 0 & 1 & \ddots \\ 0 & 1 & 0 & \ddots \\ \vdots & \ddots & \ddots & \ddots \end{bmatrix}. \tag{7.9}$$

Here $\mathbf{A}^I$ models an undirected chain network, in which each node is adjacent to its nearest neighbors. It is worth mentioning that the decomposition of the supra-adjacency matrix in (7.8) facilitates exploring the spectral properties of multi-layer networks [212].

The eigenvector centrality $\mathbf{v}_M \in \mathbb{R}^{NL}$ of the supra-adjacency matrix $\mathbf{A}_M$ can then be

defined as the solution to the following eigenvalue problem

$$\mathbf{A}_M \mathbf{v}_M = \lambda_{\max} \mathbf{v}_M, \tag{7.10}$$

where $\lambda_{\max}$ denotes the largest positive eigenvalue of $\mathbf{A}_M$. The entries of $\mathbf{v}_M$ give the centralities of each node-layer pair. It is convenient to map the eigenvector centralities $\mathbf{v}_M$ to an $N \times L$ matrix

$$V_{i\alpha} = v_{N(\alpha-1)+i},\ i \in [N], \alpha \in [L], \tag{7.11}$$

where $V_{i\alpha}$ corresponds to the joint centrality of the node-layer pair (i, α). Based on (7.11), we introduce the marginal node centrality $\hat{v}_i$ and the marginal layer centrality $\tilde{v}_\alpha$ [218]

$$\hat{v}_i = \sum_{\alpha=1}^{L} V_{i\alpha},\ i \in [N], \quad \tilde{v}_\alpha = \sum_{i=1}^{N} V_{i\alpha},\ \alpha \in [L]. \tag{7.12}$$

Similar to the supra-adjacency matrix in (7.8), we can define the decomposition of the supra-Laplacian matrix, $\mathbf{L}_M = \mathbf{L}_M^L + \mathbf{L}_M^I$, where $\mathbf{L}_M^L = \mathrm{diag}(\mathbf{A}_M^L \mathbf{1}) - \mathbf{A}_M^L$, and $\mathbf{L}_M^I = \mathrm{diag}(\mathbf{A}_M^I \mathbf{1}) - \mathbf{A}_M^I$. The decomposition of the supra-Laplacian matrix corresponds to a diffusion process over nodes of the network. Specifically, the nodal dynamics follows the differential equation [212]

$$\dot{x}_{i\alpha} = \sum_{j=1}^{N} w_{ij}^{(\alpha)} (x_{j\alpha} - x_{i\alpha}) + \sum_{\beta=1}^{L} u_{\alpha\beta} (x_{i\beta} - x_{i\alpha}) \tag{7.13}$$

for any $i \in [N]$ and $\alpha \in [L]$, where $x_{i\alpha}$ denotes the state of node i at layer α, $w_{ij}^{(\alpha)}$ is the (i, j)th entry of the graph adjacency matrix $\mathbf{A}^{(\alpha)}$ at layer α, and $u_{\alpha\beta}$ is the inter-layer coupling constant, namely, the (α, β)-th entry of the inter-layer adjacency matrix $\mathbf{A}^I$. The

discretized matrix form of the diffusion equation (7.13) yields

$$\dot{\mathbf{x}} = -(\mathbf{L}_M^L + \mathbf{L}_M^I)\mathbf{x} = -\mathbf{L}_M\mathbf{x}. \tag{7.14}$$

Here the second smallest eigenvalue of $\mathbf{L}_M$ (also known as algebraic connectivity [54]) governs the convergence properties of the diffusion process.

7.3.3 Nodal centrality via tensor decomposition

A fourth-order tensor was introduced in Section 7.2.1.2 to represent a multi-layer network. Tensor decomposition is an effective tool for multiarray data analysis, and mono-layer centrality measures can be extended in order to identify key nodes in multi-layer networks. It has been shown in [227] that the principal singular vectors obtained from the CANDECOMP/PARAFAC tensor decomposition [124] can provide hub and authority scores of all nodes in a multi-layer network.

The fourth-order adjacency tensor $\mathbf{M} \in \mathbb{R}^{N\times L\times N\times L}$ of a multi-layer network can be decomposed into a sum of rank-one tensors [124],

$$\mathbf{M} = \sum_{i=1}^{R} \sigma_i \mathbf{a}_i \circ \mathbf{b}_i \circ \mathbf{c}_i \circ \mathbf{d}_i, \tag{7.15}$$

where $\{\sigma_i\}_{i=1}^R$ are singular values of $\mathbf{M}$ sorted in a descending order, $\mathbf{a}_i \in \mathbb{R}^N$, $\mathbf{b}_i \in \mathbb{R}^L$, $\mathbf{c}_i \in \mathbb{R}^N$, and $\mathbf{d}_i \in \mathbb{R}^L$ are singular vectors corresponding to the singular value σ_i, and R is the rank of $\mathbf{M}$. Considering the principal quadruplet $\{\mathbf{a}_1, \mathbf{b}_1, \mathbf{c}_1, \mathbf{d}_1\}$ in (7.15), the entries of $\mathbf{a}_1$ and $\mathbf{c}_1$ correspond to hub and authority scores of all nodes, while the entries of $\mathbf{b}_1$ and $\mathbf{d}_1$ give hub and authority scores of all layers. Note that if $L = 1$, then the four-order adjacency tensor reduces to the second-order adjacency matrix, and the entries of $\mathbf{a}_1$ and $\mathbf{c}_1$ give the conventional hub and authority scores of nodes in a single-layer network. Given the hub and authority scores [227], one can further generalize HITS of multi-layer networks

[125].

Based on (7.15), the importance of a node-layer pair (i, α) can be evaluated as

$$H_{i,\alpha} = |a_{1,i} b_{2,\alpha}| + |c_{1,i} d_{1,\alpha}|, \tag{7.16}$$

where $a_{1,i}$ denotes the ith entry of the vector $\mathbf{a}_1$. The nodal importance measure defined in (7.16) is called EDCPTD (Essential nodes Determining based on CP Tensor Decomposition) centrality [227]. Given the joint centrality of the node-layer pair in (7.16), we can then define the marginal node centrality and the marginal layer centrality following (7.12).

In addition to the hubness and authority centrality, other generalized centrality measures such as clustering coefficient, modularity, and random walk centrality can also be defined using the tensor representation; see [67] for details. We illustrate the application of centrality to identify genes that play significant roles in an allelically biased biological process.

7.4 Clustering and community detection in multi-layer networks

Discovering meso-scale structures, such as communities, in complex networks is a wide field of study [79]. These communities are generally described as a subset of nodes in the graph that are more densely connected than other nodes in the network. This is sometimes referred to in sociology as homophily [152]. Detecting these communities in a single-layer network has and continues to be an active research field. Furthermore, research in community detection for the more general multi-layer case has become increasingly prevalent in the past decade.

Community detection in social networks facilitates the interpretation of the overall structure of the network. Generally, we expect to see communities in social networks

that strongly relate different agents to one another, such as common activities, interests, or memberships to organizations. For instance, students that attend the same university, play the same sport, or like the same music are more likely to be connected in a particular link type. The concept of communities becomes more complex when multiple layers are introduced (see Figure 7.2); communities that develop in one type of interaction may not be present in another, or may be subsumed by a larger, more prevalent super-community. It is also possible that the community structure in each layer exhibits different homophilic clusters that do not correlate across layers. Depending on the application, the main goal of analysis might be to utilize multiple layers to find communities that may have not been obvious in a single-layer slice of the network. In other applications, we may be interested in the similarities and dissimilarities of the community structure for each layer, which necessitates different approaches. Community detection in temporal networks deserves its own special treatment, as we often make temporal locality assumptions that allow for a more focused analysis. References for community detection in dynamic networks include [20], [24].

We will briefly cover three types of methods for multi-layer community detection: score-based methods, model-based methods, and aggregation methods. This list is by no means exhaustive, nor are the types of methods meant to be canonical. Rather, we find them to be useful descriptors which tend to have reasonably well understood advantages and disadvantages in the multi-layer setting. The main goal of aggregation methods is to find shared community structure by combining each slice of the multi-layer graph into a single-layer network. Score based methods rely on maximizing fitness functions based on an appropriate null model in order to detect communities. Finally, model-based methods rely on statistical models and formal inference to discover latent structure. These three types of methods are not necessarily disjoint from one another; for instance, many in the statistics community study models that are very similar to the null models that are used in score-based methods.

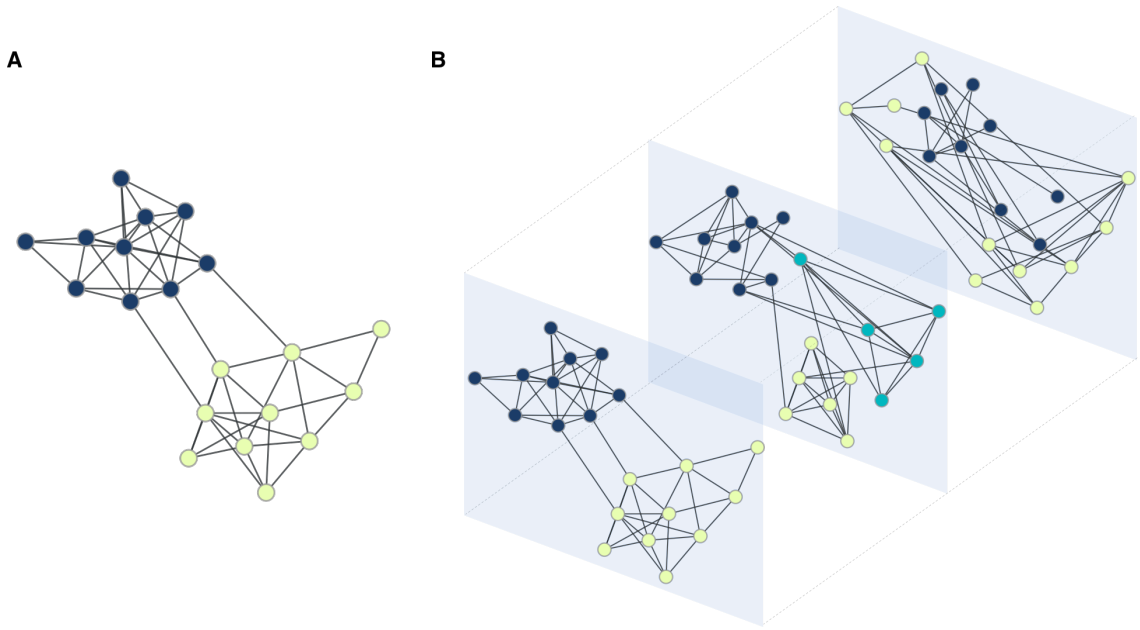


Figure 7.2: Examples of community detection with 20 nodes. (A) shows single-layer community detection, where the community structure captures homophily among the nodes. (B) displays community detection in a multi-layer setting, where more complex situations can occur. The middle layer has a sub-community in one of the larger communities displayed in the front layer, while the back layer has different latent structure altogether.

7.4.1 Score-based methods

In the single-layer setting, score based methods operate by optimizing a fitness function. Perhaps the most popular method in this category is modularity maximization. Modularity for a single-layer network is defined as follows:

$$Q = \frac{1}{2m} \sum_{i,j \in \mathcal{E}} \left(\mathbf{A}_{ij} - \frac{k_i k_j}{2m} \right) \delta(c_i, c_j), \tag{7.17}$$

where m is the number of edges in the network, $\mathbf{A}$ is the adjacency matrix, k_i is the degree of node i, and c_i is the community label of node i. Modularity is qualitatively a comparison with the structure of the network to a random null model in which every edge between every node is equally likely [163]. Extensions of modularity, such as multi-resolution variants [232] have been proposed in the literature.

In order to find community structure, we perform a maximization of Q over the com-

munity assignments c_i. Modularity maximization is typically performed using the Louvain algorithm or its appropriate variants. Modifying these fitness functions for a multi-layer setting can be done by appropriately defining a null model [162], which takes into account intra-layer connections and inter-layer connections accordingly, in which case we have the following multi-layer modularity:

$$\frac{1}{2\mu}\sum_{i,j\in\mathcal{E}_M,\alpha,\beta\in[L]}\left[\left(\mathbf{A}_{ij}^{(\alpha)}-\gamma_\alpha\frac{k_i^{(\alpha)}k_j^{(\alpha)}}{2m_\alpha}\right)\delta(\alpha,\beta)+\delta(i,j)C_{j\alpha\beta}\right]\delta(c_{i\alpha},c_{j\beta}). \tag{7.18}$$

This model only takes into account inter-layer connections between the same nodes, and their strengths are represented by $C_{j\alpha\beta}$. Further, each node in each layer has a different community label $c_{i\alpha}$, and μ is an appropriate normalization term; see [162] for details.

Another score based method that allows for extensions to any single-layer fitness function involves Pareto optimality [181]. In this case, we assume that each node has one community label for every layer, so that $c_i = c_{i\alpha} = c_{i\beta}$. In this method, we define a fitness function for each layer, $f_1(\mathbf{c}), f_2(\mathbf{c}), \ldots, f_L(\mathbf{c})$, that we wish to jointly minimize. We could, for instance, choose (negative) modularity on each layer for our cost function. Alternatively, we could choose a similar cost function that arises when attempting to reduce the inter-community connections - this is called spectral clustering [207]. Once we define these functions, we attempt to solve the multi-objective optimization problem:

$$\hat{\mathbf{c}} = \arg\min_{\mathbf{c}}[f_1(\mathbf{c}), f_2(\mathbf{c}), \ldots, f_L(\mathbf{c})]. \tag{7.19}$$

The objective is to find the Pareto optimal solution or solutions. A non-Pareto optimal solution $\mathbf{c}$ is a solution such that there exists at least one other solution $\mathbf{d}$ such that, for all $\alpha \in [L]$, $f_\alpha(\mathbf{d}) \leq f_\alpha(\mathbf{c})$, and $f_\beta(\mathbf{d}) < f_\beta(\mathbf{c})$ for at least one layer $\beta \in [L]$. The set of Pareto points is the set of solutions for which the above is not true. The special case of using the spectral clustering score function has been explored in [181]. Other methods

for finding approximate Pareto optimal points include evolutionary algorithms, and Pareto methods have been used in anomaly detection [109] and image retrieval [108].

7.4.2 Model-based methods

Model-based methods assume a specified statistical model for the network, and then use statistical methods for inference in order to discover the latent community structure. These models are often variants of a ubiquitous single-layer model called the stochastic block model (SBM) [107]. This model assumes that given the community structure, each edge is drawn independently as a Bernoulli random variable according to a parameter p_{ij}, where i and j are the communities of the nodes for the edge that is being drawn. [213] generalizes the SBM to have discrete types of layers and communities in each type. [64], [97], [185] explore different extensions of the single-layer SBM. The inference for these models can be quite difficult from both a computational and statistical persepective. Work to find provably computationally and statistically efficient algorithms in various model cases continues to be an active field of research in the multi-layer setting.

7.4.3 Aggregation-based methods

Aggregation based methods attempt to find a single-layer network that holds information about the communities in the multi-layer network, and then utilize single-layer community detection methods. Examples include [27], [66], [219]. A recent paper [47] utilizes spectral clustering and convex layer aggregation to perform community detection. Specifically, given a layer weight vector $w \in \mathcal{W}_L$, where $\mathcal{W}_L = \{w : w_\alpha \geq 0, \sum_{\alpha=1}^{L} w_\alpha = 1\}$, and a supra-adjacency matrix as defined in 7.2.1.1, we define the weighted adjacency layer matrix and associated Laplacian as:

$$\mathbf{A}_w = \sum_{\alpha=1}^{L} w_\alpha \mathbf{A}^{(\alpha)}, \quad \mathbf{L}_w = \sum_{\alpha=1}^{L} w_\alpha \mathbf{L}^{(\alpha)}. \tag{7.20}$$

The authors in [47] discuss theoretical guarantees and limits of this method under different models, and also provide a framework for model selection.

7.5 Applications

In the following sections, we will demonstrate the utility of multi-layer network methods on real data. First, we will examine a biological multi-layer network to uncover topological roles in gene contact networks. We will also describe a Twitter dataset, and use the dynamic interaction graph estimation technique discussed in Chapter 5.

7.5.1 Identifying genes encoding allelic differences in gene contact networks

Allelic differences between two homologous chromosomes (corresponding to paternal and maternal alleles) can affect the propensity of inheritance in humans [132]. Therefore, it is important to discriminate the contribution of the paternal (Pat) and maternal (Mat) genomes to the functional diploid human nucleome. In what follows, we perform multi-layer network analysis to understand allelic differences at the gene level.

Genome technologies like genome-wide chromosome conformation capture (Hi-C) can be used to measure the genomic structure [33], [45], [134]. Here, Hi-C evaluates long-range interactions between pairs of segments delimited by specific cutting sites using spatially constrained ligation [134]. As a result, we obtain a fragment read table, each row of which indicates a ligated pair of fragments from the genome, with the coordinates of both fragments. Based on that, we can construct 2D Hi-C contact maps at gene resolution [46], [139]. We refer the reader to [46] for more details on data generation and preprocessing. From the network point of view, this leads to a sequence of inter-gene interaction networks over time (namely, cell cycle phases G1, S and G2/M) under both Pat and Mat alleles. That is, we obtain an allele-specific multi-layer network, where each cell cycle stage cor-

responds to a layer. Our goal is to identify genes that yield significant contact differences between the Pat and Mat alleles.

We adopt the overlapping degree centrality and the multiplex participation coefficient to distinguish Pat allele from Mat allele. We recall from 7.3.1 that the overlapping degree centrality allows us to identify hubs from a network, and the multiplex participation coefficient can quantify the participation of a gene to different cell cycle phases. In Figure 7.3B, we present z-scores of genes' overlapping degrees versus genes' participation coefficients. As we can see, due to allelic differences, there exist genes that play different topological roles on Pat and Mat alleles. Let z_i denote the z-score of the overlapping degree for gene i, and P_i denote its multiplex participation coefficient. We distinguish hubs (interacting with many genes) from regular nodes if $z_i \geq 2$. Motivated by [22], we call genes focused if contacts associated with them were concentrated on a single cell cycle phase, corresponding to $P_1 < 1/3$, and multiplex if their connected edges were homogeneously distributed across different cell cycle phases, corresponding to $P_1 > 2/3$. In the considered experiment, genes LEPREL1 and CTSS are hubs at Pat allele, while they become regular nodes at Mat allele. And gene KBTBD2 is a multiplex node at Pat allele, but it becomes a focused node at Mat allele. We show the allelic differences in terms of contact differences of genes, e.g., LEPREL1 in Figure 7.3C.

7.5.2 Application to Twitter dataset

A Twitter dataset was extracted from a large subsampled collection of tweets spanning the month of November 2015. This dataset extracted any tweet with at least one of twenty seven hashtags. These hashtags fall into two categories: hashtags that pertain to upcoming movies at the time (creedmovie, gooddinosaur, spotlightmovie, etc.), and 5 pertinent political hashtags (bernie2016, cruz2016, cruzcrew, hillary2016, trump2016). These tweets were then aggregated on a 24-hour basis, and natural language features were extracted. These features were then used to calculate the marginal DI and ADI between each pair of

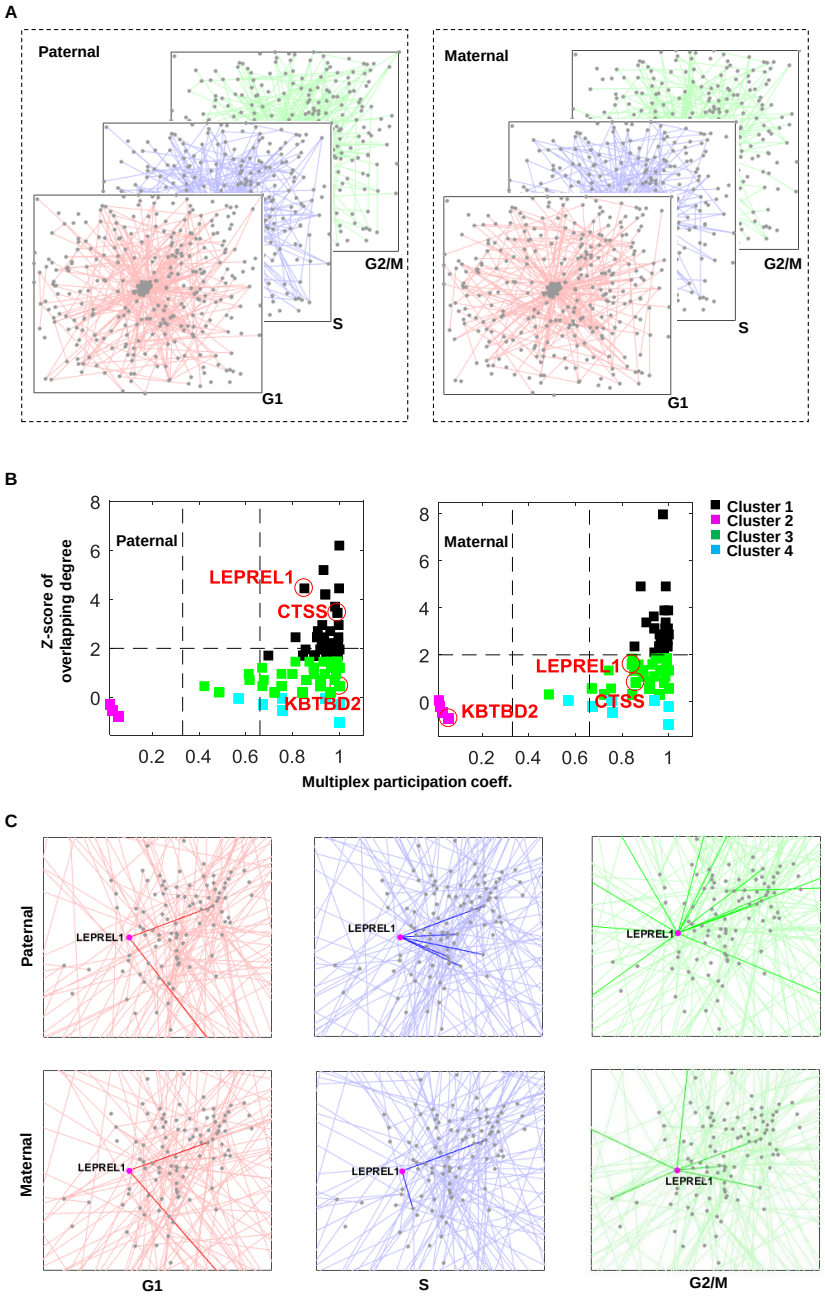


Figure 7.3: (A) Temporal network with implicit inter-layer connections between genes at one cell cycle phase and their counterparts at other cell cycle phases. (B) Overlapping degree versus multiplex participation coefficient: genes are divided into 4 clusters via K-means. (C) Representative gene LEPREL1 with allelic differences in the topological structure.

hashtags.

Figure 7.4 shows a graphical representation of the average DI over time between each

node, with the two hashtag groups differentiated by color. Note that the political hashtags exhibit a much stronger mutual influence than the movie hashtags did.

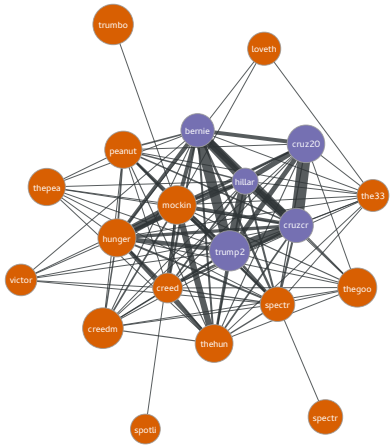


Figure 7.4: Network of average (symmetrized) DI. The political hashtags (purple) have a much stronger relationship with each other than the movie hashtags (orange), creating a strong clique in the graph.

Figure 7.5 shows the result of the same centrality analysis performed in Section 7.3, applied to the dynamic hashtag relevance network. From the plot, we can surmise that the political hashtags have stronger relationships that are sustained over time when compared to the movie hashtags.

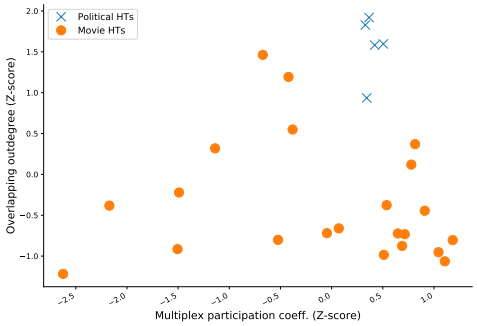


Figure 7.5: Overlapping degree versus multiplex participation coefficient for outdegree of ADI. The political hashtags have larger Z-scores in both the participation coefficient and overlapping degree. This suggests that they exert stronger relationships at each timestep than the movie hashtags and sustain these relationships consistently over the time horizon.

Figure 7.6 shows the total degree (outdegree minus indegree) of ADI over time for the

political hashtags. Note the wide variety of dyadic behavior among the political hashtags, with the trump2016 hashtag being the largest sink of influence.

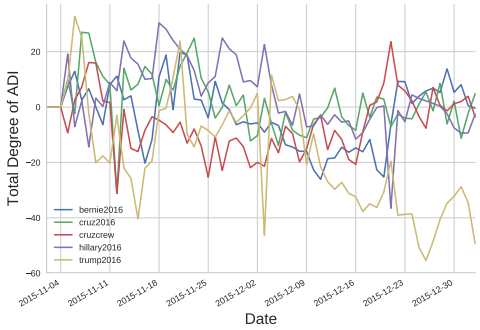


Figure 7.6: Total degree of ADI over time. Due to the use of ADI, we can see both the overall trends of influence along with occasional spikes due to a particular series of tweets. In this dataset, the hashtag trump2016 is a trending sink of relative influence coupled with negative spikes of relative influence on occasion. This would suggest that trump2016 is strongly correlated to the rest of network, and in particular that tweets with trump2016 are closely related with other tweets from the previous day.

7.6 Conclusions

Social network datasets are ubiquitous in today's data landscape. We have discussed in this chapter some methods for dealing with multi-layer social networks, and some of the difference when analyzing a single-layer network. After defining some formulations and representations of multi-layer networks and some common examples that one might encounter, we covered some measures of multi-layer network centrality. We also discussed a few types of methods for multi-layer community detection, including briefly discussing some benefits and drawbacks of each type. We finally covered the problem of multi-layer interaction graph estimation, with special focus on dynamic graphs. We then applied a few of the techniques to two datasets, a biological dataset, and finally a social network dataset.

As the field of multi-layer networks continue to grow, we expect that the methods that we have summarized here will continue to evolve and improve, and that the framework of

multi-layer graphs will become even more useful to the field of social network analysis in the years to come.

CHAPTER 8

Multi-objective Community Detection for Large Multi-layer Social Networks

Social networks often encode community structure using multiple distinct types of links between nodes. In this chapter we introduce a novel method to extract information from such multi-layer networks, where each type of link forms its own layer. Using the concept of Pareto optimality, community detection in this multi-layer setting is formulated as a multiple criterion optimization problem. We propose an algorithm for finding an approximate Pareto frontier containing a family of solutions. The power of this approach is demonstrated on a Twitter dataset, where the nodes are hashtags and the layers correspond to (1) behavioral edges connecting pairs of hashtags whose temporal profiles are similar and (2) relational edges connecting pairs of hashtags that appear in the same tweets.

Symbol	Description
$G = (\mathcal{V}, \mathcal{E})$	Multi-layer network
$\mathcal{V}$	Vertex set
$\mathcal{E} = (\mathcal{E}_1, \ldots, \mathcal{E}_M)$	Tuple of edge sets for each layer
A^k	Adjacency matrix for kth layer
D^k	Degree matrix for kth layer
$f_i(\cdot)$	Layer-specific cost function

Table 8.1: Glossary of commonly used symbols.

8.1 Introduction

Social networks have become rich sources of data for network analysis, where objectives might include community detection, edge prediction, node behavior prediction, and model inference. However, it has become increasingly difficult to extract meaningful information from these networks due to the explosion in both the volume of data collected and the diversity of available data types. In this chapter we focus on addressing the latter problem for the task of community detection; specifically, we consider networks containing multiple layers of interactions between nodes.

For many social network applications, measures of association between pairs of nodes may be available along multiple dimensions. For example, graph edges may be observed directly in the data, or they may be inferred from actions of the agents in the network. We make the distinction between *relational* links that are observed explicitly and *behavioral* links that are inferred from ancillary data describing node behavior. Examples of relational links between users might include observed interactions over a period of time, mutually established friendship connections, or email sender-reciever relationships. Likewise, behavioral links might be drawn between users who post items with similar semantic content, like the same bands or movies, or exhibit correlated activity over time. Further, it is possible to have multiple types of relational and behavioral links; for instance, there could be both a professional and personal social network over the same set of users. Networks with multiple distinct edge types have been called multi-layer [146], multi-level [210], multi-relational, or multiplex [120] networks.

In a multi-layer network, each layer may have a unique topology. The simplest way to apply existing network analysis algorithms (which generally assume homogeneous edges) is to "flatten" the data, i.e., to combine all the different types of links into a single-layer network. This can be accomplished in various ways, for instance, by performing a logical AND or OR on the layer-specific adjacency matrices, or by computing their weighted (and possibly thresholded) average. However, this approach has many hidden pitfalls; for exam-

ple, if one of the layers is noisier than the others then it probably should not receive equal consideration when attempting community detection.

A better strategy, we argue, is to directly analyze the multi-layer networks without flattening. To show how this can be done, we propose a new method of community detection for multi-layer networks. Our approach employs multi-objective optimization, taking into account multiple layers of network structure, which is then used to find a community partition. We show that this algorithm can provide significantly better community detection than that obtained by standard single-layer techniques.

The chapter proceeds as follows. In Section 8.2 we define multi-layer networks. In Section 8.4 a Pareto optimality approach to multi-layer community detection is proposed, in Section 8.5 we apply the proposed approach to simulated data, and in Section 8.6 we apply the proposed approach to three datasets. Finally, we discuss related work in Section 8.7 and give concluding remarks in Section 8.8.

8.2 Multi-layer networks

A multi-layer network $G = (\mathcal{V}, \mathcal{E})$ consists of vertices $\mathcal{V} = \{v_1, \ldots, v_p\}$, common to all layers, and edges $\mathcal{E} = (\mathcal{E}_1, \ldots, \mathcal{E}_M)$ in M layers, where $\mathcal{E}_k$ is the edge set for layer k, and $\mathcal{E}_k = \{e^k_{v_i v_j}; \quad v_i, v_j \in V\}$. Each edge is undirected, though extensions to the directed case are not difficult. The multi-layer degree of a node i is $d^i \in \mathbb{R}^M$, with each entry $[d^i]_k$ being the degree of node i on layer k.

The adjacency matrix and degree matrix are defined as usual for each layer:

$$[[A^k]]_{ij} = e^k_{v_i v_j} \tag{8.1}$$

$$D^k = \text{diag}([d^1]_k, [d^2]_k, \ldots, [d^p]_k) \tag{8.2}$$

Note that D^k is simply a $p \times p$ diagonal matrix with the layer-specific node degrees on the diagonal.

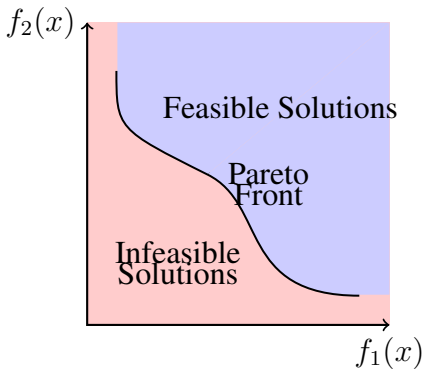


Figure 8.1: An example of a Pareto front for two objective functions. An important aspect of this example is that the Pareto front is non convex; therefore, a weighted linearization search strategy will not explore the entire front.

8.3 Multi-objective optimization

Multi-objective optimization is a general framework for solving optimization problems when there is more than one objective function to be minimized. Often, these objective functions can contradict each other, so that their individual minimizations lead to solutions that are far away. Thus, the first step in this type of optimization problem is to define what an optimal solution is; we do this with the concept of non-dominated solutions.

