University of Mississippi

eGrove

Electronic Theses and Dissertations

Graduate School

2018

A Study Of Computational Problems In Computational Biology And Social Networks: Cancer Informatics And Cascade Modelling

Christopher Ma
University of Mississippi

Recommended Citation

Ma, Christopher, "A Study Of Computational Problems In Computational Biology And Social Networks: Cancer Informatics And Cascade Modelling" (2018). *Electronic Theses and Dissertations*. 951.
https://egrove.olemiss.edu/etd/951

A STUDY OF COMPUTATIONAL PROBLEMS IN COMPUTATIONAL BIOLOGY AND SOCIAL NETWORKS: CANCER INFORMATICS AND CASCADE MODELLING

A Dissertation
presented in partial fulfillment of requirements
for the degree of Doctor of Philosophy
in the Department of Computer and Information Science
The University of Mississippi

by

CHRISTOPHER MA

May 2018

ABSTRACT

It is undoubtedly that everything in this world are related and nothing independently exists. Entities interact together to form groups, resulting in many complex networks. Examples involve functional regulation models of proteins in biology, communities of people within social network. Since complex networks are ubiquitous in daily life, network learning had been gaining momentum in a variety of discipline like computer science, economics and biology. This call for new technique in exploring the structure as well as the interactions of network since it provides insight in understanding how the network works and deepening our knowledge of the subject in hand. For example, uncovering proteins modules helps us understand what causes lead to certain disease and how protein co-regulate each others. Therefore, my dissertation takes on problems in computational biology and social network: cancer informatics and cascade model-ling. In cancer informatics, identifying specific genes that cause cancer (driver genes) is crucial in cancer research. The more drivers identified, the more options to treat the cancer with a drug to act on that gene. However, identifying driver gene is not easy. Cancer cells are undergoing rapid mutation and are compromised in regards to the body's normally DNA repair mechanisms. I employed Markov chain, Bayesian network and graphical model to identify cancer drivers. I utilize heterogeneous sources of information to discover cancer drivers and unlocking the mechanism behind cancer. Above all, I encode various pieces of biological information to form a multi-graph and trigger various Markov chains in it and rank the genes in the aftermath. We also leverage probabilistic mixed graphical model to learn the complex and uncertain relationships among various biomedical data. On the other hand, diffusion of information over the network had drawn up great interest in research community. For example, epidemiologists observe that a person becomes ill but they can neither determine who infected the patient nor the infection rate

of each individual. Therefore, it is critical to decipher the mechanism underlying the process since it validates efforts for preventing from virus infections. We come up with a new modeling to model cascade data in three different scenarios

DEDICATION

For my mother, Amy, my first teacher. She taught me how to read and write, but also encourage me to pursue my dream.

For my father, Alan, who was my hero whom I personally look up to. He is not always kind to me but he taught me to ask questions and work hard, but also not to take myself too seriously.

For my ex-girlfriends Christyle Dolan and Karen who always give me companionship and support. Thanks for always be on my side even when I rub them off the way in several occasions.

For my wife Anna who always love me with patience and compassion and of course willing to marry to me even I am still doing my Ph.D.

ACKNOWLEDGEMENTS

I would like to thank my advisers Yixin Chen, Xin Dang and Dawn Wilkins. I was incredibly fortunate to be able to work on a variety of exciting research problems while at University of Mississippi, and without a doubt my PhD would not have led to completion without you guys' collective support and guidance. Your complementary perspectives on graph modeling and scientific research have led me to be a far more well-rounded researcher and provided insights that will shape my career going forward. Special thanks to Dr Xin Dang who always helps me with statistics stuffs and points me to useful literature.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Everywhere we went these days, we realized that events and its interaction can be described by network. In the technology world, we witness the Internet, the World Wide Web and a multitude of social networks which shape people's life. In economics and social influence, we are increasingly experiencing both the positive and negative impact of a global networked economy and its influence on people's value and belief system. In epidemiology, we discover pathogen disseminating throughout our social networks, complicated by mutation of the disease agents. In the state of art bio-medical research, we are unraveling the structure of gene regulatory networks, with the prospect of unlocking the mechanism behind many human diseases. Therefore, there had been a surging interest in understanding how network can help us draw insight into extracting useful information which benefits the society and mankind. On the other hand, recent technological advances have facilitated the collection of large scale high-dimensional data in various field. These data are not only high dimensional, but also heterogeneous, where data are of various types and inter-related to each other. For example, in cancer informatics, it is a common practice to utilize multiple high-throughput technology platforms to measure genotype, RNA gene expression, CNV, mutation and methylation levels. One of the key challenges is the identification of key biological markers that can be leveraged to classify the subject into a known cancer type. There had been substantial progress in the development of some computational methods to address this challenge, however existing methods are in lack of the study of heterogeneity across different cancer types and the mixed types of measurements (binary/count/continuous) across different technology platforms. As a matter of fact, existing methods may fail to identify relevant biological patterns or mechanisms behind many complex human diseases and this call for new methods in analyzing heterogeneous data across multiple platform and discipline. In this dissertation, I studied and developed computational models to integrate data of heterogeneous nature arisen from different networks, and finally, methods we are developing can indirectly infer network structure from the measured data and enable us to extract useful information. I applied it directly to cancer genomics and social network. My dissertation is segmented into four chapters, (chp 2-5) with the first three chapters focus on a numbers of computational methods tackling problems in cancer genomics and the last chapter focus on social networks. To summarize:

Chapter 2 attempts to answer the question of distinguishing mutations in a given tumor that drive cancer from the random mutations that have no consequence for cancer. The vast majority of mutations in most cancers are largely somatic, meaning they occur during the lifetime of an individual and cannot be inherited from ancestors. The somatic mutations in a given tumor can be categorized into two types, namely drivers in which its mutations are responsible for causing cancer and passengers in which its mutations possessing no consequence for cancer. Therefore identifying the driver mutations is a starting point in understanding the mechanisms which drive uncontrolled cell growth. In this chapter, we developed a model by utilizing the patient gene mutation profile, gene expression data and gene gene interactions network to construct a graphical representation of genes and patients. We then construct a Markov chain out of these biological entities and Markov processes for mutation and patients are triggered separately within this multi-graph. After this process, cancer genes are prioritized automatically by examining their scores at their stationary distributions in the eigenvector which shed some light in identifying cancer drivers.

Chapter 3 attempts to answer questions regarding the background mutation rate and driver mutation probability of each gene out of the somatic mutation data. We leverage the power of Bayesian statistics and introduce a hierarchical Bayesian methodology to estimate candidate genes driver mutation rates and background mutation rates from somatic mutation data. We choose a suitable prior distribution for modeling the driver mutations and the background mutations for each candidate gene. We apply our method to ovarian cancer data and accordingly estimate proportion of drivers for this type of cancer. A set of candidate cancer driver genes is suggested by examining their probability of mutation at the end of this chapter.

In chapter 4, we discuss how to incorporate different biological data types by means of a mixed graphical model. The state of art in genomic technologies have collected many genomic, epigenetic, transcriptomic, and proteomic data of varied types across different biological conditions. Historically, it was always a challenge to come forward with ways to integrate data of different types. In this chapter, we leverage the node-conditional univariate exponential family distribution to capture the dependencies and interaction between different data types. The graph underlying our mixed graphical models contain both undirected and directed edges. Furthermore, we incorporate these heterogeneous data across different experimental condition which lead us to a more holistic view of the biological system and help unraveling the regulatory mechanism behind complex diseases. We then integrate the data across related biological conditions through multiple graphical models. We applied our method to cancer genomics with a goal to discover important bio-markers out of different cancer data.

In chapter 5, we study the problem of the diffusion of information, influence and disease over networks. Very often we are only capable of collecting cascade data in which an infection (receiving) time of each node is recorded but without further transmission information over the network. We didn't know where she obtained the pathogen from, nor how long it took her to get infected after exposure. As a matter of fact, the goal of this chapter is to propose a novel model to infer infection rates of diffusion processes. We successfully present three modelings with a common transmission rate, with different transmission rates and with different infection rates. We also extent our model to deal with the multiple source problem in which there exist multiple sources which contribute to the cascade data. We consider non-overlapping, partially overlapping two sources and fully overlapping multiple sources diffusion networks. For non-overlapping networks, the problem is transformed directly to the identification of the starting time of the second source. For the partially overlapping scenario, a mixture model is adopted and EM algorithm is devised for obtaining estimators. The fully overlapping case is an extension of the mixture modeling. We applied our method on real and synthetic data which demonstrate that our models can accurately estimate the transmission rates from one source as well as multiple source cascade data.

Each chapter is self-contained and readers can feel free to jump ahead to read any chapter in different orders.

CHAPTER 2

RANDOM WALK MODEL APPROACH TO IDENTIFY CANCER DRIVER GENES

Cancer is a disease driven largely by the accumulation of somatic mutations during the lifetime of a patient. Distinguishing driver mutations from passenger mutations had posed a challenge in modern cancer research. With the widespread use of microarray experiments and clinical studies, a large numbers of candidate cancer genes are produced and extracting informative genes out of them is essential. In this chapter, we aim to find the informative genes for cancer by using mutation data from ovarian cancers. We utilized the patient mutation profile, gene expression data and gene gene interactions network to build a graphical representation of genes and patients and construct Markov processes for mutation and patients separately. After this process, we can prioritize cancers genes automatically by looking into their scores at their stationary distributions. Comprehensive experiments show that the utilization of heterogeneous sources of information is very helpful in finding important cancer genes.

1 Background

Finding important genes for cancer is always an important branch of cancer research. With the advance of large scale micro-array technology, an unprecedentedly huge amount of data is produced in terms of micro-array gene expression. This not only promote research on elucidating the molecular process driving tumor progression, but also demand strong need to introduce new methodology to improve cancer therapy. Consequently, gene expression profiles had been widely analyzed and studied which had been proved effectively to identify tumor subtypes and predict outcomes in patient survival analysis. Further we postulate that the genes which cause the same disease or same cancer tend to correlate with each other in protein protein relationship networks. As a matter of fact, we employ Pearson correlation coefficients to compute the correlation of the gene expressions of different genes in an attempt to identify co-expressed genes or other important bio-makers that are responsible for cancer progression in our ranking model. On the other hand, it had been widely perceived that cancer is not a disease of individual mutation but a group of genes acting together in a molecular network. Hence the incorporation of publicly available protein protein interaction database and pathway interaction information in cancer study are crucial in understanding interactions among genes and unveiling the molecular pathway of cancer. Therefore, we incorporate the PPI interaction data in our framework to rank the cancer genes. Moreover, patient somatic mutation profile, which records the mutation profile of each patient on frequency basis is also incorporated in our framework as background information. Combining all, we aim to find the important genes which play a role in cancer by utilizing these heterogeneous sources of data. We propose a framework that encodes various heterogeneous sources including 1) gene expression profiles 2) patient somatic mutation profile 3) PPI networks from HumanNet and pathwayCommons in a graphical model. We then define separate Markov Chain on the genes and patients and then perform random walk respectively. Inspired by Google PageRank algorithm, we introduce randomness by allowing each patient to randomly choose one gene to hop and each gene to randomly choose one patient to hop for small amount of probability. In this way, not only proximity relationships between connected gene nodes can be exploited, but also gene nodes which are poorly connected can also be visited so as to discover all important but not too similar biomakers globally through some noise introduced through teleportation. Our random walk framework consists of five models which differ from each other subtlely by the sequence and order they perform the random walk on our constructed patient gene network. Noticed that in the genes network, we merge the gene gene interaction network with the gene correlation network in a multigraph which is capable of connecting multiple edges between a pair of gene nodes. Therefore following different orders to traverse this multigraph result in different transition matrices for the genes and patients as shown in our five different proposed models below. Our works sucessfully integrate multiple heterogeneous sources of data in a graphical model to find the top ranked cancer genes. Within our proposed framework, we compute the major eigenvector of each individual stationary matrix in each model and each gene is being ranked according to the value of its corresponding entry in that eigenvector. Comprehensive experiments show that the utilization of heterogeneous sources of information is very helpful in finding important cancer genes. All the five models are capable of ranking those genes as reported and discovered from other cancer studies within top positions.

2 Related Works

Most of the related works focused on ranking genes in cancer modules and biomolecular networks [2, 74, 54]. In [75], Re and Valentini utilize random walk and random walk with restart to rank genes with respect to their likelihood of belonging to each cancer module by exploiting the global topology of the functional interaction network, and local connections between genes close to genes in each cancer module. In [27], Erten, Bebek and Koyuturk come forward with a random walk based algorithm which postulates that genes which are associated with similar diseases exhibit patterns of topological similarity in PPI networks. They introduce the concept of topological profile to measure the similarity of genes and come forward with an algorithm which measures the topological similarity between the seed genes and candidate genes in ranking. In [70], Petrochilos and Deanna use random walk together with network community analysis to identify cancer-associated modules in expression data. Another group of gene ranking methods similar to random walk approach is network propagation technique. In [81], Sharan, Ulitsky and Shamir describe a numbers of computational approaches, including direct methods, which propagate functional information through the network, and module-assisted methods, which infer functional modules within the network for the annotation task. In [24], Deng utilizes Markov field to perform network propagation to predict protein function. In [63], Mostafavi and Ray make use of Guassian Random Field to perform network propagation. In [98], Zhang and Wei extends the general network propagation framework to involve graphs with nodes and edges to be initialized as positive and negative numbers for detecting differential gene expressions and DNA copy number variations (CNV) by modelling gene up/down-regulation or amplification/deletion CNV events to be positive and negative respectively. Most of the works above are based on propagating known gene labels across the network, by exploiting the weighted connections between genes, until a stopping condition is fufilled. This method has an advantage of capturing hidden clusters to recover false negatives and eliminating false positives, but it also has a disadvantage of exploring too far similarities between genes too.

3 Random Walk Markov Model

In this section, we demonstrate on how to represent all different sources of information and come forward with a framework to integrate all different sources of information in tackling a cancer gene ranking problem. We construct Markov Chains for both the mutations(genes) and patients and illustrate that the eigenvector of the stationary matrix represents the rank of each individual gene in our framework.

Heterogeneous Information

In this section, we show how to represent the three sources of information.

1) *Patient Mutation Profile*: Briefly, Patient-Mutation Profile is a two dimensional binary (0,1) matrix with rows stand for patients and columns stand for mutations of the genes. Each entry is either 0 or 1, a 1 indicates that a mutation has occured in the tumor relative to the germline(a single necleotide base change or the insertion or deletion of base) on that patient, a 0 otherwise.

2) *Gene Gene Interaction Network*: We utilize two types of gene gene interaction networks from two sources: HumanNet v.1 and PathwayCommons. HumanNet is a probabilistic func-

tional gene network comprising 18,714 validated protein-encoding genes of Homo sapiens, constructed by using naive Bayesian approach to weigh different types of data evidence collected in humans, yeast and worms according to how well those genes that are known to function together in Hommo-sapiens and combine into a single interaction score. PathwayCommons is a database of publicly available biological pathway information from multiple sources which focus on protein protein interactions and functional relationships between genes in signaling and metabolic pathways. For the sake of our problem framework, we filter out all non human genes and interactions in PathwayCommons and the remaining interactions are utilized in our problem framework. All the gene gene interaction networks mentioned above will be represented as an undirected graph $G(V, E)$ where V stands for the set of genes and edges $(i, j) \in E$ are weighted by a weight matrix W, whose element w_{ij} is the weight of the edges $(i, j) \in E$ which stands for the strength of interaction between gene i and gene j compiled from two sources of gene gene interaction networks mentioned above. 3) *Gene Expression Profiles*: Gene expression is the expression level of a gene on an individual which is measured through microarray experiment. The gene expression data show the behaviors of genes in tumor and normal samples which are used to estimate the similarity between genes, where informative genes with similar functionality are widely believed to possess similar gene expression through microarray experiments. By using gene expression data, we are capable of constructing a gene correlation graph/network which is used in our framework. A gene correlation graph/network is a graph $H(V, E)$, where V represents the set of genes and an edge $(i, j) \in E$ is weighted by the Pearsons correlation coefficient between the gene expression of gene i and gene j.

3.1 Mutual Reinforcement Model

In Mutual Reinforcement Model, we assign each mutation(gene) a driver score μ_i and each patient a patient score π_i. We allow each patient to cast a vote on each mutation(gene) and each mutation(gene) to cast a vote on each patient. Consequently, the driver score of a mutation(gene) is determined by the total votes received by the mutation(gene) and the patient score of a patient is determined by the total votes received by the patient. Therefore, a high driver score implies that the mutation is shared by patients with high patient scores whereas a high patient score implies that the patient has mutations with high driver scores. With the help of patient mutation profile, we begin to lay out some notations. The affinity matrix A of a bipartite graph is defined as $A_{ij} = 1$ if and only if patient i has mutation on gene j and 0 otherwise where m is the total numbers of patients and n stands for the total numbers of genes. Since the driver score of a mutation μ_i is directly proportional to the numbers of patients having that mutation and the patient score π_i is directly proportional to the numbers of mutations possessed by the patient, the driver score and mutation score are defined mutually to each other which justify the following equations.

$\mu_j \propto \sum_{i \in k : A_{kj}=1} \pi_i$

$\pi_i \propto \sum_{j \in k : A_{ik}=1} \mu_j$

The first of foremost, the probability in which a patient i traverses to mutation(gene) j is governed by the following matrix

$A_r[i, j] = \frac{A_{ij}}{\sum\limits_{k=1}^{n} A_{ik}}$

Similarly, the probability in which a mutation(gene) j traverses to patient i is governed by the following matrix

$A_c[i, j] = \frac{A_{ij}}{\sum\limits_{k=1}^{m} A_{kj}}$

Notice that A_r is a row stochastic matrix and A_c is a column stochastic matrix. To augment our model with randomness, we allow for most of the time, each patient will follow the outgoing edges and hop to one of his neighbors in the gene partite set with the probability governed by matrix A_r and for a small percentage of time, each patient can choose arbitrarily a mutation(gene) and teleport there. The factor $1 - \alpha$ reflects the probability that the patient quits the current matrix A_r for traversal and teleports to any gene. As a patient can teleport to any mutation(gene) j, each mutation(gene) has equal probability to be chosen. This justifies the following transition matrix for patients:

$$B_r[i, j] = \alpha * A_r + (1 - \alpha)\tfrac{1}{n} I_{m*n}$$

where I_{m*n} is a m by n matrix with all entries 1. Similarly, for the case of mutations(genes), we allow each mutation(gene) to choose arbitrarily a different patient to do the teleportation for a small amount of time and hence it justifies the following transition matrix for the mutations(genes).

$$B_c[i, j] = \alpha * A_c + (1 - \alpha)\tfrac{1}{m} I_{m*n}$$

Moreover, in the partite set of genes (mutations), we incorporate the information collected from gene gene interaction network and gene correlation network as described in previous section to augment our model. As we mentioned, a gene gene interaction network can be represented as an undirected graph $G(V, E)$ where V stands for the genes and edges $(i, j) \in E$ are weighted by a weight matrix W, whose element w_{ij} stands for the strength of interaction between gene i and gene j, we normalize the matrix W to define the transition probability matrix between the genes. We define a transition probability matrix $Q = D^{-1}W$ where D is a diagonal matrix with diagonal elements $d_{ii} = \sum_j w_{ij}$. The elements q_{ij} of Q represents the probability of a random transition from gene i to gene j. The matrix Q defines a valid transition matrix whose elements q_{ij} satisfy the probabilistic constraint $\sum_j q_{ij} = 1$. There-

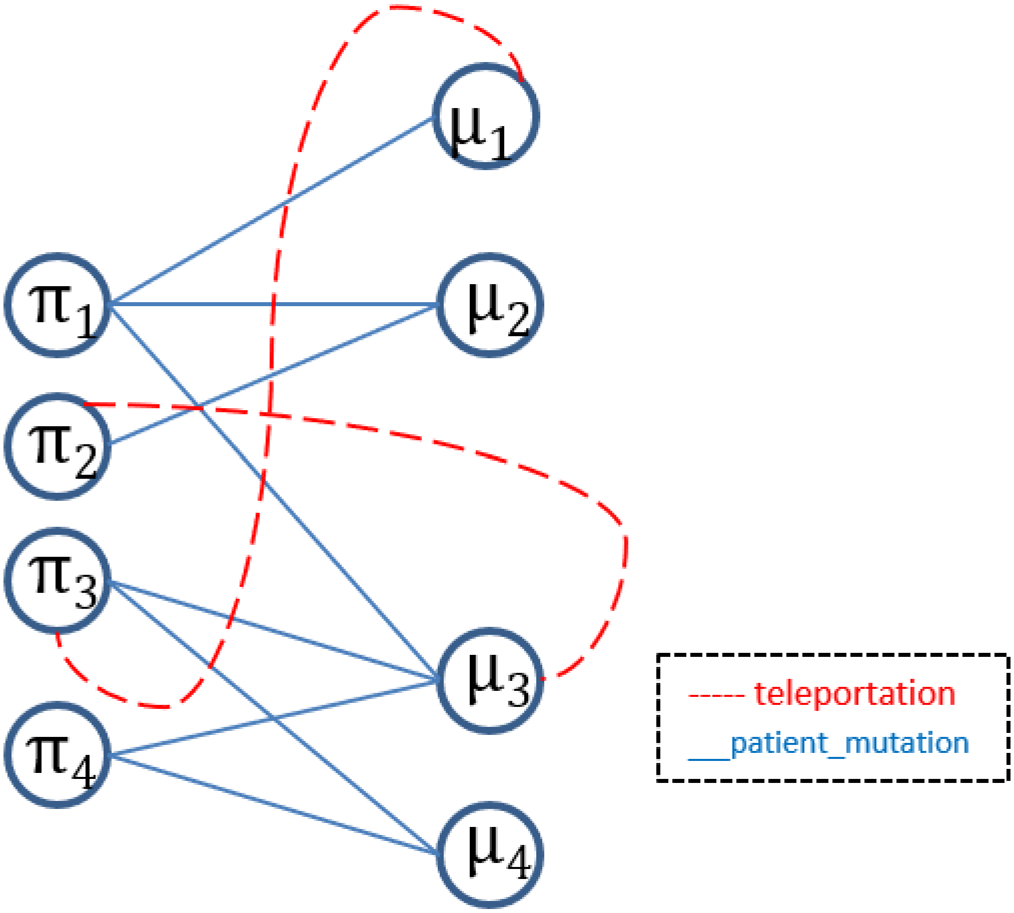

Figure 2.1. Patient Mutation Network

fore the transition matrix of the genes following the gene gene interaction network is as follow:

$$B_g[i,j] = Q = D^{-1}W$$

In the aftermath of normalizing matrix W, it is time to incorporate information we obtained from gene correlation network as described in section 3.1. Each edge in the gene correlation network is weighted by matrix H whose elements $H(i,j)$ represents the Pearsons correlation coefficient between the gene expression of gene i and gene j. To be more precise, let β_u be the gene expression vector of gene u on the patients, then

$$H(u,v) = corr(\beta_u, \beta_v)$$

$$= \frac{\sum t \in V(\beta_u(t) - \frac{1}{V})(\beta_v(t) - \frac{1}{V})}{\sqrt{\sum t \in V(\beta_u(t) - \frac{1}{V})^2}\sqrt{\sum t \in V(\beta_v(t) - \frac{1}{V})^2}}$$

where $corr(X,Y)$ denotes the Pearson correlation of random variable X and random variable Y. The idea behind this approach is that if two genes play a role in a specific cancer, their gene expression may be correlated to each other. Notice that the matrix H contains entries with value lying between -1 and 1, we take the absolute value of each entry in H. Further, we normalize the matrix H to define the transition probability between the genes like we did in the case of matrix W in gene gene interaction network. This results in the following transition matrix of the genes following the gene correlation network :

$$B_h = T = D^{-1}H$$

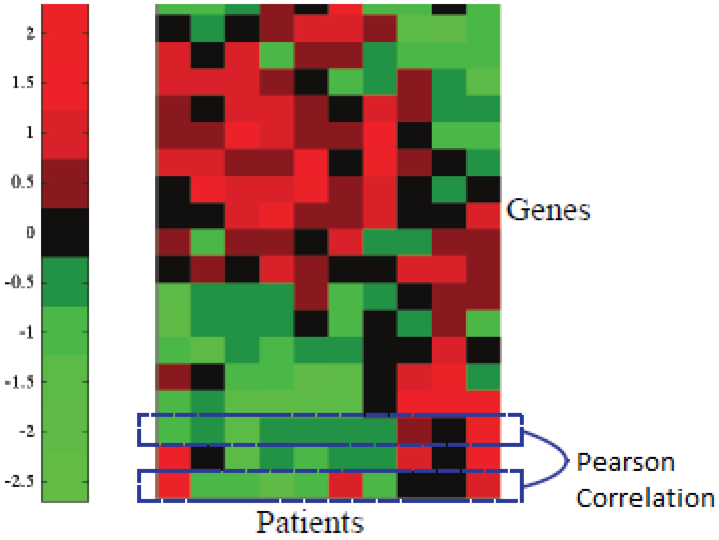

Figure 2.2. Microarray Gene Expression Data

3.2 Markov Chain

In this section, we proposed a series of models using the transition probability matrices defined above. Recall μ stands for the score vector of mutations (gene rank) and π stands for the score vector of patients (patient rank)

1.Random Walk Multiplicative Model Gene Interaction Start(RW-MMGIS):

This model starts with random walk on gene gene interaction network and next gene correlation network and then to patient and back to gene gene interaction network and repeats which defines the stationary distribution for the mutations and patients:

$$\begin{aligned} \mu &= B_r^T B_c B_h^T B_g^T \mu \\ \pi &= B_c B_h^T B_g^T B_r^T \pi \end{aligned} \tag{2.1}$$

2.Random Walk Multiplicative Model Gene Correlation Start(RW-MMGCS):

This model starts with random walk on gene correlation network and heads to gene gene interaction network and then to patient and back to gene correlation network and repeats which justifies the stationary distribution for mutations and patients:

$$\begin{aligned}\mu &= B_r^T B_c B_g^T B_h^T \mu \\ \pi &= B_c B_g^T B_h^T B_r^T \pi\end{aligned} \tag{2.2}$$

3.Random Walk Additive Model(RW-AM):

This model differs from the previous ones in a way that the overall transition matrix for the mutations is a linear combination of transition matrix following gene gene interaction network, the patient mutation profile and the gene correlation network with each transition matrix contributing a part to the overall transition matrix for mutations.

$$\begin{aligned}\mu &= (\alpha * B_r^T B_c + \beta * B_g^T + \gamma * B_h^T)\mu \\ 1 &= \alpha + \beta + \gamma \\ \pi &= B_c B_g^T B_h^T B_r^T \pi\end{aligned} \tag{2.3}$$

4. Random Walk Multiplicative Model Penalized:(RW-MMP)

In this model, we combine the gene gene interaction network and gene correlation network together in one network before performing a random walk. Recall that the gene gene interaction network is weighted by a weight matrix W, whose element w_{ij} stands for the interaction strength of the interaction between gene i and gene j and the gene correlation network is weighted by matrix H whose elements H_{ij} represents the Pearson's correlation coefficients between the gene expression of gene i and gene j. We penalize each edge in gene gene interaction network given by matrix W by the exponential value of its corresponding gene correlation value given by matrix H divided by their mean. Please be noted that each entry in H had been taken the absolute value. These justify the following equations:

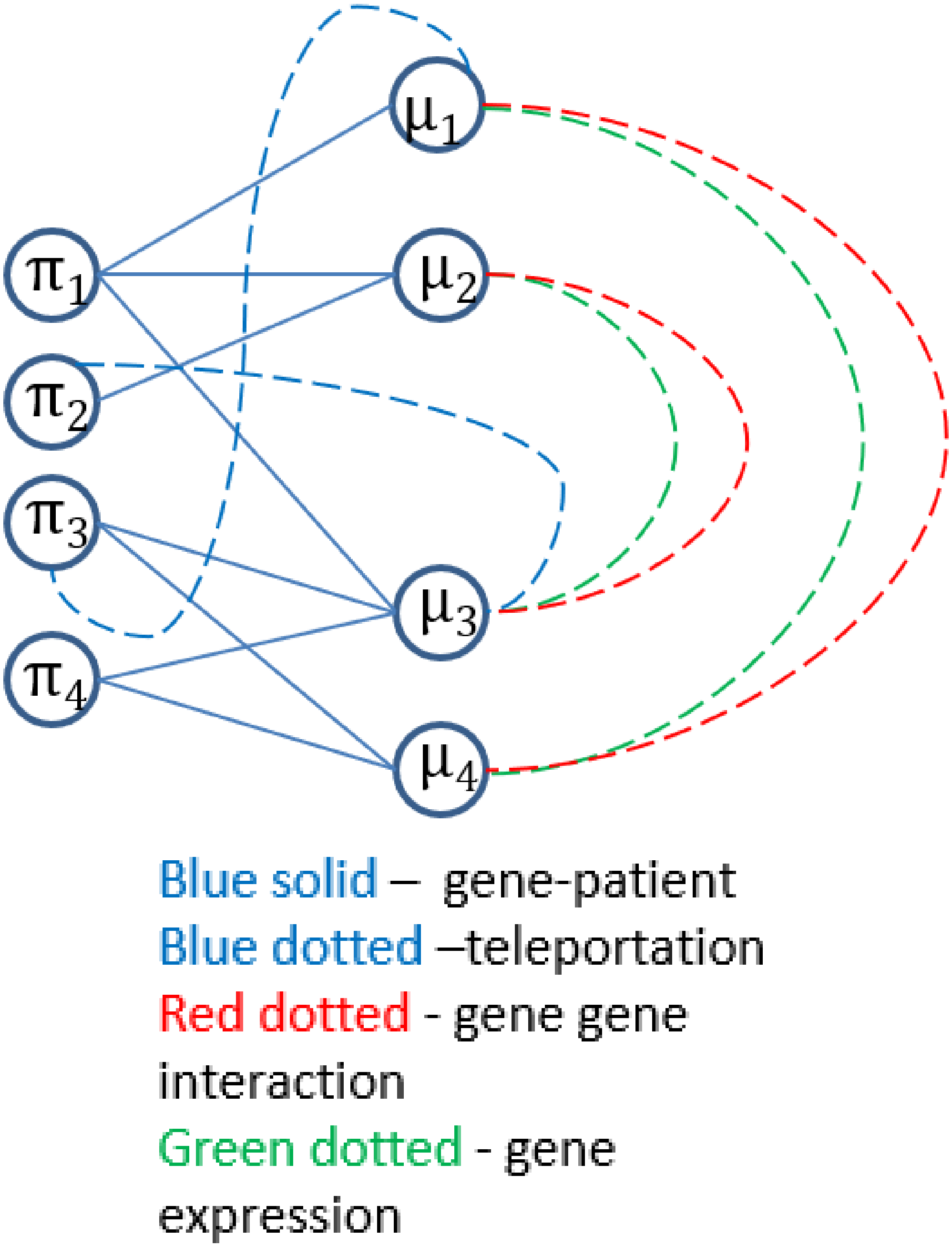
μ_1
π_1
μ_2
π_2
π_3
μ_3
π_4
μ_4
Blue solid – gene-patient
Blue dotted –teleportation
Red dotted - gene gene interaction
Green dotted - gene expression

Figure 2.3. Overall Patient Mutation Network

$$
\begin{aligned}
\sigma &= \text{mean of all entries of matrix H} \\
W_{ij} &= W_{ij}\exp(H_{ij}/\sigma) \\
B_g[i,j] &= Q = D^{-1}W \qquad (2.4) \\
\mu &= B_r^T B_c B_g^T \mu \\
\pi &= B_c B_g^T B_r^T \pi
\end{aligned}
$$

5. Random Walk Multiplicative Model Average:(RW-MMA)

In this model, we take the average output from the random walk on gene gene interaction network, gene correlation network and patient mutation profile in each step. For the sake of clarity, we will write it out as an algorithm as shown in Algorithm 1 below:

Algorithm 1 Random Walk Multiplicative Model Average algorithm (RW-MMA)

```
procedure RANDOM WALK MULTIPLICATIVE MODEL AVERAGE($l, m, r, B_g, B_r, B_c, B_h$)
    $R_0$ ← all entries are 1/n
    for $t = 1$ to max($l$,$m$,$r$) do
        if $t$ <=l then $R_{left} = B_r^T B_c * R_{t-1}$
        end if
        if $t$ <=m then $R_{mid} = B_g^T * R_{t-1}$
        end if
        if $t$ <=r then $R_{right} = B_h^T * R_{t-1}$
        end if
        $R_t = \frac{(\sigma_{t<=l} * R_{left} + \sigma_{t<=m} * R_{mid} + \sigma_{t<=r} * R_{right})}{\sigma_{t<=l} + \sigma_{t<=m} + \sigma_{t<=r})}$

        $\sigma_{t<=x} = 1$ if $t$ <=x and 0 otherwise
    end for
    return $R_t$
end procedure
```

In the aftermath of defining various models, it remains to demonstrate that all the markov chains in all the five proposed models are valid and all the corresponding transition matrices converge to unique stationary matrices which result in a unique eigenvector as our ranking vector in each model.

Lemma: All the above transition matrices define valid Markov Chains that converge to a unique stationary eigenvectors.

For the sake of simplicity, the proof of the model Random Walk Multiplicative Model Gene Interaction Start(RW-MMGIS) is outlined as below, the rest of the models can be proved similarly. Convergence: To prove convergence, we must prove the Markov chain defined by the transition matrix $C_r^T C_c C_h^T C_g^T$ is irreducible and aperiodic. Notice that each mutation is permitted to teleport to any patient and each patient is permitted to teleport to any mutation with a small probability. Coupled with the definitions of B_r and B_c, all entries in matrix C_r and C_c are strictly positive. Since C_h and C_g are also positive stochastic with nonnegative entries, the transition matrix defined by $C_r^T C_c C_h^T C_g^T$ are all strictly greater than 0 in all entries. This proves that every state in the state space S can be reached from every other state in the state space in a finite number of moves with positive probability which proves irreducibility. For aperiodicty, notice the fact that each $P_{ii} > 0$ which implies that the minimum number of steps from each state i returning to itself is 1 which proves aperiodicity. Uniqueness: To prove uniqueness,notice that C_r is a row stochastic matrix and hence C_r^T is column stochastic, in addition, C_h^T, C_r^T, C_c, C_g^T are all positive column stochastic and hence the product of positive column stochastic matrices is also positive column stochastic. By Perron-Frobenius Theorem, 1 is an eigenvalue of multiplicity one of the matrix $C_r^T C_c C_h^T C_g^T$ which is the largest and all the other eigenvalues are in modulus smaller than 1. Furthermore the eigenvector corresponding to eigenvalue 1 has all entries positive. In particular, for the eigenvalue 1 there exists a unique eigenvector with the sum of its entries equal to 1. This gives us a unique eigenvector as our rank for the genes. Similar arguments can be applied for the proof of the existence of our patient rank.

4 Results and Finding

The data sets used for the experiment were taken from the study of Integrated Genomic Analyses of Ovarian Carcinoma led by the Cancer Genome Atlas. The associated results and discussions were published in NATURE 2011 [66]. The analysis of 489 clinically annotated stage III-V HGS-OvCa samples and its corresponding normal DNA were reported in the article and posted on its associated website. The data incorporates the age at diagnosis, stage, tumour grade and surgical outcome of patients diagnosed with HGS-OvCa. We downloaded the TCGA-OV-mutations data and the unified expression profiles from the TCGA Data Portal website for our purpose. In the aftermath of data cleaning, we retain mutations containing insertion, deletion and alternation of base only. Finally a patient mutation profile table which comprises 316 patients and 8404 genes is obtained. Similar procedures were carried out on obtaining the gene expressions data from the website. Pearsons correlation coefficients are calculated on the gene expression data in pairwise fashion to obtain the gene correlation value between each pair of genes and the gene corelation graph is constructed. We utilized two different protein protein interaction networks for our experments. HumanNet is a probabilistic functional gene network which consists of 18,714 protein encoding genes and 476399 interactions between the genes of Homo sapiens. Pathway Commons is a collection of publicly available metabolic pathway database in conjunction with interactions from multiple organisms. It was filtered to retain human genes and interactions for the sake of our experiments. We obtained the required data through its web portal for download and query.

4.1 Ground Truth Data

We compiled a set of genes published in various literature on several cancer studies which are certified to be ovarian cancer genes to be our ground truth cancer genes in the evaluation of our proposed models. Afterwards, the experiments on our five proposed models are run. A gene scoring vector (gene rank μ) for each of the six models is obtained. We then evaluate our proposed models by the rankings of the ground truth genes in each of the six proposed models gene scoring vector μ and demonstrate the effects of integrating more background information in ranking. Precision/Recall graph and the top 25 genes appeared in each of the gene scoring vector (gene rank μ) of the five proposed models are presented in subsequent sections. Table 2.1 below tabulates the collection of ovarian cancer genes (ground

GENE	Literatures
BRCA1	[30],[10],[66]
BRCA2	[25]
BMPR1A	[82],[3]
BRIP1	[85]
MLH1	[78]
FHIT	[25],[6]
TFRC	[53]
FGFR2	[39],[60]
GATA3	[57]
MYST4	[88]
PTEN	[69]
FAS	[58],[6]
RB1	[85]
SEPT9	[79]
YWHAE	[32]
TP53	[66]
PIK3CA	[28]
BRAF	[28]
KRAS	[28]
AIB1	[4]
MSH2	[78]
BMP4	[56],[55]
TRIP1	[40]
MYC	[40]
EP300	[40]

Table 2.1. Ground Truth Genes

truth genes) and the associated references.

4.2 Experimental Results

We run the experiments on our six proposed models using the data set we obtained. In our experiments, we set $\alpha = 0.75$. For the additive model (RW-AM), we set $\alpha = 0.3$, $\beta = 0.3$ and $\gamma = 0.4$. Three benchmark models are utilized to evaluate our proposed models. The first one is frequency based in which each gene is awarded a rank in accordance with the occurrence of mutation which means the higher the frequency of occurrence of mutations on that gene, the higher rank will be awarded. The other two benchmark models are random

Model	Numbers of Appearances	Average Rank
RW-MMGIS	14	40
RW-MMGCS	17	38
RW-AM	15	39
RW-MMP	16	40
RW-MMA	16	42
RW-GC	9	62
RW-PG	4	17
FREQUENCY BASE	4	18

Table 2.2. Top 1 percent of the Rank

walk based in which we perform random walk on gene correlation network (RW-GC) and patient mutation (RW-PM) network respectively and a gene scoring rank vector μ for each network is attained. We present the total number of appearances of ground truth genes in the top 1 percent of the gene rank μ of each model as follows in Table 2.2:

The six proposed models outperform all the benchmark models. This can be demonstrated from the above table that the number of occurrences of ground truth genes in the above six models outnumbers the three benchmark models. We found that incorporating heterogeneous sources of biological information enhances the performance of identifying ovarian cancer genes. In the nine models, RW-MMGCS yields the best performance, followed by RW-MMA and then RW-MMP and then RW-AM and then RW-MMGIS and then RW-MMPFS and then followed by three benchmark models at last: RW-GC, RW-PG and FREQUENCY BASE. Please note that a larger gap occurs between the results of two benchmark models with RW-GC outperforming RW-PG. This underscores that the gene expression data is more informative than patient mutation profile in locating ovarian cancer genes. All in all, integrating various heterogeneous sources of information helps in locating ovarian cancer genes.

4.3 Evaluation

In this subsection, Precision/Recall graph by adjusting the threshold on the rank of the ground truth genes is presented. Precision is defined as the fraction of the ground truth genes among all genes ranked above each threshold. Recall is defined as the fraction of ground truth genes which are ranked above each threshold among all known ground truth genes. 25 ground truth genes in each experiment are used and the results are tabulated in Figure 2.4. All six proposed models outperform the three benchmark models. RWMMGCS yields the best performance, followed by RW-MMA and then RW-MMP and then RW-MMGIS and then RW-AM. RW-MMGCS, RW-MMA and RW-MMP show a very high precision rate at recall rates running from 0.1 to 0.2. This demonstrates that they are able to locate several true positive genes (ground truth genes) in topmost positions within the ranked list. Since we use only 25 ground truth genes in our experiments, we expect to achieve a better result if more candidate cancer genes are included. Almost all the models decrease their performances monotonically towards the higher recall rate except FREQUENCY BASE and RW-PG in which their precision increases a little towards a little higher recall rate and then plummets sharply. This can be explained by the fact that these two models discover a multitude of false positive at low recall rate while they obtain a little better precision towards higher recall rate when they are able to rank a few ground truth genes below the top ranked genes. Above all, we demonstrate that the integration of more heterogeneous background information in the ranking helps achieve a better recall/precision rate. There is one parameter α in our proposed models (RWMMGCS, RW-MMA, RW-MMP, RW-MMGIS) which is the probability of teleportation of genes and patients. We performed an experiment on adjusting the value of α from 0 to 1 to inspect its relation to the average rank of the ground truth genes. The result is tabulated in Figure 2.5. From above, the best α obtained is around 0.8 which achieves the lowest average ground truth genes ranking. Subsequently, in our additive model(RW-AM), we have three parameters α, β, γ that have to be determined. To evaluate these three parameters, we fix one of the parameters each time and adjust the other two parameters and record the best average ground truth genes ranking and the result

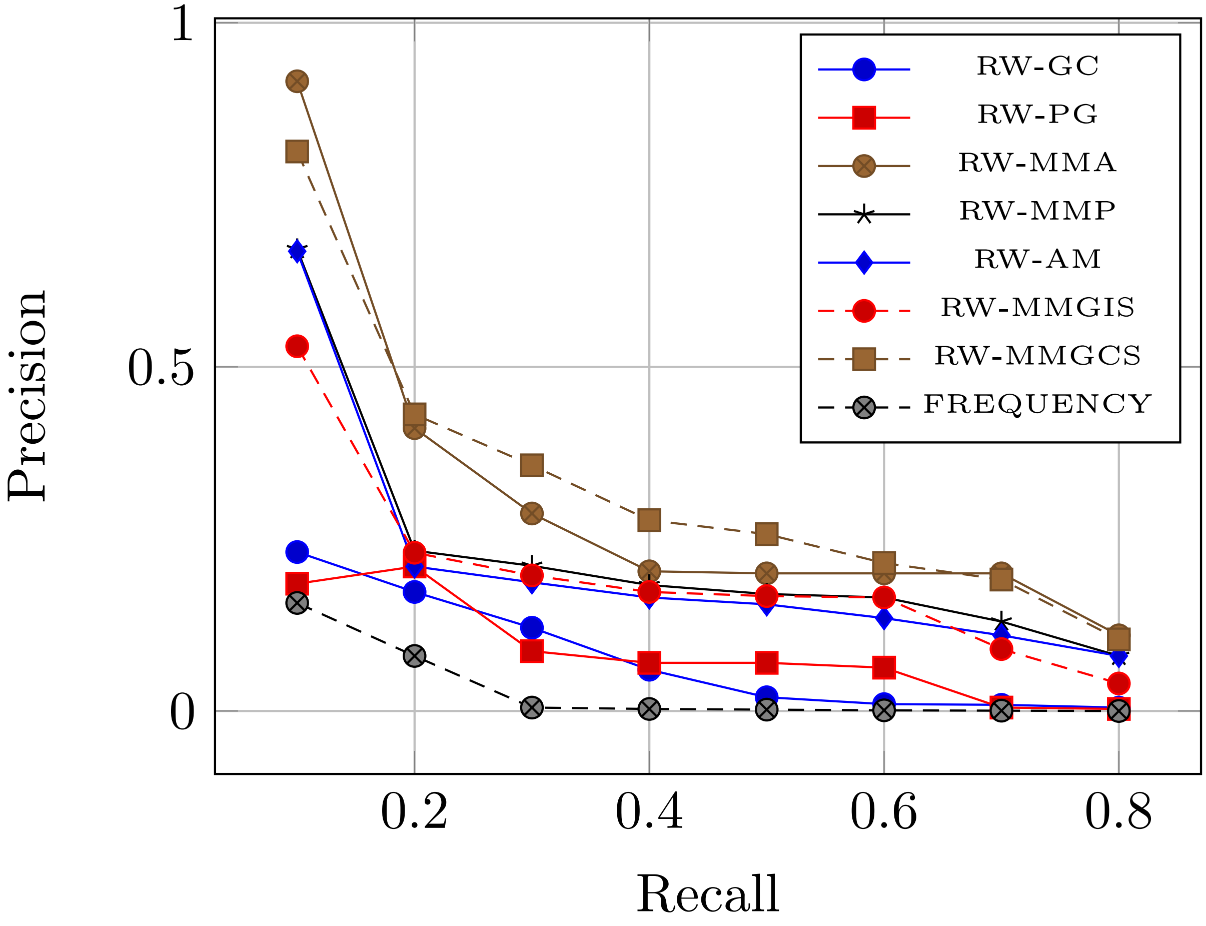

Figure 2.4. Recall/Precision of our Random Walk Markov Chain Models

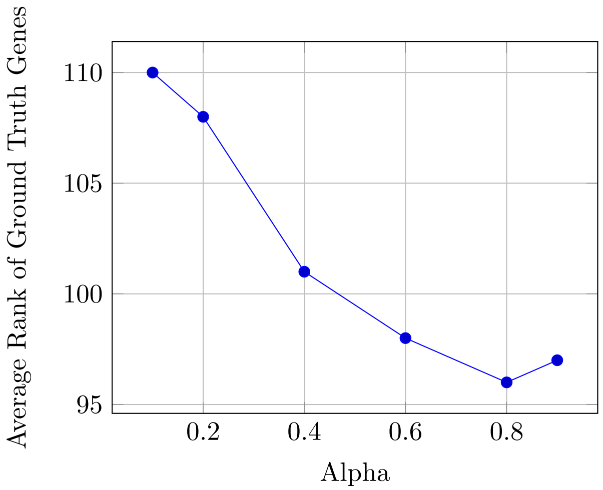

Figure 2.5. Average Rank of Ground Truth Genes By Adjusting Teleportation Parameter α

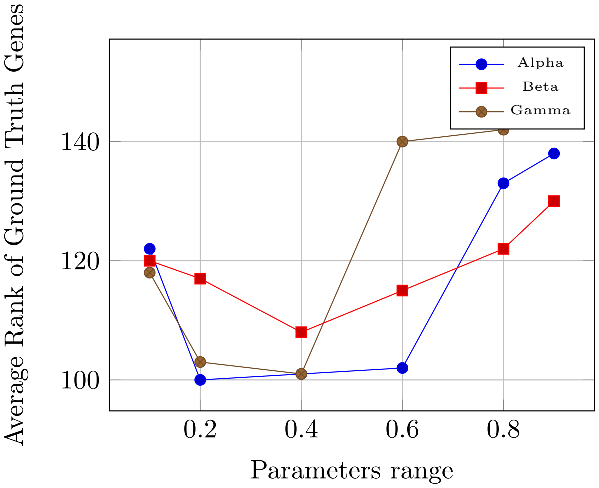

Figure 2.6. Average Rank of Ground Truth Genes Achieved By Fixing Each Parameter in RW-AM

is tabulated in Figure 2.6.

5 Conclusion

In this chapter, a Markov Chain Model for discovering important cancer genes through integration of heterogeneous sources of information are proposed: patient mutation profile, gene gene interaction network and gene correlation network in an unsupervised manner. Experimental results demonstrate that our proposed models outperform all benchmark models. Our future work will focus on developing graph Laplacian in learning cancer genes priority.

CHAPTER 3

GENERATIVE BAYESIAN MODEL APPROACH TO IDENTIFY CANCER DRIVER GENES

In this chapter, we approach the cancer driver genes identification problem by means of Bayesian modeling approach. Cancer is a disease characterized largely by the accumulation of somatic mutations during the lifetime of a patient. Distinguishing driver mutations from passenger mutations had posed a challenge in modern cancer research. With the state of art of micro-array technologies and clinical studies, a large numbers of candidate genes are extracted. Extracting informative genes out of them is essential. In this chapter, we aim to find the cancer driver genes using somatic mutation data and protein protein interaction data. We developed a generative mixture model coupled with Bayesian parameter estimation to estimate background mutation rates and driver probabilities as well as the proportion of drivers among sequenced genes. We choose suitable prior distributions for modeling both driver probabilities and background mutations of each gene. We apply our method to ovarian cancer data and numerically estimated the solution.

1 Background

Understanding cancer biology and the mechanism behind cancer progression has always been an important branch of cancer research. Another equally important question is to discover driver genes whose mutation responsible for cancer progression. There are a numbers of existing techniques which are proven to be successful in finding candidate genes related to diseases. For instance, understanding disease-associated variations in human genome had been shown to be an important step toward enhancing our understanding of the cellular and molecular mechanisms that drive cancer and other complex diseases which had enomorous applications in modeling, clinical outcome and survival prediction analysis. Authors in [34] apply genome wide linkage analysis and association studies among the genomic samples of healthy individuals and patients. They successfully pinpoint several chromosomal regions containing hundreds of polymorphisms up to 400 genes that may potentially play a role in the manifestation and progression of disease. Sequencing technologies are then employed to analyse the candidate genes. However, the drawback of this method is that it was expensive and time consuming. On the other hand, a numbers of computational methods [44, 68, 16] are primarily used to prioritize and rank the most likely disease-associated genes by utilizing a variety of heterogeneous data sources such as gene expression, patient mutation profiles and functional annotation. Authors in [52] propose a random walk based framework that encodes various heterogeneous sources in a graphical model. Markov chain on the graph is constructed and random walk is performed which results in a ranking vector in terms of an eigenvector at their stationary distributions. A numbers of candidate genes could be found by inspecting the rank. It was widely believed that cancer is not a disease of individual mutation but a group of genes interacting together in a molecular network. In addition to that, network-based analyses of diverse phenotype demonstrate that genes which are related in similar diseases are clustered together into various highly connected sub-networks in protein protein interaction networks . They will undergo interactions in these sub-networks and participate in similar biological pathways. Motivated by these findings, our proposed studies utilize the protein protein interaction network for interacting partners of known cancer ground truth genes to find a set of candidate driver genes. Our proposed method takes as input a set of seed genes (known cancer ground truth genes), candidate cancer genes (coded for the disease of interest), and a network of interactions among human proteins. Subsequently, we use protein-protein interactions to infer the relationship between candidate cancer genes and the ground truth genes by calculating the diffusion distance of the candidate cancer genes and seed cancer genes (known cancer ground truth genes) respectively. Diffusion distance, a metric based on graph diffusion property, is designed to capture finer-grained distinctions in proximity for transfer of functional annotation in protein protein interaction networks. We postulate that cancer driver genes typically participate in similar biological pathways and are expected to exhibit substantial network cross-talk to each other in terms of the aggregate strength of paths which connect the corresponding proteins within the protein protein interaction networks. As a matter of fact, diffusion distance presents an excellent measure to determine the similarity between the seed cancer driver genes (known cancer ground truth genes) and the candidate genes. We encoded the diffusion distance metric as background information in our prior probabillity distribution for the cancer driver mutations. Our proposed framework introduces a hierarchical Bayesian methodology to estimate candidate cancer genes driver mutation rates and background mutation rates from somatic mutation data and protein protein interaction networks, which shed light on the overall proportion of cancer drivers among all the candidate genes. We choose a suitable prior distribution for modelling the driver mutations and the background mutations for each candidate gene. Diffusion distance as introduced above is incorporated as background information and is carefully encoded in the prior distribution for each candidate gene. We apply our method to ovarian cancer data and estimate proportion of drivers for this type of cancer. A set of candidate cancer driver genes is suggested by examining their probability of mutation.

2 Related Work

Most of the computational methods which focus on finding candidate genes could be found in [44, 68, 16, 86] They identify and discover the most likely disease-associated genes by utilizing various data sources such as gene expression [44, 68], SNP [62] and functional annotations [86]. For instance, Sean et al. in [62] utilizes SNPs to investigate on the relationship between the change of phenotype and the alteration of molecular function so as to identify candidate genes. In [27], Erten et al. developed a random walk based algorithm which postulates that genes which are associated with similar diseases exhibit patterns of topological similarity in PPI networks. They introduce the concept of topological profile to measure the similarity of genes and developed an algorithm which measures the topological similarity between the seed genes and candidate genes in ranking. Many of these candidate genes finding algorithms utilize protein-protein interaction network to prioritize genes. Generally speaking, these algorithms take a set of seed proteins (coded by genes known to be related to the disease of interest), candidate proteins (coded by genes in the linkage interval for the disease of interest), and an interaction network among human proteins as input and then they utilize PPI to determine the relationship between seed and candidate proteins and determine the importance of the candidate gene in accordance with the score which is used to calculate the relationship between the candidate gene and seed gene. On the other hand, there are a handful of disease gene prioritization tools like [33] available online. These tools aim at distinguishing disease-associated genes from false positives in genome-wide association studies. The feature is based on human micro-array data which reveals the association between gene expression and disease-associated variants. In this chapter, we focus on developing a statistical model to spot out cancer driver genes utilizing protein protein interaction network and patient mutation profile.

3 Bayesian Modeling

In this section, we introduce hierarchical Bayesian methodology to estimate candidate cancer genes driver mutation rates and background mutation rates. First of foremost, we introduce the sources of data which are needed in our framework.

Data Sources

1) *Gene Gene Interaction*: The gene gene interaction networks are encoded as an undirected graph $G(V, E)$ where V stands for the genes and edges $(i, j) \in E$ are weighted by a weight matrix W, whose element w_{ij} is the weight of the edge $(i, j) \in E$ which represents the strength of interaction between gene i and gene j using two sources of gene gene interaction

networks described below. Two sources of protein protein interaction networks are utilized: PathwayCommons and HumanNet v.1. PathwayCommons is a database of biological pathway information compiled from multiple sources related to PPI interactions and functional relationships between genes in signalling pathways. Only human genes and interactions in PathwayCommons are utilized in our framework. HumanNet is a probabilistic functional gene network constructed using naive Bayesian method to weigh different types of data evidence collected in humans, yeast and worms in accordance with their functionality in Homo-sapiens. A single interaction score is calculated as a result.

2) *Patient Mutation Profile*: Patient-Mutation Profile is a two dimensional binary matrix with columns representing the genes and rows representing patients. Each entry is either 0 or 1, a 1 indicates that a mutation has occured in the tumor relative to the germline on that patient, a 0 otherwise.

Network Diffusion

We apply network diffusion model to incorporate gene gene interaction network (PPI) on the patient mutation profile. By utilizing this method, the discrete "patient mutation" data was smoothed out and carries the information regarding the similarity of tumor sample at the pathway and network level rather than staying only at individual gene level. We first fused the patient mutation profile with the gene gene interaction network. Then network diffusion aims at diffusing the information of each mutated gene over this network for each patient according to the function as follow:

$$F_{t+1} = \alpha F_t A_0 + (1 - \alpha) G \tag{3.1}$$

G stands for the binary patient mutation profile data, and A_0 is a normalized adjacency matrix of the gene gene interaction network. α is used to adjust the extent that the mutation signal (tumor sample) can propagate in the network. The diffusion function run continuously until F_{t+1} converges. The resulting F_{t+1} encapsulates the influence of each mutation

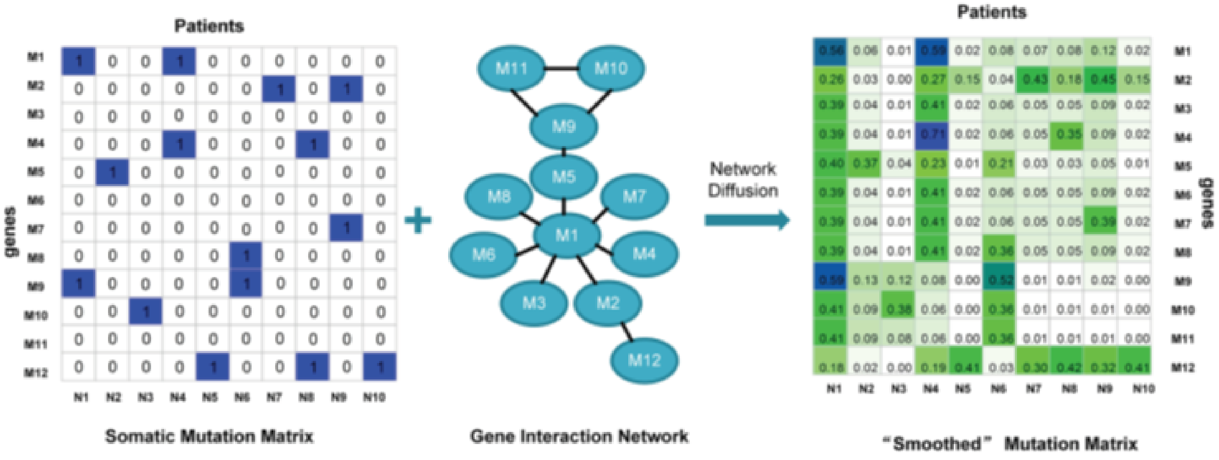

Figure 3.1. Smoothed PPI network

per patient through network diffusion. In this way, patient mutation profile can be analyzed at pathway level instead of merely at individual gene level, thus it will give a more comprehensive insight into similarity between mutated genes which can improve the identification of cancer drivers.