Formally, we define the following multi-objective optimization problem:

$$\hat{x} = \underset{x}{\operatorname{argmin}}[f_1(x), f_2(x), \ldots, f_n(x)] \, . \tag{8.3}$$

We are interested in solutions that are called *non-dominated* solutions. A solution y^* is dominated by the solution x^* if for all i between 1 and n, $f_i(x^*) \leq f_i(y^*)$, and for at least one j between 1 and n, $f_j(x^*) \leq f_j(y^*)$. We call the set of feasible solutions that are not dominated by any other solutions the first Pareto front. The Pareto front contains solutions that are at least as good as all the other solutions, and in at least one objective function they do better. In our MOO framework, we say that solutions in the first Pareto front are optimal. A visualization of a Pareto front for $n = 2$ is shown in Figure 8.1.

There are a variety of approaches to obtaining solutions for (8.3). Perhaps the most

basic is to linearize the problem and solve the corresponding scalar optimization

$$\hat{x} = \underset{x}{\operatorname{argmin}} \sum_{i=1}^{n} \alpha_i f_i(x) \tag{8.4}$$

for some set of weights $\{\alpha_i\}$. This approach is advantageous because it distills the problem down to a single optimization problem for which there are many standard methods. There are two main disadvantages, however. First, it is up to the user to choose the weights α_i in advance, or through trial and error, which can be difficult in practice. The second and perhaps more pervasive problem is that this procedure will only recover the Pareto front if the solution space and all objective functions are convex [86]. When convexity does not hold, this procedure can only find a subset of the feasible Pareto solutions. Two approaches that avoid this problem are ϵ methods and goal attainment, although both are very sensitive to parameter settings. The most popular methods for finding an approximate Pareto front are evolutionary algorithms. These algorithms use heuristic concepts from biology, along with some parameters and randomly selected seed cases to attempt to find solutions on the Pareto front by propagation. More details can be found in [40], [68] and references therein.

Another strategy is to avoid the heavy computational and analytical burden of computing an exact Pareto front. If it is possible to obtain a sample of solutions that are likely to be on or near the front, we can sort these points for non-domination. In this way, we can filter a large set of solutions to find the optimal ones that are worth further consideration. In the next section we show how, given two solutions that are assumed to be approximately Pareto optimal, a greedy, recursive algorithm can be used to find more approximately non-dominated points.

8.4 Community detection via multiobjective optimization

Many existing community detection algorithms involve optimization [78]. Methods that fall into this category include spectral algorithms, modularity methods, and methods that

rely on statistical inference, particularly those that try to maximize a likelihood function. It seems natural that a multi-layer generalization of such algorithms might somehow combine the optimization objective functions as applied to each individual layer; this is the basis of multi-objective optimization.

More formally, let community structure in a network be described by a node partition C, where $C(i) = k$ means that node i is in part k. Single-objective optimization methods of community detection seek to find the partition $\mathrm{argmin}_C f(C)$ that minimizes an objective function f (which depends internally on the network structure). In the following we consider the two community case; more communities can be found by a recursive use of the algorithm.

Now consider a two-layer network, and let f_1 and f_2 be objective functions for the two layers. One obvious way of combining the layers would be to minimize the linear combination $\alpha f_1(C)+(1-\alpha)f_2(C)$ over C, where $\alpha \in [0, 1]$. However, linear combination may be restrictive, especially when the objective functions are complex. A more general approach is instead to seek the Pareto optimal solutions of the multi-objective minimization problem:

$$\hat{C} = \mathrm{argmin}_C[f_1(C), f_2(C)] \; . \tag{8.5}$$

In order to find approximate Pareto-optimal solutions, we utilise the Kernighan-Lin node swapping technique [118]. The objective is to find solutions that are approximately Pareto optimal. Figure 8.2 shows the proposed algorithm.

For community detection, the objective is to minimize the ratio-cut f_k for each layer $k = 1, 2$:

$$f_k(C) = \frac{1}{2}\sum_{k=1}^{2} \frac{\mathrm{cut}(C)}{|\{i : C(i) = k\}|} \tag{8.6}$$

$$\mathrm{cut}(C) = \sum_{C(i)=1, C(j)=2} [A^k]_{ij} \tag{8.7}$$

Input: f_1, f_2
  Obtain optimum solutions C_1^*, C_2^* for each layer
  Initialize $C = C_1^*$
  repeat
    for $i : C(i) \neq C_2^*(i)$ **do**
      $C^{new} \leftarrow C$, $C^{new}(i) \leftarrow C_2^*(i)$
      $\text{cost}(i) \leftarrow f_2(C^{new}) - f_2(C)$
    end for
    $i^* \leftarrow \text{argmin}_i \text{cost}(i)$
    $C(i^*) \leftarrow C_2^*(i^*)$
  until $C = C_2^*$
Output: non-dominated solution values taken by C

Figure 8.2: Proposed algorithm for Pareto front identification.

A relaxed version of this objective function can be solved by performing an eigendecomposition on the Laplacian $L_i = D_i - A_i$. More details can be found in [143].

8.5 Simulation

We tested the algorithm shown in Figure 8.2 on synthetic multi-level networks. For our experiments, we used an unweighted network of 500 nodes, whose average degree was 50. The first layer was constructed using an Erdös-Rényi model with each node having an average degree of 50. We then changed a variable percentage of the edges in that layer to create the second layer, and ran the algorithm to construct an approximate Pareto front. Figure 8.3 shows some example results for differing levels of variation between layers.

Changing the variation between layers changes the nature of the solution path that is tested, as well as the resulting non-dominated set. Layers that were more similar actually were able to do better than their initializations; the Pareto front in these cases does not include the points that we assumed to be approximately optimal. As the layers become dissimilar, we are not able to improve as much on the starting points; at 80% dissimilarity almost every solution explored was part of the non-dominating set. This implies that with almost every swap, the tradeoffs to be had could almost never do better in both cut-sizes.

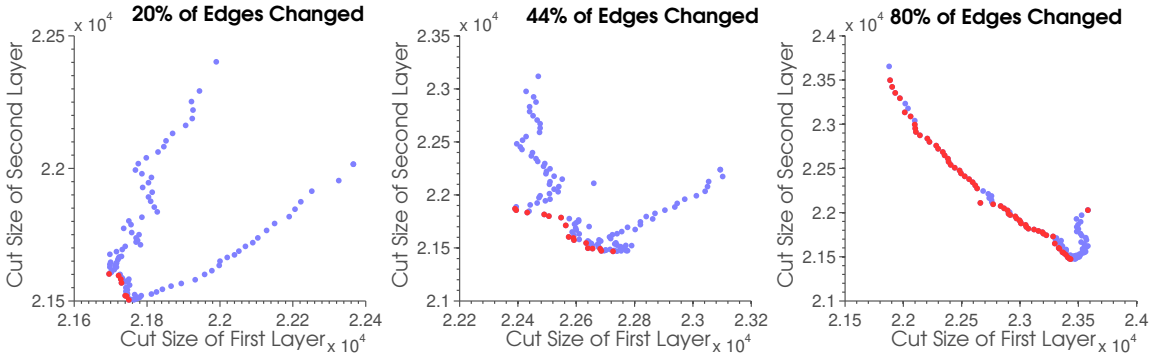


Figure 8.3: Pareto fronts for different levels of similarity. The greedy path between the spectral solutions is shown in blue; those points that are weakly non-dominated, and thus make up the approximate Pareto front, are shown in red.

Moreover, as the layer become more dissimilar, the overall cut-size increases.

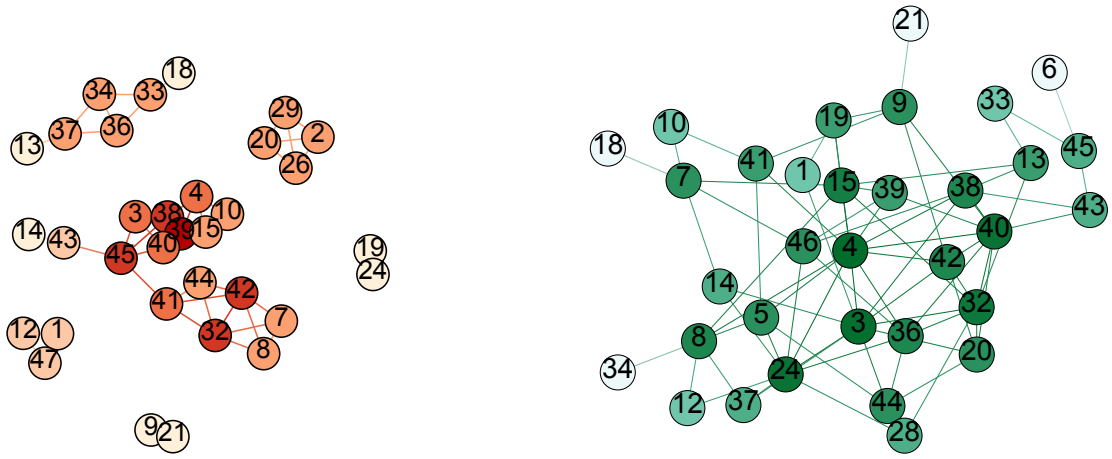
8.6 Experiments

We apply this method to three datasets. The first two are Twitter datasets, while the final dataset is the Enron email dataset.

8.6.1 Political Twitter dataset

The proposed algorithm was applied to a month of data from Twitter. A two-layer network on hashtags was developed using tweets from October 2012. The data was obtained from the Twitter stream API at gardenhose level access, which corresponds to 10% of all tweets over the month. A list of hashtags and the users who tweeted them was created for each day, as well as the volume (i.e., number of observed occurrences) of each hashtag per day.

Hashtags that were directly connected with the presidential election or politics were chosen out of a list of the most popular hashtags for the month, which yielded 48 hashtags. Figure 8.4 shows an example of two network layers for one day on the original set of 48 hashtags. In order to include some higher order connections, the list was expanded by including hashtags whose volume per day behaved similarly over the month as the first 48; this grew the network to 515 tags.



(a) Hashtag Volume Layer (b) Hashtag User Layer

Figure 8.4: A network visualization of two layers of the hashtag dataset for October 10th, 2012. This example shows the differing topologies generated by different links in a network. While we see some similarities—for instance, nodes 38, 39, and 32 have high degree centralities in both networks—these networks have many differences, the most obvious being that the volume layer is not even fully connected, while the user layer is fully connected and has a diameter of only 6.

Initially, the total volume of the hashtags was studied over time, and real events were compared with the profile; this is shown in Figure 8.5. Some events are correlated with volume; Hurricane Sandy falls on the two day period with the largest hashtag volume. The second presidential debate also corresponds to a spike in hashtag volume. In contrast, the first presidential debate is not an identifiable event in the volume plot.

A time series of two-layer networks was created with hashtags as the nodes. Specifically, 31 two-layer networks were created by aggregating daily Tweet data over each day in the month. The first layer linked two hashtags if any user used both the hashtags in that particular day. This layer is referred to as the hashtag user layer. The second layer linked two hashtags if they had similar volume profiles over time. Intuitively, two hashtags would have a link with each other if they were popular or unpopular at the same time. So as not to take into account too much past data, the volume correlation was calculated using a moving window of 5 days. A Pearson correlation coefficient was used to calculate the correlations in volume for each pair of hashtags; the correlations then underwent a Fisher

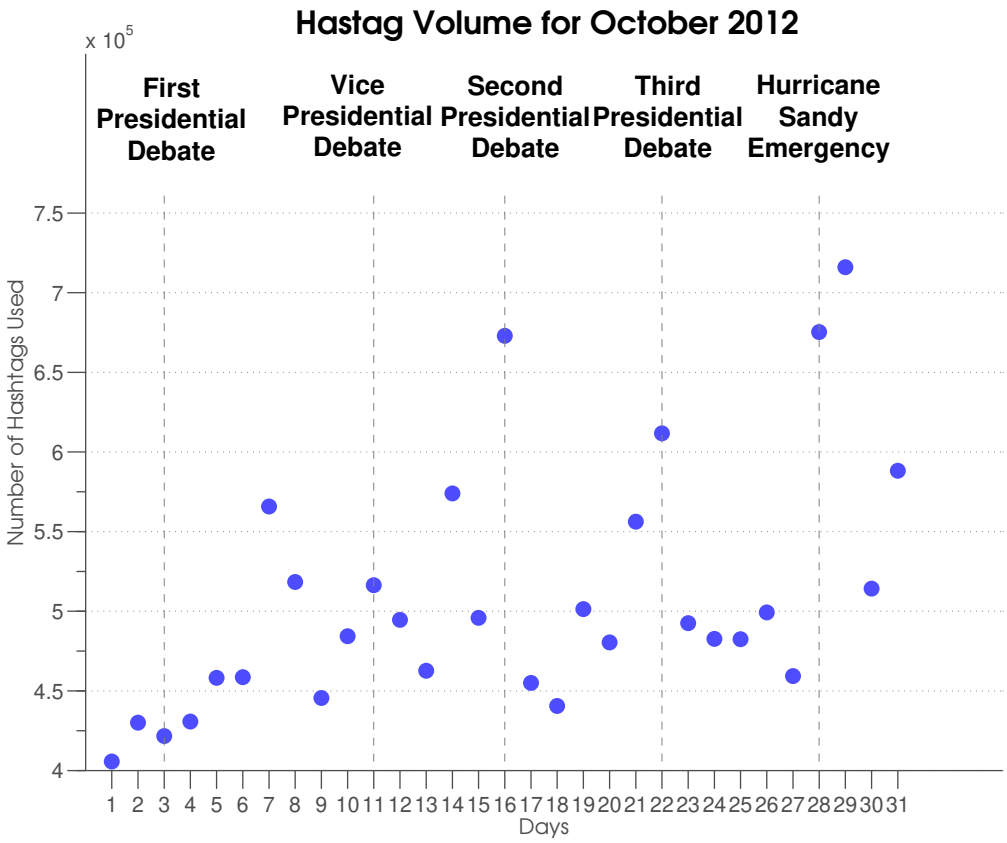


Figure 8.5: Volume of observed usage of the 515 political hashtags along with an event timeline for October 2012. Notice that while we can see that some events correlate with hashtag usage for our dataset, this is not true for all events that might be expected to affect political hashtags.

transformation and were thresholded by a value of 1.3859 which corresponds to an approximate 5% false positive rate (in the bivariate normal case) when testing for the presence of a positive correlation [77]. This layer is referred to as the hashtag volume layer. Figure 8.6 demonstrates pictorially the creation of the two layers, using a simple dataset of three hashtags.

We will show that one is able to obtain more information by the proposed Pareto multi-layer analysis methods than when the two layers are analyzed separately. To this end, the graph-cut partitions (8.7) were computed for each day. We also computed approximately Pareto-optimal partitions by combining the single-layer solutions using the algorithm in Figure 8.2, and selected a single partition by using the approximate midpoint of the Pareto

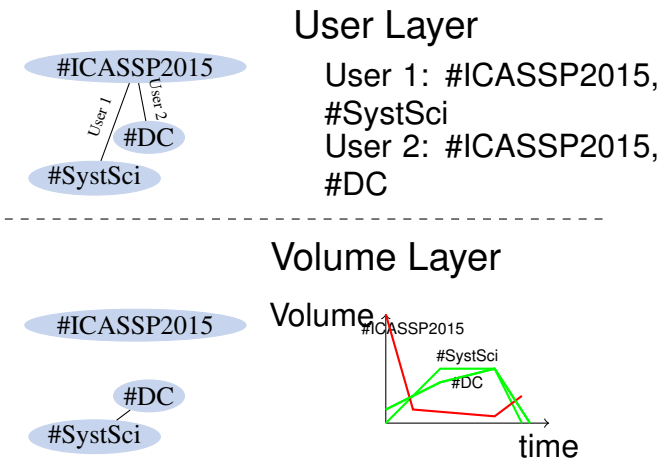
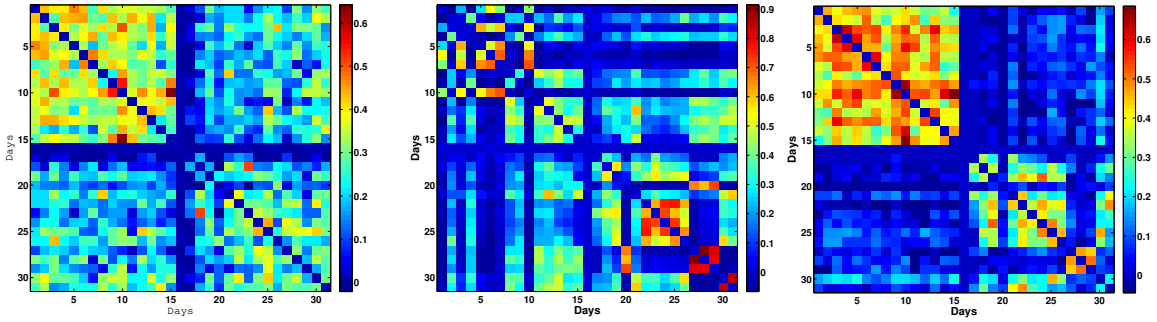


Figure 8.6: The two layers of the Twitter hashtag network are illustrated. At the top is the relational layer where a link between two hashtags indicates that at least one user used both hashtags in the same Tweet. At the bottom is the behavioral layer where a link indicates similarity in the hashtag usage volume over time.

front. The Adjusted Rand Index (ARI) [111] was then used to compare partitions on different days and see how hashtag relationships change over time. The ARI measures how similar partitions are, and can vary between -1 and 1.

Figure 8.7 shows heat maps of all the ARI indexes, both for the single layers considered separately as well as for the proposed algorithm. The hashtag user layer reflects fairly stable correlation among the two clusters until day 16, where there is a phase transition. Note that this phase transition also occurs on the volume layer heatmap. There is not much similarity between days in the user network, implying that there is not an optimal stable two cluster solution when considering the hashtag user layer alone, and it is difficult to extract real events.

In the hashtag volume layer heatmap, some community structure over days are highly correlated with each other. In particular, the days on which Hurricane Sandy occurs have communities that are highly correlated. It is also interesting to note that the communities at the end of the month are nothing like the bisected communities at the beginning, which implies considerable temporal evolution in the network. There is also more sparsity in the hashtag volume layer heatmap; consequently it may be possible to detect events more



(a) Hashtag User Network (b) Hashtag Volume Network (c) Combined Network

Figure 8.7: The more highly resolved block structure in combined network heatmap clearly indicates that the hashtag community structure remains quite stable and coherent over the first 15 days of October but then breaks up into smaller clusters of coherency over the remainder of the month. This may reflect the change of public opinions after the second Presidential debates (October 16) and the effect of Hurricane Sandy (October 28) on Twitter hashtag volume and usage.

easily using this network.

The evident block structure in the Pareto combined heatmap shows that the multi-layer algorithm eliminates similarities between the first and second half of the months. The Pareto combined solution holds attributes from both the hashtag volume layer and hashtag user layer; the structural patterns that were present in the latter half of the month of the hashtag volume network are also present in the combined solution. The first half of the month also has some self-similarity, which is seen in the hashtag user layer. However, the proposed multi-layer algorithm was able to pick out some days that were more highly correlated than in either of the single layer solutions. In particular, days 3-5 are more highly correlated in the combined solution; October 3rd was the day of the first debate. Interestingly, the layers jointly reveal correlations between days not visible in the independent single layer analyses.

8.6.2 NFL Twitter dataset

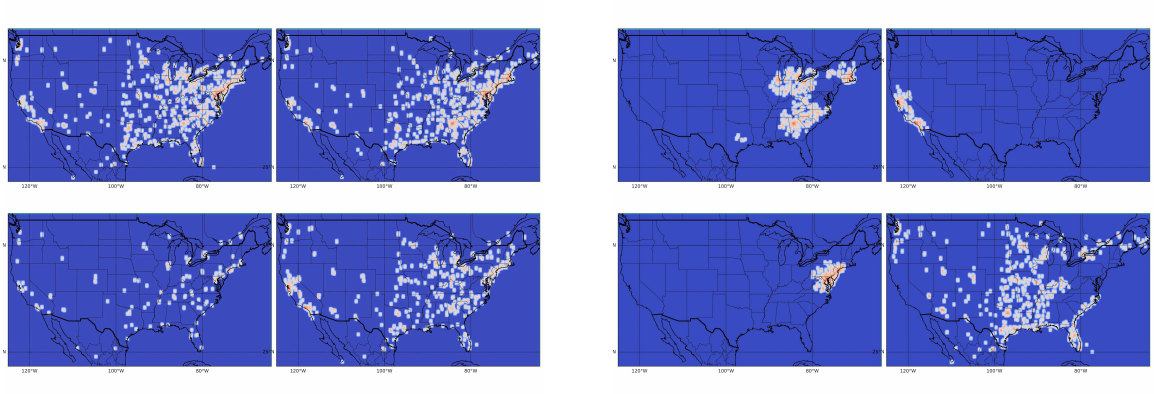
Data to create multi-layer network was obtained from the Twitter stream API at garden-hose level access during the month in January of 2013. Tweets were filtered based on the

availability of geolocation data. This geolocation information allowed for the creation of a first layer of the multi-layer network. For every pair of users i and j, they were connected ($A_{ij} = A_{ji} = 1$) if the users were closer than a certain distance threshold δ. This δ parameter changed based on the density of users and size of area that was being observed. This layer is called the coordinate network layer.

The second layer that is created utilizes hashtags to connect users. Hashtags are any words beginning with a # sign. In this layer, a user i and j are said to be connected if they use the same hashtag from a specified set of hashtags over the one month period. In order to focus on a smaller set of users, specific hashtags were chosen that applied to an event or set of events that were occurring in this period; in this case the events were the National Football League (NFL) playoffs. The dataset was created by first filtering on four of the most popular pertinent hashtags in the three month time period: #Ravens, #49ers, #Falcons, and #Patriots. These correspond to the four NFL football teams that reached the end of the NFL playoffs for that year. A two-layer network consisting of the hashtag network layer and coordinate network layer ($\delta = 50$) is analyzed. The resulting dataset contains 3456 nodes (Twitter users).

We first perform single-layer community detection. The partition resulting from spectral clustering on the hashtag network does a good job at stratifying the popular hashtags into communities, as seen in Table 8.2. Community 1 is mostly the #Ravens hashtag, while community 4 is the #49ers. Community 2 sees the #Patriots and #Falcons hashtags grouped together, while community 3 is a mixture of all four. Figure 8.8 shows a false color map of the densities of people per community. It is surprising that while there is strong community structure in this network, it is less correlated with geography than one might expect.

As expected, the coordinate network layer partitions according to high population density. Specifically, it clusters the San Francisco and LA area together, the Maryland area by itself, and the Atlanta and Boston area together. The last community seems to be a catch-all for everywhere else, i.e., those places with less density.



(a) Hashtag Network Layer

(b) Coordinate Network Layer

Figure 8.8: Density plot by community. For the hashtag network layer, the communities correspond to the numbers in Table 8.2 going from left to right and subsequently from top to bottom. Note that discussion of NFL teams are less localized than the fanbase would suggest. The communities for the coordinate network layer are highly correlated with high population density.

	Community 1	Community 2	Community 3	Community 4
#Ravens	1232	0	170	0
#49ers	57	0	155	762
#Patriots	45	291	29	10
#Falcons	49	273	29	7

Table 8.2: Hashtags per community for hashtag network layer solution.

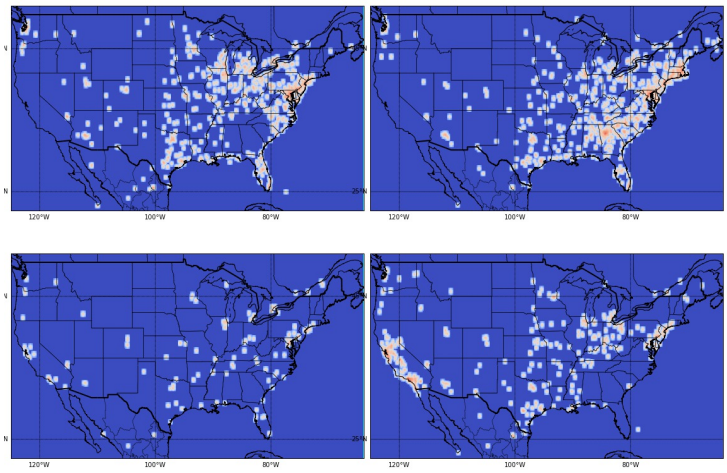


Figure 8.9: Density plot for Pareto combined community. This community partition retains attributes from both layers, while still giving a visual sense of the overall community structure. The communities in the upper left and lower right have become more concentrated about east coast and west coast, respectively. Further, the community in the upper right shows high concentration in Atlanta up to both the Maryland and Massachusetts area.

Using the described algorithm, a Pareto solution is found for the multi-layer network; the community partition is shown in Figure 8.9. The communities are more geographically localized when compared with the mention network layer solution, while still visually resembling its structure. For instance, the last community picks out the San Francisco/LA area in a single community, which the original mention network did not. Further, the second community groups the Atlanta area with the Massachusetts area, though not as well as the coordinate network layer. The Pareto community partition, however, still contains some of the interesting patterns of the hashtag network layer and is not completely given to geographic localization.

8.6.3 Enron email dataset

Figure 8.10 displays the results of running the same algorithm on the Enron email dataset. This dataset is a collection of emails that were publicly released as a result of an SEC investigation; it consists of approximately half a million messages sent to or from a set of

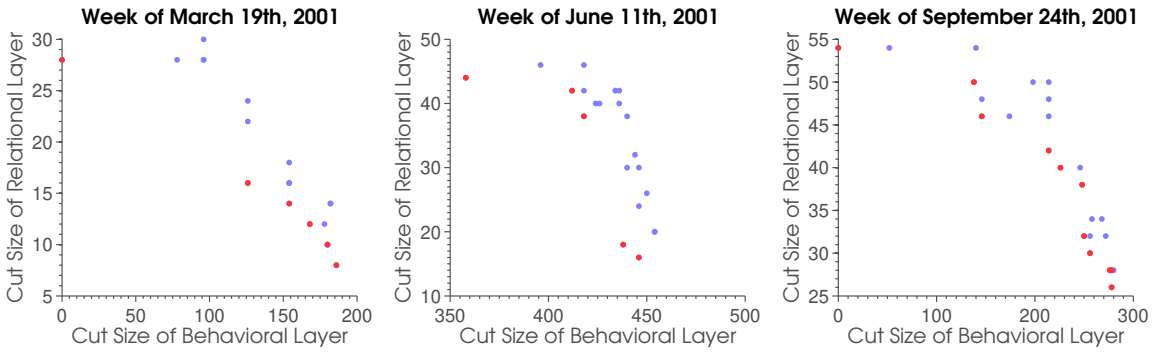


Figure 8.10: Pareto fronts from Enron email dataset. These Pareto fronts are derived from a the cut sizes of extrinsic and intrinsic layers.

150 employees. It covers a span of approximately 4 years from 1998 to 2002, though the density of emails is varied over that time. We split the emails into week-by-week periods and built a multi-layer graph for each period. The first (extrinsic) layer was created by placing an edge of weight of 1 between individuals that had correspondence over the course of that week. The second (intrinsic) layer was created by measuring semantic correlation in the email body using the TF-IDF score. These values are then thresholded to form edges, with the threshold dependent on the desired sparsity level.

Note that the Pareto fronts in Figure 8.10 do not appear to be convex. This is interesting because it implies that simply minimizing a weighted combination of objective functions would not generate the full space of potentially interesting solutions. By exploring the Pareto front we get a more nuanced view of the data. We also see a large variation in cut-sizes; in two of the cases, we see that the cut-size in one layer reaches 0, while the other cut size is still much larger. This implies that that the layers are sparse enough to be bisected almost exactly. The difference in optimal bisections between the two layers implies that that the layers have distinct properties. We also notice that the cut-size on the behavioral layer tends to be much larger than that of the relational layer; this is because the behavioral layer is less sparse.

8.7 Related work

With the advent of large data, there has been more opportunity to explore this multi-layer structure. There has been some work in the modeling and representation of multi-layer networks, and how it relates to other studied problems [67], [120]. There is a large body of work in single-layer community detection [78], consisting of many approaches. In the past few years, the multi-layer community detection literature has increased significantly. A succinct review of some of the literature on this topic can be found in [119]. Hypergraphs have been studied from a spectral perspective [153], which can be useful when dealing with a multi-layer structure. Some work in applying single-layer modularity methods to multi-layer structures is also available [19]. For more information, see [120].

Of particular interest is multi-layer community detection methods that utilize some form of tensor decomposition. As discussed in Section 7.2.1.2, the multi-layer network has a adjacency tensor representation which can be exploited in order to uncover community structure. Two examples of this approach include [65], [204] and [52]. [65], [204] use a bilinear or multilinear approach to decompose the adjacency tensor into relevant factors, including community partitions, while [52] relates a particular tensor decomposition to the concept of modularity.

Multi-objective optimization has a long history [40]. Here, we are only interested in a sorting algorithm used to find points that are possibly Pareto optimal; this is called non-dominated sorting. The method used in this chapter is part of the evolutionary algorithm described in [68]. Some interesting application work has been done using multi-objective optimization [116], including supervised and unsupervised learning.

8.8 Conclusion

Multi-level network analysis is of growing interest as we are faced with increasingly complex data. In this chapter, a method was introduced for finding communities in a multi-layer

structure; it was demonstrated on a Twitter hashtag dataset and shown to deliver results that significantly differ from single layer analysis alone. The framework described can also be applied to other single-layer algorithms for the multi-layer setting.

CHAPTER 9

Multi-Layer Graph Analysis for Dynamic Social Networks

Modern social networks frequently encompass multiple distinct types of connectivity information; for instance, explicitly acknowledged friend relationships might complement behavioral measures that link users according to their actions or interests. One way to represent these networks is as multi-layer graphs, where each layer contains a unique set of edges over the same underlying vertices (users). Edges in different layers typically have related but distinct semantics; depending on the application multiple layers might be used to reduce noise through averaging, to perform multifaceted analyses, or a combination of the two. However, it is not obvious how to extend standard graph analysis techniques to the multi-layer setting in a flexible way. In this chapter we develop latent variable models and methods for mining multi-layer networks for connectivity patterns based on noisy data.

Symbol	Description
$G = (\mathcal{V}, \mathcal{E})$	Multi-layer network
$\mathcal{V}$	Vertex set
$\mathcal{E} = (\mathcal{E}_1, \ldots, \mathcal{E}_L)$	Tuple of edge sets for each layer
A_i	True adjacency matrix
W_i	Observed adjacency matrix

Table 9.1: Glossary of commonly used symbols.

9.1 Introduction

Multi-layer networks arise naturally when we have more than one source of connectivity information for a group of users. In a social networking context, we often have knowledge of direct communication links, i.e., *relational* information. However, it is also possible to derive *behavioral* relationships based on user actions or interests. The question that this chapter attempts to address is how to deal with these multiple layers of a social network when attempting to perform tasks like inference, clustering, and anomaly detection.

We propose a generative hierarchical latent-variable model for multi-layer networks, and show how to perform inference on its parameters. Using techniques from Bayesian Model Averaging [194], we conditionally decouple the layers of the network using a latent selection variable; this makes it possible to write the posterior probability of the latent variables given the multi-layer network. The resulting mixture can be viewed as a scalarization of a multi-objective optimization problem [72], [167], [235]. When the posterior probability functions are convex, the scalarization of the multiobjective problem is both optimal and consistent with the Bayesian context [72], [86].

We then step back from the Bayesian setting and discuss how multi-objective optimization can be used to perform MAP estimation of the desired latent variables. Using the concept of Pareto optimality [167], we can define an entire front of solutions; this allows a user to define a preference over optimization functions and tune the algorithm accordingly. The result is a level of supervised optimization and inference that still utilizes the structure of multi-layer networks.

We perform experiments on a simulated example, showing that our method yields improved clustering performance in noisy conditions. We discuss how our framework can be combined with existing models, and describe the details of this process for the dynamic stochastic block model (DSBM) [234], which captures a variety of complex temporal network phenomena. Finally, we apply the multi-layer DSBM to a real-world data set drawn from the ENRON email corpus.

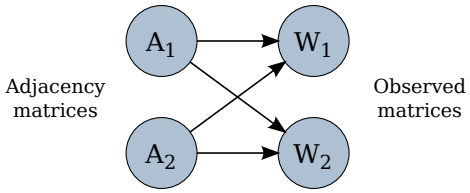


Figure 9.1: Adjacency and observation matrices. This graphical model depicts how the latent adjacency matrices can affect the observervation matrices. Note that the observation matrices are dependent on all adjacency matrices in general.

9.2 Multi-layer networks

A multi-layer graph $G = (\mathcal{V}, \mathcal{E})$ comprises vertices $\mathcal{V} = \{v_1, \ldots, v_p\}$, common to all layers, and edges $\mathcal{E} = (\mathcal{E}_1, \ldots, \mathcal{E}_L)$ on L layers, where $\mathcal{E}_i$ is the edge set for layer i.