Diffusion Distance

There are different definition of diffusion distance that one can find in various literature. The one we adopted is as follow. Given an undirected graph $G(V, E)$, where V being used in our model is $F^T F$ which is the smoothed version of protein protein interaction network described above. We define that the probability of a random hop from a node $u \in V$ to another node $v \in V$, is proportional to some kernel function $k(u, v)$ which is a decreasing function of $l(u, v)$ where $l(u, v)$ is defined in terms of $w(u, v)$ as follow. $w(u, v)$ is the weight of edge (u, v) which can represent the interaction strength of gene u and gene v in the "smoothed PPI network".

$$\begin{aligned} l(u, v) &= M - w(u, v) \\ k(u, v) &= exp(-\frac{l(u, v)^2}{\alpha}). \end{aligned} \tag{3.2}$$

where M is a positive constant and α is a positive tuning parameter. A smaller $w(u, v)$ will result in a larger $l(u, v)$ and vice versa. Therefore, starting from node u, we choose to hop to node v with probability.

$$\phi(u, v) = \frac{k(u, v)}{\sum_z k(u, z)}. \tag{3.3}$$

where the sum in the denominator runs over all vertices z in V except u itself. We set $k(x, x) = 0$ for all x in V. The notation $\phi(u, v)$ stands for the transition probability of the undirected complete graph G, and then run random walks on this graph is performed based on these transition probabilities. Choosing a positive integer m, let $\phi_m(u, v)$ be the probability that a random walk of starting at node u will end at node v in exactly m steps. Next, we define the diffusion distance $S_m : V \times V \to R$:

$$S_m(x, y) = \sqrt{\sum_u [\phi_m(x, u) - \phi_m(y, u)]^2} \tag{3.4}$$

where the sum runs over all vertices $u \in V$. $S_m(x, y)$ stands for the difference of the probability distribution of random walk between vertex x and vertex y. Intuitively, diffusion distance is considered as an average length of all the paths connecting two vertices in the graph, and it is related to the probability of arriving from one vertex to another in a random walk with a fixed number of steps. A small m represents local random walk, where diffusion distances reflect local topological structure of a graph whereas a large m represents global random walk, where diffusion distances reflect large scale cluster or connected component. One advantage of diffusion distance is that it is robust to noise, since the distance between two points depends on all possible paths of length m between the points. Another advantage of diffusion distance is that nodes possessing large common low-degree neighborhoods are considered to be similar when m is large [15]. This allows nodes that interact through a hub node in some functional modules to be identified to be similar. Therefore, diffusion distance may be able to correctly identify functionally similar node pairs.

3.1 Bayesian Model

In this subsection, we introduce a hierarchical Bayesian estimation model to estimate gene-specific background mutation rates and driver probabilities from somatic mutation data and protein protein interaction network. Figure 1 shows the model. In this model, for each gene, we flip a coin at first to determine whether we will be at driver state (which is denoted as D) or at passenger state (which is denoted as P). Depending on whether you are at driver state or passenger state, we flip another coin to determine if that gene is mutated or not. Probability m_i stands for the driver mutation probability for gene i whereas n_i represents the background mutation probability for gene i. Both m_i and n_i are determined by the patient mutation profile and protein protein interaction network data within our model that will be derived in later sections. According to the model, the probability of having gene i mutated is justified as followed:

$$\begin{aligned} Pr(g_i = 1) &= pm_i + (1-p)n_i \\ Pr(g_i = 0) &= p(1-m_i) + (1-p)(1-n_i) \end{aligned} \tag{3.5}$$

3.1.1 Bayesian Parameter Estimation

We employed Bayesian Parameter Estimation to estimate p, m_i and n_i. The reason of using it is that we can incorporate prior knowledge of m_i and n_i using diffusion distance derived above. Hence Bayesian Parameter Estimation serves as an appropriate method to estimate m_i and n_i. Next, we lay down the definition of various parameters that are needed in our model as follow:

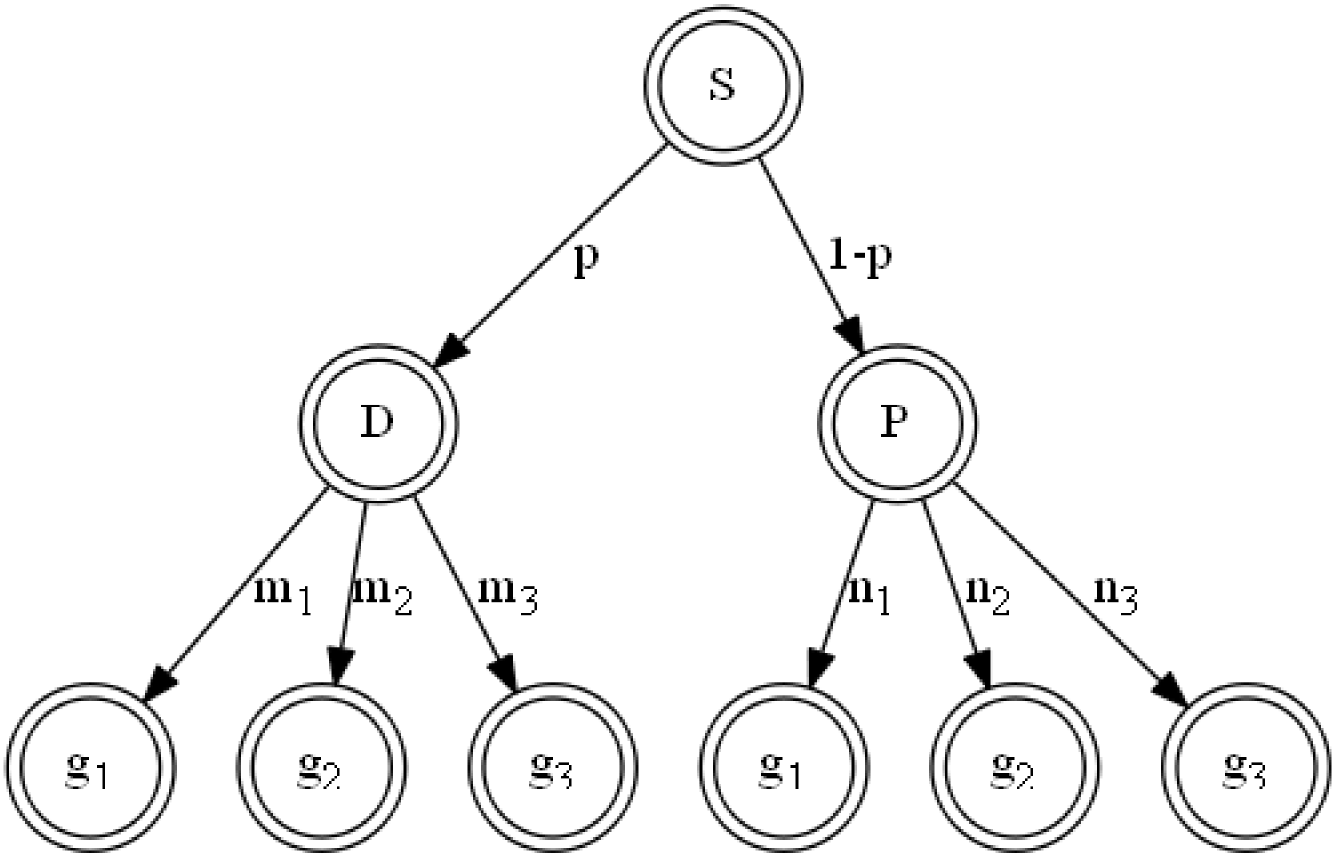

Figure 3.2. Bayesian Model Formulation

Model Parameters	
Parameter	Meaning
p	Probability of being in driver state
m_i	Driver mutation probability of gene i
n_i	Background mutation probability of gene i
G	Patient mutation profile
N	Total numbers of genes in G
M	Total numbers of patients in G

Table 3.1. Model Parameters Definition

$$\begin{aligned} & f(m_1, m_2, m_3..m_M, n_1, n_2, n_3..n_M, p|G) = \\ & \prod_{i=1}^{N}\prod_{j=1}^{M}[pm_i + (1-p)n_i]^{g_{ij}}[p(1-m_i) + (1-p)(1-n_i)]^{1-g_{ij}}\Theta_{m_i}\Theta_{n_i} \end{aligned} \tag{3.6}$$

where Θ_{m_i} and Θ_{n_i} are the prior distribution of m_i and n_i respectively.

Prior Distribution Of Driver Mutation Of Each Gene Θ_{m_i}

According to various literatures, genes of similar diseases tend to cluster together into various highly connected subnetworks in protein protein interaction networks. They interact and participate in similar biological pathways and exhibit substantial network crosstalk to each other in terms of the aggregate strength of paths in protein protein interaction network. As a matter of fact, we can use diffusion distance defined above to measure the similarity between a known cancer driver gene and a candidate gene in the "smoothed PPI network". The intuition is that if the difference between the candidate gene and a known cancer driver gene in terms of diffusion distance is small, the candidate gene is also likely a potential cancer driver gene. We model the prior distribution of m_i as Beta Distribution and incorporate the diffusion distance in it in a way that a smaller diffusion distance difference between a candidate gene and a known cancer driver gene should lead to a higher probability of mutation. Consider a gene set S=$(s_1, s_2, ...s_T)$ which encapsulates known cancer driver genes, we then define $A(i)$ for each candidate gene i (denotes as t_i) as follow:

$A(i)$=min $(S_m(t_i, s_1), S_m(t_i, s_2).....S_m(t_i, s_T))$

which firstly measures the diffusion distance difference between candidate gene i and all the known cancer driver genes in set S. We then take the minimum as $A(i)$. Afterwards, we relate $A(i)$ to the prior distribution of m_i as follow:

$$\begin{aligned} \Theta_{m_i} &= Beta(\alpha_i, \beta_i) \\ Beta(\alpha_i, \beta_i) &= \frac{m_i^{\alpha_i - 1}(1 - m_i)^{\beta_i - 1}}{B(\alpha_i, \beta_i)} \\ \alpha_i &= 1 \\ \beta_i &= A(i). \end{aligned} \tag{3.7}$$

We put $A(i)$ into β_i. The reason behind it is that a smaller diffusion distance difference between a candidate gene and a cancer driver gene should result in a larger probability value sampled from its corresponding Beta distribution in general.

Prior Distribution Of Background Mutation Of Each Gene Θ_{n_i}

We postulate that the background mutation rate of each gene is in proportion to its length. The longer the length of the gene, the higher its background mutation rate. We model the prior distribution Θ_{n_i} of the background mutation rate of each gene n_i using Beta Distribution as follow:

$$\begin{aligned} \Theta_{n_i} &= Beta(a_i, b_i) \\ Beta(a_i, b_i) &= \frac{n_i^{a_i - 1}(1 - n_i)^{b_i - 1}}{B(a_i, b_i)} \end{aligned} \tag{3.8}$$

It remains to derive both a_i and b_i. We derive them using the mean value of $Beta(a_i, b_i)$. We equate the mean value of it to the proportion of the length of the gene out of the total lengths of all genes.

$$\begin{aligned}\text{mean of } Beta(a_i, b_i) &= \frac{a_i}{a_i + b_i} \\ b_i &= 1 \\ \frac{a_i}{a_i + b_i} &= \frac{l_i}{\sum\limits_j l_j}\end{aligned} \tag{3.9}$$

Intuitively, it encodes the fact that a longer gene length would result in a higher probability value being sampled from its Beta Distribution with higher chance.

Combining the Priors and the Posterior Probabilities

After defining Θ_{m_i} and Θ_{n_i}, we can plug it into equation (5) as follow:

$$\begin{aligned}&f(m_1, m_2, m_3..m_M, n_1, n_2, n_3..n_M, p|G) \\ &= \prod_{i=1}^{N}\prod_{j=1}^{M}[pm_i + (1-p)n_i]^{g_{ij}}[p(1-m_i) + (1-p)(1-n_i)]^{1-g_{ij}}\Theta_{m_i}\Theta_{n_i} \\ &= \prod_{i=1}^{N}\prod_{j=1}^{M}[pm_i + (1-p)n_i]^{g_{ij}}[p(1-m_i) + (1-p)(1-n_i)]^{1-g_{ij}} * \\ &\frac{m_i^{\alpha_i - 1}(1-m_i)^{\beta_i - 1}}{B(\alpha_i, \beta_i)}\frac{n_i^{a_i - 1}(1-n_i)^{b_i - 1}}{B(a_i, b_i)}\end{aligned} \tag{3.10}$$

Taking logarithm:

$$\begin{aligned}l &= \log f(m_1, m_2, m_3..m_M, n_1, n_2, n_3..n_M, p|G) \\ &= \sum_{i=1}^{N}\sum_{j=1}^{M}[g_{ij}\log[pm_i + (1-p)n_i] \\ &+ (1 - g_{ij})\log[p(1-m_i) + (1-p)(1-n_i)] \\ &+ (\alpha_i - 1)\log m_i + (\beta_i - 1)\log(1-m_i) \\ &+ (a_i - 1)\log n_i + (b_i - 1)\log(1-n_i)\end{aligned} \tag{3.11}$$

Let $C_i = \sum_{j=1}^{M} g_{ij}$, which counts how many patients have gene i being mutated , we can further simplify (10):

$$\begin{aligned}
l &= \log f(m_1, m_2, m_3..m_M, n_1, n_2, n_3..n_M, p|G) \\
&= \sum_{i=1}^{N}[[C_i \log[pm_i + (1-p)n_i] \\
&+ (M - C_i)\log[p(1-m_i) + (1-p)(1-n_i)] \\
&+ M(\alpha_i - 1)\log m_i + M(\beta_i - 1)\log(1-m_i) \\
&+ M(a_i - 1)\log n_i + M(b_i - 1)\log(1-n_i)
\end{aligned} \tag{3.12}$$

Optimization

Differentiate with respect to p and set the result to 0.

$$\begin{aligned}
&\frac{\partial l}{\partial p} = 0 \\
&\sum_{i=1}^{N} \frac{(C_i - Mq_i)(m_i - n_i)}{q_i(1-q_i)} = 0
\end{aligned} \tag{3.13}$$

Differentiate with respect to m_i and set the result to 0.

$$\begin{aligned}
&\frac{\partial l}{\partial m_i} = 0 \\
&\frac{p(C_i - Mq_i)}{q_i(1-q_i)} + \frac{M[\alpha_i - 1 - [\alpha_i + \beta_i - 2]m_i]}{m_i[1-m_i]} = 0
\end{aligned} \tag{3.14}$$

Further solving (13):

$$
\begin{aligned}
&(p^2M(\alpha_i+\beta_i-1))m_i^3\\
&+(p(1-p)Mn_i - Mn_i(\alpha_i+\beta_i-2)(2p^2-2p)\\
&+(2Mp - pC_i - Mp\alpha_i - Mp\beta_i - Mp^2\alpha_i))m_i^2\\
&+(Mn_i(2p^2-2p)\alpha_i - Mn_i(2p^2-2p)\\
&-Mn_i(1-p)(\alpha_i+\beta_i-2)\\
&-p(1-p)M + (Mn_i^2(p-1)^2(\alpha_i+\beta_i-2))\\
&+(pM\alpha_i - pM + pC_i))m_i\\
&+((1-p)M(\alpha_i-1)n_i + (p-1)^2(M-M\alpha_i)n_i^2) = 0
\end{aligned}
\tag{3.15}
$$

Differentiate with respect to n_i and set the result to 0.

$$
\begin{aligned}
&\frac{\partial l}{\partial n_i} = 0\\
&\frac{(1-p)(C_i - Mq_i)}{q_i(1-q_i)} + \frac{M[a_i - 1 - [a_i + b_i - 2]n_i]}{m_i[1-m_i]} = 0
\end{aligned}
\tag{3.16}
$$

Further solving (15):

$$
\begin{aligned}
&((1-p)^2 M(a_i+b_i-1))n_i^3 \\
&+ (p(1-p)Mm_i - Mm_i(a_i+b_i-2)(2(1-p)^2-2(1-p)) \\
&+ (2M(1-p)-(1-p)C_i - M(1-p)a_i - M(1-p)b_i \\
&- M(1-p)^2 a_i))n_i^2 \\
&+ (Mm_i(2(1-p)^2-2(1-p))a_i \\
&- Mm_i(2(1-p)^2-2(1-p)) \\
&- Mm_i(p)(a_i+b_i-2) \\
&- p(1-p)M + (Mm_i^2(p-1)^2(a_i+b_i-2)) \\
&+ ((1-p)Ma_i - (1-p)M + (1-p)C_i))n_i \\
&+ ((1-p)M(a_i-1)m_i + (p-1)^2(M-Ma_i)m_i^2) = 0
\end{aligned}
\tag{3.17}
$$

Unfortunately, these equations are cubic and it is hard to solve these three simultaneous equations analytically. We resort to solve these three equations numerically.

Algorithm 2 Algorithm

```
    procedure ALGORITHM
    Sample $m_i$ from $\Theta_{m_i}$ for each $i$
    Sample $n_i$ from $\Theta_{n_i}$ for each $i$
    Repeat
    Solve $\frac{\partial l}{\partial p} = 0$ using $m_i$ and $n_i$ for all $i$
    For each $i$
    Using p and $n_i$ to solve $\frac{\partial l}{\partial m_i} = 0$ to get $m_i$
    Using p and $m_i$ to solve $\frac{\partial l}{\partial n_i} = 0$ to get $n_i$
    End For
    Until Convergence
    end procedure
```

Lastly, we have to calculate the following Hessian matrix after getting all the m_i, n_i upon convergence.

$$H_{m_i,n_i} = \begin{pmatrix} \frac{\partial^2 l}{\partial m_1^2} & \frac{\partial^2 l}{\partial m_1 m_2} & \cdots & \frac{\partial^2 l}{\partial m_1 n_N} \\ \frac{\partial^2 l}{\partial m_2 m_1} & \frac{\partial^2 l}{\partial m_2^2} & \cdots & \frac{\partial^2 l}{\partial m_2 n_N} \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 l}{\partial n_N m_1} & \frac{\partial^2 l}{\partial n_N m_2} & \cdots & \frac{\partial^2 l}{\partial n_N^2} \end{pmatrix}$$

4 Results and Findings

The data sets used for the experiment were taken from the study of Integrated Genomic Analyses of Ovarian Carcinoma led by the Cancer Genome Atlas. The associated results and discussions were published in [66]. The analysis of 489 clinically annotated stage III-V HGS-OvCa samples and its corresponding normal DNA were reported in the article and posted on its associated website. The data incorporates the age at diagnosis, stage, tumour grade and surgical outcome of patients diagnosed with HGS-OvCa. We downloaded the TCGA-OV-mutations data and the unified expression profiles from the TCGA Data Portal website for our purpose. In the aftermath of data cleaning, we retain mutations containing insertion, deletion and alternation of base only. Finally a patient mutation profile table which comprises 316 patients and 8404 genes is obtained. We utilized two different protein protein interaction networks for our experments. HumanNet is a probabilistic functional gene network which consists of 18,714 protein encoding genes and 476399 interactions between the genes of Homo sapiens. We obtained the required data through its web portal for download and query.

4.1 Known cancer driver genes

We compiled a set of genes published in various literature on several cancer studies which are certified to be ovarian cancer genes to be our known cancer driver genes in the evaluation of our proposed study. For the candidate cancer genes, we use the set of genes forming the patient mutation profile table as described above for experiment. Table 3.2 below

GENE	Literatures
BRCA1	[30],[10],[66]
BRCA2	[25]
BMPR1A	[82],[3]
BRIP1	[85]
MLH1	[78]
FHIT	[25],[6]
TFRC	[53]
FGFR2	[39],[60]
GATA3	[57]
MYST4	[88]
PTEN	[69]
FAS	[58],[6]
RB1	[85]
SEPT9	[79]
YWHAE	[32]
TP53	[66]
PIK3CA	[28]
BRAF	[28]
KRAS	[28]
AIB1	[4]
MSH2	[78]
BMP4	[56],[55]
TRIP1	[40]
MYC	[40]
EP300	[40]

Table 3.2. Ground Truth Genes

tabulates the collection of ovarian cancer genes (ground truth genes) and the associated references.

4.2 Experimental Results

Since for each gene, we have to solve three simultaneous equations which are cubic, we have many variables to estimate. As a matter of fact, it is hard to solve the equations analytically. We resort to solve the equations numerically. The idea behind the algorithm is that in each iteration, for each gene we take turns fixing two variables and solve the third variable. After multiple iterations, we hope the solution can converge. The experimental results below demonstrate that all the variables converge and fall in desirable range. Table 3.3 below shows the numbers of variables used in the experiment.

Experimental Data	
Varables Type	Numbers of variables of this type
p	1
m_i	8404
n_i	8404

Table 3.3. Experimental Data

Using mean to start the experiment

We use the mean of Θ_{n_i} and Θ_{m_i} to start the experiment and Table 3.4 below tabulates the result. The solution converges after 500 rounds and the optimal p we found is 0.0312. The largest m_i found is 0.6120 and the smallest m_i found is $4.5632e - 05$.

Experimental Data		
Numbers of Rounds	Variable	Average of difference of this variable compared to previous round
50	p	0.012
50	m_i	$2.3607e^{-02}$
50	n_i	$1.7943e^{-01}$
100	p	0.006
100	m_i	$3.6676e^{-03}$
100	n_i	$1.4591e^{-03}$
200	p	0.0002
200	m_i	$1.316e^{-04}$
200	n_i	$1.6501e^{-04}$
500	p	0.0001
500	m_i	$0.4128e^{-05}$
500	n_i	$0.6716e^{-05}$

Table 3.4. Experimental Results Using Mean

Using mode to start the experiment

We use the mode of Θ_{n_i} and Θ_{m_i} to start the experiment and Table 3.5 below tabulates the result. The solution converges after 500 rounds and the optimal p we found is 0.035. The

Experimental Data		
Numbers of Rounds	Variable	Average of difference of this variable compared to previous round
50	p	0.038
50	m_i	$2.5712e^{-02}$
50	n_i	$1.3654e^{-02}$
100	p	0.003
100	m_i	$2.3716e^{-04}$
100	n_i	$3.0861e^{-04}$
200	p	0.0002
200	m_i	$1.0871e^{-05}$
200	n_i	$2.4195e^{-05}$
500	p	0.0001
500	m_i	$0.4531e^{-06}$
500	n_i	$0.5971e^{-06}$

Table 3.5. Experimental Results Using Mode

largest m_i found is 0.6213 and the smallest m_i found is $2.1732e - 05$.

Using a value from prior to start the experiment

We sample a value out of Θ_{n_i} and Θ_{m_i} within one standard derivation randomly to start the experiment and Table 3.6 below tabulates the result. The solution converges after 500 rounds and the optimal p we found is 0.031. The largest m_i found is 0.5676 and the smallest m_i found is $3.5783e - 05$.

4.3 Findings

In this subsection, we obtain the probability vectors m_i in three experiments. We recorded the top 1 percentage (around 100) of each probability vector in each experimental setting. Table 3.7 tabulates a bunch of genes which appear in all top 1 percentage of the probability vectors that we believed to be cancer driver genes.

Biological Findings

Experimental Data		
Numbers of Rounds	Variable	Average of difference of this variable compared to previous round
50	p	0.031
50	m_i	$4.0182e^{-02}$
50	n_i	$3.6325e^{-02}$
100	p	0.002
100	m_i	$2.2665e^{-02}$
100	n_i	$1.3412e^{-02}$
200	p	0.0001
200	m_i	$1.1107e^{-04}$
200	n_i	$0.8703e^{-04}$
500	p	0.0001
500	m_i	$0.517e^{-05}$
500	n_i	$0.6316e^{-05}$

Table 3.6. Experimental Results Using Prior Distribution

In this subsection, we explain the potential roles of some of the 25 potential cancer driver genes found above from the molecular and biological points of view.

1.) AP2B1 is medium and highly expressed in tumor sample and expressed highly in protein level. A network analysis revealed a subnetwork with three genes BMP7, NR2F2 and AP2B1 that were consistently over expressed in the chemoresistant tissue or cells compared to the chemosensitive tissue or cells.

2.) It was found that the Wnt signaling coreceptor LRP6 is up-regulated in a subpopulation of human breast cancers which defines a class of breast cancer subtype and is a target for therapy.

3.) The knockdown of annexin A11 expression lead to reduced cell proliferation and colony formation ability of ovarian cancer cells. Furthermore, epigenetic silencing of annexin A11 conferred cisplatin resistance to ovarian cancer cells and hence it is believed that annexin A11 is associated with the tumor recurrence in ovarian cancer patients.

4.) GRM8 is among the 77 significantly mutated genes identified by statistical analysis to be driver genes .

Genes Commonly Found in The Above Three Experiments Within Top 1 Percentage Of Their Rank	
AP2B1	B2M
LP6	ANXA11
INSIG2	CCT3
GRM8	CRY1
DACH1	PPIC
NSUN2	PRKCZ
HIVEP3	MAPK8
DICER1	ATM
RAD51	BUB1
ADIPOQ	IGF1
THBS2	EFEMP1
MTHFD2	FBXW7

Table 3.7. Cancer Drivers Discovered Within Top 1 Percent of Their Rank

5.) SPARC is overexpressed in highly invasive subclone and ovarian cancer tissues and plays an important role in ovarian cancer growth, apoptosis and metastasis.

6.) The expression of BIRC6 in the cytoplasm is associated with epithelial ovarian cancer differentiation and is believed to be a novel predictor for poor prognosis of epithelial ovarian cancer (EOC) patients after curative resection. Univariate analyses and multivariate analyses revealed that BIRC6 was an independent significant predictor for overall survival and play an important role in oncogenesis.

7.) The HIVEP2 gene, located on 6q23-q24, belongs to a family of genes which encodes large zinc fingers containing transcription factor proteins. The overall median expression level in breast cancer was significantly lower than that in normal breast tissue (normalized median value of 4.49 versus 17.68; $p \leq 0.0001$). It was believed that the down-regulation of the HIVEP2 may be one of the genetic events responsible for breast cancer, and their transcription may be regulated by complex mechanisms involving interactions with other factors and/or by other genetic/epigenetic mechanisms.

8.) Network analysis indicates that MAPK8 is functionally connected to 3 altered genes: PIK3R1 PRKDC and TP53. TP53 is a well known cancer gene.

9.) There is interaction between gene NSUN2 and FBXW7. Both are found in our rank among top position. NSUN2 is experimentally found to be essential in cancer cells, and FBXW7, a tumour-suppressor gene, has been found to be mutated in cancer cells. In normal cells, NSUN2 and FBXW7 both function to regulate cellular differentiation via two different mechanisms. FBXW7 regulates cell differentiation by inhibiting c-Myc and proteins in Notch pathway, and NSUN2 functions to maintain normal cell differentiation when activated by LEF1/ β-catenin complex, which is part of Wnt pathway. It has been found that the loss of FBXW7 results in elevated expression of c-Myc, which results in an upregulation of NSUN2. As a matter of fact, NSUN2 stabilizes the mitotic spindle in fast cell proliferation in cancer cell growth.

10.) DACH1 protein levels increase with the invasiveness of the ovarian cancer. As the cancer progresses from benign and borderline to metastatic, DACH1 protein expression increases as well. Moreover, with the increase in expression, the subcellular distribution of DACH1 changes from nucleus in normal tissue to cytoplasm in cancer.

11.) It is suspected that DICER1 mutations in nonepithelial ovarian cancers may be oncogenic. The recurrent, focal nature of the DICER1 mutations and incomplete loss of DICER1 enzymatic activity observed in nonepithelial ovarian tumors suggest that, in certain cell types, aberrant miRNA processing may be oncogenic.

5 Conclusion

We developed a simple generative mixture model coupled with Bayesian parameter estimation to estimate background mutation rates and driver probabilities of each gene as well as the proportion of drivers among sequenced genes using somatic mutation data and protein protein interaction network data. We apply our model to ovarian cancer data and numerically estimated the solution. Upon convergence, we are able to discover and identify some new candidate cancer driver genes like SPARC, DACH1 etc.

CHAPTER 4

MIXED GRAPHICAL MODEL APPROACH TO IDENTIFY CANCER DRIVER GENES

The state of the art in bio-medical technologies has produced many genomic, epigenetic, transcriptomic, and proteomic data of varied types across different biological conditions. Historically, it has always been a challenge to produce new ways to integrate data of different types. Here, we leverage the node-conditional uni-variate exponential family distribution to capture the dependencies and interactions between different data types. The graph underlying our mixed graphical model contains both un-directed and directed edges. In addition, it is widely believed that incorporating data across different experimental conditions can lead us to a more holistic view of the biological system and help to unravel the regulatory mechanism behind complex diseases. We then integrate the data across related biological conditions through multiple graphical models. The performance of our approach is demonstrated through simulations and its application to cancer genomics.

1 Introduction

The Big Data era presents many challenges to modern data analysis. One of the inevitable consequences is the emergence of diverse and complex data sets comprising variables of different types in which the data variables are measured on the same set of samples and produce measurements of different types, such as binary, discrete, categorical, continuous, etc.

For example, one of the prominent Big Data applications is High-throughput biomedical research. The recent proliferation of genomics technologies has created large and well-characterized genomic data sets that are publicly available on many platforms such as The

Cancer Genome Atlas (TCGA), etc. Examples include mutations and aberrations such as copy number variations (CNV) and single nucleotide Polymorphisms (SNPs) which are typically binary or categorical, gene expression and miRNA expression data measured by micro-arrays and RNA-sequencing technologies that are count-valued, and epigenetics data measured by methylation arrays, which are continuous. Each of these data types, e.g. genomic, transcriptomic or proteomic data, provides a local and single layer view of the molecular system. However, all of these data are related since they belong to the same biological system under different aspects and scale. Therefore, it is beneficial to develop methods to integrate data of diverse types in order to obtain a global and holistic view of the biological system.

On one hand, multivariate distributions such as graphical models have been successfully applied to models with one type of data, typically gene expression for finding important bio-markers. However, as noted above, we have to decipher the relationships among different types of biological data in order to achieve a complete understanding of the molecular basis of disease. Therefore, it is of utmost importance to develop a class of mixed graphical models that can directly model dependencies among gene expression levels (counts), methylation data (continuous) as well as mutation (binary) data. We leverage exponential family distributions to model rich dependencies between variables of different data types. We allow the conditional distribution of each node to belong to the exponential family which could be Bernoulli, Poisson, exponential etc.

In this paper, we focus on the problem of learning regulatory relationships among heterogeneous genomic variables from various biological conditions with overlapping regulatory mechanisms. Genomic variables can be genomic variants (for instance, mutations and copy number alterations), epigenetic states (for instance, methylation status), and gene expression profiles (for instance, miRNA expression). Biological conditions can belong to different tissues or cancer types, etc. Different diseases have both shared regulatory mechanisms and disease specific regulations. Therefore, we utilize the mixed graphical model of exponential family distributions mentioned above to jointly learn conditional dependencies among a set of binary, count and continuous variables across a set of distinct but related conditions.

To summarize, we introduce our model and various background knowledge in section 2. In sections 3 and 4 we formulate the problem of inferencing multiple mixed graphical models into an optimization problem. We then propose an algorithm for parameter estimation in section 5. We then illustrate our method through simulations and cancer genomic data in section 6. We then conclude our paper in section 7.

2 Related Work

There has been an overwhelming amount of research effort in estimating sparse undirected graphical models in high-dimensional settings. Most of the work focuses on graphical models where the nodes represent either continuous or discrete variables (single type), but not both. For example, much attention has focused on estimating Gaussian graphical models among a set of random variables with a joint multivariate normal distribution, where zero entries in the precision matrix correspond to conditional independence. Meinshausen and Buhlmann [59] further estimate the precision matrix using a marginal penalized regression approach. Yuan and Lin [97], Friedman, Hastie and Tibshirani [31] and others proposed a penalized log-likelihood approach to estimate the precision matrix. Danaher, Wang and Witten [22] extend the approach to infer multiple Gaussian graphical models based on data collected from distinct but related conditions. In addition, Chun et al [19] proposed joint conditional Gaussian graphical models using multiple sources of genomic data. For discrete variables, the Ising model has been widely used to model conditional independence among variables. Hofling and Tibshirani [38] presented a pseudo-likelihood approach to estimate the sparse binary pairwise Markov networks under high-dimensional setting. In the case of both discrete and continuous variables, Lauritzen's [45] earlier seminal work focused on a mixed graphical model in the low-dimensional setting. Lee and Hastie [46] extend Lauritzen's [45] work and proposed a pairwise graphical model over continuous and discrete variables using a group lasso penalty in a high-dimensional setting. Our work is inspired by Lauritzen's work [45] to extend to exponential family distribution to allow for modelling dependencies among differing data types. On the other hand, a related line of research considers a non-parametric approach to estimate conditional dependence relationships between variables. Probabilistic graphical models using copulas [26, 50] or rank-based estimators [95, 49] are proposed for mixed data. However, they are less efficient when compared to parametric families, especially under high dimensional regimes.

3 Model

3.1 Pairwise graphical model

We consider the pairwise graphical model, in the following form:

$$p(x) \propto \Big\{\sum_{s=1}^{p} f_s(x_s) + \sum_{s=2}^{p}\sum_{t\leq s} f_{ts}(x_s, x_t)\Big\}, \tag{4.1}$$

where $x = (x_1, x_2, ..., x_p)^T$ and $f_{ts} = 0$ for $\{t, s\} \notin E$. Here, $f_s(x_s)$ denotes the node potential function, and $f_{ts}(x_s, x_t)$ denotes the edge potential function. We then simplify the pairwise interaction term by assuming that $f_{ts}(x_s, x_t) = \theta_{st}x_sx_t = \theta_{ts}x_sx_t$ so that the parameters associated with edges form a symmetric square matrix $\Theta = (\theta_{st})_{p*p}$ with the diagonal elements being zero. The joint density can then be written as

$$p(x) \propto \Big\{\sum_{s=1}^{p} f_s(x_s) + \frac{1}{2}\sum_{s=1}^{p}\sum_{t\neq s} f_{ts}(x_s, x_t) - A(\Theta)\Big\}, \tag{4.2}$$

where $A(\Theta)$ denotes the log-partition function, a function of θ. For $\{s, t\} \notin E$, the edge potentials satisfy $\theta_{st}x_sx_t = \theta_{ts}x_sx_t = 0$. We define the neighbours of the s th node as $N(x_s) = \{x_t : \theta_{st} = \theta_{ts} \neq 0\}$.

3.2 Mixed graphical model

We now consider modeling of the conditional distribution of a random vector $Y := (Y_1, ..., Y_p) \in \mathcal{Y}_1 * ... * \mathcal{Y}_p$, conditioned on a random vector $X := (X_1, ..., X_q) \in \mathcal{X}_1 * ... * \mathcal{X}_q$. Suppose that we have a graph $G_Y = (V_Y, E_Y)$, with nodes V_Y associated with variables in Y. Denote the set of neighbors in V_Y for any node $s \in V_Y$ by $N_Y(s)$. Suppose further that we also have a set of nodes V_X associated with the variables in X, and that for any node $s \in V_Y$, we denote its set of neighbors in V_X as $N_X(s)$. Suppose that the variables Y are locally Markov with respect to their specified neighbors, so that

$$P[Y_s|Y_{V_{Y-s}}, X] = P[Y_s|Y_{N_Y(s)}, X_{N_X(s)}]. \tag{4.3}$$

Moreover, suppose that the conditional distribution Y_s conditioned on the rest of $Y_{V_{Y-s}}$ and X is given by the following uni-variate exponential family:

$$\begin{aligned} P(Y_s|Y_{V_{Y-s}}, X) = \exp\Big\{ & f_s(y_s) + \sum_{x_t \in N_X(y_s)} \theta_{ts} x_t y_s \\ & + \sum_{y_t \in N_Y(y_s)} \theta^{'}_{ts} y_t y_s - A_{Y_s|Y_{V_{Y-s}},X}(Y_{V_{Y-s}}, X) \Big\} \end{aligned} \tag{4.4}$$

Suppose $f_s(y_s) = \alpha_{1s} y_s + \alpha_{2s} y_s^2 + \sum_{k=3}^{K} \alpha_{ks} B_{ks}(y_s)$, where α_{ks} is a parameter, which could be 0, and $B_{ks}(y_s)$ is a known function for $k = 3, ..., K$. Under this assumption, (4.4) belongs to the exponential family. The assumed form of $f_s(y_s)$ is quite general. We now consider some special cases of (4.4) corresponding to commonly-used distributions in the exponential family, for which $f_s(y_s)$ takes a very simple form.

For the case of Gaussian distribution with domain R,

$$P(Y_s|Y_{V_{Y-s}}, X) \propto \exp\Big\{\alpha_{1s} y_s - \frac{1}{2} y_s^2 + \sum_{s \neq t} \theta^{'}_{st} y_s y_t + \sum_{s \neq t} \theta_{st} y_s x_t \Big\}. \tag{4.5}$$

For the case of Poisson distribution with domain $\{1, 2, 3...\}$,

$$P(Y_s|Y_{V_{Y-s}}, X) \propto \exp\Big\{\alpha_{1s}y_s - \log(y_s!) + \sum_{s\neq t}\theta'_{st}y_sy_t + \sum_{s\neq t}\theta_{st}y_sx_t\Big\}. \tag{4.6}$$

Theorem 1. *Consider a p-dimensional random vector* $Y := (Y_1, ..., Y_p)$ *and a q-dimensional random vector* $X := (X_1, ..., X_q)$. *Then, the node-wise conditional distributions satisfying the Markov condition in (4.3) as well as the exponential family condition in (4.4), are indeed consistent with a graphical model joint distribution and has the form:*

$$\begin{aligned} P(Y|X) \propto \exp\Big\{&\sum_{s=1}^{p} f_s(y_s) + \sum_{s=1}^{p}\sum_{x_t\in N_X(y_s)}\theta_{ts}x_ty_s \\ &+ \sum_{s=1}^{p}\sum_{\substack{t\neq s \\ y_t\in N_Y(y_s)}}\frac{\theta'_{ts}}{2}y_ty_s\Big\} \end{aligned} \tag{4.7}$$

Proof. We now prove that any function that is capable of generating the conditional density in (4.4) is in the form (4.7). The following proof is similar to that in Besag [8].

Define $Q(Y|X) = \log(P(Y|X)/P(0|X))$

We then write $Q(Y|X)$ as

$$\begin{aligned} Q(Y|X) = &\sum_{s=1}^{p} y_sG_s(y_s, X) + \sum_{t\neq s}\frac{G_{ts}(y_t, y_s, X)}{2}y_ty_s \\ &+ \sum_{\substack{t\neq s \\ t\neq j, s\neq j}}\frac{G_{tsj}(y_t, y_s, y_j, X)}{6}y_ty_sy_j + ... \end{aligned} \tag{4.8}$$

where we write the function $Q(Y|X)$ as the sum of interactions of different orders. Note that the factor of $\frac{1}{2}$ is due to $G_{st}(y_s, y_t) = G_{ts}(y_s, y_t)$ similar factors apply for higher-order interactions.

$$\begin{aligned} Q(Y|X) - Q(Y_s^0|X) &= \log(\frac{P(Y|X)}{P(Y_s^0|X)}) \\ &= \log(\frac{P(Y_s|Y_{V_{Y-s}}, X)}{P(0|Y_{V_{Y-s}}, X)}) \end{aligned} \tag{4.9}$$

where $Y_s^0 = (Y_1, Y_2, ..Y_{s-1}, 0, Y_{s+1}, ..Y_p)$. It follows that

$$\begin{aligned} \log(\frac{P(Y_s|Y_{V_{Y-s}}, X)}{P(0|Y_{V_{Y-s}}, X)}) &= Q(Y|X) - Q(Y_s^0|X) \\ &= y_s(G_s(y_s, X) + \sum_{t \neq s} \frac{G_{ts}(y_t, y_s, X)}{2} y_t + ...) \end{aligned} \tag{4.10}$$

Set $y_i = 0$ if $i \neq s$, then

$$f_s(y_s) + \sum_{x_t \in N_X(y_s)} \theta_{ts} x_t y_s = y_s G_s(y_s, X). \tag{4.11}$$

Set $y_i = 0$ if $i \neq s$ and $i \neq t$, then

$$f_s(y_s) + \sum_{x_t \in N_X(y_s)} \theta_{ts} x_t y_s + \theta'_{ts} y_t y_s = y_s G_s(y_s, X) + y_s y_t \frac{G_{ts}(y_t, y_s, X)}{2}. \tag{4.12}$$

Combining (4.11) and (4.12),

$$\theta'_{ts} y_s y_t = y_s y_t \frac{G_{ts}(y_t, y_s, X)}{2}. \tag{4.13}$$

Replace s by t and vice versa, we have

$$\theta'_{st} y_s y_t = y_s y_t \frac{G_{st}(y_t, y_s, X)}{2}. \tag{4.14}$$

Therefore, if $\theta'_{st} = \theta'_{ts}$, then $G_{st}(y_t, y_s, X) = G_{ts}(y_t, y_s, X)$.

For higher interaction term, set $y_i = 0$ if $i \neq s$ and $i \neq t$ and $i \neq j$ then

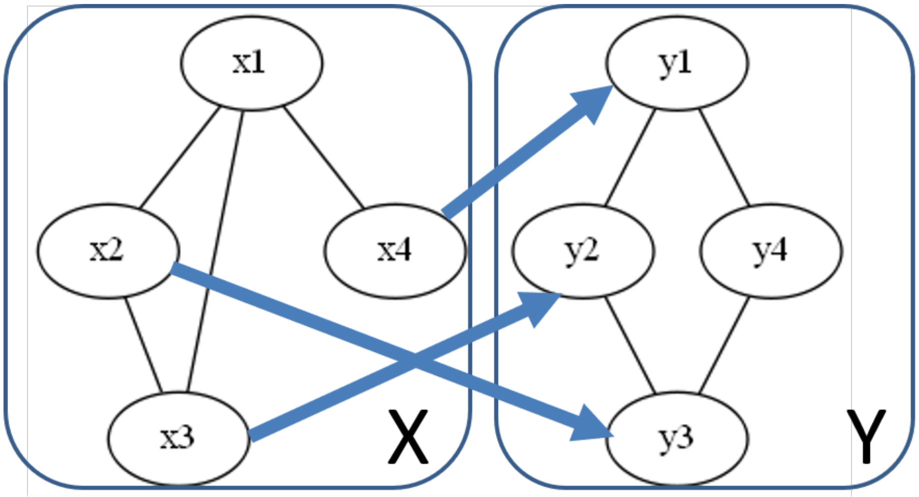

Figure 4.1. 2 Blocks MRF

$$\begin{aligned}
&f_s(y_s) + \sum_{x_t \in N_X(y_s)} \theta_{ts} x_t y_s + \theta'_{ts} y_t y_s + \theta'_{js} y_j y_s \\
&= y_s G_s(y_s, X) + y_s y_t \frac{G_{ts}(y_t, y_s, X)}{2} + y_j y_s \frac{G_{js}(y_j, y_s, X)}{2} \\
&+ y_j y_s y_t \frac{G_{tsj}(y_t, y_s, y_j, X)}{6}.
\end{aligned} \tag{4.15}$$

$G_{tsj}(y_t, y_s, y_j, X) = 0$. Similarly, we can show that fourth-and-higher-order interactions are zero. Hence, we arrive at the following formula for $Q(Y|X)$:

$$Q(Y|X) = \sum_{s=1}^{p} f_s(y_s) + \sum_{s=1}^{p} \sum_{x_t \in N_X(y_s)} \theta_{ts} x_t y_s + \sum_{s=1}^{p} \sum_{\substack{t \neq s \\ y_t \in N_Y(y_s)}} \frac{\theta'_{ts}}{2} y_t y_s. \tag{4.16}$$

Furthermore, $Q(Y|X) = log(P(Y|X)/P(0|X))$, so the function P takes the form

$$\begin{aligned}
&P(Y|X) \propto \exp Q(Y|X) \\
&= \exp\Big\{\sum_{s=1}^{p} f_s(y_s) + \sum_{s=1}^{p} \sum_{x_t \in N_X(y_s)} \theta_{ts} x_t y_s + \sum_{s=1}^{p} \sum_{\substack{t \neq s \\ y_t \in N_Y(y_s)}} \frac{\theta'_{ts}}{2} y_t y_s\Big\}.
\end{aligned} \tag{4.17}$$

X could be "cause" variables, while Y could be "effect" variables. Suppose that we have an undirected graph $G_Y = (V_Y, E_Y)$, with nodes V_Y associated with variables in Y, an undirected graph $G_X = (V_X, E_X)$, with nodes V_X associated with variables in X. Suppose in addition, we have directed edges E_{XY} from nodes in V_X to V_Y. Thus, the overall graph structure (Figure 4.1) has both undirected edges E_X and E_Y among nodes solely in X and Y respectively, as well as directed edges E_{XY}, from nodes in X to Y. For any node $s \in V_Y$, we denote its set of neighbors in G_Y by $N_Y(s)$, and likewise we denote its neighbors in V_X, by $N_{YX}(s)$. Then,

$$P(X,Y) = P(Y|X)P(X) \tag{4.18}$$

□

where $P(Y|X)$ and $P(X)$ are as follow:

$$\begin{aligned} P(Y|X) = \exp\Big\{ & \sum_{s=1}^{p} f_s(y_s) + \sum_{s=1}^{p} \sum_{x_t \in N_X(y_s)} \theta_{ts} x_t y_s \\ & + \sum_{s=1}^{p} \sum_{\substack{t \neq s \\ y_t \in N_Y(y_s)}} \frac{\theta'_{ts}}{2} y_t y_s - A_{Y|X}(\theta(X)) \Big\} \end{aligned} \tag{4.19}$$

$$P(X) = \exp\Big\{ \sum_{s=1}^{q} f_s(x_s) + \frac{1}{2} \sum_{s=1}^{q} \sum_{x_t \in N_X(x_s)} \overline{\theta_{st}} x_s x_t - A(\Theta) \Big\} \tag{4.20}$$

Biological Motivation: We can use the above design to model the interactions between binary mutation variables (SNPs) and continuous gene expression variables (microarrays) as SNPs are fixed point mutations that influence the dynamic and tissue specific gene expression. Thus, we can take X to be Bernoulli nodes representing SNP and Y to be Gaussian nodes representing gene expression and form directed edges from X to Y.

3.3 General Mixed Graphical Model

In this section, we extend our mixed graphical models to handle a chain graph structure. A DAG $G = (V, E)$ consists of a set of vertices V and a set of edges E with no directed cycle. We use $X := (X_1, .., X_q)$ to denote the set of random variables representing the vertices, V, of our network model. Suppose further that V can be partitioned into a series of disjoint exhaustive sets $V_1, ..., V_m$, such that $V_i \cap V_j = 0 \ \forall i \neq j$ and $\cup_{j=1}^m V_j = V$. These exhaustive sets V_i are connected subgraphs consisting only of undirected edges. All edges between nodes in the same set are undirected, and all edges between different sets are directed. In addition, the sets can be ordered in such a way that all arrows point from a set with a smaller number to one with a larger number. As a matter of fact, for any undirected edge $(s, t) \in E$, we have that $s, t \in V_i$, for only one $i \in [m]$ and for any directed edge $(s, t) \in E$, we have that $s \in V_i$, $t \in V_j$, with $i < j$. We then define the general class of mixed graphical models associated with the above definition. For any $i \in [m]$, indexing the m subsets $\{V_i\}_{i=1}^m$, we define the set of parents of set V_i as follow:

$$Pa(i) = \cup_{j=1}^m \{V_j \ \exists \text{directed}(s, t) \in E, s \in V_j, t \in V_i\}. \tag{4.21}$$

We also use the notation $Pa(t)$ to denote the set of parent nodes of any node $t \in V$:

$$Pa(t) = \{s \ \exists \text{directed}(s, t) \in E\}. \tag{4.22}$$

We then factorize the joint distribution in terms of the conditional distributions as follows

$$P(X) = \prod_{i=1}^m P(X_{V_i} | X_{Pa(i)}), \tag{4.23}$$

where $P(X_{V_i} | X_{Pa(i)})$ is specified by the mixed CRF detailed in (4.7).

$$\begin{aligned} P(X_{V_i} | X_{Pa(i)}) = \exp\Big\{ &\sum_{x_s \in X_{V_i}} f_s(x_s) + \sum_{x_s \in X_{V_i}} \sum_{t \in Pa(s)} \theta_{ts} x_t x_s \\ &+ \sum_{x_s \in X_{V_i}} \sum_{\substack{t \neq s \\ t \in N_{V_i}(s)}} \frac{\theta'_{ts}}{2} x_t x_s - A_i(\theta(X_{Pa(i)})) \Big\}. \end{aligned} \tag{4.24}$$

Then the overall joint distribution (4.23) is as follow:

$$P(X) = \exp \sum_{i=1}^{m} \Big\{ \sum_{x_s \in X_{V_i}} f_s(x_s) + \sum_{x_s \in X_{V_i}} \sum_{t \in Pa(s)} \theta_{ts} x_t x_s \\ + \sum_{x_s \in X_{V_i}} \sum_{\substack{t \neq s \\ t \in N_{V_i}(s)}} \frac{\theta'_{ts}}{2} x_t x_s - A_i(\theta(X_{Pa(i)})) \Big\}. \tag{4.25}$$

4 Extension

4.1 Unknown Blocks Ordering

Recovering the graph when its ordering of the blocks is unknown is a NP hard problem, when the number of parents of each block is restricted to 2. This is because of the exponential search space that one has to traverse to obtain the optimal network. Although this might seem computationally infeasible, using local search procedures we are able to provide a solution using heuristic hill-climbing [18]. The greedy hill-climbing approach starts with a prior network. The prior network could be a random DAG structure; or a DAG structure elicited by an expert. From this prior network we iteratively try to improve the structure's score defined in (4.26) by utilizing search operators. We always apply a change that improves the score until no improvement can be made.

$$\log P(X) - \frac{\log M}{2} DIM[G]. \tag{4.26}$$

where M is the number of training instances and $DIM[G]$ is the number of independent parameters in the network. The first term is a likelihood score and the second is a penalty term for complex networks. The BIC score has the following properties: (a) As we increase the number of samples, the emphasis moves from model complexity to fit to data. In other words, as we obtain more data we are more likely to consider more complicated structures.

5 Learning

In this section, we perform learning on our model (4.25). Specifically, we observe n i.i.d. samples $\{X^{(j)}\}_{j=1}^{n}$. We are interested in (1) parameter learning which is to estimate the unknown parameters θ, and (2) structure learning, which is to estimate the unknown edge-set E of the underlying mixed graph. These tasks are also referred to as graphical model estimation and selection respectively. Before we dig into the learning, we observe that our class of mixed graphical model distribution in (4.25) is specified by mixed CRF distributions (4.23) over the blocks. Therefore, we reduce the problem of estimating the overall mixed graphical model to that of estimating the corresponding mixed CRFs. In order to learn any of the mixed CRF $P(X_{V_i}|X_{Pa(i)})$, only the sample sub-vectors restricted to X_{V_i} and $X_{Pa(i)}$ are required. Therefore, the overall graphical model estimation problem can be reduced to the set of sub-problems of estimating the mixed CRFs:

$$\begin{aligned} P(X_{V_i}|X_{Pa(i)}) = \exp\Big\{ &\sum_{x_s \in X_{V_i}} f_s(x_s) + \sum_{x_s \in X_{V_i}} \sum_{t \in Pa(s)} \theta_{ts} x_t x_s \\ &+ \sum_{x_s \in X_{V_i}} \sum_{\substack{t \neq s \\ t \in N_{V_i}(s)}} \frac{\theta'_{ts}}{2} x_t x_s - A_i(\theta(X_{Pa(i)}))\Big\}. \end{aligned} \tag{4.27}$$

As the graph factors according to mixed CRFs, we therefore estimate each CRF independently. We then perform the node-wise neighborhood estimation. Neighborhood estimation approaches seek to learn the sparse network structure through an $l_1 - norm$ regularization to estimate the set of edge parameters, the non-zeros of which correspond to the selected node-neighbors. Estimating the CRF (4.27) then reduces to estimating the univariate node-conditional distribution of variable X_s for each $s \in V_i$ given all other nodes in V_i and $Pa(i)$. We have

$$
\begin{aligned}
P(X_s|X_{V_i-s}, X_{Pa(i)}, \theta_s) = \exp\Big\{ & C(X_s|X_{V_i-s}, X_{Pa(i)}, \theta_s) \\
& - D(X_{V_i-s}, X_{Pa(i)}, \theta_s)\Big\},
\end{aligned}
\tag{4.28}
$$

where

$$
\begin{aligned}
C(X_s|X_{V_i-s}, X_{Pa(i)}, \theta_s) = f_s(x_s) + & \sum_{t \in Pa(s)} \theta_{ts} x_t x_s \\
& + \sum_{\substack{t \neq s \\ t \in N_{V_i}(s)}} \frac{\theta'_{ts}}{2} x_t x_s,
\end{aligned}
\tag{4.29}
$$

and

$$
D(X_{V_i-s}, X_{Pa(i)}, \theta_s) = \log \int\limits_X \exp\Big\{C(X_s|X_{V_i-s}, X_{Pa(i)}, \theta_s)\Big\} dX_s. \tag{4.30}
$$

Furthermore, since the log partition function (4.30) is complicated, it is natural to replace the log-partition terms by a Monte-Carlo approximation through importance sampling and then minimize the resulting approximated formulation. Assume we have k ii.d samples $Y_k = \{Y^{(j)}\}_{j=1}^k$ drawn from a random vector $Y \in X^q$ with known probability density $P(Y)$ and therefore given θ, we can use importance sampling to approximate (4.30) as
$\log\{\frac{1}{k}\sum_{j=1}^k \frac{\exp\Big\{C(Y_s^{(j)}|Y_{V_i-s}^{(j)}, Y_{Pa(i)}^{(j)}, \theta_s)\Big\}}{P(Y_s^{(j)}|Y_{V_i-s}^{(j)}, Y_{Pa(i)}^{(j)}, \theta_s)}\}$.

Remark Note that there exist restrictions on θ_s [17] required for
$D(X_{V_i-s}, X_{Pa(i)}, \theta_s) \leq \infty$ in order for the conditional densities (4.28) to exist.

Therefore, for each node X_s within V_i, its node-conditional distribution is specified by three sets of parameters, namely θ_s, which is its nodewise weight, $\theta_{V_i} := \{\theta_{st}\}_{t \in V_i}$ which is the vector of intra-block edge-weights (same set) of node s and also $\theta_{Pa(i)} := \{\theta_{ts}\}_{t \in Pa(i)}$ which represents the vector of inter-block edge weights (different sets) of node s. Then, given n i.i.d. samples from our mixed graphical model of (4.28) with unknown parameters, we calculate the negative log likelihood of the node-conditional distribution as:

$$\begin{aligned}\min_{\theta_s} \frac{1}{n} \sum_{j=1}^{n} \Big\{ & -C(X_s^{(j)} | X_{V_i-s}^{(j)}, X_{Pa(i)}^{(j)}, \theta_s) \\ & + D(X_{V_i-s}^{(j)}, X_{Pa(i)}^{(j)}, \theta_s) \Big\} + \lambda \|\theta_{V_i}\|_1 + \lambda \|\theta_{Pa(i)}\|_1,\end{aligned} \tag{4.31}$$

where λ determines the degree of sparsity in the connections between X_s and X_{V_i-s} and also the degree of sparsity in the connections between X_s and nodes in $Pa(i)$. We further assume that the observation are from different classes, indexed by k varying from 1 to K, are independent. Different classes can arise from different tissue types or different cancer types, etc, which contain both shared regulations and disease specific regulations. Given the observed data $\{X^{(j)(k)}\}_{j=1}^{n_k}$ for class k with n_k samples, we then calculate the pseudo-likelihood, which is formed by the product of all node-conditional distributions as below:

$$\begin{aligned}l(\Theta^k, X^{(k)}) = \sum_{j=1}^{n_k} \sum_{s \in V} \Big\{ & -C(X_s^{(j)(k)} | X_{T(s)-s}^{(j)(k)}, X_{Pa(T(s))}^{(j)(k)}, \theta_s^{(k)}) \\ & + D(X_{T(s)-s}^{(j)(k)}, X_{Pa(T(s))}^{(j)(k)}, \theta_s^{(k)}) \Big\}.\end{aligned} \tag{4.32}$$

We use $\Theta^k = \{\theta^k\}$ to denote the set of parameters of class k. We also use $T(s)$ to denote the block where the node s belongs. The above derivation treats different biological conditions differently from biological measurements such as CNVs, and estimates multiple biological networks from different biological conditions jointly.