In the real-world network setting, we will assume that the observed data are noisy reflections of a true underlying multi-layer graph. For convenience we will work with adjacency representations, letting $A_i \in \mathbb{R}^{p \times p}$ be the true adjacency matrix of layer i, and $W_i \in \mathbb{R}^{p \times p}$ the corresponding observed adjacency matrix. Figure 9.1 depicts the model graphically.

In some cases W_i might be binary, reflecting merely the presence or absence of a connection—for instance, whether two users were seen to communicate. In other settings, such as measuring temporal or content correlation scores between users, the entries of W_i could be real-valued. We wish to estimate $A_1, \ldots, A_L$ given the observations $W_1, \ldots, W_L$. Using standard parametric methods this will require us to compute the posterior distribution of $A_1, \ldots, A_L$, which can be difficult given the number of parameters. Specifically, the influence of $A_1, \ldots, A_L$ on a single W_i is difficult to measure, as the dependencies are unspecified.

9.3 Hierarchical model description

We propose a hierarchical model that simplifies this inference procedure by conditionally decoupling $W_1, \ldots, W_L$. For simplicity, let us specialize to the case where $L = 2$. This

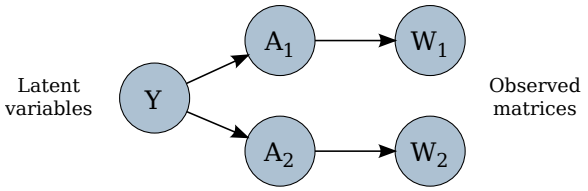


Figure 9.2: General latent variable model. This model represents a latent variable model, in which a set of variables Y control the distributions of the adjacency matrices and through them the observation matrices.

also allows us to view the networks in the setting described in the introduction: one layer of the network represents the observed extrinsic relationships between users, and the other their correlated intrinsic behaviors.

Initially, we introduce a latent variable denoted Y (see Figure 9.2) that conditionally decouples the posterior distributions of the two layers:

$$P(W_1, W_2|A_1, A_2, Y) = P(W_1|A_1, Y)P(W_2|A_2, Y), \tag{9.1}$$

$$P(W_1, W_2|A_1, A_2) = \int P(W_1, W_2|A_1, A_2, Y)P(Y|A_1, A_2)dY\ . \tag{9.2}$$

We can now shift our focus from the adjacency matrices A_1, A_2, to the latent variable Y, using Y to represent the adjacency matrices in a useful way. We can write down the posterior distribution for Y as

$$P(Y|W_1, W_2) = \sum_{A_1, A_2} P(Y|A_1, A_2)P(A_1, A_2|W_1, W_2)\ .$$

9.4 Posterior mixture modeling

Now consider the graphical model shown in Figure 9.3. We have collapsed the A_1, A_2 variables with the observed data W_1, W_2, because we are mainly interested in inferring W, and W_i can be considered a representation of the real connectivity.

Following from the previous model, we have decomposed $Y = (W, Z)$, where $W \in \mathbb{R}^{p\times p}$ is a latent adjacency or similarity matrix describing the underlying connections be-

tween vertices, and $Z \in \{1, 2\}$ is a model selection variable, $P(Z = 1) = \alpha$, and $P(Z = 2) = 1 - \alpha$. Here we are making the implicit assumption that there is a common connectivity structure W that informs all layers of the network; due to the different attributes of each layer, they may reveal this underlying structure in different ways, or obfuscate it altogether. In a sense the model produces observed matrices that correspond to multiple views of the latent variable W. The model selection variable Z will decouple the posterior distribution of W given both layers into a weighted sum of marginalized posteriors given each individual layer.

The prior for W is $P(W)$, left unspecified for now. The distributions $P(W_1|W, Z)$ and $P(W_2|W, Z)$ are in general task-dependent (e.g., they could be Gaussian, Wishart, Bernoulli, etc.), but we will make the simplifying assumption that Z acts as a selector variable, so that W and W_1 are conditionally independent given $Z = 2$, and likewise W and W_2 are conditionally independent when $Z = 1$. Formally, using the notation P_z to denote conditioning on $Z = z$, we have

$$P_2(W_1|W) = P_2(W_1) \tag{9.3}$$

$$P_1(W_2|W) = P_1(W_2)\ . \tag{9.4}$$

We are interested in the posterior distribution of the latent variable W given the observed variables W_1, W_2:

$$P(W|W_1, W_2) \tag{9.5}$$

$$= P(W, Z = 1|W_1, W_2) + P(W, Z = 2|W_1, W_2) \tag{9.6}$$

$$= P(W|W_1, W_2, Z = 1)P(Z = 1|W_1, W_2) + P(W|W_1, W_2, Z = 2)P(Z = 2|W_1, W_2) \tag{9.7}$$

$$= \xi P(W|W_1, W_2, Z = 1) + (1 - \xi)P(W|W_1, W_2, Z = 2)\ , \tag{9.8}$$

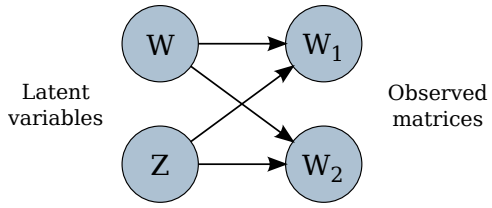


Figure 9.3: Model with similarity matrix and selection variable. We introduce the similarity matrix W and the selection variable Z to describe our latent variable model. Conditioning on W and Z, we assume that the two layers are independent from each other.

where $\xi = P(Z = 1|W_1, W_2)$. Let's consider the first term. We have

$$P(W|W_1, W_2, Z = 1) = \frac{P(W, W_1, W_2, Z = 1)}{\sum_{\hat{W}} P(\hat{W}, W_1, W_2, Z = 1)} \tag{9.9}$$

$$= \frac{P(W)P_1(W_1|W)P_1(W_2)}{\sum_{\hat{W}} P(\hat{W})P_1(W_1|\hat{W})P_1(W_2)} \, . \tag{9.10}$$

Since $P_1(W_2)$ does not depend on W, it factors out of the sum in the denominator and cancels; thus we have

$$\frac{P(W)P_1(W_1|W)}{P_1(W_1)} \, . \tag{9.11}$$

Performing the same computation on the other side and combining, we have

$$P(W|W_1, W_2) = \xi \frac{P(W)P_1(W_1|W)}{P_1(W_1)} + (1 - \xi)\frac{P(W)P_2(W_2|W)}{P_2(W_2)} \tag{9.12}$$

$$= P(W) \left[\gamma_1 P_1(W_1|W) + \gamma_2 P_2(W_2|W)\right] \, , \tag{9.13}$$

where $\gamma_1 = \xi/P_1(W_1)$ and $\gamma_2 = (1 - \xi)/P_2(W_2)$ are constants with respect to W. If we assume the prior on W is uniform, then the MAP value of W is also the maximum likelihood estimate, and can be written as

$$\text{argmax}_W \left[\gamma_1 P_1(W_1|W) + \gamma_2 P_2(W_2|W)\right] \, . \tag{9.14}$$

The above solutions describe not just one MAP estimation of W, but rather a family of

MAP estimations, based on the priors that we implicitly assign to each model by choosing a specific value of α (which affects ξ and γ in turn). Qualititatively, this can be viewed as determining a relative confidence parameter between the networks; if W_1 is more trusted than W_2, then the best choice of α would be greater than 0.5.

As an example, assume that both $P(W_1|W)$ and $P(W_2|W)$ are distributed as isometric Gaussians, i.e.,

$$P(W_1|W) = \mathcal{N}(W, \sigma_1^2 I_p) \tag{9.15}$$

$$P(W_2|W) = \mathcal{N}(W, \sigma_2^2 I_p)\ . \tag{9.16}$$

Then the solution for $\hat{W}$ has the form

$$\hat{W} = \beta W_1 + (1-\beta) W_2 \tag{9.17}$$

for some choice of $0 \leq \beta \leq 1$.

A proof of this is given in Appendix D, but intuitively this result makes sense, as an isometric normal distribution is spherically symmetric around the mean, and so there is no asymmetric density to pull the optimal point off of the line between the given means. In general, of course, the solution will not have such a simple form. However, optimization techniques can be used to find the various optimal solutions for different values of α.

9.5 Simulation example

We use simulations to show that clustering of nodes in a weighted graph can be improved using the MAP estimate of W. Two random graphs with 500 nodes are constructed with 10 known clusters. The weights between nodes in the same cluster are normally distributed as $\mathcal{N}(5, 0.5)$, and weights between nodes that are not in the same cluster are normally distributed as $\mathcal{N}(4.7, 0.5)$. Both layers come from this underlying similarity structure, but

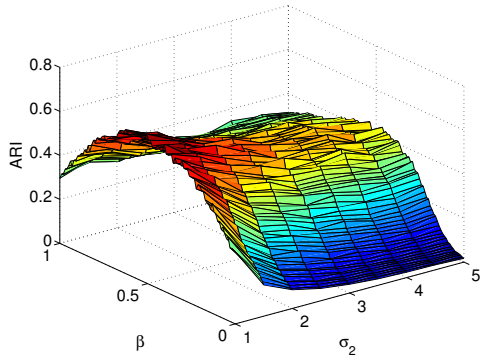


Figure 9.4: Clustering simulation. This surface plot shows the ARI for different simulations of σ_2 and β. Note that for all levels of σ_2, a β that is around 0.5 tends to produce the best clustering.

Table 9.2: Variances and ARI scores.

σ_1	σ_2	Max ARI	β
1	1	0.6782	0.5051
1	1.5	0.6199	0.5253
1	2	0.5828	0.4343
1	2.5	0.5514	0.5051
1	3	0.5073	0.4545
1	3.5	0.4878	0.4848
1	4	0.4876	0.5253
1	4.5	0.4635	0.5354
1	5	0.4429	0.4646

are corrupted with i.i.d. Gaussian noise with zero mean and different variances σ_1 and σ_2. For various choices of β, the networks are clustered using a normalized-cut spectral clustering algorithm [143], and the Adjusted Rand Indices (ARI) [111] are computed. For each of several different levels of variance, this experiment is run 50 times, and the results are averaged. Figure 9.4 shows a plot of the results. This shows that using (9.13) to estimate the mixture of networks improves the clustering, as expected. Note that even with unequal variance, optimal β is consistently near 0.5.

9.6 Pareto summarizations

Of course, in practice it may be difficult to effecitvely set the prior α directly, which is why we instead generate a family of MAP estimates, which can then all be evaluated. We can also view this procedure in a more general framework that can lead to more flexibility in the inference and estimation procedure. We can view the maximization of the combined posterior distributions as a scalarization of a multi-objective optimization problem. We can also consider other solutions to the optimization problem that are more general than linear scalarization, such as Pareto front analysis.

Let us consider the formal multi-objective optimization problem

$$\hat{W} = \text{argmin}_W[f_1(W), f_2(W)] \, . \tag{9.18}$$

For the model derived in the previous sections, we have $f_1(W) = -P_1(W|W_1)$ and $f_2(W) = -P_2(W|W_2)$. One potential solution to the multiobjective optimization problem above is a scalarization of the two objective functions, so that the new problem to be solved is

$$\hat{W} = \text{argmin}_W \gamma f_1(W) + (1-\gamma) f_2(W) \, . \tag{9.19}$$

This view leads to the objective in (9.13). However, this is not the only available approach to a multi-objective optimization problem. We can also consider the concept of Pareto optimality to find solutions for the optimization problem. A solution to an optimization problem is said to be weakly Pareto optimal (or weakly non-dominated) if it is not possible to improve any objective function without lowering some other objective function [72], [235]. More formally, we say that a solution x_1 dominates a solution x_2 if $f_i(x_1) \leq f_i(x_2)$ for every objective function f_i and there exists some j such that $f_j(x_1) < f_j(x_2)$. The first Pareto front is the set of weakly non-dominated points.

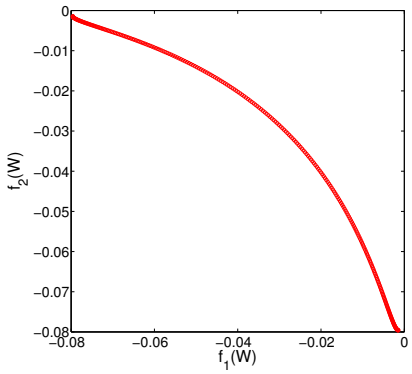


Figure 9.5: Pareto front for two Gaussians. A convex Pareto front would bulge toward the lower left corner, but this plot demonstrates that even relatively simple objective equations can have extremely non-convex Pareto fronts.

In terms of finding Pareto optimal points, the scalarization technique that is discussed above would find the Pareto front when the solution space is a convex set and the individual objective functions are convex functions on the solution space [86]. However, if these convexity conditions are not met, the scalarization technique will not find the entire Pareto front. Often times, the posterior distributions in (9.18) are not convex. So, by using the concept of Pareto optimality, we are extending our list of possible optimal solutions, and generalizing the MAP estimate that was found in earlier sections. Figure 9.5 shows an example of the Pareto front of a multiobjective optimization, where f_1 and f_2 are the two dimensional pdfs of normal distributions, as shown below:

$$f_i(W) = (2\pi)^{-n/2} \left|\Sigma_i\right|^{-\frac{1}{2}} e^{-\frac{1}{2}(W-W_i)^T \Sigma_i^{-1}(W-W_i)} \tag{9.20}$$

$$W_1 = \begin{bmatrix} 10 \\ 8 \end{bmatrix}, W_2 = \begin{bmatrix} 8 \\ 10 \end{bmatrix}, \Sigma_1 = \Sigma_2 = 2I_2 \tag{9.21}$$

Even this relatively simple distribution has a non-convex Pareto front; note that minimizing a linear combination of f_1 and f_2 can only find optima at the extremes of the curve, and does not explore the interior, which may be more useful for some applications.

This type of example motivates further research into generating MAP estimates in this

manner, as finding the Pareto front could give us an advantage when attempting to infer parameters of the model as we do above, or perform some other common task.

9.7 Stochastic block models and the DSBM

Consider for a moment a single layer network. Often times we are interested in networks which we expect to have some community structure. We also require something more general than attempting to distinguish communities from maximal clique detection. Rather, a community should be defined as disjoint subsets of nodes that behave in the same way as the other nodes in the subset. This allows for a more interesting community structure than just using the density of connections in a group, i.e., creating communities based on high intra-connectivity between nodes. For instance, one group may exhibit strong interconnection with another group, but only moderate connectivity within themselves. This model also proves interesting when the class membership is already known, and is a way to leverage that information.

Consider a network with N nodes that we expect to fall in K classes, where $c \in \mathbb{R}^N$ is a class membership vector. In this setup we are considering binary relationships between nodes, and so a connectivity matrix $A \in \mathbb{R}^{N \times N}$ is observed. The parameters for a standard SBM would then be the Bernoulli parameter matrix Θ, where θ_{ij} is the probability of a link forming between a node in class i and class j. So, while a connectivity matrix will be $N \times N$, $\Theta \in \mathbb{R}^{K \times K}$ and is symmetric. It can be shown [234] that the MLE of θ_{ij} is

$$\hat{\theta}_{ij} = \frac{m_{ij}}{n_{ij}} \tag{9.22}$$

$$m_{ij} = \sum_{x \in i} \sum_{y \in j} a_{ij} \tag{9.23}$$

$$n_{ij} = \begin{cases} |i||j|, & , i \neq j \\ |i|(|i|-1), & , i = j \end{cases} \tag{9.24}$$

This estimate of Θ can be used to explore the structure of the network. Note that this method depends heavily on a correct class membership vector; without this the nodes in the same classes will not be behaving in similar ways.

The SBM accounts for community structure, but does not account for temporal changes in the network. One solution to this problem would be to fit a SBM to every time step in the sequence. This approach has problems, however. One major concern is that there is no accounting for any noise that is introduced when including temporal information. Another downside is that it fails to take advantage of information from previous time steps, and it does not encourage the class membership to evolve smoothly over time. Recently, the Dynamic SBM (DSBM) has been introduced to account for some of these effects [234].

The DSBM employs an extended Kalman filter (EKF) to track temporal changes in the network. Two types of the DSBM are available: one that is given the class membership *a priori*, and another that estimates the class memberships along with the other SBM parameters. In the following example, only the *a priori* case is used. In order to estimate the Bernoulli parameters through the EKF, a logistic transform is used to map the estimates of Θ into the real line:

$$\psi_{ij} = \log(\theta_{ij}) - \log(1-\theta_{ij}) \in (-\infty, +\infty) \, . \tag{9.25}$$

Once the inference is complete, the Kalman estimate is then mapped back into Bernoulli parameters. For more details, see [234].

9.8 Enron example

We next look at the real-world ENRON email data set[1]. This data set consists of approximately a half million email messages sent or recieved by 150 senior employees of the ENRON Corporation. These emails were made publicly available as a result of the SEC investigation of the company in 2002, and constitute one of the largest publicly available email repositories.

This dataset represents a unique oppurtunity to examine private email messages in a corporate setting. This is rare due to privacy concerns and proprietary information, but the ENRON dataset is for the most part untouched, except for a few emails that were specifically requested to be removed. In addition to the raw emails, the dataset also contains the job title of the employees that are included. This is useful to separate the employees into classes, so that we may examine their behavior using the DSBM and its related techniques.

To explore the multi-layer structure, two layers are created from the ENRON dataset. As discussed previously, the information was chosen so that one layer represented the extrinsic, "relational" information between users, and the other represented intrinsic, "behavioral" information between users. The network layers are extracted from the data as follows. First, a *relational* network is recovered from the headers of emails by identifying the sender and reciever(s) of each message, including Cc and Bcc recipients. For each week in the dataset, a separate network of employees is constructed from the emails sent during that week.

A second set of *behavioral* networks are recovered using the contents of email messages. On the same weekly basis the contents of all emails originating from each user are combined to form long "documents". These documents combine to produce a dictionary of words from which term frequency-inverse document frequency (TF-IDF) scores are calculated [15]. TF-IDF scores are commonly used for identifying important words in text

[1] http://www.cs.cmu.edu/~enron

analysis, and are computed using

$$\text{tf}(t,d) = \frac{f(t,d)}{\max_{\hat{t}} f(\hat{t},d)} \tag{9.26}$$

$$\text{idf}(t) = \log\left(\frac{|D|}{N(t,D)}\right) \tag{9.27}$$

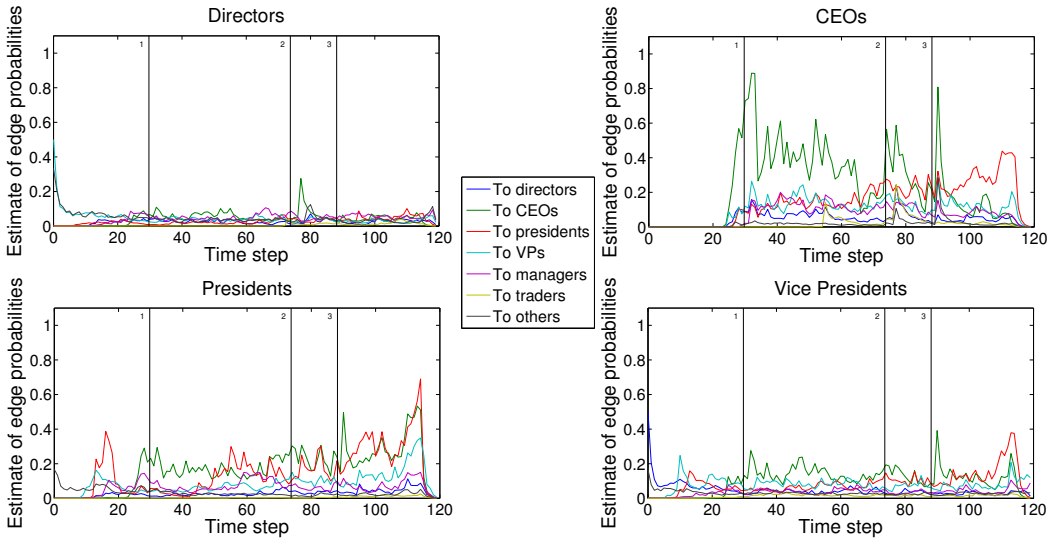
$$\text{score}(t,d) = \text{tf}(t,d)\text{idf}(t)\ , \tag{9.28}$$

where $f(t,d)$ is the frequency of term t in document d, $N(t,D)$ is the number of documents in which the term t appears, and $|D|$ is the size of the document corpus, which in this case is the number of active network nodes. For each active user (document), a TF-IDF score is computed for each word in the dictionary.

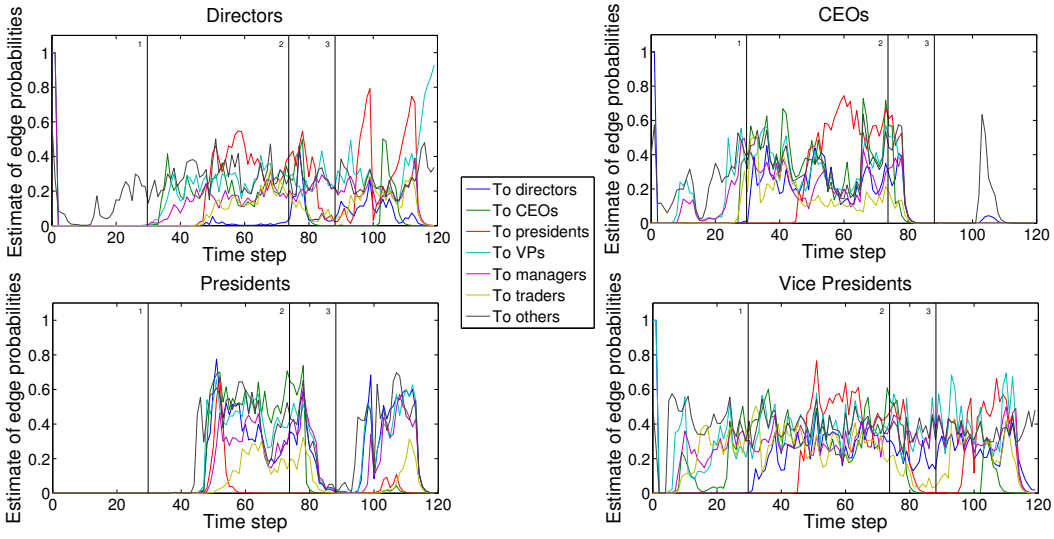
Using the vector of TF-IDF scores for each user, we measure the cosine similarity of each user by taking dot products in order to obtain a similarity matrix W. Again, this is done for every week in the relevant time period, creating a second dynamic network with weighted edges. However, since we started in the SBM framework, it is necessary to transform the weighted edge network into a binary network. To do this, the similarity scores are thresholded. To be roughly consistent with the density of the relational network, we keep the top 15% greatest correlations between users at each time step, setting all other connections to 0. This allows us to create networks of similar sparsity level.

The above procedure yields a two-layer binary dynamic network that we can use to obtain insight into the structural dynamics of the ENRON data. To do so, we will extend the dynamic stochastic block model (DSBM) [107], [234] to the multi-layer setting. In order to simplify the modeling, we assume that the DSBM groups are known *a priori*, and seek to recover the Bernoulli parameters for each class, which predict the likelihood of an edge between users from any pair of groups. In this case, we group employees by their role in the company (CEO, President, Director, etc.).

Figure 9.6a and Figure 9.6b shows some of the estimated Bernoulli parameters for different classes when the DSBM is run on the two layers separately. Figure 9.6a represents



(a) Relational DSBM Parameters.



(b) Behavioral DSBM Parameters

Figure 9.6: DSBM simulation results. These graphs show the estimated DSBM parameters for different classes, and how they evolve over time. (a) is the evolution of the DSBM parameters from the relational layer, while (b) is the evolution of parameters from the behavioral layer.

the evolution of the relational layer, while the Figure 9.6b represents the behavioral layer. The DSBM was run over a 120 week period, from December 6th, 1999 to March 27th, 2002. The vertical lines represent important events in the ENRON time line. Line 1 corresponds to ENRON releasing a code of ethics policy. It is also the first time that the

company's stock reached above $90. Line 2 their stock closing below $60. This was a critical point in the timeline, because the company began losing many partnerships, including one to create a video-on-demand system. In this same month, a few of the employees had begun to communicate the uneasiness with ENRON's accounting practices. Line 3 is the week of Jeffrey Skilling's resignation. A mere month after his resignation as CEO, the SEC began their official inquiry into ENRON. These events are chosen as a baseline to compare the two layers of the network.

For the relational DSBM parameters, the most interesting results come from the CEO's activity. Note that the CEO group combines all past and present CEO's. This evolution of parameters seems to indicate that during some of the important milestones in ENRON's demise, the CEO's were talking to each other more often, as well as sending out emails to the other employees in the network. This seems to indicate that they were at least somewhat aware of what was happening with the company during these events, and had maybe discussed matters among themselves. From the relational layer, it also appears that the CEO's were the most active in communicating with other groups, where as the Directors showed very little connectivity. One explanation for this is that because the subset of employees that were studied were higher up in the company, the Director group didn't communicate with them as much, instead managing the lower level employees. Another interesting result is that the President group had much more activity towards the end of the time period, suggesting that as the legal situation worsened, their activity increased.

The behavioral DSBM parameters appear to be more noisy than their relational counterparts. In addition, they show very different behavior than the relational layer. The Vice Presidents appear much more active during the entire period when compared with the relational layer. Because of the nature of the TF-IDF and thresholding process, there could be a number of reasons for this. One possible reason could be that the weeks in which the Vice Presidents were active, they could have been sending a lot of forwarded emails, acting as a conduit of information between parties. This would cause the TF-IDF scores for the

Vice President group to rise.

Another interesting phenomenon in the behavioral layer is that of the CEOs. Specifically, it is interesting how their activity drops off significantly, and in fact one event that is very much apparent in the relational parameters completely disappears. This can only happen if the document content for the CEOs during those weeks are completely orthogonal to the other groups. Because we consider only text that the sender has written, and we only consider sent emails, one explanation could be that the CEO's forwarded many emails without adding any additional text. This would cause the list of words for the CEO to become very small. However, a more likely explanation after some examination of the dataset shows that there is a large amount of activity in the relational dataset because many of the employees were emailing the CEO in a petition-like fashion, and so there was a lot of activity. However, the CEO group actually sent very few emails during that time.

Combining the two networks as in Section 9.4, we run the DSBM for different levels of the mixing parameter α. Because of the use of binary networks in this example, the α parameter is used as the probability that the combined data will choose to use the relational network when the two layers disagree with each other. The objective in this particular example is to show that using this method we can not only reduce noise, but also discover interesting multifaceted behavior that is not obvious from one layer alone. We expect that this form of combination will emphasize traits or attributes that occur in both networks; however, attributes that exist mostly in one network but are strong enough will also be retained. We can study these effects through various network measures; in this case we look at betweenness and degree centrality.

Figure 9.7 shows the DSBM parameters for mixing parameters $\alpha = 0.5$. Smaller values of α should be chosen because the relational netowrk seems to be less noisy, more stable. This makes sense as extrinsic relations are easier to measure. One interesting phenomenon that occurs is that much of the behavior that we saw in the relational layer is present, including the high level of CEO activity. We can also see however, that the period of

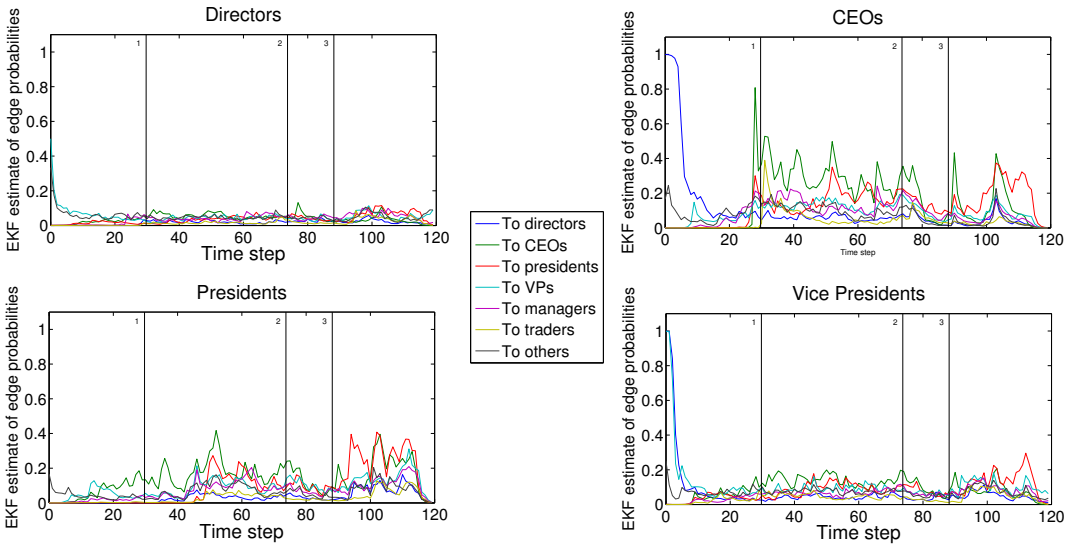


Figure 9.7: Combined DSBM results. These graphs show the results of combining the two layers of the network with a parameter $\alpha = 0.5$. Therefore, we should see attributes from both the behavioral and relational DSBM, and maybe some new, interesting results that result from combining the two layers.

inactivity that is experienced in the behavioral layer for the CEO group has an effect by dampening the some of the strong peaks that we saw towards the end of the time period.

Figure 9.8 shows the betweeness centrality of the Directors group over time as the mixing parameter is varied. In general, the betweeness rises roughly monotonically as α is varied; however, from week 95 to week 115, betweenness centrality is significantly increased when using a combined dynamic network—that is, an intermediate value of α. This time corresponds to the beginning of the company's upheaval and public disclosure of troubles. It may be concluded that by examining both network layers simultaneously we have removed some of the edges between other classes, and thus the centrality score of this particular group increased. It is true that during this time, when overall email usage increased, the betweenness centrality measure went down, as there were more shortest paths through users from other groups. Using the combination of layers, however, there appears to be an increase in the number of shortest paths through the Directors group.

On the other hand, we can also see well-behaved monotonic correlations in some cases.

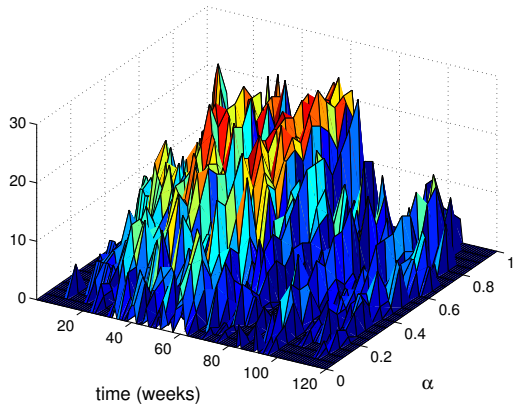


Figure 9.8: Betweenness centrality for directors. This centrality is a measure of how connected a node is to the rest of the network. Larger centrality scores often occur for intermediate values of α, particularly between time 95 and 115.

Figure 9.9 shows a transition of degree centrality for the class of CEOs (of which there were four during this time period). The behavioral network shows more connectivity for the CEO class. This phenomenon makes sense, as the behavioral data takes into account all written documents, which could be correlated with those of other users, while the relational network only takes into account direct communication between the CEOs and others. In reality, much of that communication is performed through third parties (such as assistants), and thus CEOs probably do not send as much email as the average employee. Increasingly anomalous behavior occurs toward the end of the time period. We hypothesize that this is due to a larger volume of unusual emails sent directly to the CEO during this tumultuous period.

9.9 Related work

The literature on single layer networks is large, with contributions coming from many different fields. There are many results on structural and spectral properties of a single-layer network, including community detection [164], random walk return times [170], and percolation theory results [4]. Diffusion or infection models have also been studied in the context of complex networks (see [92], for instance).

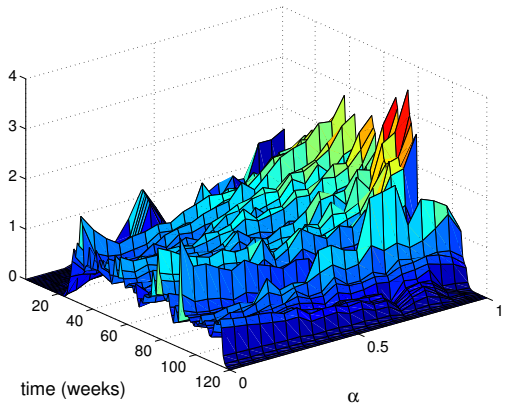


Figure 9.9: Degree centrality for CEOs. Higher degree centrality for α near one signifies greater activity in the behavioral network. Anomalous behavior can be seen in the later time steps as activity patterns shift.

Community structure of a network is a wide range of research in its own right. In this chapter, a community structure model is used in a real-world example. Specifically, the stochastic block model (SBM) [107] is used to model community structure within a network by assuming identical statistical behavior for disjoint subsets of nodes. These communities are more flexible than simple cliques because it is not required that they be heavily interconnected, but only that they interact with nodes in other subcommunities uniformly. More recently, the SBM has been extended to track temporal changes in the network, appropriately called a Dynamic SBM, or DSBM [234]. The DSBM uses an extended Kalman filter to track temporal changes between nodes, which will result in a smoothed and potentially insightful evolution of the estimated parameters.

Recently, there has been a growing interest in the multi-layer network problem. Some basic network properties have been extended to the multi-layer structure [21], [31] as well as some results that serve as an extension of single layer concepts, such as multi-layer network growth [169] and spreading of epidemics [203]. The metrics that have been proposed attempt to incorporate the dependence of the layers into the statistical framework, which will allow for a much richer view of the network. In the same vein, our research attempts to perform parameter inference on a multi-layer network, incorporating some of the dependence information that the multi-layer structure allows.

Bayesian model averaging is also related to this work; ideas from BMA are used to create conditional independence between the layers of a network [194]. Originally it was intended to address ambiguous model selection, but the conditional decoupling that occurs allows us to account for multiple layers in a multi-layer network easily. This framework accounts for the interdependent relationships between the multiple layers into latent variables, which can then be estimated.

9.10 Conclusion

We introduced a novel method for inference on multilayer networks. A hierarchical model was used to jointly describe the noisy observation matrices and MAP estimation was performed on the relevant latent variable. A simulation example using clustering demonstrated that the mixture of layers under the correct circumstances can lead to better results, and possibly a better understanding of the underlying structure between users. A real-life example was also discussed using the ENRON email dataset. This chapter also leads the way for future work; in addition to trying more noise models that are not so simply reproduced or even nonconvex, one can use multi-objective optimization to explore other objective functions that could be useful in describing a multi-layer network, such as network smoothness or the centrality distribution.

CHAPTER 10

Conclusion and Future Work

This thesis focused on methods regarding data with network structure. In particular, it touched on four separate research areas:

- edge exchangeable models for hierarchical network data
- graph-based estimation of information theoretic functionals
- estimation and summarization of time-varying interactions for agents
- multi-layer network analysis with an application to community detection

These research areas are connected in that in each of them it is imperative to utilize and respect the structure of the data, whether that structure is explicit, as in interaction data, or implicit, as in influence estimation. The thesis advanced the state-of-the-art by introducing a new framework for complex network interaction data, described new tight bounds for the multi-class Bayes error rate, introduced new graph-based estimation techniques, introduced adaptive directed information and techniques for estimation, and finally described a new approach to multi-layer community detection.

The edge-exchangeable model for hierarchical network data can be extended in multiple fruitful directions. For the edge-exchangeable interaction framework, one deficit of the model is that it allows for multi-set observations, which doesn't mirror most datasets. For instance, you would never see the same receiver twice on an email. Using techniques from

partial ranking models, we can extend our framework to remove the multisets from the possible set of observations. Additionally, more work applying the framework to prediction in networks is a logical next step, as is incorporating temporality into the model. We further suggest that models with the same invariance properties but that have better robustness properties builtin would be a useful extension to the work.

The work in this thesis of graph based information theoretic functionals can be extended in a number of ways. First, the asymptotic rates at which the estimators described in Chapters 3 and 4 have yet to be explored. For the multi-class generalization of HP divergence, utilizing this functional in other machine learning contexts could be beneficial to a learning framework. More work along the lines of Chapter 4 needs to be done to better understand the bias that occurs in these graph based estimation methods for high intrinsic dimension, along with more techniques to correct this bias.

For the estimation and summarization of time-varying interactions, we posit a number of extensions. First, more work needs to be done to understand the behavior of DI and ADI as a function of the underlying joint distribution of the system; in the theoretical results that were given in this thesis, assumptions were placed directly on the behavior of DI or the summand $\mathrm{I}(\mathbf{X}_{1:t}^i;\ X_t^j|\mathbf{X}_{1:t-1}^j)$. A more comprehensive approach would be to start with the joint distribution of $\mathbf{X}_{1:T}^k, k \in \{1,\ldots,N\}$. We leave this for future work. For the continuous estimator of DI and ADI, a dynamic covariance model was used. This model may be extended in numerous ways, including using a non-linear transformation, as in the nonparanormal family of distributions [137].

For multi-layer network analysis, more work can be done to describe how close the solutions from our community detection algorithm are from the Pareto front; it is empirically observed that they are on or very close to the Pareto front of solutions. More work can be done to extend the community detection algorithm to greater than two layers, and to find computationally efficient ways to apply it to extremely large datasets.

APPENDIX A

Supplemental Material for Chapter 2

A.1 Full Derivation for Gibbs Sampling

In this section, we assume the following prior structure on our parameters:

$$\theta \sim \text{Gamma}(a, b),\ \alpha \sim \text{Beta}(c, d) \tag{A.1}$$

$$\theta_s \sim \text{Gamma}(a_s, b_s),\ \alpha_s|\alpha \sim \text{Beta}(c_s, d_s). \tag{A.2}$$

We begin with sampling on θ and α. Rewriting the likelihood function by canceling out like terms, we have:

$$\text{pr}\left(\{R_{n,j}, T_{n,j}\}_{1\le i\le n, 1\le j\le k_n} \mid \{S_i\}_{i=1}^n; \Psi\right) = \frac{[\theta+\alpha]_\alpha^{K_n-1}}{[\theta+1]_1^{t_{\cdot\cdot}-1}} \prod_r [1-\alpha]_1^{t_{\cdot r}-1} \prod_s \frac{[\theta_s+\alpha_s]_{\alpha_s}^{t_{s\cdot}-1}}{[\theta_s+1]_1^{c_{s\cdot\cdot}-1}} \prod_{k=1}^{t_{s\cdot}} [1-\alpha_s]_1^{c_{srk}-1}$$

We can determine the posterior on θ in the following way ($\Psi^{(a)} = \Psi \setminus a$):

$$\begin{aligned}
&\text{pr}(\theta|\{R_{n,j},T_{n,j}\},\{S_i\}_{i=1}^n,\Psi^{(\theta)}) \\
&\qquad\propto \text{pr}\left(\{R_{n,j},T_{n,j}\}_{1\le i\le n,1\le j\le k_n}\mid\{S_i\}_{i=1}^n;\Psi\right)\text{pr}(\theta|\Psi^{(\theta)}) \\
&\qquad\propto \text{pr}\left(\{R_{n,j},T_{n,j}\}_{1\le i\le n,1\le j\le k_n}\mid\{S_i\}_{i=1}^n;\Psi\right)\text{pr}(\theta) \\
&\qquad\propto \frac{[\theta+\alpha]_\alpha^{K_n-1}}{[\theta+1]_1^{t_{\cdot\cdot}-1}}\,\text{pr}(\theta)
\end{aligned} \tag{A.3}$$

When $t_{\cdot\cdot}\geq 2$ the denominator is

$$\frac{1}{[\theta+1]_1^{t_{\cdot\cdot}-1}}=\frac{\Gamma(\theta+1)}{\Gamma(\theta+t_{\cdot\cdot})}=\frac{1}{\Gamma(t_{\cdot\cdot}-1)}\int_0^1 x^\theta(1-x)^{t_{\cdot\cdot}-2}dx.$$

We note that the above can be thought of as a marginalization over an auxiliary variable $x\sim\text{Beta}(\theta+1,t_{\cdot\cdot}-1)$, so that

$$\text{pr}(\theta,x|\{R_{n,j},T_{n,j}\},\{S_i\}_{i=1}^n,\Psi^{(\theta)})\propto[\theta+\alpha]_\alpha^{K_n-1}x^\theta(1-x)^{t_{\cdot\cdot}-2}\,\text{pr}(\theta).$$

Similarly, we can expand the numerator:

$$[\theta+\alpha]_\alpha^{K_n-1}=\prod_{i=1}^{K_n-1}(\theta+\alpha\cdot i)=\prod_{i=1}^{K_n-1}\sum_{y_i=0,1}\theta^{y_i}\cdot(\alpha\cdot i)^{1-y_i}.$$

We introduce the auxiliary variables $y_i\sim\text{Bernoulli}\left(\frac{\theta}{\theta+\alpha\cdot i}\right)$, so that we can rewrite the posterior once again:

$$\begin{aligned}
&\text{pr}(\theta,x,\{y_i\}|\{R_{n,j},T_{n,j}\},\{S_i\}_{i=1}^n,\Psi^{(\theta)}) \\
&\qquad\propto x^\theta(1-x)^{t_{\cdot\cdot}-2}\prod_{i=1}^{K_n-1}\theta^{y_i}\cdot(\alpha\cdot i)^{1-y_i}\theta^{a-1}e^{-b\theta}, \\
&\qquad\propto \theta^{\sum y_i+a-1}e^{-\theta(b-\log x)},
\end{aligned} \tag{A.4}$$

which is a Gamma distribution, so that the posterior for θ can be sampled as a Gamma

distribution, Gamma $(\sum y_i + a, b - \log x)$. We can do this because,

$$\text{pr}(\theta, x, \{y_i\}|\{R_{n,j}, T_{n,j}\}, \{S_i\}_{i=1}^n, \Psi^{(\theta)}) \propto \text{pr}(\theta|x, \{y_i\}, \{R_{n,j}, T_{n,j}\}, \{S_i\}_{i=1}^n, \Psi^{(\theta)}).$$

We can proceed similarly for α:

$$\text{pr}(\alpha|\{R_{n,j}, T_{n,j}\}, \{S_i\}_{i=1}^n, \Psi^{(\alpha)}) \propto [\theta + \alpha]_\alpha^{K_n - 1} \prod_r [1 - \alpha]_1^{t_{\cdot r} - 1} \text{pr}(\alpha).$$

We can take care of the first term as we had for θ. Then the final term

$$[1 - \alpha]_1^{t_{\cdot r} - 1} = \prod_{j=1}^{t_{\cdot r} - 1} (j - \alpha) = \prod_{j=1}^{t_{\cdot r} - 1} \sum_{z_{r,j}=0,1} (j - 1)^{z_{r,j}} (1 - \alpha)^{1 - z_{r,j}}.$$

So we introduce auxiliary variables $z_{r,j} \sim$ Bernoulli$(\frac{j-1}{j-\alpha})$, and can expand the posterior:

$$\begin{aligned}
&\text{pr}(\alpha, \{z_{r,j}\}, \{y_i\}|\{R_{n,j}, T_{n,j}\}, \{S_i\}_{i=1}^n, \Psi^{(\theta)}) \\
&\qquad \propto \prod_{i=1}^{K_n - 1} \theta^{y_i} \cdot (\alpha \cdot i)^{1 - y_i} \cdot \prod_{j=1}^{t_{\cdot r} - 1} (j - 1)^{z_{r,j}} (1 - \alpha)^{1 - z_{r,j}} \text{pr}(\alpha) \\
&\qquad \propto \prod_{i=1}^{K_n - 1} (\alpha \cdot i)^{1 - y_i} \prod_r \prod_{j=1}^{t_{\cdot r} - 1} (1 - \alpha)^{1 - z_{r,j}} \alpha^{c-1} (1 - \alpha)^{d-1} \\
&\qquad \propto \alpha^{c + \sum (1 - y_i) - 1} (1 - \alpha)^{\sum (1 - z_{r,j}) + d - 1}.
\end{aligned} \tag{A.5}$$

Thus, we can sample the posterior for α as Beta$(c + \sum_{i=1}^{K_n - 1} (1 - y_i), d + \sum_r \sum_{j=1}^{t_{\cdot r} - 1} (1 - z_{r,j}))$.

We may sample from the posterior distributions of α_s and θ_s in the same way. We can now write down the full sampling scheme (this would come after the c_{srk}'s and t_{sr}'s are sampled). First, for each receiver r_{si} in, we draw a table for that particular receiver:

$$\text{pr}(k_{r_{si}} = k) \propto \begin{cases} c_{srk}^{-ji} & \text{, if } k \text{ is a previously used table.} \\ \alpha_s t_{sr}^{-ji} & \text{, if } k \text{ is a new table.} \end{cases}$$

A.2 Characterization of structured interaction exchangeable networks

Here we prove a characterization of structured interaction exchangeable networks. We focus on the case where each interaction has two components, both with two constituent elements. That is for every $i \in \mathbb{N}$, the interaction $I(i)$ is of the form $\{\{a, b\}, \{c, d\}\} \in \text{fin}_2(\mathcal{P}_1) \times \text{fin}_2(\mathcal{P}_2)$. We assume the two populations are not equivalent, but use $\mathbb{N}$ as a common space for labeling purposes.

Given an interaction-labeled network $\mathbf{y}$ with $e(\mathbf{y}) = n$, let $S : [n] \to \text{fin}_2(\mathcal{P}_1) \times \text{fin}_2(\mathcal{P}_2)$ a *selection function for* $\mathbf{y}$ if $\mathbf{y}_S = \mathbf{y}$, where $\mathbf{y}_S$ is as defined by Equation 2.1; that is, S is an interaction process whose induced interaction-labeled network agrees with $\mathbf{y}$. The selection function is a means to labeling the constituent elements of the interaction-labeled network $\mathbf{y}$, i.e., its a member of the equivalence class.

Selection functions $S, S' : [n] \to \text{fin}_2(\mathcal{P}_1) \times \text{fin}_2(\mathcal{P}_2)$ are equivalent, written $S \equiv S'$, if they are elements of the same equivalence class, i.e., $\mathbf{y}_S = \mathbf{y}_{S'}$. To every interaction-labeled network $\mathbf{y}$ consisting of n interactions associate a *canonical selection function* $S_{\mathbf{y}} : [n] \to \text{fin}_2(\mathcal{P}_1) \times \text{fin}_2(\mathcal{P}_2)$ defined by labeling the constituent elements of both components *in order of appearance*, as follows.

We initialize by setting $S_{\mathbf{y}}(1)$ equal to

- $\{\{1, 1\}\}, \{1, 1\}\}$ if both components are self loops, or
- $\{\{1, 1\}\}, \{1, 2\}\}$ if first component is a self loop, or
- $\{\{1, 2\}\}, \{1, 1\}\}$ if second components is a self loop, or

- $\{\{1,2\}\},\{1,2\}\}$ if neither are self loops.

Given $S_{\mathbf{y}}(1),\ldots,S_{\mathbf{y}}(i-1)$, we define $S_{\mathbf{y}}(i) = \{\{v_{1,1}(i),v_{1,2}(i)\},\{v_{2,1}(i),v_{2,2}(i)\}\}$ by choosing $v_{1,1}(i) \le v_{1,2}(i)$ and $v_{2,1}(i) \le v_{2,2}(i)$ to be the smallest vertex labels consistent with the structure of $\mathbf{y}|_{[i]}$. Thus, $v_{1,1}(i) = s$, $v_{1,2}(i) = s'$, $v_{2,1}(i) = r$, and/or $v_{2,2}(i) = r' \ge r$ coincides with a previously observed constituent element labels if one of the elements involved in the ith interaction corresponds to the element labeled s, r, or r' in the previous interactions $S_{\mathbf{y}}(1),\ldots,S_{\mathbf{y}}(i-1)$.

The $\text{fin}_2(\mathcal{P}_1) \times \text{fin}_2(\mathcal{P}_2)$-simplex consists of all $(f_{\{\{i,j\},\{k,l\}\}})_{j\ge i\ge -1, l\ge k\ge -1}$ such that $f_{\{i,j\},\{k,l\}} \ge 0$ for all $j \ge i \ge -1$ and $k \ge j \ge -1$, such that

- $f_{\{-1,j\},\{k,l\}} = 0$ for all $j \ne 0$,
- $f_{\{i,j\},\{-1,l\}} = 0$ for all $l \ne 0$, and
- $\sum_{j\ge i\ge -1, k\ge i\ge -1} f_{\{\{i,j\},\{k,l\}\}} = 1$.

Labels -1 and 0 for both components will be used to distinguish between various types of "blips" in the future construction. For any $f = (f_{\{\{i,j\},\{k,l\}\}})_{j\ge i\ge -1, l\ge k\ge -1}$ in the $\text{fin}_2(\mathcal{P}_1)\times \text{fin}_2(\mathcal{P}_2)$-simplex and $i \in \mathcal{P}_1$, we define

$$f_\bullet^{(i)} = \sum_{j\ge i, l\ge k\ge -1} f_{\{\{i,j\},\{k,l\}\}}, \text{ and } \quad f_\bullet^{(k)} = \sum_{j\ge i\ge -1, l\ge k} f_{\{\{i,j\},\{k,l\}\}}, \text{ and } \quad f_\bullet^{(i,k)} = \sum_{j\ge i, l\ge k} f_{\{\{i,j\},\{k,l\}\}}$$

as the sum of masses involving element i and k both independently and jointly.

Every $f = (f_{\{i,j\},\{k,l\}})_{i\ge j\ge -1, l\ge k\ge -1}$ in the $\text{fin}_2(\mathcal{P}_1) \times \text{fin}_2(\mathcal{P}_2)$-simplex determines a probability distribution on interaction-labeled networks, denoted ϵ_f, as follows. Let $X_1, X_2, \ldots$ be i.i.d. random interactions $\{\{i,j\},\{k,l\}\}$ with

$$\text{pr}(X_k = \{\{i,j\},\{k,l\}\} \mid f) = f_{\{\{i,j\},\{k,l\}\}}, \quad j \ge i \ge -1, l \ge k \ge -1. \tag{A.6}$$

Given $X_1, X_2, \ldots$, we define the selection function $\mathcal{X} : \mathbb{N} \to \text{fin}_2(\mathbb{Z}) \times \text{fin}_2(\mathbb{Z})$, where $\mathbb{Z} = \{\ldots,-1,0,1,\ldots\}$, as follows. We initialize with $m_0^{(1)} = m_0^{(2)} = 0$. For $n \ge 1$,

suppose $m_{n-1}^{(j)} = z_j \leq 0$. If neither component of X_n contains 0s, then define $\mathcal{X}(n) = X_n$ and update $m_n^{(j)} = m_{n-1}^{(j)}$. We have several potential situations:

- If $X_n = \{\{0, j\}, \{k, l\}\}$ for some $j \geq 1$ and $l \geq k \geq 1$, then we put $\mathcal{X}(n) = \{\{z_1 - 1, j\}, \{k, l\}\}$ and update $m_n^{(1)} = z_1 - 1$.

- If $X_n = \{\{0, 0\}, \{k, l\}\}$ for some $l \geq k \geq 1$, then we put $\mathcal{X}(n) = \{\{z_1 - 1, z_1 - 1\}, \{k, l\}\}$ and update $m_n^{(1)} = z_1 - 1$.

- If $X_n = \{\{-1, 0\}, \{k, l\}\}$ for some $l \geq k \geq 1$, then we put $\mathcal{X}(n) = \{\{z_1 - 1, z_1 - 2\}, \{k, l\}\}$ and update $m_n^{(1)} = z_1 - 2$.

- If $X_n = \{\{i, j\}, \{0, l\}\}$ for some $j \geq i \geq 1$ and $l \geq 1$, then we put $\mathcal{X}(n) = \{\{i, j\}, \{z_2 - 1, l\}\}$ and update $m_n^{(2)} = z_2 - 1$.

- If $X_n = \{\{i, j\}, \{0, 0\}\}$ for some $j \geq i \geq 1$, then we put $\mathcal{X}(n) = \{\{i, j\}, \{z_2 - 1, z_2 - 1\}\}$ and update $m_n^{(2)} = z_2 - 1$.

- If $X_n = \{\{i, j\}, \{0, -1\}\}$ for some $j \geq i \geq 1$, then we put $\mathcal{X}(n) = \{\{i, j\}, \{z_2 - 1, z_2 - 2\}\}$ and update $m_n^{(2)} = z_2 - 2$.

- If $X_n = \{\{0, j\}, \{0, l\}\}$ for some $j \geq 1$ and $l \geq 1$, then we put $\mathcal{X}(n) = \{\{z_1 - 1, j\}, \{z_2 - 1, l\}\}$ and update $m_n^{(1)} = z_1 - 1$ and $m_n^{(2)} = z_2 - 1$.

- If $X_n = \{\{0, -1\}, \{0, l\}\}$ for some $l \geq 1$, then we put $\mathcal{X}(n) = \{\{z_1 - 1, z_1 - 2\}, \{z_2 - 1, l\}\}$ and update $m_n^{(1)} = z_1 - 2$ and $m_n^{(2)} = z_2 - 1$.

- If $X_n = \{\{0, j\}, \{0, -1\}\}$ for some $j \geq 1$, then we put $\mathcal{X}(n) = \{\{z_1 - 1, j\}, \{z_2 - 1, z_2 - 2\}\}$ and update $m_n^{(1)} = z_1 - 1$ and $m_n^{(2)} = z_2 - 2$.

- If $X_n = \{\{0, -1\}, \{0, -1\}\}$ for some $l \geq 1$, then we put $\mathcal{X}(n) = \{\{z_1 - 1, z_1 - 2\}, \{z_2 - 1, z_2 - 2\}\}$ and update $m_n^{(1)} = z_1 - 2$ and $m_n^{(2)} = z_2 - 2$.

Events with -1 or 0 involve constituent elements in each population that appear once and never again. These "blips" were ruled out in the blip-free representation theorem, ,i.e., Theorem 2.3.2. We define $\mathbf{Y} = \mathbf{y}_{\mathcal{X}} \sim \epsilon_f$ to be the interaction-labeled network induced by $\mathcal{X}$.

Proposition A.2.1. *The interaction-labeled network $\mathbf{Y} = \mathbf{y}_{\mathcal{X}}$ corresponding to $X_1, X_2, \ldots$ i.i.d. from (2.3) is interaction exchangeable for all f in the fin$_2(\mathcal{P}_1) \times$ fin$_2(\mathbb{N})$-simplex.*

For identifiability, we define the *ranked reordering of f* by $f^{\downarrow} = (f^{\downarrow}_{\{\{i,j\},\{k,l\}\}})_{j \geq i \geq -1, l \geq l \geq -1}$, the element of the fin$_2(\mathcal{P}_1) \times$ fin$_2(\mathcal{P}_2)$-simplex obtained by putting (1) $f^{\downarrow}_{\{\{-1,0\},\{-1,0\}\}} = f_{\{\{-1,0\},\{-1,0\}\}}$, (2) $f^{\downarrow}_{\{\{0,0\},\{-1,0\}\}} = f_{\{\{0,0\},\{-1,0\}\}}$, (3) $f^{\downarrow}_{\{\{-1,0\},\{0,0\}\}} = f_{\{\{-1,0\},\{0,0\}\}}$, and (4) $f^{\downarrow}_{\{\{0,0\},\{0,0\}\}} = f_{\{\{0,0\},\{0,0\}\}}$. We reorder elements $1, 2, \ldots$ for the first component so that $f^{(i)}_{\bullet} \geq f^{(i+1)}_{\bullet}$ for all $i \geq 1$ and then breaking ties $f^{(i)}_{\bullet} = f^{(i+1)}_{\bullet}$ by declaring that $(f_{\{\{i,i\},\{k,l\}}, f_{\{\{i,i+1\},\{k,l\}\}}, \ldots)$ comes before $(f_{\{\{i+1,i+1\},\{k,l\}\}}, f_{\{\{i+1,i+2\},\{k,l\}\}}, \ldots)$ in the lexicographic ordering. We also reorder elements $1, 2, \ldots$ for the second component so that $f^{(k)}_{\bullet} \geq f^{(k+1)}_{\bullet}$ for all $k \geq 1$. Here again we break ties $f^{(k)}_{\bullet} = f^{(k+1)}_{\bullet}$ using lexicographic order. We write $\mathcal{F}^{\downarrow}$ to denote the space of rank reordered elements of the fin$_2(\mathcal{P}_1) \times$ fin$_2(\mathcal{P}_2)$-simplex.

As the vertex labels other than -1 and 0 are inconsequential, it is clear that ϵ_f and $\epsilon_{f'}$ determine the same distribution for any f, f' for which $f^{\downarrow} = f'^{\downarrow}$. For any interaction-labeled network $\mathbf{y}$, we write $|\mathbf{y}|^{\downarrow} \in \mathcal{F}^{\downarrow}$ to denote its *signature*, if it exists, as follows. Let $S_{\mathbf{y}} : \mathbb{N} \to$ fin$_2(\mathcal{P}_1) \times$ fin$_2(\mathcal{P}_2)$ be the canonical selection function for $\mathbf{y}$. For every $\{\{i,j\},\{k,l\}\} \in$ fin$_2(\mathcal{P}_1) \times$ fin$_2(\mathcal{P}_2)$, $j \geq i \geq 1$ and $l \geq k \geq 1$, we define

$$
\begin{aligned}
f_{\{\{i,j\},\{k,l\}\}}(\mathbf{y}) &= \lim_{n\to\infty} n^{-1} \sum_{m=1}^{n} \mathbf{1}\left[S_{\mathbf{y}}(m) = \{i,j\},\{k,l\}\right] \quad \text{and} \\
f^{(i,k)}_{\bullet}(\mathbf{y}) &= \lim_{n\to\infty} n^{-1} \sum_{m=1}^{n} \mathbf{1}\left[i \in S_{\mathbf{y}}(m)(1), k \in S_{\mathbf{y}}(m)(2)\right], \quad \text{and} \\
f^{(i,\{k,l\})}_{\bullet}(\mathbf{y}) &= \lim_{n\to\infty} n^{-1} \sum_{m=1}^{n} \mathbf{1}\left[i \in S_{\mathbf{y}}(m)(1), S_{\mathbf{y}}(m)(2) = \{k,l\}\right], \quad \text{and}
\end{aligned}
$$

$$f_{\bullet}^{(k,\{i,j\})}(\mathbf{y}) = \lim_{n\to\infty} n^{-1} \sum_{m=1}^{n} \mathbf{1}\left[k \in S_{\mathbf{y}}(m)(2), S_{\mathbf{y}}(m)(1) = \{i,j\}\right]$$

if the limits exist, where $S_{\mathbf{y}}(m)(\cdot)$ refers to each component of the structured interaction. We also define

$$f_{\{\{0,i\},\{k,l\}\}}(\mathbf{y}) = f_{\bullet}^{(i,\{k,l\})} - \sum_{j\geq 1} f_{\{\{i,j\},\{k,l\}\}}(\mathbf{y}), i \geq 1,$$

$$f_{\{\{i,j\},\{0,k\}\}}(\mathbf{y}) = f_{\bullet}^{(k,\{i,j\})} - \sum_{l\geq 1} f_{\{\{i,j\},\{k,l\}\}}(\mathbf{y}), \quad k \geq 1,$$

$$f_{\{\{0,i\},\{0,k\}\}}(\mathbf{y}) = f_{\bullet}^{(i,k)} - \sum_{j,l\geq 1} f_{\{\{i,j\},\{k,l\}\}}(\mathbf{y}), \quad k \geq 1,$$

$$f_{\{\{0,0\},\{k,l\}\}}(\mathbf{y}) = \lim_{n\to\infty} n^{-1} \sum_{m=1}^{n} \left(\sum_{\ell\geq 1} \mathbf{1}\left[S_{\mathbf{y}}(m) = \{\{\ell,\ell\},\{k,l\}\}\right] \right) - \sum_{i=1}^{\infty} f_{\{\{i,i\},\{k,l\}\}}(\mathbf{y}), \quad l \geq k \geq 1$$

$$f_{\{\{i,j\},\{0,0\}\}}(\mathbf{y}) = \lim_{n\to\infty} n^{-1} \sum_{m=1}^{n} \left(\sum_{\ell\geq 1} \mathbf{1}\left[S_{\mathbf{y}}(m) = \{\{i,j\},\{\ell,\ell\}\}\right] \right) - \sum_{k=1}^{\infty} f_{\{\{i,j\},\{k,k\}\}}(\mathbf{y}), \quad j \geq i \geq 1$$

$$f_{\{\{0,0\},\{0,0\}\}}(\mathbf{y}) = \lim_{n\to\infty} n^{-1} \sum_{m=1}^{n} \left(\sum_{\ell',\ell\geq 1} \mathbf{1}\left[S_{\mathbf{y}}(m) = \{\{\ell,\ell\},\{\ell',\ell'\}\}\right] \right) - \sum_{i,k=1}^{\infty} f_{\{\{i,i\},\{k,k\}\}}(\mathbf{y}),$$

Provided each of the above limiting frequencies exists, we define $|\mathbf{y}| = \left(f_{\{\{i,j\},\{k,l\}\}}(\mathbf{y})_{j\geq i\geq -1, l\geq k\geq -1}\right)$ and $|\mathbf{y}|^{\downarrow} = \left(f_{\{\{i,j\},\{k,l\}\}}(\mathbf{y})^{\downarrow}_{j\geq i\geq -1, l\geq k\geq -1}\right)$. The terms $f_{\{\{i,j\},\{k,l\}\}}(\mathbf{y})$ for $i,j,k,l \in \{0,-1\}$ are used to record the residual proportion of pure "blip" interactions and do not contribute to the limiting frequencies $f_{\{\{i,j\},\{k,l\}\}}(\mathbf{y})$ for any given $j \geq i \geq 1$ and $l \geq j \geq 1$.

Theorem A.2.2. *Let $\mathbf{Y}$ be a structured interaction exchangeable network. Then there exists a unique probability measure ϕ on $\mathcal{F}^{\downarrow}$ such that $\mathbf{Y} \sim \epsilon_{\phi}$, where*

$$\epsilon_{\phi}(\cdot) = \int_{\mathcal{F}^{\downarrow}} \epsilon_f(\cdot)\phi(df). \tag{A.7}$$

That is, every structured interaction exchangeable network $\mathbf{Y}$ can be generated by first sampling $f \sim \phi$ and, given f, putting $\mathbf{Y} = \mathbf{y}_{\mathcal{X}}$ for $\mathcal{X} : \mathbb{N} \to fin_2(\mathcal{P}_1) \times fin_2(\mathcal{P}_2)$ constructed

from $X_1, X_2, \ldots$ i.i.d. according to (2.3)*. In particular, Theorem 2.3.2 follows directly by ruling out blips.*

A.2.1 Proof of Theorem A.2.2

We equip the space with the product-discrete topology induced by the metric

$$d(\mathbf{y}, \mathbf{y}') = 1/(1 + \sup\{n \in \mathbb{N} : \mathbf{y}|_{[n]} = \mathbf{y}'|_{[n]}\}),$$

with convention $1/\infty = 0$, and $\mathcal{F}^{\downarrow}$ with the topology induced by

$$d_{\mathcal{F}^{\downarrow}}(f, f') = \sum_{j \geq i \geq -1, l \geq k \geq -1} |f_{\{\{i,j\},\{k,l\}\}} - f'_{\{\{i,j\},\{k,l\}\}}|, \quad f, f' \in \mathcal{F}^{\downarrow}.$$

We then work with the respective Borel σ-fields induced by these topologies.

Let $\mathbf{Y}$ be a structured interaction exchangeable random network, let $S_{\mathbf{Y}} : \mathbb{N} \to \text{fin}_2(\mathbb{N}) \times \text{fin}_2(\mathbb{N})$ be its canonical selection function, and let $\xi_1^{(j)}, \xi_2^{(j)}, \ldots$ for $j = 1, 2$ be two i.i.d. sequences of Uniform$[0, 1]$ random variables which are independent of $\mathbf{Y}$. Given $\mathbf{Y}$ and $(\xi_i)_{i \geq 1}$, we define $\mathcal{Z} : \mathbb{N} \to \text{fin}_2([0, 1]) \times \text{fin}_2([0, 1])$ by $\mathcal{Z}(n) = \{\{\xi_i^{(1)}, \xi_j^{(1)}\}, \{\xi_k^{(2)}, \xi_l^{(2)}\}\}$ on the event $S_{\mathbf{Y}}(n) = \{\{i, j\}, \{k, l\}\}$, for $n \geq 1$.