After we derive the optimization for a single class, we then proceed to formulate our problem for joint analysis of multiple classes. The idea behind our joint mixed graphical model analysis across different biological conditions is that there exist some commonalities shared among multiple classes, such as shared regulatory mechanisms. We therefore propose two penalization approaches (fused lasso and group lasso) to facilitate borrowing information from multiple biological conditions for the estimation of the joint mixed graphical models.

$$\arg \min_{\Theta^1,...,\Theta^K} \sum_{k=1}^{K} l(\Theta^k, X^{(k)}) + P(\Theta^1, ..., \Theta^K). \tag{4.33}$$

In the case of the fused graphical lasso:

$$\begin{aligned} P(\Theta^1, ..., \Theta^K) = {} & \lambda_1 \sum_{k=1}^{K} \sum_{i=1}^{m} \|\theta_{V_i}^{(k)}\|_1 + \lambda_1 \sum_{k=1}^{K} \sum_{i=1}^{m} \|\theta_{Pa(i)}^{(k)}\|_1 \\ & + \lambda_2 \sum_{k<k'} \sum_{i=1}^{m} \|\theta_{V_i}^{(k)} - \theta_{V_i}^{(k')}\|_1 \\ & + \lambda_2 \sum_{k<k'} \sum_{i=1}^{m} \|\theta_{Pa(i)}^{(k)} - \theta_{Pa(i)}^{(k')}\|_1, \end{aligned} \tag{4.34}$$

where λ_1 and λ_2 are tuning parameters. The fused graphical lasso penalty forces the graphs from multiple biological conditions are the same except for a few edges.

In the case of the group graphical lasso:

$$\begin{aligned} P(\Theta^1, ..., \Theta^K) = {} & \lambda_1 \sum_{k=1}^{K} \sum_{i=1}^{m} \|\theta_{V_i}^{(k)}\|_1 + \lambda_1 \sum_{k=1}^{K} \sum_{i=1}^{m} \|\theta_{Pa(i)}^{(k)}\|_1 \\ & + \lambda_2 \sum_{i=1}^{m} \sum_{(u,v)\in V_i} \sqrt{\sum_{k=1}^{K} (\theta_{V_i}^{(k)})_{uv}^2} \\ & + \lambda_2 \sum_{i=1}^{m} \sum_{(u,v)\in Pa(i)} \sqrt{\sum_{k=1}^{K} (\theta_{Pa(i)}^{(k)})_{uv}^2}. \end{aligned} \tag{4.35}$$

The group graphical lasso penalty here puts the related biological conditions as one group which implies that the underlying multiple graphs are the same.

Parameters Tuning

For the selection of tuning parameters, we utilize the following Bayesian information criterion (BIC) type of approach to choose λ_1 and λ_2 which minimizes:

$$\text{BIC}(\lambda_1, \lambda_2) = -2\sum_{k=1}^{K} \text{In}l(\Theta^k_{\lambda_1,\lambda_2}, X^{(k)}) + E_k\text{In}n_k \tag{4.36}$$

where $l(\Theta^k_{\lambda_1,\lambda_2}, X^{(k)})$ is the pseudo-likelihood (4.32) for the observation from the kth class with the tuning parameters λ_1 and λ_2, and E_k is the number of edges in the kth mixed graphical model.

5.1 Algorithm

In this section, we introduce an numerical algorithm which solves the optimization problem above. This constrained optimization problem (4.33) can be simplified and solved by replacing it with a series of distributed problems through an augmented Lagrangian scheme. We first make the objective function separable by rewriting

$$\arg \min_{\{Z^{(k)}\}\{\Theta^{(k)}\}} \sum_{k=1}^{K} l(\Theta^k, X^{(k)}) + P(Z). \tag{4.37}$$

subject to the constraint that $Z^{(k)} = \Theta^{(k)}$ for $k = 1,,,,K$ where $\{Z\} = \{Z^{(1)}, ..., Z^{(K)}\}$. Then, we can perform the optimization and regularization locally and coordinate them globally via constraints by further rewriting the problem using the scaled augmented Lagrangian, [9, 37] that is,

$$L(\{\Theta\}, \{Z\}, \{U\}) = \sum_{k=1}^{K} l(\Theta^k, X^{(k)}) + P(Z) + \frac{d}{2}\sum_{k=1}^{K} \|\Theta^k + U^k - Z^k\|_F^2, \tag{4.38}$$

where $\{U\} = \{U^1, ..., U^K\}$ are dual feasibility-tolerance variables, and d is a scalar constant. The augmented Lagrangian optimization problem can be solved by the alternating direction method of multipliers (ADMM) which guarantees converge to the global optimum. The skeleton of the algorithm at the i^{th} iteration includes the following three steps:

$$\begin{aligned}
\{\Theta_{(i)}\} &= \arg\ \min_{\Theta} L(\{\Theta\}, \{Z_{(i-1)}\}, \{U_{(i-1)}\}) \\
\{Z_{(i)}\} &= \arg\ \min_{Z} L(\{\Theta_{(i)}\}, \{Z\}, \{U_{(i-1)}\}) \\
\{U_{(i)}\} &= \{U_{(i-1)}\} + (\{\Theta_{(i)}\} - \{Z_{(i)}\})
\end{aligned} \tag{4.39}$$

Briefly, to estimate Θ, we utilize a coordinate-wise descent approach to obtain each parameter in Θ, and directly apply a well-suited proximal gradient algorithm, which can achieve ϵ optimality within $O(\frac{1}{\epsilon})$ iterations. To update Z, the optimization problem is separable with respect to each pair of elements in the matrix, and thus can be solved using the fused lasso signal approximator or the group lasso operator, depending on the choice of penalty P in (4.34) or (4.35).

6 Experiment

6.1 Simulated Data

We consider three blocks of variables in our simulation as follows: We consider two mixed graphical models $G^1 = (V^1, E^1)$ $G^2 = (V^2, E^2)$ (representing two classes) each consisting of three blocks of variables (block A, block B and block C). Block A comprises 50 Gaussian variables, block B contains 50 Bernoulli variables and block C contains 50 Poisson variables. We order the directionality such that block A points to block B, and block C points to both block A and block B. The topologies of the two simulated networks are generated as follows: In both models, each node in block A is connected to its two nearest neighbors in block A and each node in block B is connected to its two nearest neighbors in block B and likewise for block C. For the inter-block edges, for each node x in block A, we randomly picked three nodes in block B and form an edge from node x to the three chosen nodes. For each node y in block C, we randomly picked three nodes in block B and block A and form an edge from node y to each of the chosen nodes. We then proceed to set the edge

Algorithm 3 ADMM algorithm

```
Require: Data $\{X^{(j)(k)}\}_{j=1}^{n_k}$
Initialize $\Theta^{(k)} = 0$,$U^{(k)} = 0$, $Z^{(k)} = 0$, $t = 0.8$, $\beta = 0.5$ for $k = 1, ...K$
Choose a scalar $d > 0$

for $i = 1, 2, 3, ....$ and repeat until convergence do
    for $k = 1, ..K$ do
        for each parameter $\theta^{(k)}$ in $\Theta^{(k)}$ do
            $t = t_{(i-1)}$;
            Repeat until convergence;
            $v = \frac{1}{d+t}(d(z^{(k)}_{(i-1)} - u^{(k)}_{(i-1)}) + t(\theta^{(k)}_{(i-1)} - t\frac{dl(\Theta^{(k)},X^{(k)})}{d\theta^{(k)}_{(i-1)}})));$
            $t = \beta t$;
            Return $t_{(i)} = t$ and $\theta^{(k)}_{(i)} = v$;
        end for
    end for
    for $k = 1, ..K$ do
        $Z^{(k)}_{(i)}$=arg $\min_{Z^{(k)}} \left\{ \frac{d}{2} \sum_{k=1}^{K} \|Z^{(k)} - (\Theta^{(k)}_{(i)} + U^{(k)}_{(i)})\|_F^2 + P(Z) \right\}$
        and solve it depending on the penalty type $P$;
        Set $A^{(k)} = \Theta^{(k)}_{(i)} + U^{(k)}_{(i-1)}$;
    end for
    If $(P =$ fused lasso penalty$)\{$
      for each $(u, v)$
       Solve $\min_{\{Z^{(k)}_{uv}\}} \left\{ \frac{d}{2} \sum_{k=1}^{K} (Z^{(k)}_{uv} - A^{(k)}_{uv})^2 + \lambda_1 \sum_{k=1}^{K} \mid Z^k_{uv} \mid + \lambda_2 \sum_{k \leq k^,} \mid Z^k_{uv} - Z^{k^,}_{uv} \mid \right\}$

       using fused lasso signal approximator

    $\}$

    else $\{$
      for each $(u, v)$
       Solve $\min_{\{Z^{(k)}_{uv}\}} \left\{ \frac{d}{2} \sum_{k=1}^{K} (Z^{(k)}_{uv} - A^{(k)}_{uv})^2 + \lambda_1 \sum_{k=1}^{K} \mid Z^k_{uv} \mid + \lambda_2 \sqrt{\sum_{k=1}^{K} (Z^{(k)}_{uv})^2} \right\}$

       using group lasso signal approximator

    $\}$

    $U^{(k)}_{(i)} = U^{(k)}_{(i-1)} + (\Theta^{(k)}_{(i)} - Z^{(k)}_{(i)})$
end for
```

potential θ_{ij} for the two classes as follow:

$$\theta_{st}^{(k)} = \begin{cases} 0.3 & \text{if } (s,t) \in E^k \text{ and } s \in A \text{ and } t \in A \text{ and } k = 1 \\ 0.1 & \text{if } (s,t) \in E^k \text{ and } s \in B \text{ and } t \in B \text{ and } k = 1 \\ -0.7 & \text{if } (s,t) \in E^k \text{ and } s \in C \text{ and } t \in C \text{ and } k = 1 \\ 0.1 & \text{if } (s,t) \in E^k \text{ and } s \in A \text{ and } t \in B \text{ and } k = 1 \\ 0.1 & \text{if } (s,t) \in E^k \text{ and } s \in C \text{ and } t \in A \text{ and } k = 1 \\ 0.1 & \text{if } (s,t) \in E^k \text{ and } s \in C \text{ and } t \in B \text{ and } k = 1 \\ 0.1 & \text{if } (s,t) \in E^k \text{ and } s \in A \text{ and } t \in A \text{ and } k = 2 \\ 0.4 & \text{if } (s,t) \in E^k \text{ and } s \in B \text{ and } t \in B \text{ and } k = 2 \\ -0.3 & \text{if } (s,t) \in E^k \text{ and } s \in C \text{ and } t \in C \text{ and } k = 2 \\ 0.3 & \text{if } (s,t) \in E^k \text{ and } s \in A \text{ and } t \in B \text{ and } k = 2 \\ 0.3 & \text{if } (s,t) \in E^k \text{ and } s \in C \text{ and } t \in A \text{ and } k = 2 \\ 0.3 & \text{if } (s,t) \in E^k \text{ and } s \in C \text{ and } t \in B \text{ and } k = 2 \end{cases}$$

6.1.1 Results

For both classes, we employ a Gibbs sampler. Speaking, we iterate through the nodes, and sample from each nodes conditional distribution. To ensure independence, after a burn-in period of 3000 iterations, we select samples from the chain 500 iterations apart. Using the proposed method, we discovered the network structure for two classes over a range of tuning parameters (λ_1 and λ_2). We recorded the total number of identified edges for each pair of tuning parameters (λ_1 and λ_2) and calculated the number of true positive edges and the number of false positive edges. We then investigate the performance of the tuning parameter selection procedure by checking the sensitivity and specificity of the selected model. The sensitivity and specificity are defined below.

$$\begin{aligned} sensitivity &= \frac{TP}{TP + FN} \\ specificity &= \frac{TN}{TN + FP} \end{aligned} \tag{4.40}$$

where TP refers to true positives, FP refers to false positives, TN refers to true negatives and FN refers to false negatives. For each pair of tuning parameters, we calculated the corresponding sensitivity, specificity. Figure 4.2(a) shows the sensitivities of our mixed graphical model with fused lasso penalty over a range of tuning parameters, while Figure 4.2(b) shows the sensitivities of our model with group lasso penalty. Figure 4.3(c) shows the specificities of our model with fused lasso penalty over a range of tuning parameters, while Figure 4.3(d) shows the specificities of our model with group lasso penalty. The results demonstrate that our method achieves high sensitivities (0.927 for group lasso and 0.88 for fused lasso penalties) and specificities (0.91 for group lasso penalty, 0.83 for fused lasso penalty) using the BIC-type model selection approach.

6.2 Real Data

Brain cancer and Colorectal cancer We applied our method to the publicly available TCGA datasets of two cancer types: colorectal carcinoma (coadread) and glioblastoma multiforme (GBM). We obtained the mutation, copy number variation (CNV) and the gene expression data from TCGA data portal resulting in 768 subjects for coadread and 702 GBM subjects. For both GBM and coadread, we use level III RNA-sequencing data for the gene expression so that the gene expression levels can be modeled with the Poisson distribution. For the CNV and mutation data, we utilize both the Level II nonsilent somatic mutations in conjunction with Level III copy number variation data. We merge them together forming a binary matrix with the rows and columns corresponding to samples and genes respectively. We then filtered out genes with mutation rate occur less than 8% of the patients in the

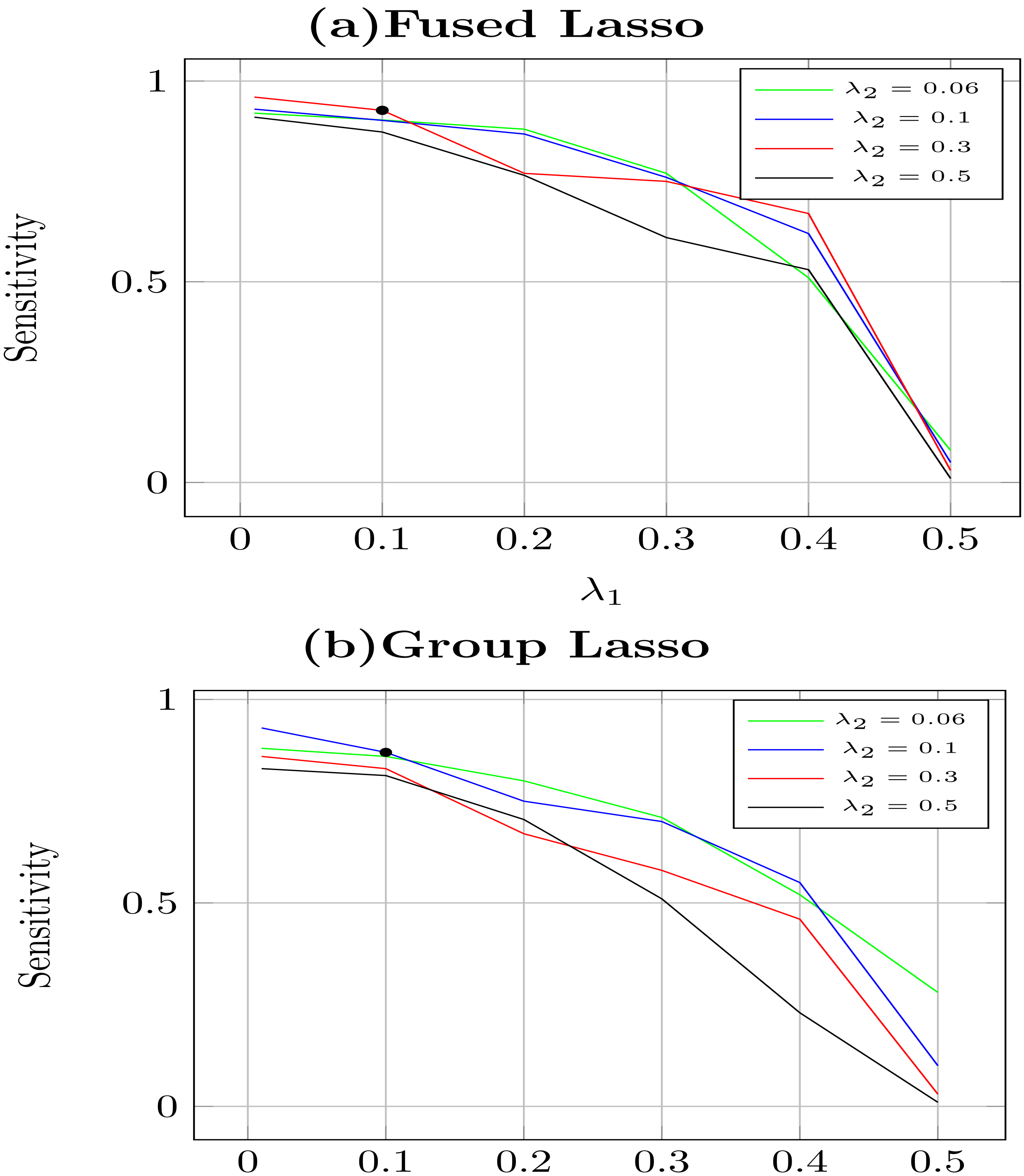

Figure 4.2. Performance of our mixed graphical model on three blocks with various λ_1 and λ_2. BIC value depicted as a black dot

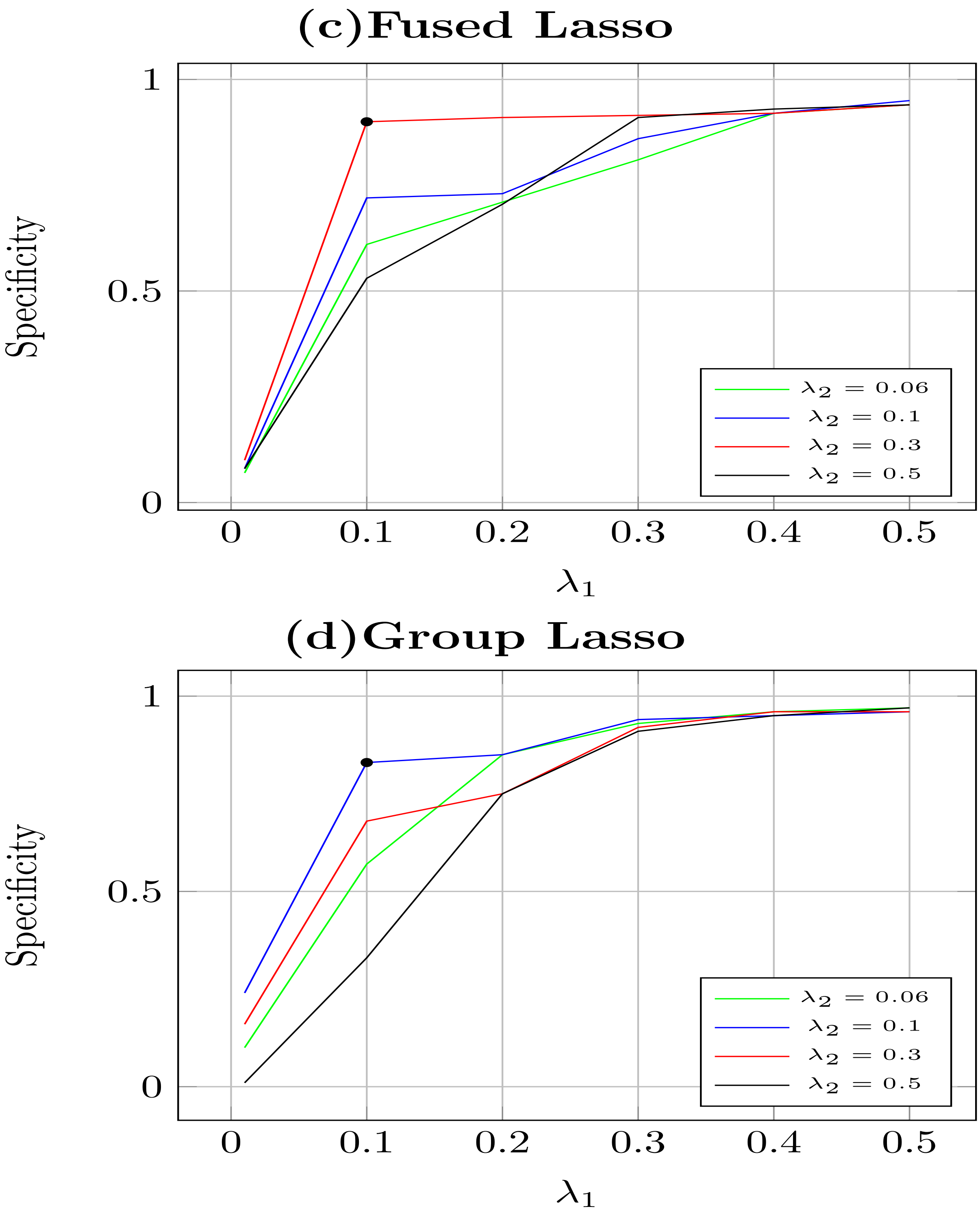

Figure 4.3. Performance of our mixed graphical model on three blocks with various λ_1 and λ_2. BIC value depicted as a black dot

mutation CNV data.

Since it is well known that both the CNV and fixed point mutation (block X) affect gene expression level (block Y), we can use this relationship to order two blocks of variables as follow: $P[X, Y] = P[Y|X]P[X]$, where $P[Y|X]$ is a pairwise Poisson conditional random field and $P[X]$ stands for the pair-wise Ising model in both classes.

We set $\lambda_2 = 0$ assuming that there is no any similarity between coadread and GBM. We also select the optimal value for the tuning parameter controlling sparsity as $\lambda_1 = 0.5$ by BIC. We discovered 736 edges for coadread and 617 edges for GBM respectively. Among the interactions of coadread, 27.9% are CNV-CNV interactions, 53.4% are gene-gene interactions and 18.7% are CNV-gene interactions. Among the interactions of GBM, 28.2% are CNV-CNV interactions, 44.2% are gene-gene interactions and 27.6% are CNV-gene interactions. In addition, there are 551 coadread-specific interactions, 432 GBM-specific interactions, and 185 common interactions. Among the 185 common interactions are common oncogenes such as $PTEN, TP53, PIK3CA$ etc which play important roles in different cancers. Our overall estimated networks for GBM and coadread are shown in Figures 4.6 and 4.7.

Among the coadread-specific interactions, we identify several connections between biological data of different types which could be bio-markers or driver genes. Examples include: The mutation of $MLH1$ is connected to $TCF7L2$ gene expression. MLH1 [71] is a tumor suppressor gene involved in DNA mismatch repair. Germline mutations in this gene are known to cause Lynch syndrome. The most common malignancies in Lynch syndrome are colorectal and endometrial carcinomas. In addition to germline mutations, somatic mutations in this gene have been described in colorectal and endometrial cancers. $TCF7L2$ gene [84] is involved in the Wnt β signaling pathway, and all factors are thought to be important in the etiology of colon cancer. MCC gene mutation is linked to the gene expression of $TCF7L1$. MCC is a candidate colorectal tumor suppressor gene that is thought to negatively regulate cell cycle progression. In [42], it is discovered that a high level of $TCF7L1$ mRNA expression correlates with shorter survival of patients and knocking out $TCF7L1$ can reduce growth of a colorectal tumor cell line in vitro. The mutation of $PIK3R1$ is linked to the expression of $GSK3B$. $PIK3R1$ [48] has been reported as an oncogene in colon cancer. $GSK3B$ [83] is part of the GABAB R/GSK-3β/NF-KB signaling pathway on regulating proliferation of colorectal cancer cell, and the down-regulation of $GSK3B$ triggers the proliferation of colon cancer cell. The mutation of both the $KRAS$ gene and the APC gene are linked to the expression of $PIK3CG$. $KRAS$ [5] is involved in the modulation of several downstream effectors, that include: $RAF/MEK/ERK$, $PI3K/AKT$, $RAlGDS/p38MAPK$ pathway and is found to be frequently dysregulated in colorectal cancer. APC [43] is a tumor suppressor gene which is commonly mutated in colon cancer. For $PIK3CG$, a reduction of $PIK3CG$ [80] expression is detected immunohistochemically in 85% of human colorectal cancers and was closely associated with invasion, metastasis, and poor differentiation. In addition, down-regulation of $PIK3CG$ expression and hypermethylation of promoter regions are also detected in primary colon cancers. The mutation of gene $TGFB1$ is linked to the expression of $CCND1$. $TGFB1$ [14] is a tumor suppressor gene regarding tumor initiation and over-expression of $CCND1$ [47] is significantly associated with both poor overall survival and disease free survival among the colon cancer patients.

On the other hand, among the GBM-specific interactions, we found the following connections among different biomarkers: $CDKN2A$ mutation is linked to the gene expression of $ERBB3$. $CDKN2A$ [76] is a tumor suppressor gene whose loss is associated with shortened overall survival in lower grade Astrocytomas and $ERBB3$ is found to be differentially expressed between the proneural tumors and the mesenchymal tumors in GBM [87]. The mutation of $PIK3CA$ is linked to both the $GAB1$ and $GAB2$ gene expression. $PIK3CA$ [20] is a well known oncogene. $GAB1$ and $GAB2$ [51] play important roles in cancer cell signaling. In particular, it has been demonstrated that the up-regulation of $GAB2$ is correlated with the World Health Organization (WHO) grade of gliomas and that patients with high $GAB2$ expression levels exhibited shorter survival time. The mutation of $ERBB2$ is linked to the expression of $PDGFRB$ gene. $ERBB2$ [87] is a proto-oncogene which is frequently mutated in GBM. $PDGFRB$ [93] is over-expressed in GBM microvascular proliferation. Another mutation $LZTR1$ [72] which is thought to act as a tumor suppressor is connected to $SPRY2$ expression. $SPRY2$ [90] is a known regulator of receptor tyrosine kinases (RTKs) which promotes glioma cell and tumor growth and cellular resistance to targeted inhibitors of oncogenic RTKs; $SPRY2$ [90] is also related to glioblastoma subtypes and patient survival. It is also found that $SPRY2$ is under-expressed in the proneural tumors in GBM. Mutation of $PTEN$ is linked to the expression of $ADAM10$ gene. $PTEN$ [11] is frequently mutated in mesenchymal tumor of GBM. And over-expression of $ADAM10$ gene [13] has been observed in human glioma tissue and especially in tumor sphere cultures. The mutation of $PDGFRA$ is linked to the gene expression of $FOXO3$. $PDGFRA$ [87] is frequently mutated in proneural tumors of GBM. For $FOXO3$, a recent study [73] examining $FOXO3$ expression in patient HGGs suggests a strong association between high expression and poor prognosis, while another study [94] using two human GBM cell lines demonstrates that $FOXO3$ expression induces TMZ resistance.

Gene Enrichment Analysis We performed gene enrichment analysis on the 40 genes with the highest degree (hub nodes) in their respective gene network in both GBM and Colorectal cancer. As shown in Figure 4.4 and 4.5, we discovered that the identified functions in both cancers are related to tumor development like pathway signalling events and cell apoptosas events etc.