By independence of $\mathbf{Y}$ and $(\xi_i^{(j)})_{i \geq 1, j=1,2}$ and interaction exchangeability of $\mathbf{Y}$, $(\mathcal{Z}(n))_{n \geq 1}$ is an exchangeable sequence taking values in the Polish space $\text{fin}_2([0, 1]) \times \text{fin}_2([0, 1])$. By de Finetti's theorem, see, for example, [7], there exists a unique measure μ on the space $\mathcal{P}(\text{fin}_2([0, 1]) \times \text{fin}_2([0, 1]))$ of probability measures on $\text{fin}_2([0, 1]) \times \text{fin}_2([0, 1])$ such that $\mathcal{Z} =_{\mathcal{D}} \mathcal{Z}^* = (\mathcal{Z}^*(n))_{n \geq 1}$ with

$$\text{pr}(\mathcal{Z}^* \in \cdot) = \int_{\mathcal{P}(\text{fin}_2([0,1]) \times \text{fin}_2([0,1]))} m^{\infty}(\cdot) \mu(dm),$$

where m^{∞} denotes the infinite product measure of m. In particular, there exists a random

measure ν on $\mathrm{fin}_2([0,1]) \times \mathrm{fin}_2([0,1])$ such that

$$\mathrm{pr}(\mathcal{Z} \in \cdot \mid \nu) = \nu^\infty \quad \text{a.s.}$$

Given ν, we define

$$f_{\{\{i,j\},\{k,l\}\}} = \nu(\{\{\xi_i^{(1)}, \xi_j^{(1)}\}, \{\xi_k^{(2)}, \xi_l^{(2)}\}\}), \quad i,j,k,l \geq 1,$$

$$f_\bullet^{(i,k)} = \nu(\{\{w,x\},\{y,z\}\}) \in \mathrm{fin}_2([0,1]) \times \mathrm{fin}_2([0,1]) : \xi_i^{(1)} \in \{w,x\}, \xi_k^{(2)} \in \{y,z\}\}), i,k \geq 1,$$

$$f_{\{\{0,i\},\{0,k\}\}} = f_\bullet^{(i,k)} - \sum_{j,l=1}^{\infty} f_{\{\{i,j\},\{k,l\}\}}, \quad i,k \geq 1,$$

$$f_\bullet^{(i,\{k,l\})} = \nu(\{\{w,x\},\{\xi_k^{(2)}, \xi_l^{(2)}\}\}) \in \mathrm{fin}_2([0,1]) \times \mathrm{fin}_2([0,1]) : \xi_i^{(1)} \in \{w,x\}, \quad i \geq 1,$$

$$f_{\{\{0,i\},\{k,l\}\}} = f_\bullet^{(i,\{k,l\})} - \sum_{j=1}^{\infty} f_{\{\{i,j\},\{k,l\}\}}, \quad i,k,l \geq 1,$$

$$f_\bullet^{(k,\{i,j\})} = \nu(\{\{\xi_i^{(1)}, \xi_j^{(1)}\},\{y,z\}\}) \in \mathrm{fin}_2([0,1]) \times \mathrm{fin}_2([0,1]) : \xi_k^{(2)} \in \{y,z\}, \quad i,j,k \geq 1,$$

$$f_{\{\{i,j\},\{k,0\}\}} = f_\bullet^{(k,\{i,j\})} - \sum_{l=1}^{\infty} f_{\{\{i,j\},\{k,l\}\}}, \quad i,k,l \geq 1,$$

and

$$f_{\{\{0,0\},\{k,l\}\}} = \nu(\{\{\{u,u\},\{\xi_k^{(2)}, \xi_l^{(2)}\}\} \in \mathrm{fin}_2([0,1]) \times \mathrm{fin}_2([0,1])\}) - \sum_{i=1}^{\infty} f_{\{\{i,i\},\{k,l\}\}},$$

$$f_{\{\{i,j\},\{0,0\}\}} = \nu(\{\{\{\xi_i^{(1)}, \xi_j^{(1)}\},\{u,u\}\} \in \mathrm{fin}_2([0,1]) \times \mathrm{fin}_2([0,1])\}) - \sum_{k=1}^{\infty} f_{\{\{i,j\},\{k,k\}\}},$$

$$f_{\{\{0,0\},\{0,0\}\}} = \nu(\{\{\{u,u\},\{u',u'\}\} \in \mathrm{fin}_2([0,1]) \times \mathrm{fin}_2([0,1])\}) - \sum_{i,k=1}^{\infty} f_{\{\{i,i\},\{k,k\}\}},$$

and

$$f_{\{\{-1,0\},\{k,l\}\}} = \nu(\{\{\{u,v\},\{\xi_k^{(2)}, \xi_l^{(2)}\} \in \mathrm{fin}_2([0,1]) \times \mathrm{fin}_2([0,1]) : v \neq u\}) - \sum_{j>i\geq 1} f_{\{\{i,j\},\{k,l\}\}},$$

$$f_{\{\{i,j\},\{-1,0\}\}} = \nu(\{\{\{\xi_k^{(1)}, \xi_l^{(1)}\},\{u,v\} \in \mathrm{fin}_2([0,1]) \times \mathrm{fin}_2([0,1]) : v \neq u\}) - \sum_{l>k\geq 1} f_{\{\{i,j\},\{k,l\}\}},$$

$$f_{\{\{-1,0\},\{-1,0\}\}} = \nu(\{\{\{u,v\},\{u',v'\}\} \in \text{fin}_2([0,1]) \times \text{fin}_2([0,1]) : u \neq v, u' \neq v'\}) - \sum_{j>i\geq 1, l>k\geq 1} f_{\{\{i,j\},\{k,l\}\}}.$$

By construction $(f_{\{\{i,j\},\{k,l\}\}})_{j\geq i\geq -1, l\geq k\geq -1}$ is in the $\text{fin}_2(\mathbb{N}) \times \text{fin}_2(\mathbb{N})$-simplex and, therefore, $f^{\downarrow} \in \mathcal{F}^{\downarrow}$. Note that $\mathcal{F}^{\downarrow}$ is a subset of the $\text{fin}_2(\mathbb{N}) \times \text{fin}_2(\mathbb{N})$-simplex and $f \mapsto f^{\downarrow}$ is measurable with respect to the Borel σ-field induced by the metric $d_{\mathcal{F}^{\downarrow}}(\cdot,\cdot)$ given above.

Given ν, we let $(\mathcal{Z}', S')$ be an i.i.d. copy of $(\mathcal{Z}, S_{\mathbf{Y}})$ and let $\mathbf{Y}' = \mathbf{y}_{S'}$ be the interaction-labeled network induced by S'. We complete the proof by showing $\text{pr}(\mathbf{Y}' \in \cdot \mid \nu) = \epsilon_{f^{\downarrow}}$, for $f^{\downarrow}$ as defined above from ν.

First, let $A = \{(i,k) \in \mathbb{N} \times \mathbb{N} : f_{\bullet}^{(i,k)} > 0\}$ and $\xi_A = \{(\xi_i^{(1)}, \xi_k^{(2)}) : (i,k) \in A\}$. Second, for $j = 1, 2$ let $A^{(j)} = \{i \in \mathbb{N} : f_{\bullet}^{(j)} > 0\}$ and $\xi_A^{(j)} = \{\xi_i^{(j)} : i \in A^{(j)}\}$. It follows that

$$\begin{aligned}
&\text{pr}(\mathcal{Z}'(1) \cap \xi_A = \emptyset \mid \nu) = f_{\{\{0,0\},\{0,0\}\}} + f_{\{\{-1,0\},\{0,0\}\}} + \\
&\qquad f_{\{\{0,0\},\{-1,0\}\}} + f_{\{\{-1,0\},\{0,0\}\}} + f_{\{\{-1,0\},\{-1,0\}\}}, \\
&\text{pr}(\mathcal{Z}'(1)(1) \cap \xi_A^{(1)} = \emptyset \text{ and } \mathcal{Z}'(1)(2) \cap \xi_A^{(2)} = \{\xi_k^{(2)}\} \mid \nu) = f_{\{\{0,0\},\{0,k\}\}} + f_{\{\{-1,0\},\{0,k\}\}} + \\
&\qquad f_{\{\{0,0\},\{k,k\}\}} + f_{\{\{-1,0\},\{k,k\}\}}, \\
&\text{pr}(\mathcal{Z}'(1)(1) \cap \xi_A^{(1)} = \{\xi_i^{(1)}\} \text{ and } \mathcal{Z}'(1)(2) \cap \xi_A^{(2)} = \emptyset \mid \nu) = f_{\{\{0,i\},\{0,0\}\}} + f_{\{\{0,i\},\{-1,0\}\}} + \\
&\qquad f_{\{\{i,i\},\{0,0\}\}} + f_{\{\{i,i\},\{-1,0\}\}}, \\
&\text{pr}(\mathcal{Z}'(1)(1) \cap \xi_A^{(1)} = \{\xi_i^{(1)}\} \text{ and } \mathcal{Z}'(1)(2) \cap \xi_A^{(2)} = \{\xi_k^{(2)}\} \mid \nu) = f_{\{\{0,i\},\{0,k\}\}} + f_{\{\{i,i\},\{0,k\}\}} + \\
&\qquad f_{\{\{0,i\},\{k,k\}\}} + f_{\{\{i,i\},\{k,k\}\}},
\end{aligned}$$

and

$$\begin{aligned}
&\text{pr}(\mathcal{Z}'(1)(1) \cap \xi_A = \{\xi_i^{(1)}\} \text{ and } \mathcal{Z}'(1)(2) \cap \xi_A = \{\xi_k^{(2)}, \xi_l^{(2)}\} \mid \nu) = f_{\{\{0,i\},\{k,l\}\}} + f_{\{\{i,i\},\{k,l\}\}}, \\
&\text{pr}(\mathcal{Z}'(1)(1) \cap \xi_A = \{\xi_i^{(1)}, \xi_j^{(2)}\} \text{ and } \mathcal{Z}'(1)(2) \cap \xi_A = \{\xi_k^{(2)}\} \mid \nu) = f_{\{\{i,j\},\{0,k\}\}} + f_{\{\{i,j\},\{k,k\}\}},
\end{aligned}$$

$\text{pr}(\mathcal{Z}'(1)(1) \cap \xi_A = \{\xi_i^{(1)}, \xi_j^{(2)}\}$ and $\mathcal{Z}'(1)(2) \cap \xi_A = \{\xi_k^{(2)}, \xi_l^{(2)}\} \mid \nu) = f_{\{\{i,j\},\{k,l\}\}}$.

By exchangeability, $i \notin A^{(j)}$ implies the pair $\xi_i^{(j)}$ appears at most once in $\mathcal{Z}$ with probability 1 for $j = 1, 2$. We further have that

$$\begin{aligned}
&\text{pr}(\mathcal{Z}_1' \cap \xi_A = \emptyset \text{ and } \mathcal{Z}'(1)(1) = \{u, u\},\\
&\qquad \mathcal{Z}'(1)(1) = \{u', u'\} \text{ for some } u \neq u' \in [0,1] \mid \nu) = f_{\{\{0,0\},\{0,0\}\}},\\
&\text{pr}(\mathcal{Z}_1' \cap \xi_A = \emptyset \text{ and } \mathcal{Z}'(1)(1) = \{u, v\},\\
&\qquad \mathcal{Z}'(1)(1) = \{u', u'\} \text{ for some } u \neq v \neq u' \in [0,1] \mid \nu) = f_{\{\{-1,0\},\{0,0\}\}}\\
&\text{pr}(\mathcal{Z}_1' \cap \xi_A = \emptyset \text{ and } \mathcal{Z}'(1)(1) = \{u, u\},\\
&\qquad \mathcal{Z}'(1)(1) = \{u', v'\} \text{ for some } u \neq u' \neq v' \in [0,1] \mid \nu) = f_{\{\{0,0\},\{-1,0\}\}}\\
&\text{pr}(\mathcal{Z}_1' \cap \xi_A = \emptyset \text{ and } \mathcal{Z}'(1)(1) = \{u, v\},\\
&\qquad \mathcal{Z}'(1)(1) = \{u', v'\} \text{ for some } u \neq v \neq u' \neq v' \in [0,1] \mid \nu) = f_{\{\{-1,0\},\{-1,0\}\}}
\end{aligned}$$

Similar statements can be written to uniquely identify each element $f_{\{\{i,j\},\{k,l\}\}}$ for $j \geq i \geq -1$ and $l \geq k \geq -1$. We omit the rest simply to conserve space.

Now, define $\mathcal{X}' : \mathbb{N} \to \text{fin}_2(\mathbb{N} \cup \{-1, 0\}) \times \text{fin}_2(\mathbb{N} \cup \{-1, 0\})$ and the random selection function $S_{\mathcal{X}'} : \mathbb{N} \to \text{fin}_2(\mathbb{Z}) \times \text{fin}_2(\mathbb{Z})$ as follows. Let $m_0^{(1)} = m^{(2)} = 0$. For $n \geq 1$, suppose $m_{n-1}^{(j)} = z_j \leq 0$ for $j = 1, 2$. Then

1. If $\mathcal{Z}'(n)(1) \cap \xi_A^{(1)} = \{\xi_i^{(1)}, \xi_j^{(1)}\}$ for some $i, j \in \mathbb{N}$ and if $\mathcal{Z}'(n)(2) \cap \xi_A^{(2)} = \{\xi_k^{(2)}, \xi_l^{(2)}\}$ for some $k, l \in \mathbb{N}$, then put $\mathcal{X}'(n) = S_{X'}(n) = \{\{i, j\}, \{k, l\}\}$.

2. If $\mathcal{Z}'(n)(1) \cap \xi_A^{(1)} = \{\xi_i^{(1)}, \xi_j^{(1)}\}$ for some $i, j \in \mathbb{N}$ and if $\mathcal{Z}'(n)(2) \cap \xi_A^{(2)} = \{\xi_k^{(2)}\}$ for some $k \in \mathbb{N}$, then put $\mathcal{X}'(n) = \{\{i, j\}, \{0, k\}\}$, $S_{X'}(n) = \{\{i, j\}, \{z_2 - 1, k\}\}$ and $m_n^{(2)} = z_2 - 1$.

3. If $\mathcal{Z}'(n)(1) \cap \xi_A^{(1)} = \{\xi_i^{(1)}, \xi_j^{(1)}\}$ for some $i, j \in \mathbb{N}$ and if $\mathcal{Z}'(n)(2) \cap \xi_A^{(2)} = \emptyset$ and $\mathcal{Z}'(n)(2) = \{u, u\}$ for some $u \in [0, 1]$, then put $\mathcal{X}'(n) = \{\{i, j\}, \{0, 0\}\}$,

$S_{X'}(n) = \{\{i, j\}, \{z_2 - 1, z_2 - 1\}\}$ and $m_n^{(2)} = z_2 - 1$.

4. If $\mathcal{Z}'(n)(1) \cap \xi_A^{(1)} = \{\xi_i^{(1)}, \xi_j^{(1)}\}$ for some $i, j \in \mathbb{N}$ and if $\mathcal{Z}'(n)(2) \cap \xi_A^{(2)} = \emptyset$ and $\mathcal{Z}'(n)(2) = \{u, v\}$ for some $u \neq v \in [0, 1]$, then put $\mathcal{X}'(n) = \{\{i, j\}, \{-1, 0\}\}$, $S_{X'}(n) = \{\{i, j\}, \{z_2 - 2, z_2 - 1\}\}$ and $m_n^{(2)} = z_2 - 2$.

5. If $\mathcal{Z}'(n)(2) \cap \xi_A^{(1)} = \{\xi_k^{(2)}, \xi_l^{(2)}\}$ for some $k, l \in \mathbb{N}$ and if $\mathcal{Z}'(n)(1) \cap \xi_A^{(1)} = \{\xi_i^{(1)}\}$ for some $i \in \mathbb{N}$, then put $\mathcal{X}'(n) = \{\{0, i\}, \{k, l\}\}$, $S_{X'}(n) = \{\{z_1 - 1, i\}, \{k, l\}\}$ and $m_n^{(1)} = z_1 - 1$.

6. If $\mathcal{Z}'(n)(2) \cap \xi_A^{(1)} = \{\xi_k^{(2)}, \xi_l^{(2)}\}$ for some $k, l \in \mathbb{N}$ and if $\mathcal{Z}'(n)(1) \cap \xi_A^{(1)} = \emptyset$ and $\mathcal{Z}'(n)(1) = \{u, u\}$ for some $u \in [0, 1]$, then put $\mathcal{X}'(n) = \{\{0, 0\}, \{k, l\}\}$, $S_{X'}(n) = \{\{z_1 - 1, z_1 - 1\}, \{k, l\}\}$ and $m_n^{(1)} = z_1 - 1$.

7. If $\mathcal{Z}'(n)(2) \cap \xi_A^{(2)} = \{\xi_k^{(2)}, \xi_l^{(2)}\}$ for some $k, l \in \mathbb{N}$ and if $\mathcal{Z}'(n)(1) \cap \xi_A^{(1)} \emptyset$ and $\mathcal{Z}'(n)(1) = \{u, v\}$ for some $u \neq v \in [0, 1]$, then put $\mathcal{X}'(n) = \{\{-1, 0\}, \{k, l\}\}$, $S_{X'}(n) = \{\{z_1 - 2, z_1 - 1\}, \{k, l\}\}$ and $m_n^{(1)} = z_1 - 2$.

8. If $\mathcal{Z}'(n)(1) \cap \xi_A^{(1)} = \emptyset$ and and $\mathcal{Z}'(n)(1) = \{u, u\}$ for some $u \in [0, 1]$, and if $\mathcal{Z}'(n)(2) \cap \xi_A^{(2)} = \emptyset$ and $\mathcal{Z}'(n)(1) = \{u', u'\}$ for some $u' \in [0, 1]$, then put $\mathcal{X}'(n) = \{\{0, 0\}, \{0, 0\}\}$, $S_{X'}(n) = \{\{z_1 - 1, z_1 - 1\}, \{z_2 - 1, z_2 - 1\}\}$ and $m_n^{(1)} = z_1 - 1$ and $m_n^{(2)} = z_2 - 1$.

9. If $\mathcal{Z}'(n)(1) \cap \xi_A^{(1)} = \emptyset$ and and $\mathcal{Z}'(n)(1) = \{u, u\}$ for some $u \in [0, 1]$, and if $\mathcal{Z}'(n)(2) \cap \xi_A^{(2)} = \emptyset$ and $\mathcal{Z}'(n)(1) = \{u', v'\}$ for some $u' \neq v' \in [0, 1]$, then put $\mathcal{X}'(n) = \{\{0, 0\}, \{-1, 0\}\}$, $S_{X'}(n) = \{\{z_1 - 1, z_1 - 1\}, \{z_2 - 2, z_2 - 1\}\}$ and $m_n^{(1)} = z_1 - 1$ and $m_n^{(2)} = z_2 - 2$.

10. If $\mathcal{Z}'(n)(1) \cap \xi_A^{(1)} = \emptyset$ and and $\mathcal{Z}'(n)(1) = \{u, v\}$ for some $u \neq v \in [0, 1]$, and if $\mathcal{Z}'(n)(2) \cap \xi_A^{(2)} = \emptyset$ and $\mathcal{Z}'(n)(1) = \{u', u'\}$ for some $u' \in [0, 1]$, then put $\mathcal{X}'(n) = \{\{-1, 0\}, \{0, 0\}\}$, $S_{X'}(n) = \{\{z_1 - 2, z_1 - 1\}, \{z_2 - 1, z_2 - 1\}\}$ and $m_n^{(1)} = z_1 - 2$ and $m_n^{(2)} = z_2 - 1$.

11. If $\mathcal{Z}'(n)(1) \cap \xi_A^{(1)} = \emptyset$ and and $\mathcal{Z}'(n)(1) = \{u, v\}$ for some $u \neq v \in [0, 1]$, and if $\mathcal{Z}'(n)(2) \cap \xi_A^{(2)} = \emptyset$ and $\mathcal{Z}'(n)(1) = \{u', v'\}$ for some $u' \neq v' \in [0, 1]$, then put $\mathcal{X}'(n) = \{\{-1, 0\}, \{-1, 0\}\}$, $S_{X'}(n) = \{\{z_1 - 2, z_1 - 1\}, \{z_2 - 2, z_2 - 1\}\}$ and $m_n^{(1)} = z_1 - 2$ and $m_n^{(2)} = z_2 - 2$.

By construction, we have $S_{\mathcal{X}'} \equiv S'$ a.s. and, given f, $\mathcal{X}'$ is conditionally i.i.d. from distribution (2.3). The integral representation in (A.7) follows by de Finetti's theorem, completing the proof.

A.3 Technical Arguments

A.3.1 Proof of Thm. 4.2 and Lemma 5.1

Without loss of generality, we can assume that the number of recievers for each correspondence is equal to 1, i.e. $v_l^{(s)} = 1$ for $l = 1$ and 0 otherwise - it is trivial to show that the results hold for the more general model. Thus, the indexing is simplified, for instance $V_{n,j}$ is replaced by V_n.

Theorem A.3.1. Let $(S_1, R_1, V_1), (S_2, R_2, V_2), \ldots,$ be distributed according to the extended canonical HVCM model. Then, pr$(R_{n,j} = r|\mathcal{H}_{n,j})$ is proportional to

$$\frac{D_n(s,r) - \alpha_s V_n(s,r) + (\theta_s + \alpha_s V_n(s,r)) \left(\frac{V_n(\cdot,r) - \alpha}{m_n + \theta}\right)}{m_n(s) + \theta_s}, r \in R_n(s)$$

and

$$\frac{\theta_s + \alpha_s V_n(s,r)}{m_n(s) + \theta_s} \cdot \frac{\theta + \alpha V_n(\cdot,r)}{m_n + \theta}, r \notin R_n(s).$$

Further,

$$
\begin{aligned}
\text{pr}((S_n, R_n)_{n=1}^N, (V_n)_{n=1}^N) &= \text{pr}((R_n)_{n=1}^N, (V_n)_{n=1}^N \mid (S_n)_{n=1}^N)\, \text{pr}((S_n)_{n=1}^N), \\
\text{pr}((R_n)_{n=1}^N, (V_n)_{n=1}^N \mid (S_n)_{n=1}^N) &= \frac{[\theta+\alpha]_\alpha^{K_N-1}}{[\theta+1]_1^{m_N-1}} \prod_r [1-\alpha]_1^{V_N(\cdot,r)-1} \cdot \\
&\quad \prod_s \frac{[\theta_s+\alpha_s]_{\alpha_s}^{V_N(s,\cdot)-1}}{[\theta_s+1]_1^{m_N(s)-1}} \prod_{r=1}^{K_N} s_{\alpha_s}(V_N(s,r), D_N(s,r)), \\
\text{pr}((S_n)_{n=1}^N) &= \frac{[\tilde{\theta}+\tilde{\alpha}]_\alpha^{\mathcal{S}_N}}{[\tilde{\theta}+1]_1^N} \prod_s [1-\tilde{\alpha}]_1^{D_N^{out}(s)-1}
\end{aligned}
\tag{A.8}
$$

Proof. Let $V_{n,j}$ be the vertex that $R_{n,j}$ is assigned to. Then we have:

$$
P(R_{n,j} = r) = \sum_{i=1}^{V_{n,j}(s,\cdot)} I(l_s(i) = r) P(V_{n,j} = i),
$$

for which the first equality follows. The full distribution result is similar to that in [220], and depends on the following lemma from that paper:

Lemma A.3.2 ([220])**.** *Let v be the number of latent vertices and d be the number of observed recievers r for a particular sender s. Further, for $w = 1, 2, \ldots, v$ be a particular latent vertex and d_w its degree, and let $\bar{d} = d_1, d_2, \ldots, d_w \in C$, where C are the possible configurations of recievers assigned to latent vertices, where $d_w > 0$ and $\sum_{w=1}^v = d$, then we have the following relation:*

$$
\sum_{\bar{d} \in C} \prod_{w=1}^{v} [1-\alpha_s]_1^{d_w-1} = s_{\alpha_s}(d, v),
$$

where $s_{\alpha_s}(t_{s\cdot}, c_{s\cdot\cdot})$ is the generalized Stirling number of the first kind [110] and parameters (-1, -α_s, 0).

Combining this lemma with the joint distribution of the extended HVCM achieves the stated result. □

In order to show distributional equivalence with the stick-breaking process for $\alpha_s = 0$,

we present a modified results of [39], which demonstrates a similar distribution for the partition when dealing directly with hierarchical Pitman-Yor processes:

Proposition A.3.3. *Let (X_{ij}) be a sample drawn from the following hierarchical model:*

$$\tilde{\pi}|\theta, \alpha \sim PY(\theta, \alpha, H) \tag{A.9}$$

$$f_{r|s}|\tilde{\pi}, \theta_s, \alpha_s \sim PY(\theta_s, \alpha_s, G), \tag{A.10}$$

$$X_{rs}|f_{r|s} \sim f_{r|s}, i.i.d. \tag{A.11}$$

Then the sample generates a exchangeable partition probability function:

$$\Pi(n_1, n_2, \ldots, n_k) = \frac{[\theta+\alpha]_{\alpha}^{K_n-1}}{[\theta+1]_1^{t_{\cdot\cdot}-1}} \prod_r [1-\alpha]_1^{t_{\cdot r}-1} \prod_s \frac{[\theta_s+\alpha_s]_{\alpha_s}^{t_{s\cdot}-1}}{[\theta_s+1]_1^{c_{s\cdot\cdot}-1}} \prod_{r=1}^{k} s_{\alpha_s}(t_{sr}, c_{sr\cdot}),$$

where $s_{\alpha_s}(t_{s\cdot}, c_{s\cdot\cdot})$ is the generalized Stirling number of the first kind [110] and parameters (-1, -α_s, 0).

The above proposition is an extension to the result in [39]. From the previous proposition, note that the EPPF is distributionally equivalent to the HVCM model, as displayed in (A.8), which completes the equivalency. Note also that this result is similar to [220], although the model is slightly different, as there is no underlying discrete measure. It is modified from [39], as their result does not allow for different α_s and θ_s.

A.3.2 Proof of Lemma 4.1

This result is an extension of [221] for hierarchical Dirichlet processes. We note that a general result for $\alpha_s > 0$ is not known.

Proof. We start by stating an alternative hierarchical representation of the model

$$\tilde{\pi}|\theta,\alpha \sim \text{GEM}(\theta,\alpha) \tag{A.12}$$

$$f_{r|s}|\tilde{\pi},\theta_s,\alpha_s \sim \text{PY}(\theta_s,\alpha_s,G), \tag{A.13}$$

$$G(x) = \sum_{r=1}^{\infty} \tilde{\pi}_r \delta_r(x), \tag{A.14}$$

Where δ_y is the Dirac delta function.

We assume that for each sender s, $\alpha_s = 0$, and so $\text{PY}(\theta_s, 0, G) = \text{DP}(\theta_s, G)$, e.g., a Dirichlet process. From the aggregation principle of the Dirichlet distribution, we have:

$$\left(\sum_{r=1}^{k-1} f_{r|s}, f_{ks}, \sum_{r=k+1}^{\infty} f_{r|s}\right) \sim Dir\left(\theta_s \sum_{r=1}^{k-1} \tilde{\pi}_r, \theta_s \tilde{\pi}_k, \theta_s \sum_{r=k+1}^{\infty} \tilde{\pi}_r\right)$$

Further, by removing the first element, we have (given standard properties of the Dirichlet distribution:

$$\frac{1}{1-\sum_{r=1}^{k-1} f_{r|s}}\left(f_{k|s}, \sum_{r=k+1}^{\infty} f_{r|s}\right) \sim Dir\left(\theta_s \tilde{\pi}_k, \theta_s \sum_{r=k+1}^{\infty} \tilde{\pi}_r\right)$$

Define $\beta'_{k|s} = \frac{f_{k|s}}{1-\sum_{r=1} k-1 f_{r|s}}$, and note that that, again by standard properties of the Dirichlet distribution:

$$\beta'_{k|s} \sim Beta\left(\theta_s \tilde{\pi}_k, \theta_s \sum_{r=k+1}^{\infty} \tilde{\pi}_r\right),$$

and finally that $\sum_{r=k+1}^{\infty} \tilde{\pi}_r = 1 - \sum_{r=1}^{k} \tilde{\pi}_r$. It is straightforward algebra to show then that $\beta'_{r|s} \prod_{k=1}^{r-1}(1-\beta'_{k|s}) = f_{r|s}$. □

A.3.3 Proof of Theorem 2.5.3

For a sequence of real-valued random variables $(X_n)_{n\geq 0}$ and a sequence real-valued non-random variables $(\lambda(n))_{n\geq 0}$, we say $X_n \simeq \lambda(n)$ if, for $n \to \infty$, $\lim_n X_n/\lambda(n)$ exists

almost surely and equals a finite and positive random variable. Theorem 2.5.3 is a generalization of Theorem 7 in [39].

Proof. K_n denotes the number of unique receivers seen after n e-mails in the HVCM. Let $K_{0,n}$ denote the number of *unique* receivers seen from the *global* distribution. Finally, let $\mu_s = \sum_{l=1}^{\infty} l\nu_l^{(s)}$ and $\mu := \sum_s p_s\mu_s$. It can be shown that $K_n = K_{0,\eta(\boldsymbol{n})}, a.s.$, where $\eta(\boldsymbol{n}) = K_{1,n_1} + K_{2,n_2} + \ldots + K_{d,n_d}$, and K_{s,n_s} is the number of unique receivers for the sth sender distribution. By [189], we have the following relations:

$$\frac{K_{0,n}}{\lambda_0(\mu n)} \xrightarrow{a.s.} M_0, \quad \frac{K_{s,n_s}}{\lambda_s(\mu_s\, n_s)} \,\&\, \frac{K_{s,n_s}}{\lambda_s(\mu_s\, p_s\, n)} \xrightarrow{a.s.} M_s, \tag{A.15}$$

where $\lambda_0(x) = x^{\alpha_0}, \lambda_s(x) = x^{\alpha_s}$, and $M_0, \{M_s\}_{s=1}^d$ are Mittag-Leffler random variables. Now define $\lambda_\star(x) = \lambda_{s^\star}(x)$, $\mu_\star = \mu_{s^\star}$, $n_\star = n_{s^\star}$, and $\alpha_\star = \alpha_{s^\star}$ where $s^\star = \arg\max_{s\in[d]} \alpha_s$. Then

$$\frac{\eta(\boldsymbol{n})}{\lambda_\star(\mu_\star n_\star)} = \sum_{s=1}^{d} \frac{K_{s,n_s}}{\lambda_\star(\mu_\star n_\star)} = \sum_{s=1}^{d} \frac{K_{s,n_s}/\lambda_s(\mu_s n_s)}{(\mu_\star n_\star)^{\alpha_\star}(\mu_s n_s)^{-\alpha_s}} \to \sum_{s=1}^{d} \frac{K_{s,n_s}/\lambda_s(\mu_s n_s)}{(\mu_\star p_\star)^{\alpha_\star}(\mu_s p_s)^{-\alpha_s}} n^{\alpha_s-\alpha_\star}$$

For $\alpha_s = \alpha_\star = \max_s \alpha_s$, the summand goes to M_s. For $\alpha_s \neq \alpha_*$, the summand goes to 0 almost surely. Therefore,

$$\frac{\eta(\boldsymbol{n})}{\lambda_\star(\mu_\star n_\star)} \xrightarrow{a.s.} M_{s^\star} =: M_\star. \tag{A.16}$$

This implies

$$\frac{K_{0,\eta(\boldsymbol{n})}}{K_{0,M_\star\lambda_\star(\mu_\star n_\star)}} = \frac{\lambda_0(\mu\eta(\boldsymbol{n}))}{\lambda_0(\mu M_\star\lambda_\star(\mu_\star n_\star))} \frac{K_{0,\eta(\boldsymbol{n})}/\lambda_0(\mu\eta(\boldsymbol{n}))}{K_{0,M_\star\lambda_\star(\mu_\star n_\star)}/\lambda_0(\mu M_\star\lambda_\star(\mu_\star n_\star))} \xrightarrow{a.s.} 1.$$

where the first term goes to 1 almost surely by (A.16) and the second by (A.15). This implies

$$\frac{K_n}{\lambda_0(\mu M_\star\lambda_\star(\mu_\star n_\star))} = \frac{K_{0,\eta(\boldsymbol{n})}}{K_{0,M_\star\lambda_\star(\mu_\star n_\star)}} \frac{K_{0,M_\star\lambda_\star(\mu_\star n)}}{\lambda_0(\mu M_\star\lambda_\star(\mu_\star n_\star))} \xrightarrow{a.s.} M_0$$

so $K_n \simeq \left(\mu^{1/\alpha_\star}\mu_\star p_\star n\right)^{\alpha_0\alpha_\star}$. □

	Num. Verts.	Deg. 1	Deg. 10
Hier. Edge-Ex Model	**(412429, 415870)**	(207581, 210207)	(4081, 4353)
Hollywood Model	**(411375, 414523)**	(225586, 228006)	**(3695, 3943)**
Real Value	413029	224791	3733

Table A.1: Posterior predictive intervals for global statistics.

	Num. Verts.	Deg. 1	Deg. 10
Hier. Edge-Ex Model	9 / 98	6 / 98	68 / 93
Hollywood Model	0 / 98	0 / 98	0 / 98

Table A.2: Posterior predictive coverage rates of the local distributions when using the 95% posterior predictive interval.

A.4 ArXiv Dataset Details

A.4.1 Posterior Predictive Validation (PPV)

Table A.1, Table A.2, and Figure A.1 summarize the posterior predictive validation for the ArXiv dataset. The proposed model performs well on the number of unique vertices, and number of authors with 10 papers. It also improves over the Hollywood model for local posterior predictive coverage.

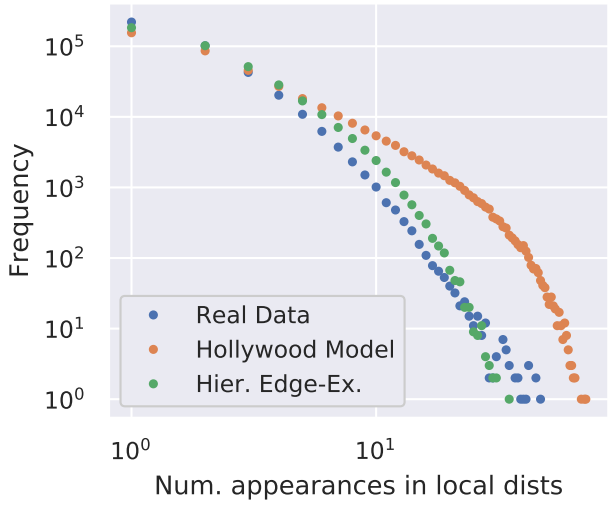


Figure A.1: Distribution of nodes that are in x number of local subject distributions

A.4.2 Comparison of Subject Overlap Results to Co-authorship network

One comparator to the results in Section 2.8.1 is to contruct a weighted network, where each edge between subjects s_1 and s_2 is weighted by the amount of articles that are listed in both s_1 and s_2. Figure A.2 shows the results of normalized spectral clustering [166] on this network that is also regularized according to the recommendation in [131], which is adding d/n to each edge weight, there d is the maximum degree and n is the number of nodes. Six clusters were used, as that was what was used in the subject overlap study.

This clustering omits a significantly different result than that in the main chapter. While the subject overlap clustering from the HVCM in Section 2.8.1 is able to reconstruct the arXiv-created meta classes of computer science, physics, and statistics, the naive spectral clustering on the coauthorship network lumps these metaclasses together. Further, the result in Figure A.2 has two clusters of size 2 that have high degree, whereas the subject overlap clustering appropriately normalizes this scale free degree structure and so does not suffer from this bias towards high degree. Finally, the HVCM was able to uncover two clusters that represent the relatively new (and cross-disciplinary) machine-learning class, where the clustering on the coauthorship network does not.

Number of Subjects	Number of Articles	Percentage (%)
1	234916	45.9
2	169138	33.1
3	70301	13.8
4	26203	5.1
5	7836	1.5
6	1925	0.3
$\geq$7	493	0.01

Table A.3: Number of articles with x amount of subject subclasses.

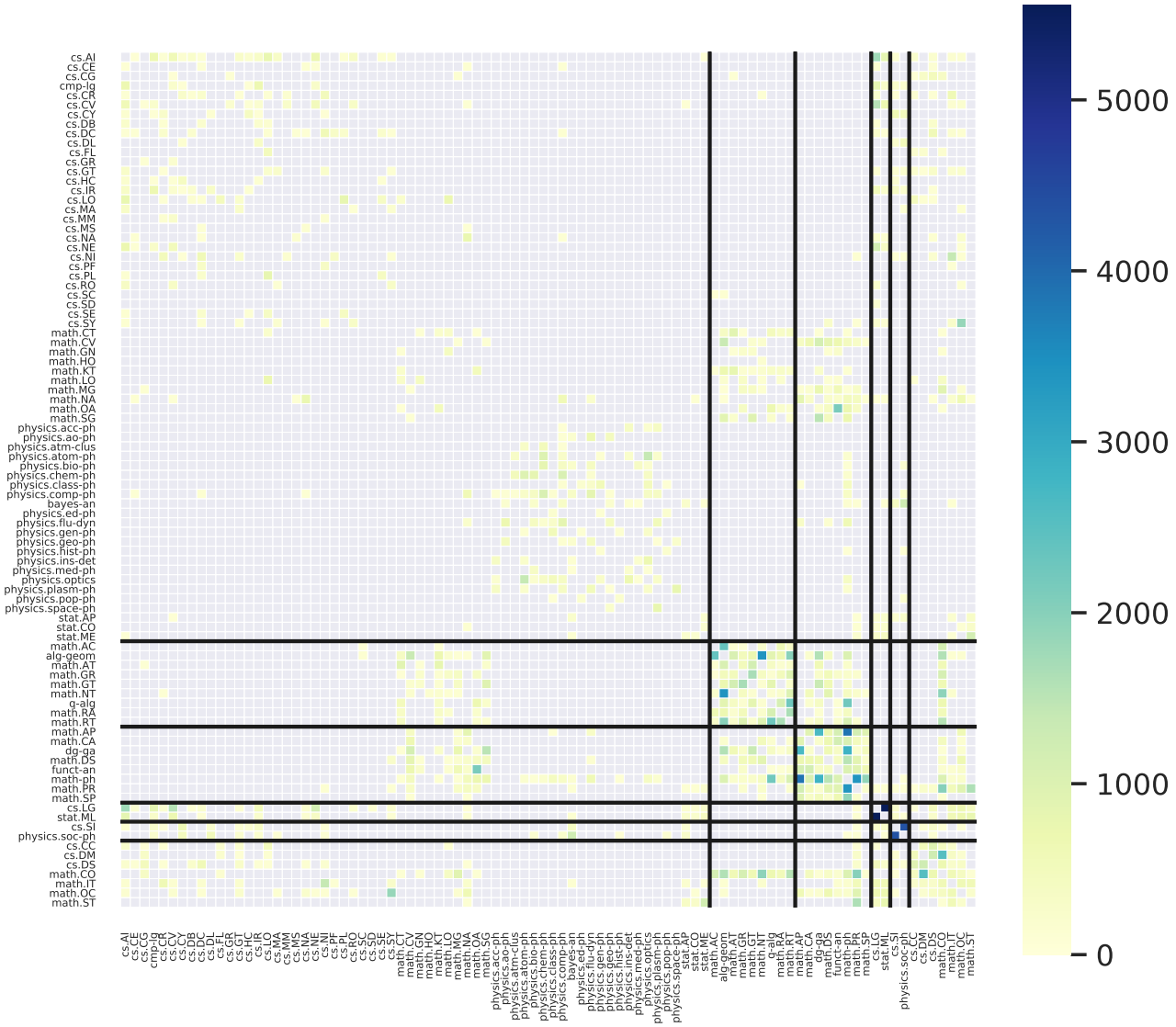


Figure A.2: Results of spectral clustering on the co-authorship network. The resulting clusters combines most of the computer science, math, and physics subjects, and also creates two clusters each with two high degree members only.

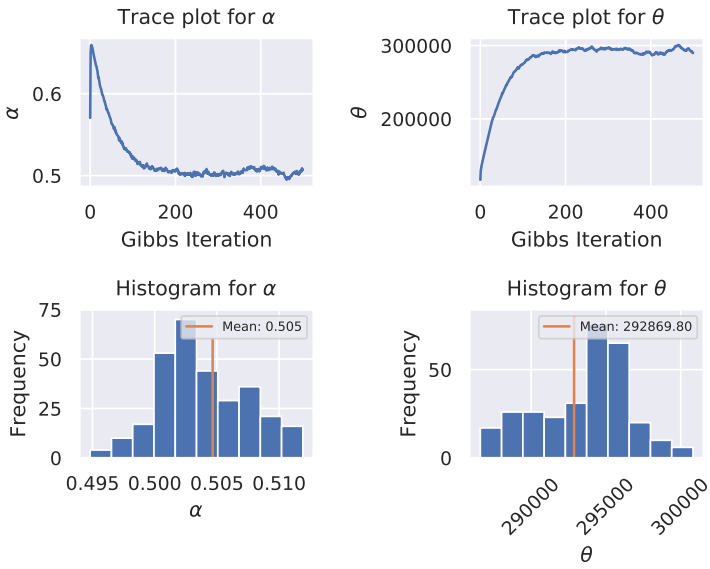


Figure A.3: Trace plot for α and θ, and posterior mean for the arXiv dataset. The posterior mean excluded the first 200 samples of burn-in.

APPENDIX B

Supplemental Material for Chapter 3

B.1 Proofs of Main Theorems

Here we prove Theorems 3.2.1-3.4.1. Throughout this section, we use notations δ_{ij} and δ_{ij}^m for FR and generalized FR test statistic, as defined in the chapter. D represents the HP divergence and $f^{(m)}(\mathbf{x})$ is the marginal distribution of random vector $\mathbf{X}$; $\mathbb{E}$ stands for expectation.

B.1.1 Theorem 3.2.1

The part (a) can be easily derived. Here we provide the proof of part (b). It can be seen that there exists a constant C_1 depending on the p_i and p_j such that for every f_i and f_j

$$f_i(\mathbf{x})f_j(\mathbf{x}) \leq C_1 \left(p_i f_i(\mathbf{x}) + p_j f_j(\mathbf{x})\right)^2. \tag{B.1}$$

Set

$$g_{ij}(\mathbf{x}) := \left(p_i f_i(\mathbf{x}) + p_j f_j(\mathbf{x})\right) \sum_{k \neq i,j} p_k f_k(\mathbf{x}) \big/ f^{(m)}(\mathbf{x}). \tag{B.2}$$

The inequality (3.8) is equivalent to

$$0 \leq f_i(\mathbf{x})f_j(\mathbf{x}) \left(\frac{1}{p_i f_i(\mathbf{x}) + p_j f_j(\mathbf{x})} - \frac{1}{f^{(m)}(\mathbf{x})} \right) \leq C_1 \, g_{ij}(\mathbf{x}). \tag{B.3}$$

Therefore

$$\int \frac{f_i(\mathbf{x})f_j(\mathbf{x})}{p_i f_i(\mathbf{x}) + p_j f_j(\mathbf{x})}\mathrm{d}\mathbf{x} \leq \int \frac{f_i(\mathbf{x})f_j(\mathbf{x})}{f^{(m)}(\mathbf{x})}\mathrm{d}\mathbf{x} + C_1\Big(\int g_{ij}(\mathbf{x})\ \mathrm{d}\mathbf{x}\Big) \tag{B.4}$$

On the other hand, we have

$$\begin{aligned} D\Big(\widetilde{p}_{ij}f_i + \widetilde{p}_{ji}f_j, \sum_{k\neq i,j}\widetilde{p}_k^{ij}f_k\Big) = \\ 1 - \frac{1}{(p_i+p_j)\sum_{r\neq i,j}p_r}\int g_{ij}(\mathbf{x})\ \mathrm{d}\mathbf{x}, \end{aligned} \tag{B.5}$$

where $\widetilde{p}_{ij}$ and $\widetilde{p}_{ji}$ are as before and $\widetilde{p}_k^{ij} = p_k/\sum_{r\neq i,j}p_r$. Hence,

$$\int g_{ij}(\mathbf{x})\ \mathrm{d}\mathbf{x} = \mathrm{C}_2\Big\{1 - D\Big(\widetilde{p}_{ij}f_i + \widetilde{p}_{ji}f_j, \sum_{k\neq i,j}\widetilde{p}_k^{ij}f_k\Big)\Big\}. \tag{B.6}$$

where C_2 is a constant depending on priors $p_1, p_2, \ldots p_m$. This together with (B.4) implies that there exists a constant C depending only on priors $p_1, p_2, \ldots, p_m$ such that

$$\begin{aligned} \int \frac{f_i(\mathbf{x})f_j(\mathbf{x})}{p_i f_i(\mathbf{x}) + p_j f_j(\mathbf{x})}\mathrm{d}\mathbf{x} \leq \int \frac{f_i(\mathbf{x})f_j(\mathbf{x})}{f^{(m)}(\mathbf{x})}\mathrm{d}\mathbf{x} \\ +C\Big(1 - D\Big(\widetilde{p}_{ij}f_i + \widetilde{p}_{ji}f_j, \sum_{k\neq i,j}\widetilde{p}_k^{ij}f_k\Big)\Big), \end{aligned} \tag{B.7}$$

By recalling HP_{ij} (3.4) and GHP_{ij}^m (3.5) we conclude the result.

B.1.2 Theorem 3.3.1

To derive the inequality in (3.17), first we need to prove the following lemma:

Lemma B.1.1. *Let $a_1, a_2, \ldots, a_m$ be a probability distribution on m classes so that $\sum_{i=1}^{m} a_i =$*

1. *Then*

$$1 - \max_i a_i \leq 2 \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} a_i a_j. \tag{B.8}$$

Proof. Assume, without loss of generality, that the a_i have been reordered in such a way that a_m is the largest. So it is sufficient to prove that

$$1 - a_m \leq 2 \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} a_i a_j. \tag{B.9}$$

Since $\sum_{i=1}^{m} a_i = 1$ then

$$1 - a_m = \sum_{i=1}^{m-1} a_i = \sum_{i=1}^{m-1} \sum_{j=1}^{m} a_i a_j.$$

Therefore we need to show that

$$\sum_{i=1}^{m-1} \sum_{j=1}^{m} a_i a_j \leq 2 \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} a_i a_j. \tag{B.10}$$

The LHS in (B.10) is

$$\sum_{i=1}^{m-1} \sum_{j=1}^{m} a_i a_j = 2 \sum_{i=1}^{m-2} \sum_{j=i+1}^{m-1} a_i a_j + \sum_{i=1}^{m-1} a_i a_m + \sum_{i=1}^{m-1} a_i^2. \tag{B.11}$$

And the RHS in (B.10) is written as

$$2 \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} a_i a_j = 2 \sum_{i=1}^{m-2} \sum_{j=i+1}^{m-1} a_i a_j + 2 \sum_{i=1}^{m-1} a_i a_m. \tag{B.12}$$

Recalling our assumption that a_m is the largest we have

$$\sum_{i=1}^{m-1} a_i^2 \leq \sum_{i=1}^{m-1} a_i a_m. \tag{B.13}$$

This implies that $(B.11) \leq (B.12)$. This concludes (B.10) and proves our Lemma. □

Going back to prove upper bound (3.17) in Theorem 3.3.1, let $p_1 f_1(\mathbf{x}), p_2 f_2(\mathbf{x}), \ldots, p_m f_m(\mathbf{x})$ be joint probabilities of $\mathbf{x}$ and i. And denote $p(i|\mathbf{x}) := P(y = i|\mathbf{x})$ where variable $y \in \{1, 2, \ldots, m\}$ is class label with priors p_i. The BER for m classes is given by

$$\begin{aligned} \epsilon^m &= 1 - \int \max\left\{p_1 f_1(\mathbf{x}), \ldots, p_m f_m(\mathbf{x})\right\} \mathrm{d}\mathbf{x} \\ &= \mathbb{E}_{\mathbf{X}}\Big[1 - \max_{i=1,\ldots,m} p(i|\mathbf{x})\Big], \end{aligned} \tag{B.14}$$

Moreover the marginal density for random vector $\mathbf{X}$ is

$$f_{\mathbf{X}}(\mathbf{x}) = \sum_{i=1}^{m} p_i f_i(\mathbf{x}) = f^{(m)}(\mathbf{x}),$$

And

$$\begin{aligned} \int \frac{p_i p_j f_i(\mathbf{x}) f_j(\mathbf{x})}{f^{(m)}(\mathbf{x})} \mathrm{d}\mathbf{x} &= \int \left(\frac{p_i f_i(\mathbf{x})}{f^m(\mathbf{x})}\right) \left(\frac{p_j f_j(\mathbf{x})}{f^{(m)}(\mathbf{x})}\right) f^{(m)}(\mathbf{x}) \mathrm{d}\mathbf{x} \\ &= \int p(i|\mathbf{x}) p(j|\mathbf{x}) f^{(m)}(\mathbf{x}) \mathrm{d}\mathbf{x} = \mathbb{E}_{\mathbf{X}}\left[p(i|\mathbf{X}) p(j|\mathbf{X})\right]. \end{aligned} \tag{B.15}$$

Therefore (3.17) turns into the following claim:

$$\mathbb{E}_{\mathbf{X}}\Big[1 - \max_{i=1,\ldots,m} p(i|\mathbf{x})\Big] \leq \mathbb{E}_{\mathbf{X}}\left[2 \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} p(i|\mathbf{X}) p(j|\mathbf{X})\right]. \tag{B.16}$$

We know that $\sum_{i=1}^{m} p(i|\mathbf{x}) = 1$. Using Lemma B.1.1 where a_i represents $p(i|\mathbf{x})$ we have

$$1 - \max_{i=1,\ldots,m} p(i|\mathbf{x}) \leq 2 \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} p(i|\mathbf{X}) p(j|\mathbf{X}). \tag{B.17}$$

Hence, we prove the inequality (B.16) and consequently our claim (3.17).

Next we prove the lower bound (3.18). The following lemma is required:

Lemma B.1.2. *For all $a_1, a_2, ..., a_m$ such that $\sum_{i=1}^{m} a_i = 1$, we have the following:*

$$\frac{m-1}{m}\left[1-\left(1-2\frac{m}{m-1}\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}a_i a_j\right)^{1/2}\right] \leq 1-\max_i a_i. \tag{B.18}$$

Proof. After some algebra, we rewrite the inequality in the following form:

$$m(\max_i a_i)^2 - 2\max_i a_i \leq m-2-(m-1)b,$$

where $b = 2\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} a_i a_j$. Without loss of generality, we can assume that the a_is are ordered, so that a_m is the largest. Then we have that $\max_i a_i = 1 - \sum_{i=1}^{m-1} a_i$.

Using this equality on the left side, expanding the square, and subtracting $m-2$ from both sides, we have:

$$m(\sum_{i=1}^{m-1} a_i)^2 - (2m-1)\sum_{i=1}^{m-1} a_i \leq -(m-1)b. \tag{B.19}$$

Expanding terms once again:

$$\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} a_i a_j = \sum_{i=1}^{m-2}\sum_{j=i+1}^{m-1} a_i a_j + \sum_{i=1}^{m-1} a_i a_m, \tag{B.20}$$

and collecting like terms:

$$\begin{aligned} m\sum_{i=1}^{m-1} a_i^2 + (4m-2)\sum_{i=1}^{m-2}\sum_{j=i+1}^{m-1} a_i a_j - (2m-1)\sum_{i=1}^{m-1} a_i \\ \leq -2(m-1)\sum_{i=1}^{m-1} a_i a_m \end{aligned} \tag{B.21}$$

We note, that since $\sum_{i=1}^{m} a_i = 1$, we have the following:

$$\sum_{i=1}^{m-1} a_i = \sum_{i=1}^{m-1}\sum_{j=1}^{m} a_i a_j = 2\sum_{i=1}^{m-2}\sum_{j=i+1}^{m-1} a_i a_j + \sum_{i=1}^{m-1} a_i a_m + \sum_{i=1}^{m-1} a_i^2.$$

Plugging in once more:

$$(1-m)\sum_{i=1}^{m-1} a_i^2 - (2m-1)\sum_{i=1}^{m-1} a_i a_m \leq -2(m-1)\sum_{i=1}^{m-1} a_i a_m,$$

or equivalently:

$$(1-m)\sum_{i=1}^{m-1} a_i^2 - \sum_{i=1}^{m-1} a_i a_m \leq 0.$$

Note that since $a_m = \max_i a_i$, $\sum_{i=1}^{m-1} a_i a_m \geq \sum_{i=1}^{m-1} a_i^2$, so that

$$(1-m)\sum_{i=1}^{m-1} a_i^2 - \sum_{i=1}^{m-1} a_i a_m \leq -m\sum_{i=1}^{m-1} a_i^2 \leq 0,$$

since $\sum_{i=1}^{m-1} a_i^2 \geq 0$. □

Now to prove (3.18), let $p_1 f_1(\mathbf{x}), p_2 f_2(\mathbf{x}), \ldots, p_m f_m(\mathbf{x})$ be joint probabilities of $\mathbf{x}$ and i. And denote $p(i|\mathbf{x}) := P(y = i|\mathbf{x})$ where variable $y \in \{1, 2, \ldots, m\}$ is class label with priors p_i. By taking the expectation from both sides of (B.18) when $a_i = p(i|\mathbf{x})$, we have

$$\begin{aligned} &\mathbb{E}_{\mathbf{X}}[1 - \max_i p(i|\mathbf{x})] \\ &\quad \geq \frac{m-1}{m}\left[1 - \mathbb{E}_{\mathbf{X}}\left(1 - 2\frac{m}{m-1}\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} p(i|\mathbf{x})p(j|\mathbf{x})\right)^{1/2}\right], \end{aligned} \tag{B.22}$$

Further, since $\phi(\mathbf{x}) = \sqrt{\mathbf{x}}$ is a concave function, by applying Jensen inequality the RHS in (B.22) is lower bounded by

$$\frac{m-1}{m}\left[1 - \left(1 - 2\frac{m}{m-1}\mathbb{E}_{\mathbf{X}}\left[\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} p(i|\mathbf{x})p(j|\mathbf{x})\right]\right)^{1/2}\right], \tag{B.23}$$

And we know that

$$\mathbb{E}_{\mathbf{X}}\left[p(i|\mathbf{x})p(j|\mathbf{x})\right] = \delta_{ij}^{m},$$

and

$$\mathbb{E}_{\mathbf{X}}[1 - \max_{i} p(i|\mathbf{x})] = \epsilon^{m},$$

then this proves our proposed lower bound in (3.18).

B.1.3 Theorem 3.3.2

To derive (3.19), the following lemma is required to be proved:

Lemma B.1.3. *Let $a_1, a_2, \ldots, a_m$ be probability distributions on m classes so $\sum_{i=1}^{m} a_i = 1$. Then, for $m \geq 3$ and log basis 2, we have*

$$2\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} a_i a_j \leq -\frac{1}{2}\sum_{i=1}^{m} a_i \log a_i. \tag{B.24}$$

Proof. The claim in (B.24) can be rewritten as

$$4\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} a_i a_j \leq \sum_{i=1}^{m} a_i \log \frac{1}{a_i}, \tag{B.25}$$

where $0 \leq a_i \leq 1$. In addition we have

$$4\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} a_i a_j = 4\sum_{i=1}^{m-2}\sum_{j=i+1}^{m-1} a_i a_j + 4\sum_{i=1}^{m-1} a_i a_m, \tag{B.26}$$

and

$$\sum_{i=1}^{m-1} a_i - \sum_{i=1}^{m-1} a_i a_m - \sum_{i=1}^{m-1} a_i^2 = 2\sum_{i=1}^{m-2}\sum_{j=i+1}^{m-1} a_i a_j. \tag{B.27}$$

Combining (B.26) and (B.27), we have

$$\begin{aligned} 4\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} a_i a_j &= 2\sum_{i=1}^{m-1} a_i + 2\sum_{i=1}^{m-1} a_i a_m - 2\sum_{i=1}^{m-1} a_i^2 \\ &= 2(1-a_m^2) - 2\sum_{i=1}^{m-1} a_i^2. \end{aligned} \tag{B.28}$$

Hence we need to show that

$$2(1-a_m^2) - 2\sum_{i=1}^{m-1} a_i^2 \leq \sum_{i=1}^{m} a_i \log\frac{1}{a_i}. \tag{B.29}$$

Equivalently

$$2 - 2\sum_{i=1}^{m} a_i^2 \leq \sum_{i=1}^{m} a_i \log\frac{1}{a_i}. \tag{B.30}$$

Or

$$g(m) := \sum_{i=1}^{m} a_i(2 - 2a_i + \log a_i) \leq 0. \tag{B.31}$$

Since for $a_i \leq 1/2$ the function $a_i\big(2-2a_i+\log a_i\big)$ is negative and we know that $\sum_{i=1}^{m} a_i = 1$, therefore $g(m)$ is a decreasing function in m i.e. $g(m) \leq g(3)$ for $m \geq 3$. And it can be easily checked that $g(3) \leq 0$. Hence the proof is completed.

□

Now, Following arguments in [135], one can check that

$$\frac{1}{2}\big(H(p) - JS(f_1, f_2, \ldots, f_m)\big) = -\frac{1}{2}\mathbb{E}_{\mathbf{X}}\left[\sum_{i=1}^{m} p(i|\mathbf{X}) \log p(i|\mathbf{X})\right]. \tag{B.32}$$

Further, in Theorem 3.3.1, we derived

$$2\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} \delta_{ij}^{m} = \mathbb{E}_{\mathbf{X}}\left[2\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} p(i|\mathbf{X})p(j|\mathbf{X})\right], \tag{B.33}$$

such that $\sum_{i=1}^{m} p(i|\mathbf{x}) = 1$. Using Lemma B.1.3, where again $a_i = p(i|\mathbf{x})$, we have

$$2\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} p(i|\mathbf{X})p(j|\mathbf{X}) \leq -\frac{1}{2}\sum_{i=1}^{m} p(i|\mathbf{X})\log p(i|\mathbf{X}). \tag{B.34}$$

Taking expectation from both sides of (B.34) proves our claim in (3.19).

Next, we prove the lower bound in (3.20). Similar to Appendices B and C let $p(i|\mathbf{x})$ be the posterior probabilities. Therefore we can rewrite (3.22) in terms of $p(i|\mathbf{x})$ as

$$\begin{aligned}\frac{m-1}{m}\left[1-\left(1-2\frac{m}{m-1}\mathbb{E}_{\mathbf{X}}\left[\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} p(i|\mathbf{x})p(j|\mathbf{x})\right]\right)^{1/2}\right] \\ \geq \frac{1}{4(m-1)}\left(\mathbb{E}_{\mathbf{X}}\left[\sum_{i=1}^{m} p(i|\mathbf{x})\log p(i|\mathbf{x})\right]\right)^2.\end{aligned} \tag{B.35}$$

Analogous to other proofs, let $a_i = p(i|\mathbf{x})$ and to shorten the formula set

$$A(\mathbf{x}) = 1 - 2\frac{m}{m-1}\left[\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} a_i a_j\right],$$

therefore (B.35) can be rewritten as

$$\frac{m-1}{m}\left[1-\sqrt{\mathbb{E}_{\mathbf{X}}[A(\mathbf{X})]}\right] \geq \frac{1}{4(m-1)}\left(\mathbb{E}_{\mathbf{X}}\left[\sum_{i=1}^{m} a_i \log a_i\right]\right)^2. \tag{B.36}$$

Equivalently

$$\left[1-\sqrt{\mathbb{E}_{\mathbf{X}}[A(\mathbf{X})]}\right] \geq \frac{m}{4(m-1)^2}\left(\mathbb{E}_{\mathbf{X}}\left[\sum_{i=1}^{m} a_i \log a_i\right]\right)^2. \tag{B.37}$$

Multiple the both sides of (B.35) in $1+\sqrt{\mathbb{E}_{\mathbf{X}}[A(\mathbf{X})]}$:

$$\begin{aligned}&[1-\mathbb{E}_{\mathbf{X}}[A(\mathbf{X})]]\\&\quad\geq\frac{m}{4(m-1)^2}\left(\mathbb{E}_{\mathbf{X}}\left[\sum_{i=1}^{m}a_i\log a_i\right]\right)^2\left(1+\sqrt{\mathbb{E}_{\mathbf{X}}[A(\mathbf{X})]}\right).\end{aligned}\tag{B.38}$$

And we have

$$1-\mathbb{E}_{\mathbf{X}}[A(\mathbf{X})]=2\frac{m}{m-1}\mathbb{E}_{\mathbf{X}}\left[\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}a_ia_j\right].$$

And since $\sqrt{\mathbb{E}_{\mathbf{X}}[A(\mathbf{X})]}\leq 1$ then $1+\sqrt{\mathbb{E}_{\mathbf{X}}[A(\mathbf{X})]}\leq 2$, so it is sufficient to prove that

$$\mathbb{E}_{\mathbf{X}}\left[\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}a_ia_j\right]\geq\frac{1}{4(m-1)}\left(\mathbb{E}_{\mathbf{X}}\left[\sum_{i=1}^{m}a_i\log a_i\right]\right)^2.\tag{B.39}$$

On the other hand we know that by using Jensen inequality

$$\left(\mathbb{E}_{\mathbf{X}}\left[\sum_{i=1}^{m}a_i\log a_i\right]\right)^2\leq\mathbb{E}_{\mathbf{X}}\left[\sum_{i=1}^{m}a_i\log a_i\right]^2,$$

so we only need to show that

$$\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}a_ia_j\geq\frac{1}{4(m-1)}\left[\sum_{i=1}^{m}a_i\log a_i\right]^2.\tag{B.40}$$

Or

$$4(m-1)\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}a_ia_j\geq\left[\sum_{i=1}^{m}a_i\log a_i\right]^2.\tag{B.41}$$

Recalling (B.28) in Appendix B.1.3 this is equivalent to

$$2(m-1)\left(1-\sum_{i=1}^{m}a_i^2\right)\geq\left[\sum_{i=1}^{m}a_i\log a_i\right]^2.\tag{B.42}$$

Now let $g(m)$ be

$$2(m-1)\left(1-\sum_{i=1}^{m} a_i^2\right)-\left[\sum_{i=1}^{m} a_i \log a_i\right]^2,$$

this is non-negative when $m=3$, $g(3)\geq 0$. In addition g is an increasing function in m i.e. $g(m)\geq g(3)$. Therefore following similar arguments as showing (B.31) the proof of (3.20) is completed.

B.1.4 Theorem 3.3.3

Recalling the pairwise bound (3.11), the multi-class classification Bayes error HP bound is given as

$$\epsilon^m \leq 2\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} \delta_{ij}. \tag{B.43}$$

Since $\delta_{ij}^m \leq \delta_{ij}$, our proposed bound (3.17) is tighter than (B.43). This implies (3.21).

To derive (3.22), let us first focus on $u_{\tilde{p}_{ij}}$:

$$\begin{aligned}
u_{\tilde{p}_{ij}} &= 4\widetilde{p}_{ij}\widetilde{p}_{ji}\, D_{\tilde{p}_{ij}}(f_i,f_j)+(\widetilde{p}_{ij}-\widetilde{p}_{ji})^2 \\
&= 1-\frac{4p_ip_j}{p_i+p_j}\int \frac{f_i(\mathbf{x})f_j(\mathbf{x})}{p_if_i(\mathbf{x})+p_jf_j(\mathbf{x})}\,\mathrm{d}\mathbf{x} \\
&= 1-\frac{4}{p_i+p_j}\int \frac{p_if_i(\mathbf{x})p_jf_j(\mathbf{x})/(f^m(\mathbf{x}))^2}{\big(p_if_i(\mathbf{x})+p_jf_j(\mathbf{x})\big)/f^m(\mathbf{x})} f^m(\mathbf{x})\,\mathrm{d}\mathbf{x} \\
&= 1-\frac{4}{p_i+p_j}\mathbb{E}_{\mathbf{x}}\left[\frac{a_ia_j}{a_i+a_j}\right],
\end{aligned} \tag{B.44}$$

where $a_i=P(i|\mathbf{x})=p_if_i/f^{(m)}$. Therefore the RHS in (3.22) can be written as

$$\frac{1}{m}\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}(p_i+p_j)\left[1-\sqrt{1-\frac{4}{p_i+p_j}\mathbb{E}_{\mathbf{x}}\left[\frac{a_ia_j}{a_i+a_j}\right]}\right]. \tag{B.45}$$

Furthermore the LHS in (3.22) can be rewrite in terms of a_i and a_j as

$$\frac{m-1}{m}\left[1-\left(1-2\frac{m}{m-1}\mathbb{E}_{\mathbf{X}}\left[\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}a_i a_j\right]\right)^{1/2}\right]. \tag{B.46}$$

Note that since $\sum_{i}^{m} p_i = 1$, we have $\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}(p_i+p_j) = m-1$, so that it is sufficient to show that

$$\begin{aligned}&\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}\left((p_i+p_j)^2-4(p_i+p_j)\mathbb{E}_{\mathbf{X}}\left[\frac{a_i a_j}{a_i+a_j}\right]\right)^{1/2} \\ &\geq (m-1)\left(1-2\frac{m}{m-1}\mathbb{E}_{\mathbf{X}}\left[\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}a_i a_j\right]\right)^{1/2}.\end{aligned} \tag{B.47}$$

In addition we have

$$\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}(\ldots)^{1/2} \geq \left(\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}\ldots\right)^{1/2},$$

then from (B.47), we need to prove that

$$\begin{aligned}&\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}(p_i+p_j)^2-4\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}(p_i+p_j)\left[\frac{a_i a_j}{a_i+a_j}\right] \\ &\geq (m-1)^2-2m(m-1)\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}a_i a_j. \qquad \text{(B.48)} \\ &= (m-1)\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}(p_i+p_j)-2m(m-1)\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}a_i a_j.\end{aligned}$$

The following inequality implies (B.48)

$$(p_i+p_j)-4\left[\frac{a_i a_j}{a_i+a_j}\right] \geq m-1-\frac{2m(m-1)}{p_i+p_j}\,a_i a_j. \tag{B.49}$$

We know that $p_i + p_j \in (0,1)$ and $a_i + a_j \in (0,1)$ and since $\sum_{l=1}^{m} p_l = 1$ and $\sum_{l=1}^{m} a_l = 1$. One can check that for $m \geq 3$ the inequality (B.49) holds true. This proves our initial claim in (3.22).