Comparison with other methods We applied the method by Danaher [22] to the above cancer dataset (coadread and GBM) by treating both CNV/mutation and gene expression as continuous variables. It results in 451 interactions for coadread, and 348 interactions for GBM. Among the interactions of coadread, 1.4% are CNV-CNV interactions, 93.8% are gene-gene interactions and 4.8% are CNV-gene interactions. Among the interactions of GBM, 2.1% are CNV-CNV interactions, 91.5% are gene-gene interactions and 6.4% are CNV-gene interactions. This highlights our method is better at capturing interactions

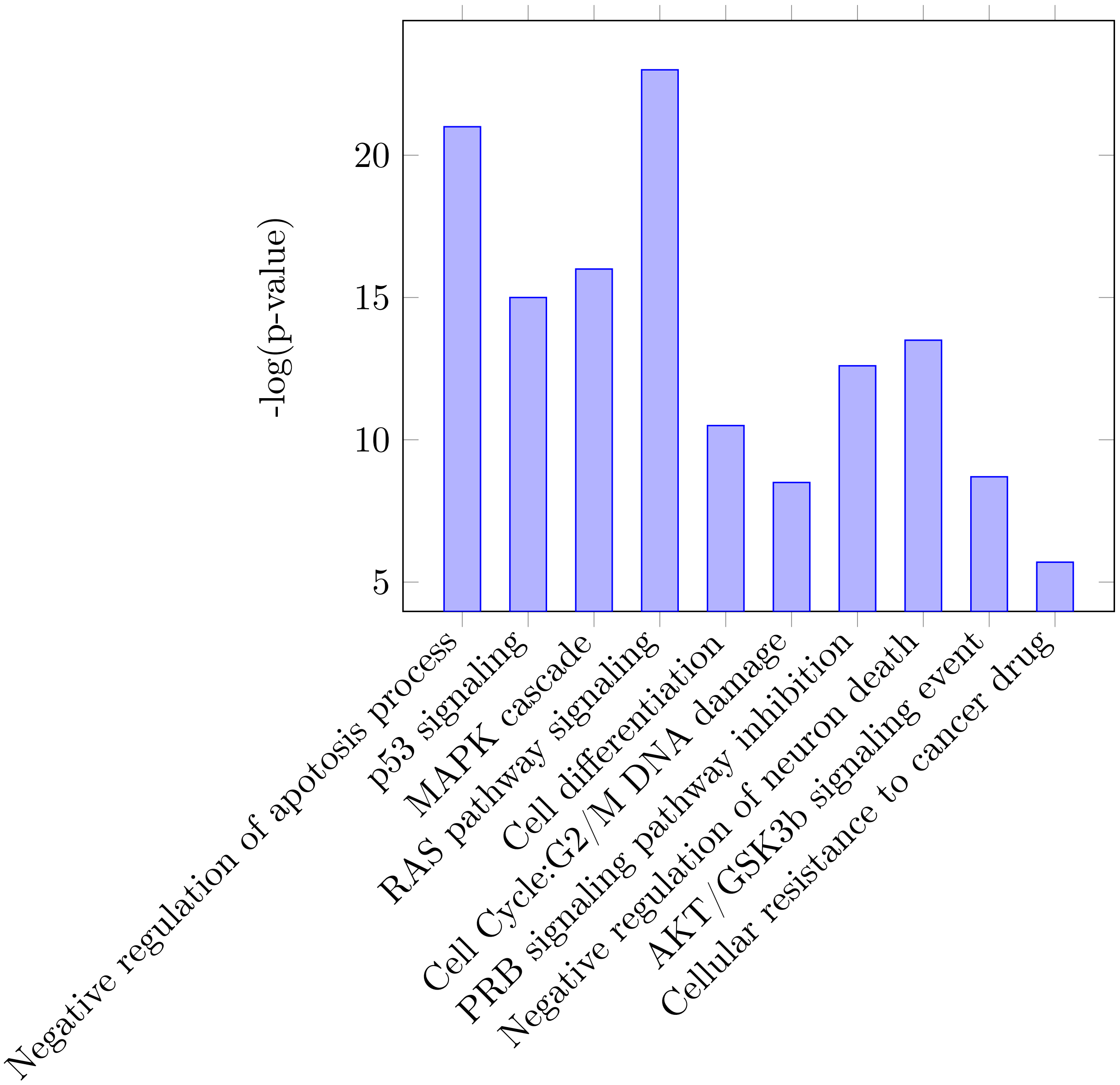

Figure 4.4. Functions discovered by Gene Enrichment Analysis on hub nodes of GBM

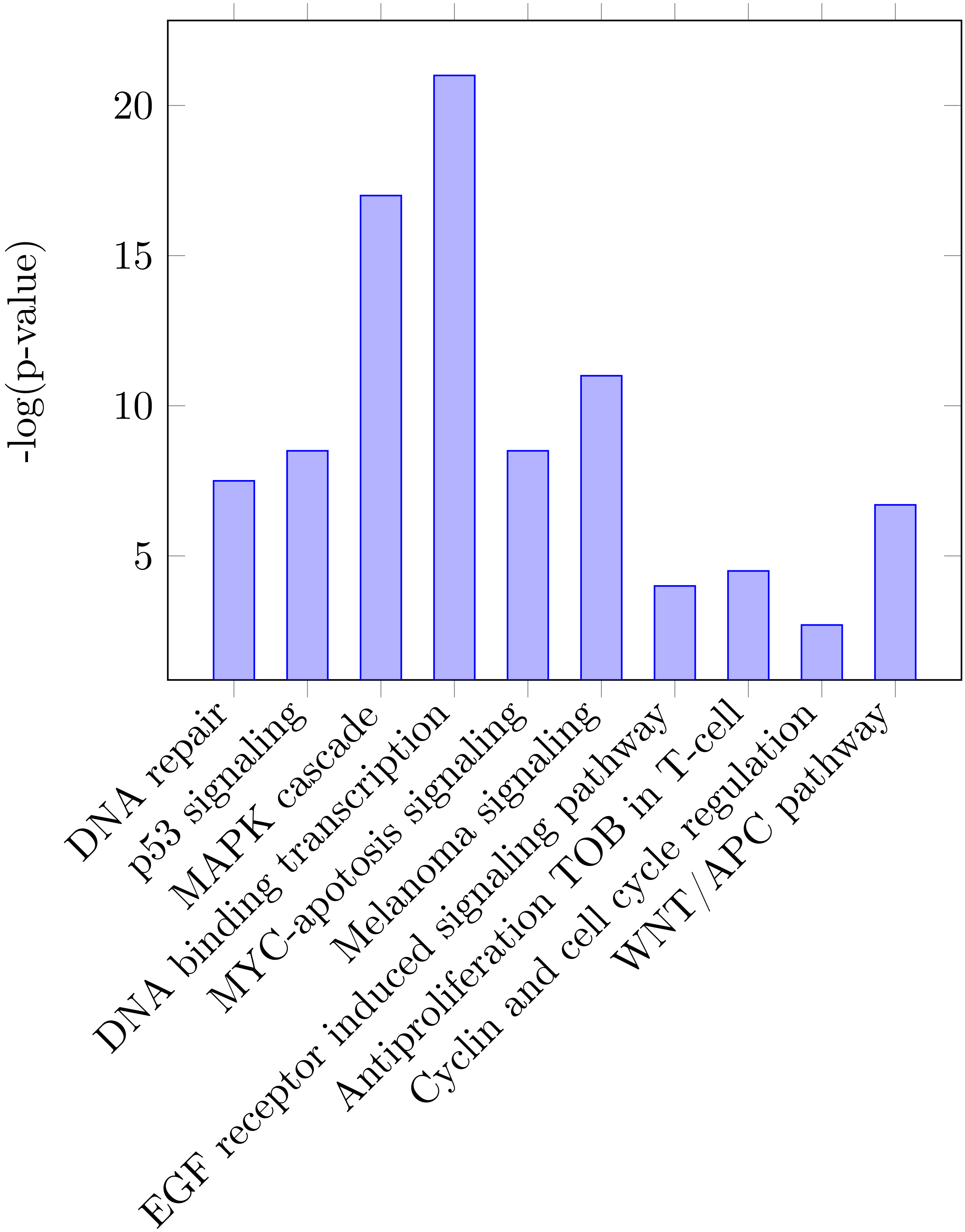

Figure 4.5. Functions discovered by Gene Enrichment Analysis on hub nodes of Colorectal cancer

and also more capable of identifying CNV-CNV interactions and CNV-gene interactions.

Luminal-A breast cancer and Basal-like breast cancer We applied our method to two major subtypes of breast cancer: luminal A cancers and basal-like cancers. Luminal A cancers tend to grow slowly and possess good prognosis, while basal-like cancers tend to grow quickly and have poor prognosis. As usual, we obtained the CNV, mutation as well as gene expression data from TCGA data portal resulting in 232 luminal A cancer subjects and 174 basal cancer subjects. We used the fused lasso penalty to the data-sets for the two cancer subtypes, and the proposed BIC approach to choose the tuning parameters ($\lambda_1 = 0.4$, $\lambda_2 = 0.5$). We identified 321 edges for luminal A cancer and 296 edges for basal cancer, respectively. Among them, there are 118 luminal-specific interactions, 93 basal-specific interactions, and 203 common interactions. Among the common interactions were genes such as PTEN, BRCA1, EP300, SMAD4, TP53, PIK3CD, etc. The overall networks for the two sub-types are shown in Figures 4.8 and 4.9 . We then identify genetic variants that are potentially implicated in each cancer subtype among the subtypes specific interactions. We identify 4 hub nodes in the two networks which can differentiate between the two sub-types. We discovered that gene KIT has a higher degree in basal cancer gene network than in luminal A cancer gene network (30-13). Gene $EGFR$ also has a higher degree in basal cancer gene network than in luminal A cancer gene network (16-6). Both genes KIT and $EGFR$ are amplified and over-expressed in basal type cancer [65]. On the other hand, gene $CDK6$ has a higher degree in luminal A cancer gene network than in basal cancer gene network (13-5) and $ESR1$ gene is only present in the luminal A gene network but not in the basal. $CDK6$ is one member of the cyclin-dependent kinase family, which is found to be amplified in luminal A cancer [65]. Similarly, gene $ESR1$ [65] is one of two main types of estrogen receptors which is typically highly expressed in luminal A cancer. The module containing the four hub nodes for luminal A and basal cancer are shown in Figure 4.10 and Figure 4.11 respectively.

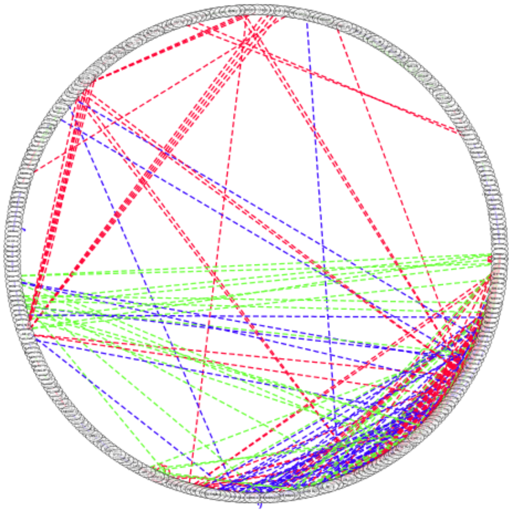

Figure 4.6. Graphical model estimation of GBM (Blue:GENE-GENE Red:CNV-CNV Green:CNV-GENE)

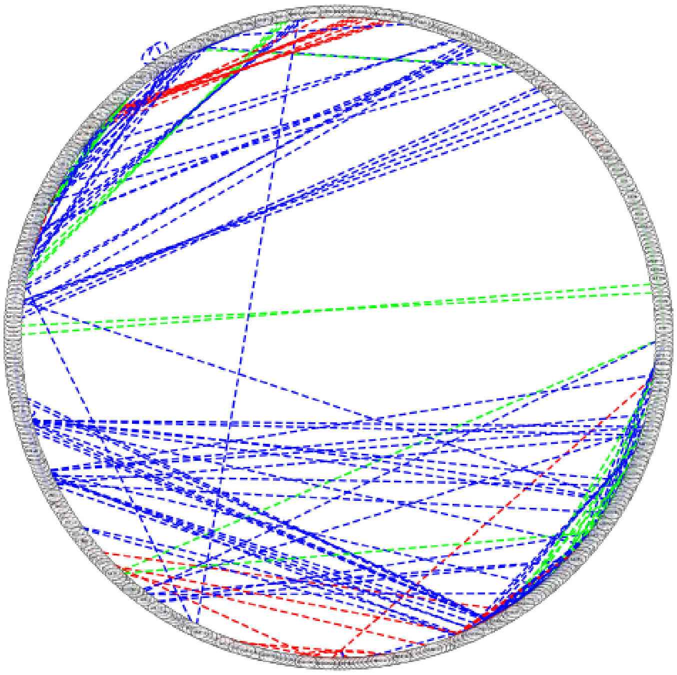

Figure 4.7. Graphical model estimation of COADREAD (Blue:GENE-GENE Red:CNV-CNV Green:CNV-GENE)

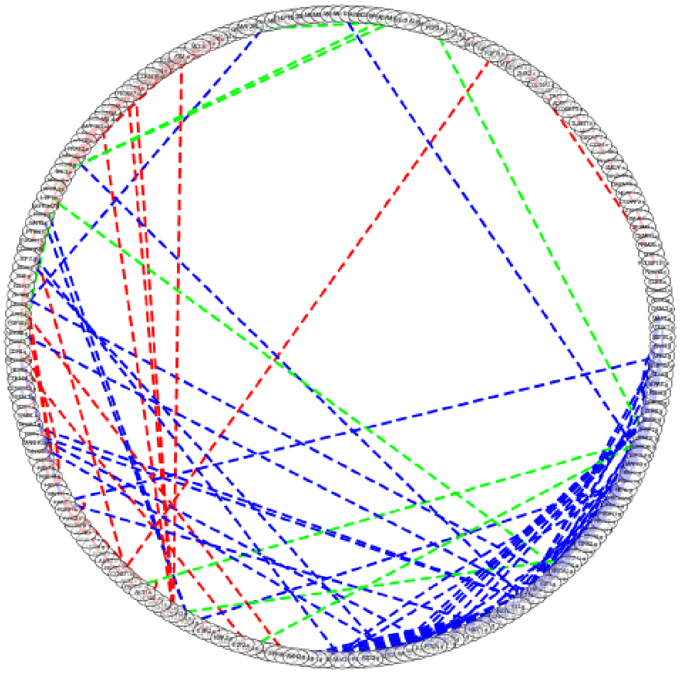

Figure 4.8. Graphical model estimation of BASAL (Blue:GENE-GENE Red:CNV-CNV Green:CNV-GENE)

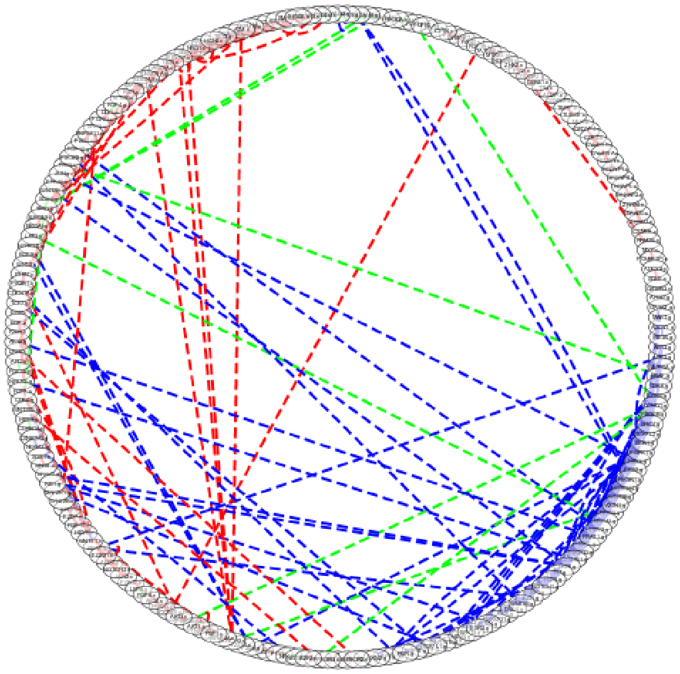

Figure 4.9. Graphical model estimation of LUMINAL-A (Blue:GENE-GENE Red:CNV-CNV Green:CNV-GENE)

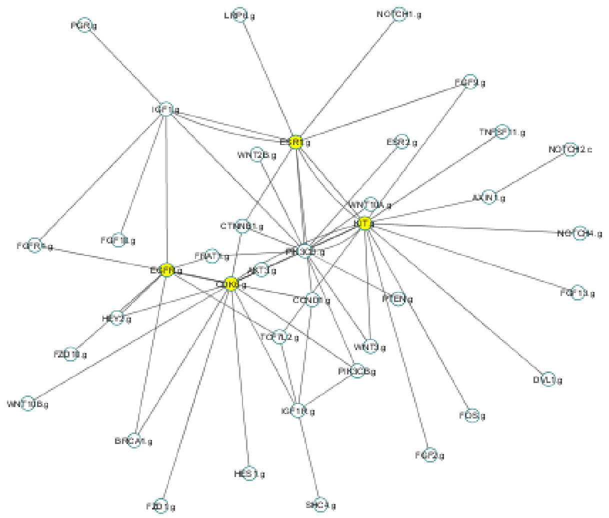

Figure 4.10. Gene network of 4 hubs nodes (KIT, CDK6, EGFR, ESR1) across Luminal-A cancer

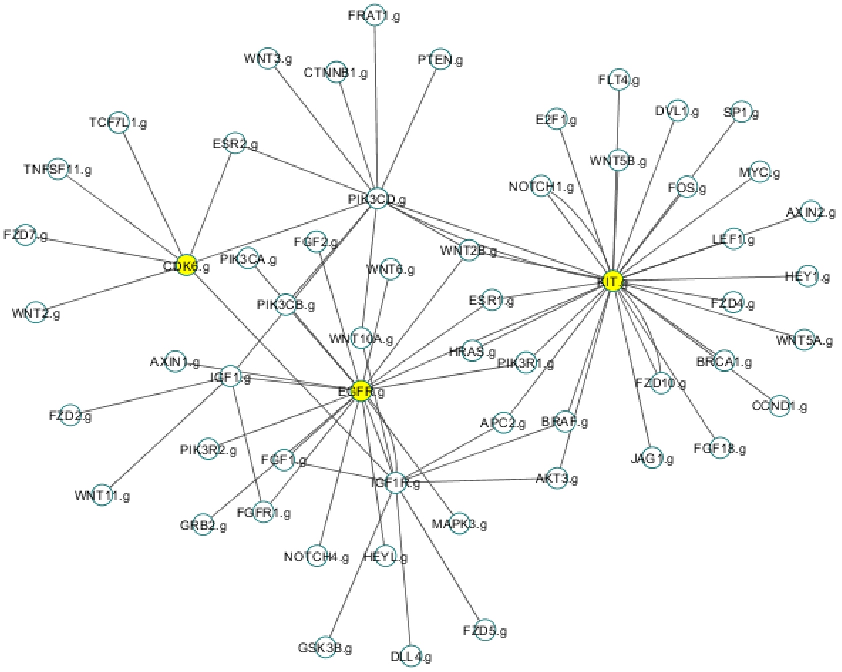

Figure 4.11. Gene network of 4 hubs nodes (KIT, CDK6, EGFR, ESR1) across Basal cancer

7 Conclusion

In this paper, we employ the exponential families for the problem of estimating multiple related mixed graphical models from high dimensional data with different discrete and continuous variables and with observation across distinct and related biological conditions. This framework had been demonstrated using real cancer data. For future work, it is natural to develop hypothesis testing such that the mixed graphical models can be supplemented by a p-value on each edge to determine the confidence level of its existence in the overall mixed graphical model.

CHAPTER 5

EFFICIENT CASCADE MODELLING OF DIFFUSION NETWORK BY PARETO DISTRIBUTION

Time plays an essential role in the diffusion of information, influence and disease over networks. Usually we are only able to collect cascade data in which an infection (receiving) time of each node is recorded but without any transmission information over the network. In this chapter, we given all the fer the transmission rates among nodes by Pareto distributions. Pareto modeling has several advantages. It is naturally motivated and has a nice interpretability. The scale parameter of a Pareto distribution naturally fits in the starting time of a transition, i.e., the infection time of a parent node in the cascade data is the starting point for a transition from the parent to its receiver. The shape parameter (alpha) serves as the transition rate. The larger the alpha is, the faster the transition is and the larger probability for disease or information to spread in a short time period is. Pareto modeling is mathematically simple and computationally easy. It has explicit solutions for the optimization problem which maximizes time-dependent pairwise transmission likelihood between all pairs of nodes. We present three modelings with a common transmission rate, with different transmission rates and with different infection rates. We also extent the Pareto modeling to deal with the multiple source problem. We consider non-overlapping, partially overlapping two sources and fully overlapping multiple sources diffusion networks. For non-overlapping networks, the problem is transformed to the identification of the starting time of the second source. For the partially overlapping scenario, a mixture model is adopted and EM algorithm is utilized for obtaining estimators. The fully overlapping case is an extension of the mixture modeling. The number of sources can be selected by an usual Akaike or Bayesian information criterion. Experiments on real and synthetic data show that our models accurately estimate the transmission rates from one source as well as multiple source cascade data.

1 Background

Diffusion network and its propagation have attracted a great deal of research attention recently [1, 7, 21, 35, 41, 64, 89, 92]. It has been applied to many problem domains varying from social networks to viral marketing. Inferring diffusion network from cascades has become one of the major tools to understand social behaviors or virus infection. Cascade data about a diffusion process in a network often record the diffusion traces but without any information on network structure. For example, epidemiologists can observe that a person becomes ill at what time but they can neither determine who infected the patient nor the infection rate of each individual in virus infection. In information dissemination, we observe when a blog posts a piece of news or when a piece of information is tweeted by Twitter. However, as is often the case, the blogger does not link to her source and we had no idea where she obtained the information, how long it took her to post it or the extent in which the piece of information could be spread further. In viral marketing, viral marketers can track when customers purchase products or subscribe to services, but it is hard to determine who influence the customers' decisions, how long it takes for them to make up their decision, or the extent they pass their opinion or recommendation on to other customers. In all these circumstances, we observe where and when but not so much on how a piece of information or antigen can propagate through a network. As a matter of fact, it is of utmost interest to decipher the mechanism underlying the process since understanding diffusion process validates efforts for preventing from virus infections, predicting information propagation, or maximizing the profit of selling a product. Our goal of this paper is to propose a novel modeling to infer infection rates of diffusion processes.

2 Related Works

Most of the previous works have focused on developing network inference algorithms and evaluating their performance experimentally on different synthetic and real networks. The models which are most related to our works are [35, 64, 77]. In [77], authors developed a method called NETRATE to model the underlying diffusion process. It infers the transmission rates between nodes of a network by computing the model which maximizes the likelihood of the observed data in terms of temporal traces observed by cascades of infections. Another similar line of work is [64] in which the authors utilized a generative probabilistic model for inferring diffusion networks using sub-modular optimization. Meyers and Leskovec in [64] developed an algorithm called CONNIE in which they infer the connectivity of the network as well as the prior probability of infection of each node using a convex program and heuristics. Both papers assumed the transmission rate between all nodes to be fixed and their models are only applicable to one source. Works on the similar line that utilizes a generative probability model to infer the network which generate all the cascades with the maximum likelihood formulation include NETINF [35] and InfoPath [36], which have been demonstrated to perform incredibly well on synthetic data. Authors in [23] further extended the work to investigate the condition in which the network structure could be recovered from the traces of cascade. They are capable of identifying a natural incoherence condition for such a model which depends on the network structure, the diffusion parameters as well as the sampling process of the cascades. This condition captures intuitions that the network structure could be recovered if the co-occurrence of a node and its non-parent nodes is small in the cascades and with enough cascades, the probability of success in recovery is approaching one in a rate exponential in the number of cascades. On the other hand, another line of research has done by [29] in which the authors aim to discover the source of infection using incomplete and partially observed cascade traces. They developed a two stage graphical model, which at first learns a continuous time diffusion network model based on historical diffusion traces and then identifies the source of an incomplete trace of cascades by maximizing the likelihood of the trace under the learned model. Furthermore,[91] studies the problem of transferring structure knowledge from an external diffusion network with sufficient cascade data to help predict the hidden links of the diffusion network. In our work, we employ a similar line of reasoning and assumption of a generative probabilistic model. What we contribute are (1) new modelings (2) extension to multiple sources, which to the best of our knowledge hadn't appeared in the above schemes and in the literature.

An overview of the proposed model

This article presents a model for inferring the mechanisms underlying diffusion processes based on historical diffusion traces. To achieve this goal, we make some basic assumptions about the temporal structures which generate the diffusion process and incorporate into our model. First, diffusion process occur over on a static unknown network. Second, infections along the edges (between each pair of individuals) of the network occur independently of each other. Third, infection can occur at different times and the probability of a parent node infecting a child node is determined by a probability density function depending on the time of infection of the parent node, the time of infection of the child node and infection rate. Finally, we observe the time of occurrence of all infections in the network during the time window recorded. Our objective is to infer the infection rate and the likelihood of infection across its edges after recording the times of infection of each individual node within the time window in a network. We cast the problem as a maximum likelihood problem and are able to calculate the infection rate efficiently. In this consideration, we are motivated to use Pareto distribution. An important characteristic of the Parate distribution is its slow convergence to zero, which enables occasional long-range transmissions of infectious agents in addition to principal short-range infections [12, 96, 61]. The scale parameter of the Pareto distribution naturally fits in the starting time of a transition, i.e., the infection time of a parent node in the cascade data is the starting point for a transition from the parent to its child. The shape parameter (alpha) serves as the transition rate. The larger the alpha is, the faster the transition is and the larger probability for disease or information to spread in a short time period is. Not only Pareto modeling has an intuitive motivation and nice interpretation, but also is mathematical simple and computational easy. We present three modelings: (1) with a common transmission rate; (2) with different transmission rates; (3) with different infection rates. All of them have explicit solutions for the optimization problem. On the other hand, in real world scenarios, there may exist multiple sources contributing to the dissemination of content or pathogens in a network. For instance, in virus outbreak, one source of pathogen can start in North America whereas another one starts in Europe and the two sources converge after people from the two geographical regions commute through common access point like airport. We extended our model to include multiple sources. Three cases are considered: (1) nonoverlaping two source; (2) partially overlapping two sources; (3) fully overlapping sources. In the first case, the problem is translated to the identification of the starting time of the second source. For the latter two scenarios, mixture distributions are adopted and and EM algorithms are utilized to obtain estimates of the infection rates of multiple sources accordingly. Our model differs from the traditional ones such as in [23] and [77]. They intend to recover the network and try to estimate variations among all pairwise edges, which unavoidably leads to models with a large number of parameters. In contrast, our works focus on modeling the diffusion rate among the network. Our models are simple and easy to interpret, as well as easy to compute. Moreover, we extend our modeling to multiple source diffusion processes, while to the best of our knowledge, it is extremely difficult to extend their model to consider more than one source.

3 Framework

Before introducing our model, we first give some basic concepts which are essential to information diffusion. Then we describe the cascade data and assumption of cascade modeling assumptions.

3.1 Basic terminology and Pareto distribution

Let $f(t)$ be the probability density function (pdf) of T. Then the cumulative distribution function (cdf) can be denoted as $F(t) = P(T \leq t) = \int_0^t f(x)dx$. The *survival function* $S(t)$ is the probability that an event does not happen by time t:

$$S(t) = P(T > t) = 1 - F(t) = \int\limits_t^\infty f(x)dx. \tag{5.1}$$

Given functions $f(t)$ and $S(t)$, we further define the *hazard function* $H(t)$, which represents the instantaneous rate that an event occurs just right after time t given that it already survives up to time t. That is,

$$H(t) = \lim_{\Delta t \to 0} \frac{P(t \leq T \leq t + \Delta t | T \geq t)}{\Delta t} = \frac{f(t)}{S(t)}. \tag{5.2}$$

A random variable T is said to have a Pareto (Type I) distribution if its survival function (also called tail function) is of the form

$$S(t) = \left(\frac{t_0}{t}\right)^\alpha I(t \geq t_0) + I(t < t_0),$$

where $I(\cdot)$ is the indictor function, t_0 is the scale parameter which is (necessarily positive) minimum possible value of T and α is the shape parameter which is positive. It follows (by differentiation) that the probability density function and hence the hazard function are

$$f(t) = \frac{\alpha t_0^\alpha}{t^{\alpha+1}} I(t \geq t_0); \quad H(t) = \frac{t_0}{t} I(t \geq t_0).$$

The Pareto distribution is a simple model for nonnegative data with a power law probability tail [67]. An important characteristic of the Parate distribution is its slow convergence to zero, which enables occasional long-range transmissions of infectious agents in addition to principal short-range infections [12]. Two parameters of the Pareto distribution have an intuitive interpretation when modeling diffusion of information or disease over networks. The scale parameter naturally fits in the starting time of a transition, i.e., the infection time of a parent node in the cascade data is the starting point for a transition from the parent to its receiver. The shape parameter (alpha) serves as the transition rate. The larger the alpha is, the faster the transition is and the larger probability for disease or information to spread in a short time period is. The fact that the hazard rate of the Pareto distribution decreasing with the time makes the Pareto distribution more realistic in applications than some other distributions such as the exponential distribution, which has a constant hazard rate over the time. In epidemiology, the Pareto distribution has been widely used for describing epidemic behavior such as the probability of outbreaks of different sizes or the rate of incidence [96, 61].

Data

Observations are recorded on a fixed set of N objects and result in a cascade $\mathcal{T} = \{t_1, t_2, ..., t_N\}$. It is an N dimensional vector recording when the i^{th} node is infected at t_i, where $t_i \in (0, T_{\max}] \cup \{\infty\}$. Symbol ∞ labels that the node is not infected during the observation window $[0, T_{\max}]$. Without loss of generalization, we can assume $0 \leq t_1 \leq t_2 \leq ... \leq t_N$.