B.1.5 Theorem 3.4.1

Let $\mathbf{X} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$ be an i.i.d. m-multiclass labeled sample. Let N_{n_k} be Poisson variables with mean $n_k = \sum_{i=1}^{n} I(y_i = k)$, for $k = 1, \ldots, m$ and independent of one another and of $\mathbf{X}^{(k)} = \{(\mathbf{x}_i, y_i)\}_{i=1, y_i=k}^{n}$. Now let $\overline{\mathbf{X}}^{(k)} = \{(\mathbf{x}_i, y_i)\}_{i=1, y_i=k}^{N_{n_k}}$, $k = 1, \ldots, m$ be the Poisson process with FR statistic $\overline{\mathfrak{R}}_{n_i,n_j}^{(ij)}$ defined in Section 3.4 and constructed by global MST over $\bigcup_{k=1}^{m} \overline{\mathbf{X}}^{(k)} = \bigcup_{k=1}^{m} \{(\mathbf{x}_i, y_i)\}_{i=1, y_i=k}^{N_{n_k}}$. Following the arguments in [99] one yields that

$$n^{-1}\mathbb{E}\left|\overline{\mathfrak{R}}_{n_i,n_j}^{(ij)} - \mathfrak{R}_{n_i,n_j}^{(ij)}\right| \to 0,$$

because of

$$\left|\overline{\mathfrak{R}}_{n_i,n_j}^{(ij)} - \mathfrak{R}_{n_i,n_j}^{(ij)}\right| \leq c_d \left(\sum_{k=1}^{m} \left|N_{n_k} - n_k\right|\right),$$

where c_d is the largest possible degree of any vertex in global MST over $\bigcup_{k=1}^{m} \mathbf{X}^{(k)}$, $\mathbf{X}^{(k)} = \{(\mathbf{x}_i, y_i)\}_{i=1, y_i=k}^{n}$. Hence it remains to prove that

$$\frac{\mathbb{E}\left[\overline{\mathfrak{R}}_{n_i,n_j}^{(ij)}\right]}{2n} \to \delta_{ij}^{m}. \tag{B.50}$$

For $n_1^m := (n_1, \ldots, n_m)$ let $\mathbf{Z}_1^{n_1^m}, \mathbf{Z}_2^{n_1^m}, \ldots$ be independent vectors with common densities $g_n^{(m)}(\mathbf{x}) = \sum_{k=1}^{m} n_k f_k(\mathbf{x})/n$. Next let K_n be an independent Poisson variable with mean n. Consider $\mathbf{Z}_n = \left\{\mathbf{Z}_1^{n_1^m}, \mathbf{Z}_2^{n_1^m}, \ldots, \mathbf{Z}_{K_n}^{n_1^m}\right\}$ a nonhomogeneous Poisson process of rate $\sum_{k=1}^{m} n_k f_k(\mathbf{x})$. Assign a mark from the set $\{1, 2, \ldots, m\}$ to each point of $\mathbf{Z}_n$. A point at $\mathbf{x}$, independently of other points, being assigned the mark s with probability

$n_s f_s/(\sum_{k=1}^{m} n_k f_k(\mathbf{x}))$, for $t = 1, \ldots, m$. Let $\widetilde{\mathbf{X}}_{n_s}^{(s)}$ denotes the set of points in $\mathbf{Z}_n$ with mark s for $s = 1, 2, \ldots, m$ i.e. $\widetilde{\mathbf{X}}_{n_s}^{(s)} = \{(\mathbf{Z}_i, y_i)\}_{i=1, y_i=s}^{n_s}$. Introduce $\widetilde{\mathfrak{R}}_{n_i,n_j}^{(ij)}$ as the FR statistics for data set $\widetilde{\mathbf{X}}_{n_1}^{(1)} \cup \widetilde{\mathbf{X}}_{n_2}^{(2)} \cup \cdots \cup \widetilde{\mathbf{X}}_{n_m}^{(m)}$, applying the global MST and counting edges connecting a point with mark i to a point with mark j. Using the marking theorem $\widetilde{\mathbf{X}}_{n_s}^{(s)}$, for all $s = 1, \ldots, m$ are independent Poisson process with the same distribution as $\overline{\mathbf{X}}^{(s)}$. Therefore we prove (B.50) for $\widetilde{\mathfrak{R}}_{n_i,n_j}^{(ij)}$, see [99], once again. Given points of $\mathbf{Z}_n$ at $\mathbf{x}$ and $\mathbf{z}$, the probability that they have marks i and j

$$W_{n_i,n_j}^{(m)}(\mathbf{x}, \mathbf{z}) := \frac{n_i f_i(\mathbf{x}) n_j f_j(\mathbf{z}) + n_j f_j(\mathbf{x}) n_i f_i(\mathbf{z})}{\left(\sum_{k=1}^{m} n_k f_k(\mathbf{x})\right)\left(\sum_{k=1}^{m} n_k f_k(\mathbf{z})\right)}.$$

Then for $1 \leq i < j \leq m$

$$\mathbb{E}\left[\widetilde{\mathfrak{R}}_{n_i,n_j}^{(ij)} | \mathbf{Z}_n\right] = \sum_{1 \leq t < l \leq K_n} \sum W_{n_i,n_j}^{(m)}(\mathbf{Z}_t^{n_1^m}, \mathbf{Z}_l^{n_1^m}) \times \mathbf{1}\big\{(\mathbf{Z}_t^{n_1^m}, \mathbf{Z}_l^{n_1^m}) \in \mathfrak{F}(\mathbf{Z}_n)\big\}, \tag{B.51}$$

here $\mathfrak{F}(\mathbf{Z}_n)$ represents the global MST over nodes in $\mathbf{Z}_n$. Hence, we have

$$\mathbb{E}\left[\widetilde{\mathfrak{R}}_{n_i,n_j}^{(ij)} | \mathbf{Z}_n, \right] = \sum_{1 \leq t < l \leq K_n} \sum W_{n_i,n_j}^{(m)}(\mathbf{Z}_t^{n_1^m}, \mathbf{Z}_l^{n_1^m}) \mathbf{1}\big\{(\mathbf{Z}_t^{n_1^m}, \mathbf{Z}_l^{n_1^m}) \in \mathfrak{F}(\mathbf{Z}_n)\big\}.$$

Further, set

$$W^{(m)}(\mathbf{x}, \mathbf{z}) := \frac{p_i p_j \big(f_i(\mathbf{x}) f_j(\mathbf{z}) + f_j(\mathbf{x}) f_i(\mathbf{z})\big)}{\left(\sum_{k=1}^{m} p_k f_k(\mathbf{x})\right)\left(\sum_{k=1}^{m} p_k f_k(\mathbf{z})\right)}.$$

One can check that $W_{n_i,n_j}^{(m)} \to W^{(m)}$ and they range in $[0, 1]$. Next by taking expectation from (B.51), we can write

$$\mathbb{E}\left[\widetilde{\mathfrak{R}}_{n_i,n_j}^{(ij)}\right] = \mathbb{E} \sum_{1 \leq t < l \leq K_n} \sum W^{(m)}(\mathbf{Z}_t^{n_1^m}, \mathbf{Z}_l^{n_1^m}) \mathbf{1}\big\{(\mathbf{Z}_t^{n_1^m}, \mathbf{Z}_l^{n_1^m}) \in \mathfrak{F}(\mathbf{Z}_n)\big\} + o(n). \tag{B.52}$$

By taking into account the non-Poisson process $\mathbf{Z}'_n = \left\{\mathbf{Z}_1^{n_1^m}, \mathbf{Z}_2^{n_1^m}, \ldots, \mathbf{Z}_n^{n_1^m}\right\}$ and the fact that $\mathbb{E}\big[\big|\sum_{k=1}^{m} N_{n_k} - n\big|\big] = o(n)$, one yields:

$$\mathbb{E}\left[\widetilde{\mathfrak{R}}_{n_i,n_j}^{(ij)}\right] = \mathbb{E}\sum\sum_{1\leq t<l\leq n} W^{(m)}(\mathbf{Z}_t^{n_1^m}, \mathbf{Z}_l^{n_1^m})\mathbf{1}\big\{(\mathbf{Z}_t^{n_1^m}, \mathbf{Z}_l^{n_1^m}) \in \mathfrak{F}(\mathbf{Z}'_n)\big\} + o(n). \tag{B.53}$$

Also, we can write that $g_n^{(m)}(\mathbf{x}) \to g^{(m)}(\mathbf{x})$ where $g^{(m)}(\mathbf{x}) = \sum_{k=1}^{m} p_k f_k(\mathbf{x})$. Consequently by Proposition 1 in [99], we have

$$\frac{\mathbb{E}\left[\widetilde{\mathfrak{R}}_{n_i,n_j}^{(ij)}\right]}{n} \to \int W^{(m)}(\mathbf{x},\mathbf{x}) g^{(m)}(\mathbf{x})\,\mathrm{d}\mathbf{x} = 2\int \frac{p_i p_j f_i(\mathbf{x}) f_j(\mathbf{x})}{\sum_{k=1}^{m} p_k f_k(\mathbf{x})}\mathrm{d}\mathbf{x}. \tag{B.54}$$

This completes the proof.

APPENDIX C

Supplemental Material for Chapter 6

C.1 Proof of Theorem 2.1

To aid in the proof, we prove two propositions and restate Theorem 2 in [206] as Lemma C.1.1. Since we assume that i_t is piecewise constant with m changes, we can define $[t_1, t_2, \ldots, t_m]$ as the (unknown) transition points, where $t_1 = 1$. We further define the "oracle mean estimator" $u^*(t)$:

$$u^*(t) = \sum_{k=1}^{m} \sum_{t=1}^{T} \frac{1}{t - t_k + 1} \mathbf{1}(t_k \leq t < t_{k+1}) i_t,$$

where $t_{m+1} = T + 1$. The proof is based on the following result found in [206], restated in terms of ADI:

Lemma C.1.1. *The tracking regret of the ensemble ADI estimator in comparison with $u^*(t)$, defined as:*

$$R(u^*(T)) = \sum_{t=1}^{T} (\overline{\text{ADI}}(t) - i_t)^2 - \sum_{t=1}^{T} (u^*(t) - i_t)^2,$$

is at most

$$R(u^*(T)) \leq \frac{m}{\gamma} \ln n_t - \frac{1}{\gamma} \ln \beta^m (1 - \beta)^{T-m} + \frac{\gamma}{8} T. \tag{C.1}$$

Proposition C.1.2.

$$\mathbb{E}\left[(u^*(t) - i_t)^2\right] \leq \sigma_t^2 + \frac{1}{t - t_k + 1} \sigma_*^2. \tag{C.2}$$

Proof.

$$\begin{aligned}
\mathbb{E}\left[(u^*(t) - i_t)^2\right] &= \mathbb{E}\left[\left(\frac{1}{t-t_k+1}\sum_{i=t_k}^{t}(\theta_i+\epsilon_i) - (\theta_t+\epsilon_t)\right)^2\right] \\
&= \mathbb{E}\left[\left(\frac{1}{t-t_k+1}\sum_{i=t_k}^{t-1}\epsilon_i - \epsilon_t\right)^2\right] \\
&= \frac{1}{(t-t_k+1)^2}\left(\sum_{i=t_k}^{t-1}\epsilon_i + (t-t_k)^2\epsilon_t\right) \\
&\leq \frac{(t-t_k)}{(t-t_k+1)^2}\sigma_*^2 + \frac{(t-t_k)^2}{(t-t_k+1)^2}\sigma_t^2 \\
&\leq \frac{\sigma_*^2}{t-t_k+1} + \sigma_t^2.
\end{aligned}$$

□

Proposition C.1.3.

$$\mathbb{E}\left[(\overline{\text{ADI}}(t) - i_t)^2\right] \geq \mathbb{E}\left[(\overline{\text{ADI}}(t) - \theta_t)^2\right] + \sigma_t^2. \tag{C.3}$$

Proof. We first decompose the left side using the definition of i_t:

$$(\overline{\text{ADI}(t)} - i_t)^2 = (\overline{\text{ADI}}(t) - \theta_t)^2 + 2\epsilon_t(\overline{\text{ADI}}(t) - \theta_t) + \epsilon_t^2.$$

The result follows from taking the expectation of both sides, along with the following observation:

$$\begin{aligned}
\mathbb{E}\left[2\epsilon_t(\overline{\text{ADI}}(t) - \theta_t)\right] &= \mathbb{E}\left[2\epsilon_t\sum_{j=1}^{n_t} w_{j,t-1}\sum_{i=1}^{t} g_j(i,T;t_0)i_i\right] &\text{(C.4)} \\
&= \mathbb{E}\left[2\sum_{j=1}^{n_t} w_{j,t-1}g_j(i,T;t_0)i_t\epsilon_t\right] &\text{(C.5)} \\
&= 2\sigma_t^2\sum_{j=1}^{n_t} w_{j,t-1}g_j(i,T;t_0) \geq 0, &\text{(C.6)}
\end{aligned}$$

where the last inequality is due to the fact that $w_{j,t-1}$ and $g_j(i, T; t_0)$ are non-negative, $\forall i, j, t$. $\square$

Using the definition of $R(u^*(T))$ and Props. C.1.3, C.1.2, we obtain:

$$\sum_{t=1}^{T} \left(\mathbb{E}\left[(\overline{\text{ADI}}(t) - \theta_t)^2 \right] + \sigma_t^2 \right) - \sum_{t=1}^{T} \left(\sigma_t^2 + \sum_{k=1}^{m} \frac{1}{t - t_k + 1} \mathbf{1}(t_k \leq t < t_{k+1}) \sigma_*^2 \right) \leq R(u^*(T)).$$

Finally, note the following inequality:

$$\sum_{k=1}^{m} \sum_{t=1}^{T} \frac{1}{t - t_k + 1} \leq m \ln \left(\frac{T}{e} \right).$$

Combining this with Lemma C.1.1, and rearranging terms, achieves the desired bound.

APPENDIX D

Supplemental Material for Chapter 9

D.1 Appendix A: Solution of two gaussian distributions

Theorem 1: Let $W \in \mathbb{R}^n$ The solution to the maximization problem

$$\text{argmax}_W f(W) = \left[\gamma_1 P_1(W_1|W) + \gamma_2 P_2(W_2|W)\right], \tag{D.1}$$

with $P(W_i|W)$ of the multivariate Normal distribution

$$P(W_1|W) = \mathcal{N}(W, \sigma_1^2 I_n) \tag{D.2}$$

$$P(W_2|W) = \mathcal{N}(W, \sigma_2^2 I_n), \tag{D.3}$$

is of the form

$$\hat{W} = \beta W_1 + (1-\beta) W_2. \tag{D.4}$$

Proof: The proof is separated into two steps. First, we show that for any arbitrary point $x \in \mathbb{R}^n$, the point $x_\parallel$ which is the projection of x onto the line $g(x) = W_1 + \lambda(W_2 - W_1)$ increases the value of f, that is

$$f(x) < f(x_{\parallel}). \tag{D.5}$$

Then we show that for all points on the line $g(x)$, f is maximized for some point on the line segment between W_1 and W_2, corresponding to $\lambda \in [0, 1]$

Let $x \in \mathbb{R}^n$. There exists a unique decomposition of x into a vector parallel to $g(x)$ and one perpindicular to $g(x)$:

$$x = x_{\parallel} + x_{\perp}. \tag{D.6}$$

Plugging x into $f(W)$, we have

$$\begin{aligned} f(x) &= (2\pi)^{-n/2} \left|\sigma_1^2 I_n\right|^{-\frac{1}{2}} e^{-\frac{1}{2}(x-W_1)^T(\sigma_1^2 I_n)^{-1}(x-W_1)} \\ &\quad + (2\pi)^{-n/2} \left|\sigma_2^2 I_n\right|^{-\frac{1}{2}} e^{-\frac{1}{2}(x-W_2)^T(\sigma_2^2 I_n)^{-1}(x-W_2)} \qquad \text{(D.7)} \\ &= \left(2\pi\sigma_1^2\right)^{-n/2} e^{\frac{1}{2\sigma_1}(x_{\parallel}-W_1+x_{i\perp}-(W_1)_i)^2} \\ &\quad + \left(2\pi\sigma_2^2\right)^{-n/2} e^{\frac{1}{2\sigma_2}\sum_{i=1}^{n}(x_{i\parallel}+x_{i\perp}-(W_2)_i)^2}. \qquad \text{(D.8)} \end{aligned}$$

The exponent can be decomposed as follows:

$$\begin{aligned} &(x_{\parallel} + x_{\perp} - W_1)^T(x_{\parallel} + x_{\perp} - W_1) && \text{(D.9)} \\ &\quad = (x_{\parallel} - W_1)^T(x_{\parallel} - W_1) + 2x_{\perp}(x_{\parallel} - W_1) + x_{\perp}^T x_{\perp} && \text{(D.10)} \\ &\quad = (x_{\parallel} - W_1)^T(x_{\parallel} - W_1) + x_{\perp}^T x_{\perp} && \text{(D.11)} \\ &\quad \geq (x_{\parallel} - W_1)^T(x_{\parallel} - W_1). && \text{(D.12)} \end{aligned}$$

Note that due to the orthogonality of $x_\perp$ with the line $g(x)$, the cross term goes to 0. The same can be shown for the other exponential term with W_2. Since the term with x is greater than with just $x_\parallel$, so

$$f(x_\parallel) \geq f(x). \tag{D.13}$$

Finally, let us show that the maximum for f must be between W_1 and W_2. This can easily be seen by the fact that both summation terms in f decrease as the distance between x and the means W_1 and W_2 increases. When on the line g, but outside the line segment between W_1 and W_2, moving closer to the means will increase both terms. Therefore, the maximum of f must be on the line g, with λ restricted between 0 and 1.

Bibliography

[1] E. Airoldi, D. Blei, S. Fienberg, and E. Xing, "Mixed membership stochastic blockmodels," *Journal of Machine Learning Research*, vol. 9, pp. 1981–2014, 2008.

[2] E. Airoldi, D. Choi, and P. Wolfe, "Confidence sets for network structure," *Statistical Analysis and Data Mining*, vol. 4, no. 5, pp. 461–469, 2011.

[3] E. M. Airoldi, T. B. Costa, and S. H. Chan, "Stochastic blockmodel approximation of a graphon: Theory and consistent estimation," in *Advances in Neural Information Processing Systems*, 2013, pp. 692–700.

[4] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Rev. Mod. Phys.*, vol. 74, pp. 47–97, 1 Jan. 2002. DOI: 10.1103/RevModPhys.74.47. [Online]. Available: http://link.aps.org/doi/10.1103/RevModPhys.74.47.

[5] D. Aldous and J. M. Steele, "Asymptotic for euclidean minimal spanning trees on random points," *Probab. Theory Related Fields*, vol. 92, pp. 247–258, 1992.

[6] D. J. Aldous, "Representations for partially exchangeable arrays of random variables," *Journal of Multivariate Analysis*, vol. 11, no. 4, pp. 581–598, 1981.

[7] D. J. Aldous, "Exchangeability and related topics," in *École dété de probabilités de Saint-Flour, XIII—1983*, ser. Lecture Notes in Math. Vol. 1117, Berlin: Springer, 1985, pp. 1–198.

[8] A. A. Alemi, I. Fischer, J. V. Dillon, and K. Murphy, "Deep variational information bottleneck," *arXiv preprint arXiv:1612.00410*, 2016.

[9] A. Aleta, S. Meloni, and Y. Moreno, "A multilayer perspective for the analysis of urban transportation systems," *Scientific reports*, vol. 7, p. 44 359, 2017.

[10] S. Ali and S. D. Silvey, "A general class of coefficients of divergence of one distribution from another," *J. Royal Statist. Soc. Ser. B (Methodology.)*, pp. 131–142, 1966.

[11] P.-O. Amblard and O. J. J. Michel, "On directed information theory and granger causality graphs," *Journal of Computational Neuroscience*, vol. 30, no. 1, pp. 7–16, 2011, ISSN: 1573-6873.

[12] S. Aminikhanghahi and D. J. Cook, "A survey of methods for time series change point detection," *Knowledge and information systems*, vol. 51, no. 2, pp. 339–367, 2017.

[13] A. Anandkumar, V. Y. Tan, F. Huang, A. S. Willsky, *et al.*, "High-dimensional structure estimation in ising models: Local separation criterion," *The Annals of Statistics*, vol. 40, no. 3, pp. 1346–1375, 2012.

[14] A. Athreya, D. E. Fishkind, M. Tang, C. E. Priebe, Y. Park, J. T. Vogelstein, K. Levin, V. Lyzinski, and Y. Qin, "Statistical inference on random dot product graphs: A survey," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 8393–8484, 2017.

[15] M. Baena-García, J. M. Carmona-Cejudo, G. Castillo, and R. Morales-Bueno, *Tf-sidf: Term frequency, sketched inverse document frequency*, Nov. 2011.

[16] B. Baingana, G. Mateos, and G. B. Giannakis, "Proximal-gradient algorithms for tracking cascades over social networks," *IEEE Journal on Selected Topics in Signal Processing*, vol. 8, no. 4, pp. 563–575, 2014, ISSN: 19324553.

[17] T. Banerjee, H. Firouzi, and A. O. Hero, "Quickest detection for changes in maximal knn coherence of random matrices," *IEEE Transactions on Signal Processing*, vol. 66, no. 17, pp. 4490–4503, 2018.

[18] A.-L. Barabási *et al.*, *Network science*. Cambridge university press, 2016.

[19] M. Barigozzi, G. Fagiolo, and G. Mangioni, "Identifying the community structure of the international-trade multi-network," *Physica A: Statistical Mechanics and its Applications*, vol. 390, no. 11, pp. 2051–2066, 2011, ISSN: 0378-4371. DOI: http://dx.doi.org/10.1016/j.physa.2011.02.004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378437111001129.

[20] D. S. Bassett, M. A. Porter, N. F. Wymbs, S. T. Grafton, J. M. Carlson, and P. J. Mucha, "Robust detection of dynamic community structure in networks," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 23, no. 1, p. 013 142, 2013.

[21] F. Battiston, V. Nicosia, and V. Latora, "Metrics for the analysis of multiplex networks," DOI: arXiv:1308.3182v1.

[22] ——, "Structural measures for multiplex networks," *Physical Review E*, vol. 89, no. 3, p. 032 804, 2014.

[23] J. Beardwood, J. H. Halton, and J. M. Hammersley, "The shortest path through many points," in *Mathematical Proceedings of the Cambridge Philosophical Society*, 1959, pp. 299–327.

[24] P. Bedi and C. Sharma, "Community detection in social networks," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 6, no. 3, pp. 115–135, 2016.

[25] V. Berisha and A. O. Hero, "Empirical non-parametric estimation of the fisher information," *IEEE Signal Process. Lett.*, vol. 22, no. 7, pp. 988–992, 2015.

[26] V. Berisha, A. Wisler, A. O. Hero, and A. Spanias, "Empirically estimable classification bounds based on a nonparametric divergence measure," *IEEE Transactions on Signal Processing*, vol. 64, no. 3, pp. 580–591, 2016.

[27] M. Berlingerio, M. Coscia, and F. Giannotti, "Finding redundant and complementary communities in multidimensional networks," in *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, ser. CIKM '11, Glasgow, Scotland, UK: ACM, 2011, pp. 2181–2184, ISBN: 978-1-4503-0717-8.

[28] A. Bertrand and M. Moonen, "Seeing the bigger picture: How nodes can learn their place within a complex ad hoc network topology," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 71–82, 2013.

[29] K. Bharath, "Discussion of "bayesian nonparametric models of sparse and exchangeable random graphs"," *Journal of the Royal Statistical Society, Series B*, vol. 79, no. 5, 2017.

[30] A. Bhattacharyya, "On a measure of divergence between two multinomial populations," *Sankhyā: the indian journal of statistics*, pp. 401–406, 1946.

[31] G. Bianconi, "Statistical mechanics of multiplex networks: Entropy and overlap," *Phys. Rev. E*, vol. 87, p. 062 806, 6 Jun. 2013. DOI: 10.1103/PhysRevE.87.062806. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevE.87.062806.

[32] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[33] R. E. Boulos, N. Tremblay, A. Arneodo, P. Borgnat, and B. Audit, "Multi-scale structural community organisation of the human genome," *BMC bioinformatics*, vol. 18, no. 1, p. 209, 2017.

[34] P. Bródka, P. Kazienko, K. Musiał, and K. Skibicki, "Analysis of neighbourhoods in multi-layered dynamic social networks," *International Journal of Computational Intelligence Systems*, vol. 5, no. 3, pp. 582–596, 2012.

[35] P. Bródka, K. Skibicki, P. Kazienko, and K. Musiał, "A degree centrality in multi-layered social network," in *International Conference on Computational Aspects of Social Networks (CASoN)*, IEEE, 2011, pp. 237–242.

[36] C. T. Butts, "A relational event framework for social action," *Sociological Methodology*, vol. 38, no. 1, pp. 155–200, 2008.

[37] D. Cai, Z. Shao, X. He, X. Yan, and J. Han, "Community mining from multi-relational networks," in *European Conference on Principles of Data Mining and Knowledge Discovery*, Springer, 2005, pp. 445–452.

[38] D. Cai, T. Campbell, and T. Broderick, "Edge-exchangeable graphs and sparsity," in *Advances in Neural Information Processing Systems*, 2016, pp. 4249–4257.

[39] F. Camerlenghi, A. Lijoi, P. Orbanz, and I. Prünster, "Distribution theory for hierarchical processes," *Annals of Statistics*, 2018.

[40] M. Caramia and P. Dell'Olmo, "Multi-objective optimization," English, in *Multi-objective Management in Freight Logistics*, Springer London, 2008, pp. 11–36, ISBN: 978-1-84800-381-1.

[41] G. Carneiro and N. Vasconcelos, "Minimum bayes error features for visual recognition by sequential feature selection and extraction," in *Proceedings. The 2nd Canadian Conference on*, 2005, pp. 253–260.

[42] F. Caron and E. B. Fox, "Sparse graphs using exchangeable random measures," *Journal of the Royal Statistical Society, Series B*, vol. 79, no. 5, pp. 1295–1366, 2017.

[43] N. Cesa-Bianchi and G. Lugosi, *Prediction, learning, and games*. Cambridge university press, 2006.

[44] P. K. Chan and S. J. Stolfo, "Meta-learning for multistrategy and parallel learning," in *in Proc. 2nd. Int. Workshop on Multi-strategy Learning*, 1993, pp. 150–165.

[45] H. Chen, J. Chen, L. A. Muir, S. Ronquist, W. Meixner, M. Ljungman, T. Ried, S. Smale, and I. Rajapakse, "Functional organization of the human 4d nucleome," *Proceedings of the National Academy of Sciences*, vol. 112, no. 26, pp. 8002–8007, 2015.

[46] H. Chen, S. Liu, L. Seaman, C. Najarian, W. Wu, M. Ljungman, G. Higgins, A. Hero, M. Wicha, and I. Rajapakse, "Parental allele-specific genome architecture and transcription during the cell cycle," *bioRxiv*, p. 201 715, 2017.

[47] P.-Y. Chen and A. O. Hero, "Multilayer spectral graph clustering via convex layer aggregation: Theory and algorithms," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 3, no. 3, pp. 553–567, 2017.

[48] X. Chen, A. Hero, and S. Savarese, "Shrinkage optimized directed information using pictorial structures for action recognition," *arXiv preprint arXiv:1404.3312*, 2014.

[49] X. Chen, Z. Syed, and A. Hero, "EEG spatial decoding with shrinkage optimized directed information assessment," *ICASSP 2012 Proceedings*, pp. 577–580, 2012, ISSN: 15206149.

[50] ——, "Eeg spatial decoding and classification with logit shrinkage regularized directed information assessment (l-soda)," *arXiv preprint arXiv:1404.0404*, 2014.

[51] Z. Chen and C. Leng, "Dynamic covariance models," *Journal of the American Statistical Association*, vol. 111, no. 515, pp. 1196–1207, 2016.

[52] Z. Chen, C. Chen, Z. Zheng, and Y. Zhu, "Tensor decomposition for multilayer networks clustering," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 3371–3378.

[53] H. Chernoff, "A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations," *Annalls of Mathematical Statistics*, vol. 23, pp. 493–507, 1952.

[54] F. R. K. Chung, *Spectral graph theory*, 92. American Mathematical Soc., 1997.

[55] W. W. Cohen, *Enron email dataset*, Aug. 2009. [Online]. Available: http://www.cs.cmu.edu/~enron/.

[56] T. M. Cover and J. a. Thomas, *Elements of Information Theory*. 2005, pp. 1–748, ISBN: 9780471241959.

[57] D. R. Cox, "Regression models and life-tables (with discussion).," *Journal of the Royal Statistical Society, Series B*, vol. 34, pp. 187–220, 1972.

[58] E. Cozzo, M. Kivelä, M. De Domenico, A. Solé-Ribalta, A. Arenas, S. Gómez, M. A. Porter, and Y. Moreno, "Structure of triadic relations in multiplex networks," *New Journal of Physics*, vol. 17, no. 7, p. 073 029, 2015.

[59] H. Crane, "Discussion of "bayesian nonparametric models of sparse and exchangeable random graphs"," *Journal of the Royal Statistical Society, Series B*, vol. 79, no. 5, 2017.

[60] H. Crane and W. Dempsey, "Relational exchangeability," *Journal of Applied Probability*, 2019 (To appear).

[61] ——, "A framework for statistical network modeling," *arXiv preprint arXiv:1509.08185*, 2015.

[62] ——, "Edge exchangeable models for interaction networks," *Journal of the American Statistical Association*, vol. 113, no. 523, pp. 1311–1326, 2018.

[63] I. Csiszár and P. C. Shields, "Information theory and statistics: A tutorial," *J. Royal Statist. Soc. Ser. B (Methodology.)*, vol. 1, no. 4, pp. 417–528, 2004.

[64] C. De Bacco, E. A. Power, D. B. Larremore, and C. Moore, "Community detection, link prediction, and layer interdependence in multilayer networks," *Phys. Rev. E*, vol. 95, p. 042 317, 4 Apr. 2017.

[65] C. De Bacco, E. A. Power, D. B. Larremore, and C. Moore, "Community detection, link prediction, and layer interdependence in multilayer networks," *Physical Review E*, vol. 95, no. 4, p. 042 317, 2017.

[66] M. De Domenico, V. Nicosia, A. Arenas, and V. Latora, "Structural reducibility of multilayer networks," vol. 6, p. 6864, Apr. 2015.

[67] M. De Domenico, A. Solé-Ribalta, E. Cozzo, M. Kivelä, Y. Moreno, M. A. Porter, S. Gómez, and A. Arenas, "Mathematical formulation of multilayer networks," *Physical Review X*, vol. 3, no. 4, p. 041 022, 2013.

[68] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *Evolutionary Computation, IEEE Transactions on*, vol. 6, no. 2, pp. 182–197, 2002, ISSN: 1089-778X.

[69] W. Dempsey, B. Oselio, and A. Hero, "Hierarchical network models for structured exchangeable interaction processes," *Journal of the American Statistical Association (In review)*, 2019.