3.2 Modeling pairwise infection likelihood

Define $f(t_i|t_j)$ as the conditional likelihood of transmission from node j and node i. The conditional transmission likelihood depends on the infection times (t_j, t_i). A node cannot be infected by a node infected later in time. In other words, a node j that has been infected at a time t_j may infect a node i at a time t_i only if $t_j \leq t_i$. We first give a general framework of modeling the likelihood of a cascade. We then proceed to the three different modelings. Consider a cascade $\mathcal{T} = \{t_1, t_2, ..., t_N\}$. We first compute the likelihood of the observed infections $t^{\leq T} = (t_1, t_2, ..., t_N | t_i \leq T_{max})$. Since we assume infections are conditionally independent given the parents of the infected nodes, the likelihood factorizes over nodes as

$$f(t^{\leq T}) = \prod_{t_i \leq T_{max}} f(t_i|t_1, t_2, ..., t_{i-1}, t_{i+1}, ..., t_N) \tag{5.3}$$

Computing the likelihood of a cascade thus boils down to computing the conditional likelihood of the infection time of each node given the rest of the cascade. Following the independent cascade model proposed by Kempe [41], we assume that a node gets infected once the first parent infects the node. Given an infected node i, we compute the likelihood of a potential parent j to be the first parent,

$$f(t_i|t_j) \times \prod_{k \neq j, t_k \leq t_i} S(t_i|t_k) \tag{5.4}$$

We now compute the conditional likelihood by summing over the likelihoods of the mutually disjoint events considering each potential parent as the first parent in turn resulting in

$$f(t_i|t_1, t_2, ..., t_N) = \sum_{t_j < t_i} f(t_i|t_j) \times \prod_{j \neq k, t_k \leq t_i} S(t_i|t_k) \tag{5.5}$$

and therefore the likelihood of the infection in a cascade is

$$f(\mathcal{T}) = \prod_{t_i \leq T_{\max}} \sum_{t_j < t_i} f(t_i|t_j) \times \prod_{k \neq j, t_k \leq t_i} S(t_i|t_k) \tag{5.6}$$

In the next section, we have three Pareto modelings for one source diffusion network and in Section IV, we consider two sources modeling based on Pareto distributions.

4 One Source Diffusion Modeling

With one source diffusion network, we first consider a simple Pareto model in which every node in the network has the same dissemination rate α. Then extend models to deal with different dissemination rate on each parent node and with different infected rate on each child node.

4.1 Same infection rate α for all nodes

We employ the Pareto distribution to model the diffusion process with same infection rate α, which is the scale parameter of the Pareto distribution and is needed to be estimated.

The another scale parameter of the Pareto distribution can naturally be interpreted as onset time of infection of the parent node. If one node is infected by node j, the inflection time of the node follows a Pareto distribution with parameters t_j and α. In other words, its density function is of form

$$f(t|t_j,\alpha) = \frac{\alpha t_j^{\alpha}}{t^{\alpha+1}}, \qquad \text{for } t > t_j,$$

where t_j is the infection time of node j. The condition of $t_j < t$ means that the infection time of a parent node must be earlier than the infection time of a child node. On the other hand, if node k is not responsible for the infection of a node, i.e, the node is not infected by node k, then its survival function is modeled as

$$S(t|t_k,\alpha) = \left(\frac{t_k}{t}\right)^{\alpha}, \quad \text{for } t \geq t_k,$$

where t_k is the infection time of node k. With this modeling, we first consider an ordered cascade data $\mathcal{T} : t_1 \leq t_2 \leq t_3 \leq \cdots \leq t_N \leq T_{\max}$, in which all nodes are infected before $T_{\max}$. The cases with uninfected nodes can be easily extended later.

We can obtain the following likelihood function (5.5) for the i^{th} $(i > 1)$ node.

$$\begin{aligned} f(t_i|t_1,t_2,...,t_{i-1}) &= \sum_{j=1}^{i-1}\left[f(t_i|t_j,\alpha) \times \prod_{k=1,k\neq j}^{i-1} S(t_i|t_k,\alpha)\right] \\ &= \sum_{j=1}^{i-1}\left[\frac{\alpha t_j^{\alpha}}{t_i^{\alpha+1}} \prod_{k=1,k\neq j}^{i-1} \frac{t_k^{\alpha}}{t_i^{\alpha}}\right] \\ &= \frac{(i-1)\alpha}{t_i}\left(\frac{t_1}{t_i}\frac{t_2}{t_i}\cdots\frac{t_{i-1}}{t_i}\right)^{\alpha}. \end{aligned}$$

Then the likelihood function of (5.6) can be written as

$$f(\mathcal{T}|\alpha) = \prod_{i=2}^{N} f(t_i|t_1, ..., t_{i-1}; \alpha)$$
$$= (\prod_{i=2}^{N} \frac{i-1}{t_i})\alpha^{N-1}(\frac{t_1}{t_N})^{(N-1)\alpha}(\frac{t_2}{t_{N-1}})^{(N-2)\alpha}...(\frac{t_{N-1}}{t_2})^{\alpha}.$$

Taking logarithm, the log-likelihood is

$$\begin{aligned} l &= \log f(\mathcal{T}|\alpha) \\ &= (N-1)\log\alpha + \alpha\sum_{k=1}^{N-1}(N-k)\log\frac{t_k}{t_{N-k+1}} + c, \end{aligned} \tag{5.7}$$

where c is a term free of parameter α. Differentiating l with respect to α and solving $dl/d\alpha = 0$ gives the maximum likelihood estimator of α as follows.

$$\hat{\alpha} = \frac{N-1}{-\sum_{k=1}^{N-1}(N-k)\log\frac{t_k}{t_{N-k+1}}}. \tag{5.8}$$

proposition $\hat{\alpha} > 0$ in (5.8) is optimal which gives the maximum value of l.

Proof: $\hat{\alpha} > 0$ follows directly from

$$\sum_{k=1}^{N-1}(N-k)\log\frac{t_k}{t_{N-k+1}} = \sum_{i=2}^{N}\sum_{j=1}^{i-1}\log\frac{t_j}{t_i}$$

with each term $\log(t_j/t_i) < 0$. The second derivative of l,

$$\frac{d^2l}{d\alpha^2} = \frac{-(N-1)}{\alpha^2} < 0,$$

implies that $\hat{\alpha}$ is the maximizer of l. □

The above cascade likelihood argument can be easily extended to include uninfected nodes which survive at $T_{\max}$. Suppose that there are $N-K$ infected nodes and K uninfected nodes in the cascade data. Then the survival log-likelihood term for those uninfected nodes, $K \log S(t_{\max}|t_1, ..., t_{N-K})$, is

$$K\alpha \sum_{k=1}^{N-K} \log \frac{t_k}{T_{\max}}.$$

The derivation of the optimal α in this case follows similarly and yields

$$\hat{\alpha} = \frac{N-K-1}{-\sum_{k=1}^{N-K} \left[(N-K-k) \log \frac{t_k}{t_{N-K-k+1}} + K \log \frac{t_k}{T_{\max}}\right]}.$$

Due to this easy extension, we only consider cascades with all nodes being infected later on. Assume a set of C independent cascades $\mathcal{C} = \{\mathcal{T}^{(1)}, \mathcal{T}^{(2)}, \cdots, \mathcal{T}^{(C)}\}$ with $\mathcal{T}^{(c)} = \{t_1^{(c)}, t_2^{(c)}, \cdots, t_N^{(c)}\}$. The log-likelihood of $\mathcal{C}$ is the sum of the log-likelihoods of the individual cascade given as following

$$\sum_{c=1}^{C} \log f(\mathcal{T}^{(c)}|\alpha),$$

and the maximum likelihood estimator of α is

$$\hat{\alpha} = \frac{C(N-1)}{-\sum_{c=1}^{C} \sum_{k=1}^{N-1} (N-k) \log \frac{t_k^{(c)}}{t_{N-k+1}^{(c)}}}. \tag{5.9}$$

We have obtained estimator of the transition rate of the network. The modeling is mathematically convenient and easy to interpret. However, a common infection rate for all nodes in the network may be too restrictive. For example, some diseases may have different infection rates at the different periods after the first burst. Individuals in a network may disseminate information at different rates. A more realistic modeling shall have different α_i for each parent node or for each child node.

4.2 Different α_j for each sender

In this model, instead of having the same infection rate α for each node, it allows α_j for each sender which encodes the infection ability of each parent node j. A large α_j of node j means its higher risk of infecting others at the onset after being infected and the risk subsides substantially after that. A smaller α_j means node j possesses a longer duration to infect others. Let $\alpha = (\alpha_1, ..., \alpha_{N-1})$. The likelihood of the whole cascade of (5.6) is derived as follows

$$\begin{aligned} f(\mathcal{T}|\alpha) &= \prod_{i=2}^{N} \left[\sum_{j=1}^{i-1} f(t_i|t_j, \alpha_j) \times \prod_{j \neq k, k=1}^{i-1} S(t_i|t_k, \alpha_k) \right] \\ &= \frac{\alpha_1 t_1^{\alpha_1}}{t_2^{\alpha_1+1}} \times \left(\frac{\alpha_2 t_2^{\alpha_2}}{t_3^{\alpha_2+1}} \frac{t_1^{\alpha_1}}{t_3^{\alpha_1}} + \frac{\alpha_1 t_1^{\alpha_1}}{t_3^{\alpha_1+1}} \frac{t_2^{\alpha_2}}{t_3^{\alpha_2}} \right) \times \Bigg(\frac{\alpha_3 t_3^{\alpha_3}}{t_4^{\alpha_3+1}} \frac{t_1^{\alpha_1}}{t_4^{\alpha_1}} \frac{t_2^{\alpha_2}}{t_4^{\alpha_2}} \\ &+ \frac{\alpha_2 t_2^{\alpha_2}}{t_4^{\alpha_2+1}} \frac{t_3^{\alpha_3}}{t_4^{\alpha_3}} \frac{t_1^{\alpha_1}}{t_4^{\alpha_1}} + \frac{\alpha_1 t_1^{\alpha_1}}{t_4^{\alpha_1+1}} \frac{t_2^{\alpha_2}}{t_4^{\alpha_2}} \frac{t_3^{\alpha_3}}{t_4^{\alpha_3}} \Bigg) \times \cdots \\ &= \prod_{i=2}^{N} \left[\frac{\left(\sum_{j=1}^{i-1} \alpha_j \right)}{t_i} \prod_{j=1}^{i-1} \left(\frac{t_j}{t_i} \right)^{\alpha_j} \right]. \end{aligned} \tag{5.10}$$

Taking logarithm $l = \log f(\mathcal{T}|\alpha)$ and derivative with respect to each α_i, we have

$$\frac{\partial l}{\partial \alpha_1} = \sum_{i=2}^{N} \left[\frac{1}{\sum_{k=1}^{i-1} \alpha_k} + \log\left(\frac{t_1}{t_i} \right) \right] \tag{5.11}$$

$$\frac{\partial l}{\partial \alpha_2} = \sum_{i=3}^{N} \left[\frac{1}{\sum_{k=1}^{i-1} \alpha_k} + \log\left(\frac{t_1}{t_i} \right) \right] \tag{5.12}$$

$$\frac{\partial l}{\partial \alpha_3} = \sum_{i=4}^{N} \left[\frac{1}{\sum_{k=1}^{i-1} \alpha_k} + \log\left(\frac{t_1}{t_i} \right) \right] \tag{5.13}$$

Setting (5.11) and (5.12) to be zero, we obtain the estimator of α_1

$$\hat{\alpha}_1 = \frac{1}{(N-1)(\log t_2 - \log t_1)}.$$

With (5.12 and (5.13) being zero, we get the following derivation of α_2

$$\hat{\alpha}_2 = \frac{1}{(N-2)(\log t_3 - \log t_2)} - \frac{1}{(N-1)(\log t_2 - \log t_1)}.$$

In general, we have

$$\frac{\partial l}{\partial \alpha_j} = \sum_{i=j+1}^{N} \left[\frac{1}{\sum_{k=1}^{i-1} \alpha_k} + \log\left(\frac{t_1}{t_i}\right) \right] \tag{5.14}$$

and for $j = 2, \cdots, N-1$

$$\hat{\alpha}_j = \frac{1}{(N-j)\log(t_{j+1}/t_j)} - \frac{1}{(N+1-j)\log(t_j/t_{j-1})}.$$

Note that except for α_1, the estimator of α_j is determined by three infection times at t_{j-1}, t_j and t_{j+1}. To ensure positive $\hat{\alpha}_j$, the condition

$$t_j^{2N-2j+1} > t_{j+1}^{N-j} t_{j-1}^{N+1-j} \tag{5.15}$$

must hold. If the cascade data satisfy this condition (5.15) for all consecutive three infection time periods, this modeling is useful and mathematically sound. However, this condition may not be satisfied for some cascade data in which the inference of this model is invalid and the maximum likelihood estimator does not exist. In such cases, we would like to model a different infection rate for each child node.

4.3 Different α_i for each receiver

In this model, we assign α_i for each receiver which encodes the susceptibility of each node. A large α_i for node i means node i has a much higher chance of getting infected at the beginning than later. A smaller α_i for node i means node i is subject to infection for a longer period of time. Let $\alpha = (\alpha_2, ..., \alpha_N)$. Then the likelihood of the whole cascade is derived as follow:

$$
\begin{aligned}
f(\mathcal{T}|\alpha) &= \prod_{i=2}^{N}\left[\sum_{j=1}^{i-1} f(t_i|t_j,\alpha_i) \times \prod_{k\neq j;k=1}^{i-1} S(t_i|t_k,\alpha_i)\right] \\
&= \frac{\alpha_2 t_1^{\alpha_2}}{t_2^{\alpha_2+1}} \times \left(\frac{\alpha_3 t_2^{\alpha_3}}{t_3^{\alpha_3+1}}\frac{t_1^{\alpha_3}}{t_3^{\alpha_3}} + \frac{\alpha_3 t_1^{\alpha_3}}{t_3^{\alpha_3+1}}\frac{t_2^{\alpha_3}}{t_3^{\alpha_3}}\right) \times \left(\frac{\alpha_4 t_3^{\alpha_4}}{t_4^{\alpha_4+1}}\frac{t_1^{\alpha_4}}{t_4^{\alpha_4}}\frac{t_2^{\alpha_4}}{t_4^{\alpha_4}}\right. \\
&\left. + \frac{\alpha_4 t_2^{\alpha_4}}{t_4^{\alpha_4+1}}\frac{t_3^{\alpha_4}}{t_4^{\alpha_4}}\frac{t_1^{\alpha_4}}{t_4^{\alpha_4}} + \frac{\alpha_4 t_1^{\alpha_4}}{t_4^{\alpha_4+1}}\frac{t_2^{\alpha_4}}{t_4^{\alpha_4}}\frac{t_3^{\alpha_4}}{t_4^{\alpha_4}}\right) \times \cdots \\
&= \prod_{i=2}^{N}\frac{(i-1)\alpha_i}{t_i}\prod_{j=1}^{i-1}\left(\frac{t_j}{t_i}\right)^{\alpha_i}.
\end{aligned}
$$

Take the derivative of the log-likelihood $l = \log f(\mathcal{T}|\alpha)$ to be zero, we have the solution

$$
\begin{aligned}
\frac{\partial l}{\partial \alpha_2} &= \frac{1}{\alpha_2} + \log t_1 - \log t_2 := 0 \Rightarrow \hat{\alpha}_2 = \left(\log\frac{t_2}{t_1}\right)^{-1} \\
\frac{\partial l}{\partial \alpha_3} &= \frac{1}{\alpha_3} + \log(t_1 t_2) - 2\log t_3 := 0 \Rightarrow \hat{\alpha}_3 = \left(\log\frac{t_3^2}{t_1 t_2}\right)^{-1}
\end{aligned}
$$

Continuing the calculation, we obtain an estimator of α_i which gives the maximum likelihood of l.

$$
\hat{\alpha}_i = \left[\sum_{j=1}^{i-1}\log\left(\frac{t_i}{t_j}\right)\right]^{-1} > 0. \tag{5.16}
$$

proposition The estimator $\hat{\alpha}_i$ in (5.16) is optimal which gives the maximum value of l.

Proof: We have to calculate the Hessian matrix H as follows.

$$\begin{aligned}
H &= \left(\frac{\partial^2 l}{\partial\alpha_i\partial\alpha_j}\right)_{i,j=2}^{N} \\
&= \begin{pmatrix} \frac{\partial^2 l}{\partial\alpha_2^2} & \frac{\partial^2 l}{\partial\alpha_2\alpha_3} & \cdots & \frac{\partial^2 l}{\partial\alpha_2\alpha_N} \\ \frac{\partial^2 l}{\partial\alpha_3\alpha_2} & \frac{\partial^2 l}{\partial\alpha_3^2} & \cdots & \frac{\partial^2 l}{\partial\alpha_3\alpha_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 l}{\partial\alpha_N\alpha_2} & \frac{\partial^2 l}{\partial\alpha_N\alpha_3} & \cdots & \frac{\partial^2 l}{\partial\alpha_N^2} \end{pmatrix} \\
&= \begin{pmatrix} -\alpha_2^{-2} & 0 & \cdots & 0 \\ 0 & -\alpha_3^{-2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -\alpha_N^{-2} \end{pmatrix}.
\end{aligned}$$

It is invertible and negative definite obviously. This proves that $\hat{\alpha}$ maximizes l. □

Note that $\hat{\alpha}_i$ is determined by log ratios of t_i and t_j for $j = 1, 2, \cdots, i-1$. This makes sense since the i^{th} infected node can be infected from any of the first $i-1$ infected nodes and its infected rate is determined by the infection times of its parent nodes.

5 Multiple Source Modeling

In this section, we extend our model to the case where cascade $\mathcal{T}$ is contributed by two sources and further to multiple source cascades. We want to model the circumstance that there are two or more sources of pathogens which cause the infection among the population. We first consider two cases about two source cascades. One is that the cascade contributed by each source does not overlap with each other. Therefore the problem is translated to the identification of the starting time of the second source. The other is to deal with a a partially overlapped cascade from two sources. We use a mixture distribution to model this case and propose an EM algorithm to obtain the estimators. Later we consider the case that has fully overlapping cascade from multiple sources.

5.1 Non-overlap two source cascade modeling

This case depicts a scenario in which one source starts the dissemination process and begin to infect other nodes. All the nodes infected by the source cease to infect others before the second source starts the dissemination process. Therefore the whole cascade $\mathcal{T} = \{t_1, t_2, \cdots, t_N\}$ consists of two sub-cascades $\mathcal{T}_1 = \{t_1, t_2, \cdots, t_{K-1}\}$ and $\mathcal{T}_2 = \{t_K, t_2, \cdots, t_N\}$ which are contributed by source 1 and source 2 respectively. Let α and β be the infection rate of source 1 and source 2 respectively. When a node is infected by node j within $\mathcal{T}_1$, the probability of infection time t ($t > t_j$) is denoted as

$$f(t|t_j, \alpha) = \frac{\alpha t_j^{\alpha}}{t^{\alpha+1}}.$$

Similarly, when a node is infected by node m within $\mathcal{T}_2$, the probability of infection is denoted as

$$f(t|t_m, \beta) = \frac{\beta t_m^{\beta}}{t^{\beta+1}}.$$

Collectively, the likelihood of the whole cascade $\mathcal{T}$ is derived as follow:

$$\begin{aligned} f(\mathcal{T}|\alpha, \beta) &= \prod_{i=2}^{K-1} \left(\sum_{j=1}^{i-1} f(t_i|t_j, \alpha) \times \prod_{k=1, k\neq j}^{i-1} S(t_i|t_k, \alpha) \right) \\ &\times \prod_{i=K+1}^{N} \left(\sum_{j=K}^{i-1} f(t_i|t_j, \beta) \times \prod_{k=K, k\neq j}^{i-1} S(t_i|t_k, \beta) \right) \\ &= (\prod_{i=2}^{K-1} \frac{i-1}{t_i}) \alpha^{K-2} (\frac{t_1}{t_{K-1}})^{(K-2)\alpha} (\frac{t_2}{t_{K-2}})^{(K-3)\alpha} ... (\frac{t_{K-2}}{t_2})^{\alpha} \\ &(\prod_{i=K+1}^{N} \frac{i-K}{t_i}) \beta^{M-1} (\frac{t_K}{t_N})^{(M-1)\beta} (\frac{t_{K+1}}{t_{N-1}})^{(M-2)\beta} ... (\frac{t_{N-1}}{t_{K+1}})^{\beta}, \end{aligned} \tag{5.17}$$

where $M = N - K$. Taking logarithm, we have

$$\begin{aligned} l(\alpha, \beta, K|\mathcal{T}) &= \log f(\mathcal{T}|\alpha, \beta) \\ &= \log f(\mathcal{T}_1|\alpha) + \log f(\mathcal{T}_2|\beta) \\ &:= l_1(1, K-1, \alpha) + l_2(K, N, \beta). \end{aligned}$$

Given K, solving $\partial l/\partial\alpha = 0$ yields

$$\hat{\alpha}(K) = -\frac{K-2}{\sum_{i=1}^{K-2}(K-1-i)\log\frac{t_i}{t_{K-i}}} \tag{5.18}$$

Similarly, solving $\partial l/\partial\beta = 0$ provides an estimator of β given K, that is,

$$\hat{\beta}(K) = -\frac{N-K-1}{\sum_{i=K}^{N-1}(N-1-i)\log\frac{t_i}{t_{N-i+K}}}. \tag{5.19}$$

It can be easily shown that $\hat{\alpha}(K)$ and $\hat{\beta}(K)$ are positive and maximize l_1 and l_2 respectively for each $K(1 < K < N)$. To determine the starting point of the second source, we choose the index K such that the likelihood of the cascade l is maximized. The algorithm to find the maximum likelihood estimator is summarized as follows.

Algorithm 4 Non-overlapping Algorithm

```
Input: $\mathcal{T} = \{t_1, t_2, \cdots, t_N\}$
Initialization: MaxL $= -\infty$
For $K = 2$ To $N - 1$
    1 Calculate $\hat{\alpha}$ using (5.18) ;
    2 Calculate $\hat{\beta}$ using (5.19);
    3 $l(K, \alpha, \beta) = l_1(1, K-1, \hat{\alpha}) * l_2(K, N, \hat{\beta})$;
    4 MaxL $= \max\{l(K, \alpha, \beta), \text{MaxL}\}$
End For
Output: $\hat{K}$, $\hat{\alpha}$ and $\hat{\beta}$ that maximizes MaxL.
```

5.2 Overlapping two source cascade modeling

This case depicts a scenario in which one source with infection rate α starts the dissemination process to infect nodes and then after a while the second source with infection rate β starts the dissemination process. Therefore the whole cascade $\mathcal{T} = \{t_1, t_2, \cdots, t_N\}$ can be divided into two sub-cascades namely $\mathcal{T}_1 = \{t_1, t_2, \cdots, t_{K-1}\}$ and $\mathcal{T}_2 = \{t_K, t_{K+1}, \cdots, t_N\}$ in which timestamps in $\mathcal{T}_1$ are contributed only by source 1 and timestamps in $\mathcal{T}_2$ are contributed by both source 1 and source 2. When node i $(i < K)$ within $\mathcal{T}_1$ is infected by node j, its density probability of infection time is denoted as

$$f(t|t_j, \alpha) = \frac{\alpha t_j^\alpha}{t^{\alpha+1}}.$$

The log-likelihood of $\mathcal{T}_1$ then can be derived similarly as before

$$\begin{aligned} l_1(K, \alpha) &= \log\left[\prod_{i=2}^{K-1} \sum_{j=1}^{i-1} f(t_i|t_j, \alpha) \times \prod_{k=1,k\neq j}^{i-1} S(t_i|t_k, \alpha)\right] \\ &= (K-2)\log\alpha + \sum_{i=2}^{K-1}\left[(K-i)\alpha\log\frac{t_{i-1}}{t_{K-i+1}} + \log\frac{i}{t_{i+1}}\right]. \end{aligned}$$

One the other hand, if node i $(i > K)$ within $\mathcal{T}_2$ is infected by node j, its density distribution of the infection time follows

$$\begin{aligned} f(t|t_j, \alpha, \beta, \pi) &= \pi f_1(t|t_j, \alpha) + (1-\pi) f_2(t|t_j, \beta) \\ &= \pi\frac{\alpha t_j^\alpha}{t^{\alpha+1}} + (1-\pi)\frac{\beta t_j^\beta}{t^{\beta+1}} \end{aligned}$$

where π is the probability that a node is infected due to source 1. Since we do not know which source infects node i, we employ an EM framework for the mixture model. The observed sub-cascade data $\mathcal{T}_2$ are viewed as incomplete. The complete sub-cascade data shall be $\mathcal{Z} = \{t_i, z_i\}_{i=K+1}^N$ where z_i is an "unobserved" indicator vector with $z_i = 1$ if node i is infected by source 1 and 0 otherwise. Let $\theta = (\alpha, \beta, \pi, K)$. The complete log-likelihood of

$\mathcal{Z}$ is then derived by

$$
\begin{aligned}
& l_2^c(\theta|\mathcal{Z}) \\
&= \sum_{i=K+1}^{N} z_i \log[\pi f_1(t_i|\theta)] + (1-z_i)\log[(1-\pi)f_2(t_i|\theta)] && (5.20)\\
&= \sum_{i=K+1}^{N} \log\left\{\sum_{j=K}^{i-1}\left[\pi f_1(t_i|t_j,\alpha)\prod_{k\neq j,k=K}^{i-1} S(t_i|t_k,\alpha)\right]^{z_i}\right. && (5.21)\\
&\left.\times\left[(1-\pi)f_2(t_i|t_j,\beta)\prod_{k\neq j,k=K}^{i-1} S(t_i|t_k,\beta)\right]^{1-z_i}\right\} && (5.22)\\
&= \sum_{i=K+1}^{N}\left[\log(\frac{i-K}{t_i}) + (1-z_i)\log((1-\pi)\beta) + \right.\\
& \left. z_i\log(\pi\alpha) + [z_i\alpha + (1-z_i)\beta]\sum_{j=K}^{i-1}\log\left(\frac{t_j}{t_i}\right)\right]
\end{aligned}
$$

The terms in (5.21) and (5.22) stand for the likelihood of node i being infected by $\mathcal{T}_2$ from source 1 and source 2, respectively. Hence the complete log-likelihood of the whole cascade is

$$l^c(\theta|\mathcal{T}) = l_1(K,\alpha) + l_2^c(\theta|\mathcal{Z}). \tag{5.23}$$

The EM algorithm obtains a sequence of estimates $\{\theta^{(s)}, s = 0, 1, 2, \cdots\}$ by alternating E step and M step until some convergence criterion is met. We first provide the EM algorithm for each K, then grid-search from 2 to $N-1$ for the optimal K to maximize $l^c(\theta|\mathcal{T})$.

E-step: Calculate Q function, the conditional expectation of the complete log-likelihood, given $\mathcal{T}$ and the current estimate $\theta^{(s)}$. For $i > K$, since z_i is either 1 or 0, $E(z_i|\theta^{(s)},\mathcal{T}) = Pr(z_i = 1|\theta^{(s)},\mathcal{T})$, which is denoted by $y_i^{(s)}$. By the Bayes rule, we have

$$y_i^{(s)} = \frac{\hat{\pi}^{(s)}L_1(t_i|\theta^{(s)})}{\hat{\pi}^{(s)}L_1(t_i|\theta^{(s)}) + (1-\hat{\pi}^{(s)})L_2(t_i|\theta^{(s)})}, \tag{5.24}$$

where

$$L_1(t_i|\theta^{(s)}) = \frac{(i-1)\alpha^{(s)}}{t_i}\prod_{j=1}^{i-1}(\frac{t_j}{t_i})^{\alpha^{(s)}}$$

and

$$L_2(t_i|\theta^{(s)}) = \frac{(i-K)\beta^{(s)}}{t_i}\prod_{m=K}^{i-1}(\frac{t_m}{t_i})^{\beta^{(s)}}.$$

$y_i^{(s)}$ can be interpreted as soft labels at the s^{th} iteration. Replacing z_i with $y_i^{(s)}$ in (5.23), we obtained $Q(\theta|\theta^{(t)})$.

M-step: Update the estimate of the parameters by maximizing the Q function, i.e,

$$\theta^{(t+1)} = \text{argmax}_\theta Q(\theta|\theta^{(s)}),$$

which yields the following updates:

$$\hat{\pi}^{(s+1)} = \frac{1}{N-K-1}\sum_{i=K+1}^{N} y_i^{(s)} \tag{5.25}$$

$$\hat{\alpha}^{(s+1)} = \frac{-[(K-2)+\sum_{i=K+1}^{N} y_i^{(s)}]}{\sum_{i=2}^{K-1}\sum_{j=1}^{i-1}\log\frac{t_j}{t_i}+\sum_{i=K+1}^{N} y_i^{(s)}\sum_{m=K}^{i-1}\log\frac{t_m}{t_i}} \tag{5.26}$$

$$\hat{\beta}^{(s+1)} = \frac{-\sum_{i=K+1}^{N}(1-y_i^{(s)})}{\sum_{i=K+1}^{N}(1-y_i^{(s)})\sum_{m=K}^{i-1}\log\frac{t_m}{t_i}} \tag{5.27}$$

For given K, we obtain an approximated maximum likelihood estimator $\theta(K)$ when iterations of E step and M step converge. We determine the index K such that the log-likelihood of the whole cascade is maximized and The algorithm to approximate the maximum likelihood estimator is summarized as follows. Although there are explicit solutions for each of updating E and M steps, Algorithm (5) may be inefficient since it involves two loops, which may be computationally expansive, especially when N is large. To overcome this limitation, we consider the completely overlapping multiple source cascades under the assumption that all sources have a roughly same start point.