[70] D. Durante, D. B. Dunson, and J. T. Vogelstein, "Nonparametric bayes modeling of populations of networks," *Journal of the American Statistical Association*, vol. 112, no. 520, pp. 1516–1530, 2017.

[71] N. Eagle and A. Pentland, "Reality mining: Sensing complex social systems.," *Pers. Ubiquit. Comput.*, vol. 10, pp. 255–268, 2006.

[72] M. Ehrgott, "Multiobjective optimization," English, *AI Magazine*, vol. 29, no. 4, pp. 47–57, Winter 2008. [Online]. Available: http://search.proquest.com.proxy.lib.umich.edu/docview/208128027?accountid=14667.

[73] M. Escobar and M. West, "Bayesian density estimation and inference using mixtures," *Journal of the American Statistical Association*, vol. 90, pp. 577–588, 1995.

[74] E. Estrada, *The structure of complex networks: theory and applications*. Oxford University Press, 2012.

[75] J. Etesami, N. Kiyavash, and T. P. Coleman, "Learning minimal latent directed information trees," *IEEE International Symposium on Information Theory - Proceedings*, pp. 2726–2730, 2012.

[76] J. Fan, Y. Liao, and H. Liu, *An overview of the estimation of large covariance and precision matrices*, 2016.

[77] R. A. Fisher, "On the probable error of a coefficient of correlation deduced from a small sample," *Metron*, vol. 1, pp. 3–32, 1921.

[78] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, pp. 75–174, 2010, ISSN: 0370-1573. DOI: DOI:10.1016/j.physrep.2009.11.002.

[79] S. Fortunato and D. Hric, "Community detection in networks: A user guide," *Physics reports*, vol. 659, pp. 1–44, 2016.

[80] J. H. Fowler, "Connecting the congress: A study of cosponsorship networks.," *Polit. Anal.*, vol. 14, pp. 456–487, 2006.

[81] J. H. Friedman and L. C. Rafsky, "Multivariate generalizations of the wald-wolfowitz and smirnov two-sample tests," *The Annals of Statistics*, pp. 697–717, 1979.

[82] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432–441, 2008.

[83] F. Garber and A. Djouadi, "Bounds on the bayes classification error based on pairwise risk functions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 2, pp. 281–288, 1988.

[84] L. Gauvin, A. Panisson, and C. Cattuto, "Detecting the community structure and activity patterns of temporal networks: A non-negative tensor factorization approach," *PloS one*, vol. 9, no. 1, e86028, 2014.

[85] A. Gelman, X.-L. Meng, and H. Stern, "Posterior predictive assessment of model fitness via realized discrepancies," *Statistica Sinica*, vol. 6, pp. 733–807, 1996.

[86] A. M. Geoffrion, "Proper efficiency and the theory of vector maximization," *Journal of Mathematical Analysis and Applications*, vol. 22, no. 3, pp. 618–630, 1968, ISSN: 033-247X. DOI: http://dx.doi.org/10.1016/0022-247X(68)90201-1. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0022247X68902011.

[87] D. F. Gleich, "Pagerank beyond the web," *SIAM Review*, vol. 57, no. 3, pp. 321–363, 2015.

[88] A. Goldenberg, A. X. Zheng, S. E. Fienberg, E. M. Airoldi, *et al.*, "A survey of statistical network models," *Foundations and Trends® in Machine Learning*, vol. 2, no. 2, pp. 129–233, 2010.

[89] M. Gomez-Rodriguez, J. Leskovec, and A. Krause, "Inferring networks of diffusion and influence," *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '10*, vol. 5, no. 4, pp. 1019–1028, 2010, ISSN: 15564681.

[90] K. Greenewald and A. O. Hero, "Robust kronecker product pca for spatio-temporal covariance estimation," *IEEE Transactions on Signal Processing*, vol. 63, no. 23, pp. 6368–6378, 2015.

[91] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future generation computer systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[92] A. Guille, H. Hacid, C. Favre, and D. A. Zighed, "Information diffusion in online social networks: A survey," *SIGMOD Rec.*, vol. 42, no. 1, pp. 17–28, Jul. 2013, ISSN: 0163-5808. DOI: 10.1145/2503792.2503797. [Online]. Available: http://doi.acm.org.proxy.lib.umich.edu/10.1145/2503792.2503797.

[93] R. Guimera and L. A. N. Amaral, "Functional cartography of complex metabolic networks," *nature*, vol. 433, no. 7028, p. 895, 2005.

[94] X. Guorong, C. Peiqi, and W. Minhui, "Bhattacharyya distance feature selection," in *In Pattern Recognition, Proceedings of the 13th International Conference on IEEE*, vol. 2, 1996, pp. 195–199.

[95] A. Gupta, B. Eysenbach, C. Finn, and S. Levine, "Unsupervised meta-learning for reinforcement learning," in *Under review as a conference paper at ICLR 2019, available at arXiv:1806.04640*, 2018.

[96] A. B. Hamza and H. Krim, "Image registration and segmentation by maximizing the jensen-rényi divergence," in *Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2003, pp. 147–163.

[97] Q. Han, K. S. Xu, and E. M. Airoldi, "Consistent estimation of dynamic and multi-layer block models," in *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ser. ICML'15, Lille, France: JMLR.org, 2015, pp. 1511–1520.

[98] N. Henze, "A multivariate two-sample test based on the number of nearest neighbor type coincidences," *The Annals of Statistics*, pp. 772–783, 1988.

[99] N. Henze, M. D. Penrose, *et al.*, "On the multivariate runs test," *The Annals of Statistics*, vol. 27, no. 1, pp. 290–298, 1999.

[100] M. Herbster and M. K. Warmuth, "Tracking the best expert," *Machine learning*, vol. 32, no. 2, pp. 151–178, 1998.

[101] A. Hero, B. Ma, O. Michel, and J. Gorman, "Applications of entropic spanning graphs," *IEEE Signal Processing Magazine*, vol. 19, no. 5, pp. 85–95, 2002.

[102] A. Hero and B. Rajaratnam, "Large-scale correlation screening," *Journal of the American Statistical Association*, vol. 106, no. 496, pp. 1540–1552, 2011.

[103] ——, "Hub discovery in partial correlation graphs," *IEEE Transactions on Information Theory*, vol. 58, no. 9, pp. 6064–6078, 2012.

[104] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, A. Trischler, and Y. Bengio, "Learning deep representations by mutual information estimation and maximization," *arXiv preprint arXiv:1808.06670*, 2018.

[105] P. Hoff, A. Raftery, and M. Handcock, "Latent space approaches to social network analysis," *J. Amer. Statist. Assoc.*, vol. 97, no. 460, pp. 1090–1098, 2002.

[106] R. v. d. Hofstad, *Random Graphs and Complex Networks*, ser. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2016, vol. 1.

[107] P. W. Holland, K. B. Laskey, and S. Leinhardt, "Stochastic blockmodels: First steps," *Social networks*, vol. 5, no. 2, pp. 109–137, 1983.

[108] K. J. Hsiao, J. Calder, and A. O. Hero, "Pareto-depth for multiple-query image retrieval," *IEEE Transactions on Image Processing*, vol. 24, no. 2, pp. 583–594, Feb. 2015, ISSN: 1057-7149.

[109] K. J. Hsiao, K. S. Xu, J. Calder, and A. O. Hero, "Multicriteria similarity-based anomaly detection using pareto depth analysis," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 6, pp. 1307–1321, Jun. 2016, ISSN: 2162-237X.

[110] L. C. Hsu and P. J.-S. Shiue, "A unified approach to generalized stirling numbers," *Advances in Applied Mathematics*, vol. 20, no. 3, pp. 366–384, 1998.

[111] L. Hubert and P. Arabie, "Comparing partitions," English, *Journal of Classification*, vol. 2, no. 1, pp. 193–218, 1985, ISSN: 0176-4268. DOI: 10.1007/BF01908075. [Online]. Available: http://dx.doi.org/10.1007/BF01908075.

[112] H. Ishwaran and L. James, "Gibbs sampling methods for stick-breaking priors," *Journal of the American Statistical Association*, vol. 96, no. 453, pp. 161–173, 2001.

[113] S. Janson, "On edge exchangeable random graphs," *Journal of Statistical Physics*, vol. 173, no. 3-4, pp. 448–484, 2018.

[114] J. Jiao, H. H. Permuter, L. Zhao, Y.-H. Kim, and T. Weissman, "Universal Estimation of Directed Information," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6220–6242, 2013.

[115] J. Jiao, H. H. Permuter, L. Zhao, Y.-H. Kim, and T. Weissman, "Universal estimation of directed information," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6220–6242, 2013.

[116] Y. Jin and B. Sendhoff, "Pareto-Based Multiobjective Machine Learning: An Overview and Case Studies," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 38, no. 3, pp. 397–415, Apr. 2008.

[117] T. Kailath, "The divergence and bhattacharyya distance in signal selection," *IEEE Trans. Communications Technology*, vol. COM-15, pp. 52–60, 1967.

[118] B. W. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs," *Bell System Technical Journal*, vol. 49, no. 2, pp. 291–307, 1970, ISSN: 1538-7305. DOI: 10.1002/j.1538-7305.1970.tb01770.x. [Online]. Available: http://dx.doi.org/10.1002/j.1538-7305.1970.tb01770.x.

[119] J. Kim and J.-G. Lee, "Community detection in multi-layer graphs: A survey," *ACM SIGMOD Record*, vol. 44, no. 3, pp. 37–48, 2015.

[120] M. Kivelä, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, "Multilayer networks," *Journal of complex networks*, vol. 2, no. 3, pp. 203–271, 2014.

[121] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604–632, 1999.

[122] B. Klimt and Y. Yang, "The enron corpus: A new dataset for email classification research," in *European Conference on Machine Learning*, Springer, 2004, pp. 217–226.

[123] E. D. Kolaczyk and G. Csárdi, *Statistical analysis of network data with R*. Springer, 2014, vol. 65.

[124] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

[125] T. G. Kolda, B. W. Bader, and J. P. Kenny, "Higher-order web link analysis using multilinear algebra," in *Data Mining, Fifth IEEE International Conference on*, 2005, 8–pp.

[126] D. Koller, N. Friedman, and F. Bach, *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[127] V. Latora, V. Nicosia, and G. Russo, *Complex networks: principles, methods and applications*. Cambridge University Press, 2017.

[128] P. Latouche, E. Birmelé, C. Ambroise, *et al.*, "Overlapping stochastic block models with application to the french political blogosphere," *The Annals of Applied Statistics*, vol. 5, no. 1, pp. 309–336, 2011.

[129] ——, "Model selection in overlapping stochastic block models," *Electronic journal of statistics*, vol. 8, no. 1, pp. 762–794, 2014.

[130] P. Latouche, S. Robin, and S. Ouadah, "Goodness of fit of logistic models for random graphs," *arXiv preprint arXiv:1508.00286*, 2015.

[131] C. M. Le, E. Levina, and R. Vershynin, "Concentration and regularization of random graphs," *Random Structures & Algorithms*, vol. 51, no. 3, pp. 538–561, 2017.

[132] D. Leung, I. Jung, N. Rajagopal, A. Schmitt, S. Selvaraj, A. Y. Lee, C.-A. Yen, S. Lin, Y. Lin, Y. Qiu, *et al.*, "Integrative analysis of haplotype-resolved epigenomes across human tissues," *Nature*, vol. 518, no. 7539, p. 350, 2015.

[133] M. Lichman, *UCI machine learning repository*, 2013. [Online]. Available: http://archive.ics.uci.edu/ml.

[134] E. Lieberman-Aiden *et al.*, "Comprehensive mapping of long-range interactions reveals folding principles of the human genome," *Science*, vol. 326, no. 5950, pp. 289–293, 2009. DOI: 10.1126/science.1181369.

[135] J. Lin, "Divergence measures based on the shannon entropy," *IEEE Transactions on Information Theory*, vol. 37, no. 1, pp. 145–151, 1991.

[136] T. Lissack and K.-S. Fu, "Error estimation in pattern recognition via L_α-distance between posterior density functions," *IEEE Transactions on Information Theory*, vol. 22, no. 1, pp. 34–45, 1976.

[137] H. Liu, F. Han, M. Yuan, J. Lafferty, L. Wasserman, *et al.*, "High-dimensional semiparametric gaussian copula graphical models," *The Annals of Statistics*, vol. 40, no. 4, pp. 2293–2326, 2012.

[138] H. Liu, J. Lafferty, and L. Wasserman, "The nonparanormal: Semiparametric estimation of high dimensional undirected graphs," *Journal of Machine Learning Research*, vol. 10, no. Oct, pp. 2295–2328, 2009.

[139] S. Liu, H. Chen, S. Ronquist, L. Seaman, N. Ceglia, W. Meixner, L. A. Muir, P.-Y. Chen, G. Higgins, P. Baldi, S. Smale, A. Hero, and I. Rajapakse, "Genome architecture leads a bifurcation in cell identity," *bioRxiv*, p. 151 555, 2017.

[140] Y. Liu and S. Aviyente, "Directed information measure for quantifying the information flow in the brain," in *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, IEEE, 2009, pp. 2188–2191.

[141] B. Lu and P. R. Rosenbaum, "Optimal pair matching with two control groups," *Journal of Computational and Graphical Statistics*, vol. 13, no. 2, pp. 422–434, 2004.

[142] M. Luckie, Y. Hyun, and B. Huffaker, "Traceroute probe method and forward ip path inference," in *Proc. Internet Measurement Conference*, Oct. 2008, pp. 311–324.

[143] U. Luxburg, "A tutorial on spectral clustering," English, *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007, ISSN: 0960-3174. DOI: 10.1007/s11222-007-9033-z. [Online]. Available: http://dx.doi.org/10.1007/s11222-007-9033-z.

[144] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[145] D. J. MacKay and D. J. Mac Kay, *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[146] M. Magnani and L. Rossi, "The ml-model for multi-layer social networks," in *Advances in Social Networks Analysis and Mining (ASONAM), 2011 International Conference on*, IEEE, 2011, pp. 5–12.

[147] M. Mariadassou, S. Robin, and C. Vacher, "Uncovering latent structure in valued graphs: A variational approach," *Annals of Applied Statistics*, vol. 4, no. 2, pp. 715–742, 2010.

[148] J. Massey, "Causality, feedback and directed information," in *Proc. Int. Symp. Inf. Theory Applic.(ISITA-90)*, Citeseer, 1990, pp. 303–305.

[149] J. L. Massey and P. C. Massey, "Conservation of mutual and directed information," in *Information Theory, 2005. ISIT 2005. Proceedings. International Symposium on*, IEEE, 2005, pp. 157–158.

[150] A. McCallum, X. Wang, and A. Corrada-Emmanuel, "Topic and role discovery in social networks with experiments on enron and academic email," *Journal of Artificial Intelligence Research*, vol. 30, pp. 249–272, 2007.

[151] P. McCullagh, "What is a statistical model?" *Annals of Statistics*, pp. 1225–1267, 2002.

[152] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annual review of sociology*, vol. 27, no. 1, pp. 415–444, 2001.

[153] T. Michoel and B. Nachtergaele, "Alignment and integration of complex networks by hypergraph-based spectral clustering," *Phys. Rev. E*, vol. 86, p. 056 111, 5 Nov. 2012. DOI: 10.1103/PhysRevE.86.056111. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevE.86.056111.

[154] K. Moon and A. Hero, "Ensemble estimation of multivariate f-divergence," in *IEEE International Symposium on Information Theory (ISIT)*, 2014, pp. 356–360.

[155] ——, "Multivariate f-divergence estimation with confidence," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2420–2428.

[156] K. Moon, K. Sricharan, K. Greenewald, and A. Hero, "Improving convergence of divergence functional ensemble estimators," in *IEEE International Symposium on Information Theory (ISIT)*, 2016, pp. 1133–1137.

[157] K. R. Moon and A. O. Hero, "Ensemble estimation of multivariate f-divergence," in *IEEE International Symposium on Information Theory*, 2016, pp. 356–360.

[158] K. R. Moon, M. Noshad, S. Yasaei Sekeh, and A. O. Hero, "Information theoretic structure learning with confidence," in *in Proc. IEEE Int. Conf. Acoust Speech Signal Process*, 2017.

[159] K. R. Moon, K. Sricharan, K. Greenewald, and A. O. Hero, "Nonparametric ensemble estimation of distributional functionals," *arXiv preprint arXiv:1601.06884*, 2016.

[160] K. R. Moon, K. Sricharan, and A. O. Hero, "Ensemble estimation of mutual information," in *2017 IEEE International Symposium on Information Theory*, IEEE, 2017, pp. 3030–3034.

[161] T. Morimoto, "Markov processes and the h-theorem," *J. Phys. Soc. Jpn.*, vol. 18, no. 3, pp. 328–331, 1963.

[162] P. J. Mucha, T. Richardson, K. Macon, M. A. Porter, and J.-P. Onnela, "Community structure in time-dependent, multiscale, and multiplex networks," *Science*, vol. 328, no. 5980, pp. 876–878, 2010, ISSN: 0036-8075.

[163] M. E. J. Newman, "Modularity and community structure in networks.," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 103, no. 23, pp. 8577–82, Jun. 2006, ISSN: 0027-8424.

[164] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Phys. Rev. E*, vol. 69, p. 066 133, 6 Jun. 2004. DOI: 10.1103/PhysRevE.69.066133. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevE.69.066133.

[165] M. Newman, *Networks*. Oxford university press, 2018.

[166] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in neural information processing systems*, 2002, pp. 849–856.

[167] P. Ngatchou, A. Zarei, and M. El-Sharkawi, "Pareto multi objective optimization," in *Intelligent Systems Application to Power Systems, 2005. Proceedings of the 13th International Conference on*, 2005, pp. 84–91. DOI: 10.1109/ISAP.2005.1599245.

[168] A. Nguyen, J. Wainwright, and M. I. Jordan, "Estimating divergence functionals and the likelihood ratio by convex risk minimization," *IEEE Transactions on Information Theory*, vol. 56, no. 11, pp. 5847–5861, 2010.

[169] V. Nicosia, G. Bianconi, V. Latora, and M. Barthelemy, "Growing multiplex networks," *Phys. Rev. Lett.*, vol. 111, p. 058 701, 5 Jul. 2013. DOI: 10.1103/PhysRevLett.111.058701. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevLett.111.058701.

[170] J. D. Noh and H. Rieger, "Random walks on complex networks," *Phys. Rev. Lett.*, vol. 92, p. 118 701, 11 Mar. 2004. DOI: 10.1103/PhysRevLett.92.118701. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevLett.92.118701.

[171] M. Noshad, K. Moon, S. Yasaei Sekeh, and A. Hero, "Direct estimation of information divergence using nearest neighbor ratios," in *IEEE International Symposium on Information Theory (ISIT)*, 2017.

[172] M. Noshad and A. O. Hero, "Scalable hash-based estimation of divergence measures," in *2018 Information Theory and Applications Workshop*, IEEE, 2018, pp. 1–10.

[173] P. Orbanz and D. M. Roy, "Bayesian models of graphs, arrays and other exchangeable random structures," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 2, pp. 437–461, 2015.

[174] B. Oselio, A. Hero, A. Sadeghian, and S. Savarese, "Time-varying interaction estimation using ensemble methods," in *2019 IEEE Data Science Workshop (DSW)*, Jun. 2019, pp. 69–75.

[175] B. Oselio and A. Hero, "Dynamic reconstruction of influence graphs with adaptive directed information," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2017, pp. 5935–5939.

[176] B. Oselio and A. Hero, "Dynamic Directed Influence Networks: A Study of Campaigns on Twitter," in *Social, Cultural, and Behavioral Modeling, 9th International Conference*, 2016, pp. 152–161.

[177] B. Oselio, A. Kulesza, and A. Hero, "Multi-objective optimization for multi-level networks," in *International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction*, Springer, 2014, pp. 129–136.

[178] ——, "Information extraction from large multi-layer social networks," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, 2015, pp. 5451–5455.

[179] ——, "Socio-spatial pareto frontiers of twitter networks," in *International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction*, Springer, 2015, pp. 388–393.

[180] B. Oselio, A. Kulesza, and A. O. Hero, "Multi-layer graph analytics for social networks," in *2013 5th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, IEEE, 2013, pp. 284–287.

[181] ——, "Multi-layer graph analysis for dynamic social networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 4, pp. 514–523, 2014.

[182] B. Oselio, S. Liu, and A. Hero, "Multi-layer relevance networks," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, IEEE, 2018, pp. 1–5.

[183] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. Prentice Hall: Englewood Cliffs, 1982.

[184] S. L. van der Pas and A. W. van der Vaart, "Bayesian community detection," *Bayesian Anal.*, vol. 13, no. 3, pp. 767–796, Sep. 2018.

[185] T. P. Peixoto, "Inferring the mesoscale structure of layered, edge-valued, and time-varying networks," *Phys. Rev. E*, vol. 92, p. 042 807, 4 Oct. 2015.

[186] H. H. Permuter, Y. H. Kim, and T. Weissman, "On directed information and gambling," *IEEE International Symposium on Information Theory - Proceedings*, pp. 1403–1407, 2008, ISSN: 21578101.

[187] H. H. Permuter, Y.-H. Kim, and T. Weissman, "Interpretations of Directed Information in Portfolio Theory, Data Compression, and Hypothesis Testing," *IEEE Transactions on Information Theory*, vol. 57, no. 6, pp. 3248–3259, 2011.

[188] P. Perry and P. Wolfe, "Point process modelling for directed interaction networks," *Journal of the Royal Statistical Society, Series B*, vol. 75, pp. 821–849, 2013.

[189] J. Pitman, *Combinatorial Stochastic Processes: Ecole d'Eté de Probabilités de Saint-Flour XXXII-2002*. Springer, 2006.

[190] C. E. Priebe, J. M. Conroy, D. J. Marchette, and Y. Park, "Scan statistics on enron graphs," *Computational & Mathematical Organization Theory*, vol. 11, no. 3, pp. 229–247, 2005.

[191] A. Prodromidis, P. Chan, and S. Stolfo, "Meta-learning in distributed data mining systems: Issues and approaches," *Advances in distributed and parallel knowledge discovery*, vol. 3, pp. 81–114, 2000.

[192] C. J. Quinn, T. P. Coleman, N. Kiyavash, and N. G. Hatsopoulos, "Estimating the directed information to infer causal relationships in ensemble neural spike train recordings," *Journal of computational neuroscience*, vol. 30, no. 1, pp. 17–44, 2011.

[193] C. Quinn, N. Kiyavash, and T. P. Coleman, "Directed information graphs," *IEEE Transactions on Information Theory*, vol. 61, no. 12, pp. 6887–6909, 2015, ISSN: 0018-9448.

[194] A. Raftery, "Bayesian model selection in social research," *Sociological methodology*, vol. 25, pp. 111–164, 1995.

[195] A. Rao, A. O. Hero, D. J. States, and J. D. Engel, "Using directed information for influence discovery in interconnected dynamical systems," *Proc. SPIE 7074, Advanced Signal Processing Algorithms, Architectures, and Implementations XVIII, 70740P*, 2008, ISSN: 0277786X.

[196] C. Rao, "Diversity and dissimilarity coefficients: A unified approach," *Theoretical Population Biol.*, vol. 21, pp. 24–43, 1982.

[197] C. Rao and T. Nayak, "Cross entropy, dissimilarity measures, and characterizations of quadratic entropy," *IEEE Trans. Inform. Theory*, vol. IT-31, no. 5, pp. 589–593, 1985.

[198] A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese, "Learning social etiquette: Human trajectory understanding in crowded scenes," in *European conference on computer vision*, Springer, 2016, pp. 549–565.

[199] G. Robins and J. S. Salowe, "On the maximum degree of minimum spanning trees," in *Proceedings of the tenth annual symposium on Computational geometry*, ACM, 1994, pp. 250–258.

[200] G. Robins, P. Pattison, Y. Kalish, and D. Lusher, "An introduction to exponential random graph (p*) models for social networks," *Social networks*, vol. 29, no. 2, pp. 173–191, 2007.

[201] P. R. Rosenbaum, "An exact distribution-free test comparing two multivariate distributions based on adjacency," *Journal of Royal Statistics Society B*, vol. 67, no. 4, pp. 515–530, 2005.

[202] L. A. Rossi and O. Gnawali, "Language independent analysis and classification of discussion threads in coursera mooc forums," in *Proceedings of the IEEE International Conference on Information Reuse and Integration*, Aug. 2014.

[203] A. Saumell-Mendiola, M. Á. Serrano, and M. Boguñá, "Epidemic spreading on interconnected networks," *Phys. Rev. E*, vol. 86, p. 026 106, 2 Aug. 2012. DOI: 10.1103/PhysRevE.86.026106. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevE.86.026106.

[204] A. Schein, M. Zhou, D. M. Blei, and H. Wallach, "Bayesian poisson tucker decomposition for learning the structure of international relations," *arXiv preprint arXiv:1606.01855*, 2016.

[205] K. Scricharan, R. Raich, and A. O. Hero, "Estimation of nonlinear functionals of densities with confidence," *IEEE Trans. Inf. Theory*, vol. 57, no. 7, pp. 4135–4159, 2012.

[206] C. R. Shalizi, A. Z. Jacobs, K. L. Klinkner, and A. Clauset, "Adapting to non-stationarity with growing expert ensembles," *arXiv preprint arXiv:1103.0949*, 2011.

[207] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000, ISSN: 01628828. eprint: 0703101v1.

[208] M. Signorelli and E. Wit, "A penalized inference approach to stochastic block modelling of community structure in the italian parliament," *Journal of the Royal Statistical Society: Series C*, vol. 67, pp. 355–369, 2 2018.

[209] J. Silva and R. Willett, "Hypergraph-based anomaly detection of high-dimensional co-occurrences," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 3, pp. 563–569, 2009.

[210] T. A. Snijders and C. Baerveldt, "A multilevel network study of the effects of delinquent behavior on friendship evolution," *Journal of mathematical sociology*, vol. 27, no. 2-3, pp. 123–151, 2003.

[211] L. Solá, M. Romance, R. Criado, J. Flores, A. García del Amo, and S. Boccaletti, "Eigenvector centrality of nodes in multiplex networks," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 23, no. 3, p. 033 131, 2013.

[212] A. Sole-Ribalta, M. De Domenico, N. E. Kouvaris, A. Díaz-Guilera, S. Gómez, and A. Arenas, "Spectral properties of the laplacian of multiplex networks," *Physical Review E*, vol. 88, no. 3, p. 032 807, 2013.

[213] N. Stanley, S. Shai, D. Taylor, and P. J. Mucha, "Clustering network layers with the strata multilayer stochastic block model," *IEEE Transactions on Network Science and Engineering*, vol. 3, no. 2, pp. 95–105, Apr. 2016, ISSN: 2327-4697.

[214] Y. Sun, Y. Yu, and J. Han, "Ranking-based clustering of heterogeneous information networks with star network schema," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2009, pp. 797–806.

[215] T. Sweet, "Incorporating covariates into stochastic blockmodels," *Journal of Educational and Behavioral Statistics*, vol. 40, no. 6, pp. 634–664, 2015.

[216] M. Tahani, A. Hemmatyar, and H. R. Rabiee, "Inferring Dynamic Diffusion Networks in Online Media," *ACM Transactions on Knowledge Discovery from Data*, vol. 10, no. 4, p. 44, 2016, ISSN: 1556472X.

[217] C. Tallberg, "A bayesian approach to modeling stochastic block- structures with covariates," *Journal of Mathematical Sociology*, vol. 29, no. 1, pp. 1–23, 2004.

[218] D. Taylor, S. A. Myers, A. Clauset, M. A. Porter, and P. J. Mucha, "Eigenvector-based centrality measures for temporal networks," *Multiscale Modeling & Simulation*, vol. 15, no. 1, pp. 537–574, 2017.

[219] D. Taylor, S. Shai, N. Stanley, and P. J. Mucha, "Enhanced detectability of community structure in multilayer networks through layer aggregation," *Phys. Rev. Lett.*, vol. 116, p. 228 301, 22 Jun. 2016.

[220] Y. W. Teh, "A bayesian interpretation of interpolated kneser-ney nus school of computing technical report tra2/06," *National University of Singapore*, 2006.

[221] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical dirichlet processes," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1566–1581, 2006.

[222] N. Tishby and N. Zaslavsky, "Deep learning and the information bottleneck principle," in *2015 IEEE Information Theory Workshop (ITW)*, IEEE, 2015, pp. 1–5.

[223] J. R. Tyler, D. Wilkinson, and B. Huberman, "E-mail as spectroscopy: Automated discovery of community structure within organizations.," *Inform. Soc.*, vol. 21, pp. 143–153, 2005.

[224] V. Veitch and D. M. Roy, "The class of random graphs arising form exchangeable random," *arXiv preprint arXiv:1512.03099*, 2015.

[225] J. R. Vergara and P. A. Estévez, "A review of feature selection methods based on mutual information," *Neural computing and applications*, vol. 24, no. 1, pp. 175–186, 2014.

[226] P. Viola and W. M. Wells III, "Alignment by maximization of mutual information," *International journal of computer vision*, vol. 24, no. 2, pp. 137–154, 1997.

[227] D. Wang, H. Wang, and X. Zou, "Identifying key nodes in multilayer networks based on tensor decomposition," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 27, no. 6, p. 063 108, 2017.

[228] S. Wasserman and P. Pattison, "Logit models and logistic regressions for social networks: I. an introduction to markov graphs andp," *Psychometrika*, vol. 61, no. 3, pp. 401–425, 1996.

[229] T. Weissman, Y.-H. Kim, and H. H. Permuter, "Directed information, causal estimation, and communication in continuous time," *IEEE Transactions on Information Theory*, vol. 59, no. 3, pp. 1271–1287, 2012.

[230] A. Wisler, V. Berisha, D. Wei, K. Ramamurthy, and A. Spanias, "Empirically-estimable multi-class classification bounds," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016.

[231] W. H. Wolberg and O. L. Mangasarian, "Multisurface method of pattern separation for medical diagnosis applied to breast cytology," *Proceedings of the National Academy of Sciences*, vol. 87, no. 23, pp. 9193–9196, 1990. DOI: 10.1073/pnas.87.23.9193. eprint: http://www.pnas.org/content/87/23/9193.full.pdf. [Online]. Available: http://www.pnas.org/content/87/23/9193.abstract.

[232] J. Xiang, X. G. Hu, X. Y. Zhang, J. F. Fan, X. L. Zeng, G. Y. Fu, K. Deng, and K. Hu, "Multi-resolution modularity methods and their limitations in community detection," *European Physical Journal B*, vol. 85, no. 10, 2012, ISSN: 14346028.

[233] T. P. Xie, N. Nasrabadi, and A. O. Hero, "Learning to classify with possible sensor failure," *IEEE Transactions on Signal Processing*, vol. 65, no. 4, pp. 836–849, 2017.

[234] K. S. Xu and A. O. Hero, "Dynamic stochastic blockmodels: Statistical models for time-evolving networks," in *Social Computing, Behavioral-Cultural Modeling and Prediction*, A. M. Greenberg, W. G. Kennedy, and N. D. Bos, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 201–210.

[235] X.-S. Yang, "Multiobjective optimization," in *Engineering Optimization*. John Wiley and Sons, Inc., 2010, pp. 231–246, ISBN: 9780470640425. DOI: 10.1002/9780470640425.ch18. [Online]. Available: http://dx.doi.org/10.1002/9780470640425.ch18.

[236] S. Yasaei Sekeh, B. Oselio, and A. O. Hero, "Multi-class bayes error estimation with a global minimal spanning tree," in *56th Allerton Conference on Communication, Control, and Computing*, 2018.

[237] S. Yasaei Sekeh, B. Oselio, and A. Hero, "A dimension-independent discriminant between distributions," in *proc. IEEE Int. Conf. on Image Processing (ICASSP)*, 2018.

[238] S. Yasaei Sekeh, B. Oselio, and A. Hero, "Learning to bound the multi-class bayes error," *IEEE Transactions of Signal Processing (In review)*, 2019.

[239] Z. Yuan, C. Zhao, W.-X. Wang, Z. Di, and Y.-C. Lai, "Exact controllability of multiplex networks," *New Journal of Physics*, vol. 16, no. 10, p. 103 036, 2014.

[240] J. E. Yukich, *Probability theory of classical Euclidean optimization problems*. Springer, 2006.

[241] Y. Zhang, E. Levina, and J. Zhu, "Community detection in networks with node features," *Electronic journal of statistics*, vol. 10, no. 2, 2016.