Algorithm 5 Partially Overlapping Algorithm

```
Input: $\mathcal{T} = \{t_1, t_2, \cdots, t_N\}$, $\varepsilon$, maxit
Initialization: $\text{MaxL} = -\infty$
For $K = 2$ To $N - 1$
    Set: $\pi^{(1)} = 0.5$, $\pi^{(0)} = 0$, $s = 1$, $\alpha^{(1)} = \beta^{(1)} = 1$
    While ($|\pi^{(s)} - \pi^{(s-1)}| > \varepsilon$ and $s <$ maxit) Do
        1 Update $\pi^{(s+1)}$ by (5.24) and (5.25);
        2 Update $\alpha$ and $\beta$ using (5.26) and (5.27);
        3 $s = s + 1$;
    End While
    1 Compute $l^c(\hat{\theta}|\mathcal{T}) = l_1(K, \hat{\alpha}) + l_2^c(\hat{\theta}|\mathcal{Z})$;
    2 $\text{MaxL} = \max\{l^c(\hat{\theta}|\mathcal{T}), \text{MaxL}\}$
End For
Output: $\hat{K}$, $\hat{\pi}$, $\hat{\alpha}$ and $\hat{\beta}$ that maximizes MaxL.
```

5.3 Fully Overlapping Multiple Source Modeling

Suppose that an incident occurs and is broadcasted by D different social media immediately. Each media has its own rate to disseminate information in the diffusion network. We only know the diffusion traces, without any knowledge on the network structure and source of information. This is a case of fully overlapping multiple source cascade problem and can be dealt with a mixture model and EM algorithm. Assume that source d has dissemination rate α_d for $d = 1, 2, \cdots, D$. Denote the probability of a node infected by source d as π_d. Let $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_D)^T$ and $\pi = (\pi_1, \pi_2, \cdots, \pi_D)^T$. Similarly as before, the infection time of a node infected by node j follows a mixture distribution with the density function

$$f(t|t_j, \pi, \alpha) = \sum_{d=1}^{D} \pi_d \frac{\alpha_d}{t} \left(\frac{t_j}{t}\right)^{\alpha_d} \tag{5.28}$$

Treat t_1 as the starting time and each of the following t_i associated with an “unobserved” indicator vector $z_i = (z_{i1}, z_{i2}, \cdots, z_{iD})^T$, where $z_{id} = 1$ if node i is infected by source d and

0 otherwise. Hence the complete log-likelihood of the cascade $\mathcal{T}$ is

$$
\begin{aligned}
& l^c(\alpha, \pi|\mathcal{T}) \\
&= \sum_{i=2}^{N} \log \left\{ \sum_{j=1}^{i-1} \prod_{d=1}^{D} \left[\pi_d f_d(t_i|t_j, \alpha_d) \prod_{k \neq j, k=1}^{i-1} S(t_i|t_k, \alpha_d) \right]^{z_{id}} \right\} \\
&= \sum_{d=1}^{D} \left[\log(\alpha_d) \left(\sum_{i=2}^{N} z_{id} \right) + \alpha_d \left(\sum_{i=2}^{N} z_{id} \right) \sum_{j=1}^{i-1} \log \left(\frac{t_j}{t_i} \right) \right]
\end{aligned}
$$

Working out the E step and M step, we have the following updates for $d = 1, \cdots, D$:

$$
\begin{aligned}
y_{id}^{(s)} &= \frac{\pi_d^{(s)} L_d(t_i|\alpha_d^{(s)})}{\sum_{k=1}^{D} \pi_k^{(s)} L_k(t_i|\alpha_k^{(s)})} \\
\pi_d^{(s+1)} &= \frac{1}{N-1} \sum_{i=2}^{N} y_{id}^{(s)} \\
\alpha_d^{(s+1)} &= \frac{\sum_{i=2}^{N} y_{id}^{(s)}}{\sum_{i=2}^{N} y_{id}^{(s)} \sum_{j=1}^{i-1} \log(t_j/t_i)},
\end{aligned}
$$

where $L_d(t_i|\alpha_d^{(s)}) = \frac{(i-1)\alpha_d^{(s)}}{t_i} \prod_{j=1}^{i-1} (\frac{t_j}{t_i})^{\alpha^{(s)}}$. The procedure continues until it converges.

The next question arises naturally: how to determine D, the number of sources. For $D = 1$, this model reduces to the one source modeling with a common infection rate, while for $D = N - 1$, this model is equivalent to the one with different infection rate for each receiver. Hence this question is also to be asked in another way: how to choose model?

5.4 Experiments

5.4.1 Simulation

We generate cascades data to mimic the diffusion process. To construct the ground truth model for our analysis, we generate a cascade of N timestamps based various values of infection rate α and produce the infection time of each node accordingly. We fix a α value and t_1 value. For each node i ($i = 2, \cdots, N$), we randomly select its parent node m from its parent list $\{t_1, t_2, \cdots, t_{i-1}\}$ with the probability of

$$\frac{(t_1 t_2 \cdots t_{m-1} t_{m+1} \cdots t_{i-1})^\alpha}{\sum_{k=1}^{i-1} (t_1 t_2 \cdots t_{k-1} t_{k+1} \cdots t_{i-1})^\alpha}.$$

Once its parent node m is chosen, we generate the timestamp t_i by sampling from the Pareto distribution with the starting point being t_m and tail index α. We repeat the generation process for C times. Upon generating all the t_i, we applied the equation (5.9) to obtain $\hat{\alpha}$. Then we compute the normalized mean absolute error (MAE) as an assessment criterion. The normalized MAE is defined as

$$\text{MAE} = \left| \frac{\alpha - \alpha^*}{\alpha} \right|,$$

where α is the true infection rate from the ground truth model whereas α^* is the averaged value of 100 infection rate estimates ($\hat{\alpha}$). We examine the effects of the normalized MAE's of the estimator on different values of α, on different values of N and on different values of C We observed that utilizing more cascades leads to more accurate estimate of the normalized MAE and the error rate can be bought down to around 20% when the number of cascades reach around 1000 and Figure (5.2) shows the result.

5.4.2 Comparison with NETRATE

We compare our model with the widely used NETRATE model [77]. Since our approach and NETRATE are based on likelihoods, it is natural to select model based on some criteria with the common form of log-likelihood augmented by a model complexity penalty term. For example, Akaike information criterion (AIC), Bayesian information criterion (BIC), the normalized entropy criterion (NEC) etc. have yielded good results for model choice in a range of applications. Here, we use BIC for model selection and comparison. BIC is defined as negative twice of the the log-likelihood plus $p \log p$, where p is the number of independent parameters. That is, $BIC = -2l + p \log(N)$. A model with a smaller BIC

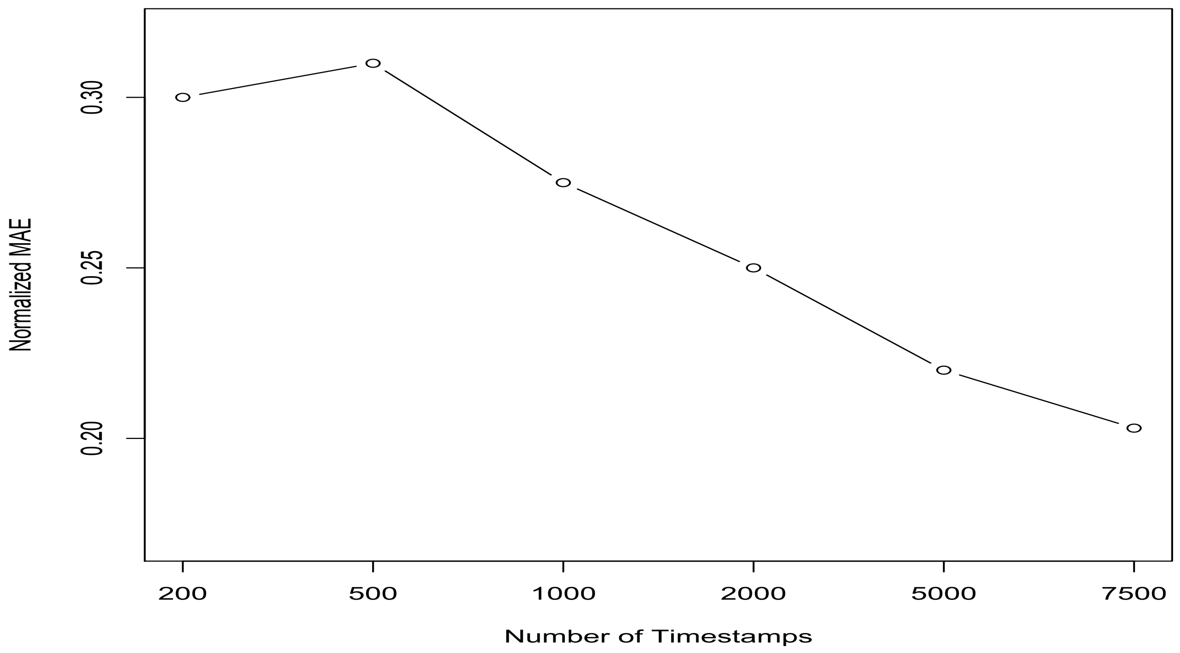

Figure 5.1. Normalized MAE vs the cascade size

Size	NETRATE	Our Model
200	-58.945	-247.652
400	-376.184	-606.475
600	-776.347	-1538.85

Table 5.1. BIC of our model comparing with NETRATE. Smaller BIC implies a better modeling.

is preferred. Table (5.1) lists BIC values of our model and NETRATE in the cascade data generated in the same way as previously described. Our model has a much smaller BIC than NETRATE for all cases. The results can be explained by the difference of the parameter number in two approaches. Our model is much simpler than theirs and the model complexity penalty in our model is much smaller than that in their model. As a result, our model has a better generalization performance than theirs also has a better interpretation than theirs.

5.4.3 Twitter data application

We obtained real cascade data from Twitter. By using the Tweepy API of Python, we

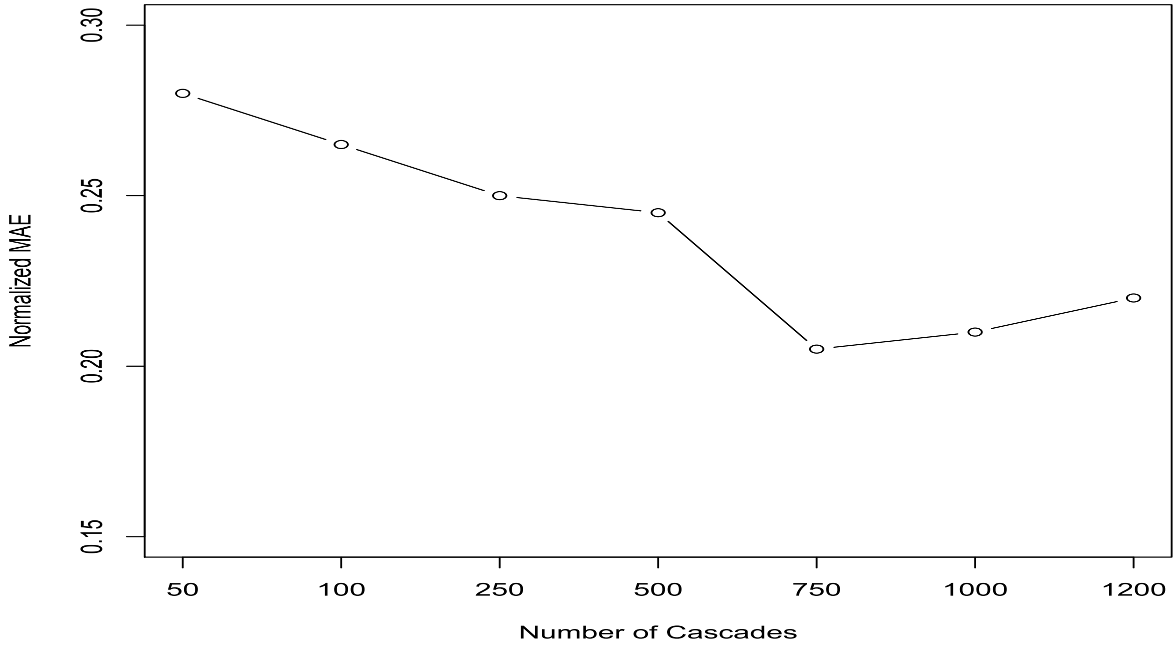

Figure 5.2. Normalized MAE with respects to numbers of cascades

were able to compile three sets of data on February 14, 2017 from Donald Trump's Twitter profile and Senator Bernie Sanders' Twitter profile. The first data set tracked the tweet from Trump: "Obamacare continues to fail. Humana to pull out in 2018. Will repeal, replace & save healthcare for ALL Americans." The second data set tracked the tweet from Trump: "The real story here is why are there so many illegal leaks coming out of Washington? Will these leaks be happening as I deal on N.Korea etc?" The third data set tracked the tweet from Sanders: "Talk about cowardice. Republicans are trying to ram through Pruitt's confirmation before the American people find out what is in his emails." We extract the timestamps of each cascade and calculate $\hat{\alpha}$. The result is tabulated in Table (5.2). As shown in the Table (5.2), the first and the second tweets from Trump have a higher $\hat{\alpha}$ value than the third tweet from Sanders. That implies that Trump's messages are more easily

Tweet	# of Timestamps	$\hat{\alpha}$
Trump (Obamacare)	1560	3.083
Trump (Illegal Leak)	2272	2.789
Sanders (Cowardice)	1102	1.382

Table 5.2. Estimated α on real twitter data

disseminated in a very short burst of time compared to Sanders.

6 Conclusion

We have developed a flexible model structure underlying diffusion processes that assume the infection time following the Pareto power-law. This modeling not only provides intuitive interpretation but also brings in mathematical and computational ease. It infers transmission rates between nodes of a network by computing a model which maximizes time dependent pairwise transmission likelihood between all pairs of nodes. We present three different modelings to account for different transmission rates and infection rates of each node. Experiments on real and synthetic data show that our models accurately estimate their transmission rates. Moreover, our model has a advantage compared to the widely used NETRATE [77] model due to its simplicity. It usually produces a much smaller BIC than NETRATE, which indicates our model is simpler and fits data better.

CHAPTER 6

FUTURE WORKS

Nowadays, deep learning methods have achieved state-of-the-art accuracy on many prediction tasks such as image classification . A deep learning model automatically learns complex functions which map inputs to outputs. The advantage of it is it can eliminate the necessity to use hand-crafted features or rules. One version of deep learning is called Convolutional Neural Networks (CNNs), which capture both local and global representations in the input samples to learn the most crucial features which help make better predictions. CNNs have been used successfully in computer vision; natural language processing and bioinformatics.

1 Deep Learning on Image and Bio-medical data for cancer classification

Inspired by the above mentioned success of using deep convolutional network, we propose to train deep multi-instance networks for cancer classification and predictions using both image data like mammogram and other high throughput biomedical data like gene expression, methylation etc. The proposed deep architecture should have multiple convolutional layers, one linear regression layer, one ranking layer, and one loss layer.

2 Deep Learning on understanding gene regulation and histone chromatin mark

Histone modifications are among the most important factors which control gene regulation in epigenetics. These chromatin marks are typically high-dimensional and highly structured and our prime objective is to understand what the relevant factors or marks are and how they interact and work together. I propose to use deep learning to model the complex dependencies among input signals. I propose to use Long Short-Term Memory

(LSTM) modules to encode the input signals and to model how various chromatin marks work together to control the gene expression.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Adar, E., and L. A. Adamic (2005), Tracking information epidemics in blogspace, in *Proceedings of the 2005 IEEE/WIC/ACM international conference on web intelligence*, pp. 207–214, IEEE Computer Society.

[2] Aerts, S., et al. (2006), Gene prioritization through genomic data fusion, *Nature biotechnology*, 24(5), 537–544.

[3] Alarmo, E.-L., T. Kuukasjärvi, R. Karhu, and A. Kallioniemi (2007), A comprehensive expression survey of bone morphogenetic proteins in breast cancer highlights the importance of bmp4 and bmp7, *Breast cancer research and treatment*, 103(2), 239–246.

[4] Anzick, S. L., et al. (1997), Aib1, a steroid receptor coactivator amplified in breast and ovarian cancer, *Science*, 277(5328), 965–968.

[5] Bahrami, A., et al. (2017), Targeting the ras signaling pathway as a potential therapeutic target in the treatment of colorectal cancer, *Journal of cellular physiology*.

[6] Baldwin, R. L., H. Tran, and B. Y. Karlan (1999), Primary ovarian cancer cultures are resistant to fas-mediated apoptosis, *Gynecologic oncology*, 74(2), 265–271.

[7] Barabási, A.-L., and R. Albert (1999), Emergence of scaling in random networks, *science*, 286(5439), 509–512.

[8] Besag, J. (1974), Spatial interaction and the statistical analysis of lattice systems, *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 192–236.

[9] Boyd, S., N. Parikh, E. Chu, B. Peleato, and J. Eckstein (2011), Distributed optimization and statistical learning via the alternating direction method of multipliers, *Foundations and Trends® in Machine Learning*, 3(1), 1–122.

[10] BRCA, S. G. (1994), A strong candidate for the breast and ovarian cancer susceptibility gene brca1, *Science*, 266, 7.

[11] Brennan, C. W., et al. (2013), The somatic genomic landscape of glioblastoma, *Cell*, 155(2), 462–477.

[12] Brockmann, D., L. Hufnagel, and T. Geisel (2006), The scaling laws of human travel, *Nature*, 439(7075), 462–465.

[13] Bulstrode, H., L. M. Jones, E. J. Siney, J. M. Sampson, A. Ludwig, W. P. Gray, and S. Willaime-Morawek (2012), A-disintegrin and metalloprotease (adam) 10 and 17 promote self-renewal of brain tumor sphere forming cells, *Cancer letters*, 326(1), 79–87.

[14] Calon, A., et al. (2012), Dependency of colorectal cancer on a tgf-β-driven program in stromal cells for metastasis initiation, *Cancer cell*, 22(5), 571–584.

[15] Cao, M., C. M. Pietras, X. Feng, K. J. Doroschak, T. Schaffner, J. Park, H. Zhang, L. J. Cowen, and B. J. Hescott (2014), New directions for diffusion-based network prediction of protein function: incorporating pathways with confidence, *Bioinformatics*, 30(12), i219–i227.

[16] Chen, J., E. E. Bardes, B. J. Aronow, and A. G. Jegga (2009), Toppgene suite for gene list enrichment analysis and candidate gene prioritization, *Nucleic acids research*, 37(suppl_2), W305–W311.

[17] Chen, S., D. M. Witten, and A. Shojaie (2014), Selection and estimation for mixed graphical models, *Biometrika*, 102(1), 47–64.

[18] Chickering, D., D. Geiger, and D. Heckerman (1995), Learning bayesian networks: Search methods and experimental results, in *proceedings of fifth conference on artificial intelligence and statistics*, pp. 112–128.

[19] Chun, H., M. Chen, B. Li, and H. Zhao (2013), Joint conditional gaussian graphical models with multiple sources of genomic data, *Frontiers in genetics*, 4.

[20] Cizkova, M., A. Susini, S. Vacher, G. Cizeron-Clairac, C. Andrieu, K. Driouch, E. Fourme, R. Lidereau, and I. Bièche (2012), Pik3ca mutation impact on survival in breast cancer patients and in erα, pr and erbb2-based subgroups, *Breast Cancer Research*, 14(1), R28.

[21] Clauset, A., C. Moore, and M. E. Newman (2008), Hierarchical structure and the prediction of missing links in networks, *Nature*, 453(7191), 98–101.

[22] Danaher, P., P. Wang, and D. M. Witten (2014), The joint graphical lasso for inverse covariance estimation across multiple classes, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(2), 373–397.

[23] Daneshmand, H., M. Gomez-Rodriguez, L. Song, and B. Schoelkopf (2014), Estimating diffusion network structures: Recovery conditions, sample complexity & soft-thresholding algorithm., in *ICML*, pp. 793–801.

[24] Deng, M., T. Chen, and F. Sun (2004), An integrated probabilistic model for functional prediction of proteins, *Journal of Computational Biology*, 11(2-3), 463–475.

[25] Dhillon, V. S., M. Shahid, and S. A. Husain (2004), Cpg methylation of the fhit, fancf, cyclin-d2, brca2 and runx3 genes in granulosa cell tumors (gcts) of ovarian origin, *Molecular cancer*, 3(1), 33.

[26] Dobra, A., A. Lenkoski, et al. (2011), Copula gaussian graphical models and their application to modeling functional disability data, *The Annals of Applied Statistics*, 5(2A), 969–993.

[27] Erten, S., G. Bebek, and M. Koyutürk (2011), Vavien: an algorithm for prioritizing candidate disease genes based on topological similarity of proteins in interaction networks, *Journal of computational biology*, 18(11), 1561–1574.

[28] Fadare, O., and D. Khabele (2014), Md molecular profiling of epithelial ovarian cancer.

[29] Farajtabar, M., M. Gomez-Rodriguez, M. Zamani, N. Du, H. Zha, and L. Song (2015), Back to the past: Source identification in diffusion networks from partially observed cascades., in *AISTATS*.

[30] Ford, D., et al. (1998), Genetic heterogeneity and penetrance analysis of the brca1 and brca2 genes in breast cancer families, *The American Journal of Human Genetics*, 62(3), 676–689.

[31] Friedman, J., T. Hastie, and R. Tibshirani (2008), Sparse inverse covariance estimation with the graphical lasso, *Biostatistics*, 9(3), 432–441.

[32] Gagné, J.-P., et al. (2005), Proteome profiling of human epithelial ovarian cancer cell line tov-112d, *Molecular and cellular biochemistry*, 275(1), 25–55.

[33] George, R. A., J. Y. Liu, L. L. Feng, R. J. Bryson-Richardson, D. Fatkin, and M. A. Wouters (2006), Analysis of protein sequence and interaction data for candidate disease gene prediction, *Nucleic acids research*, 34(19), e130–e130.

[34] Glazier, A. M., J. H. Nadeau, and T. J. Aitman (2002), Finding genes that underlie complex traits, *Science*, 298(5602), 2345–2349.

[35] Gomez Rodriguez, M., J. Leskovec, and A. Krause (2010), Inferring networks of diffusion and influence, in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1019–1028, ACM.

[36] Gomez Rodriguez, M., J. Leskovec, and B. Schölkopf (2013), Structure and dynamics of information pathways in online media, in *Proceedings of the sixth ACM international conference on Web search and data mining*, pp. 23–32, ACM.

[37] Hestenes, M. R. (1969), Multiplier and gradient methods, *Journal of optimization theory and applications*, 4(5), 303–320.

[38] Höfling, H., and R. Tibshirani (2009), Estimation of sparse binary pairwise markov networks using pseudo-likelihoods, *Journal of Machine Learning Research*, 10(Apr), 883–906.

[39] Hunter, D. J., et al. (2007), A genome-wide association study identifies alleles in fgfr2 associated with risk of sporadic postmenopausal breast cancer, *Nature genetics*, 39(7), 870–874.

[40] Hwang, T. H., G. Atluri, R. Kuang, V. Kumar, T. Starr, K. A. Silverstein, P. M. Haverty, Z. Zhang, and J. Liu (2013), Large-scale integrative network-based analysis identifies common pathways disrupted by copy number alterations across cancers, *BMC genomics*, 14(1), 440.

[41] Kempe, D., J. Kleinberg, and É. Tardos (2003), Maximizing the spread of influence through a social network, in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 137–146, ACM.

[42] Ku, A. T., et al. (2017), Tcf7l1 promotes skin tumorigenesis independently of β-catenin through induction of lcn2, *eLife*, 6.

[43] Kwong, L. N., and W. F. Dove (2009), Apc and its modifiers in colon cancer, in *APC Proteins*, pp. 85–106, Springer.

[44] Lage, K., et al. (2007), A human phenome-interactome network of protein complexes implicated in genetic disorders, *Nature biotechnology*, 25(3), 309–316.

[45] Lauritzen, S. L. (1996), *Graphical models*, vol. 17, Clarendon Press.

[46] Lee, J. D., and T. J. Hastie (2015), Learning the structure of mixed graphical models, *Journal of Computational and Graphical Statistics*, 24(1), 230–253.

[47] Li, Y., J. Wei, C. Xu, Z. Zhao, and T. You (2014), Prognostic significance of cyclin d1 expression in colorectal cancer: a meta-analysis of observational studies, *PloS one*, 9(4), e94,508.

[48] Lin, Y., et al. (2015), Pik3r1 negatively regulates the epithelial-mesenchymal transition and stem-like phenotype of renal cancer cells through the akt/gsk3β/ctnnb1 signaling pathway, *Scientific reports*, 5, 8997.

[49] Liu, H., J. Lafferty, and L. Wasserman (2009), The nonparanormal: Semiparametric estimation of high dimensional undirected graphs, *Journal of Machine Learning Research*, 10(Oct), 2295–2328.

[50] Liu, H., F. Han, M. Yuan, J. Lafferty, L. Wasserman, et al. (2012), High-dimensional semiparametric gaussian copula graphical models, *The Annals of Statistics*, 40(4), 2293–2326.

[51] Liu, H., G. Li, W. Zeng, P. Zhang, F. Fan, Y. Tu, and Y. Zhang (2014), Combined detection of gab1 and gab2 expression predicts clinical outcome of patients with glioma, *Medical Oncology*, 31(8), 77.

[52] Ma, C., Y. Chen, and D. Wilkins (2014), Ranking of cancer genes in markov chain model through integration of heterogeneous sources of data, in *Bioinformatics and Biomedicine (BIBM), 2014 IEEE International Conference on*, pp. 248–253, IEEE.

[53] Majidzadeh-A, K., R. Esmaeili, and N. Abdoli (2011), Tfrc and actb as the best reference genes to quantify urokinase plasminogen activator in breast cancer, *BMC research notes*, 4(1), 215.

[54] McDermott, J., R. Bumgarner, and R. Samudrala (2005), Functional annotation from predicted protein interaction networks, *Bioinformatics*, 21(15), 3217–3226.

[55] McLean, K. (2013), Bmps morph into new roles in ovarian cancer, *Cell Cycle*, 12(3), 389–389.

[56] McLean, K., et al. (2011), Human ovarian carcinoma–associated mesenchymal stem cells regulate cancer stem cells and tumorigenesis via altered bmp production, *The Journal of clinical investigation*, 121(8), 3206.

[57] Mehra, R., S. Varambally, L. Ding, R. Shen, M. S. Sabel, D. Ghosh, A. M. Chinnaiyan, and C. G. Kleer (2005), Identification of gata3 as a breast cancer prognostic marker by global gene expression meta-analysis, *Cancer research*, 65(24), 11,259–11,264.

[58] Meinhold-Heerlein, I., et al. (2001), Expression and potential role of fas-associated phosphatase-1 in ovarian cancer, *The American journal of pathology*, 158(4), 1335–1344.

[59] Meinshausen, N., and P. Bühlmann (2006), High-dimensional graphs and variable selection with the lasso, *The annals of statistics*, pp. 1436–1462.

[60] Meyer, K. B., A.-T. Maia, M. O'Reilly, A. E. Teschendorff, S.-F. Chin, C. Caldas, and B. A. Ponder (2008), Allele-specific up-regulation of fgfr2 increases susceptibility to breast cancer, *PLoS biology*, 6(5), e108.

[61] Meyer, S., L. Held, et al. (2014), Power-law models for infectious disease spread, *The Annals of Applied Statistics*, 8(3), 1612–1639.

[62] Mooney, S. D., V. G. Krishnan, and U. S. Evani (2010), Bioinformatic tools for identifying disease gene and snp candidates, *Genetic Variation: Methods and Protocols*, pp. 307–319.

[63] Mostafavi, S., D. Ray, D. Warde-Farley, C. Grouios, and Q. Morris (2008), Genemania: a real-time multiple association network integration algorithm for predicting gene function, *Genome biology*, 9(1), S4.

[64] Myers, S., and J. Leskovec (2010), On the convexity of latent social network inference, in *Advances in Neural Information Processing Systems*, pp. 1741–1749.

[65] Network, C. G. A., et al. (2012), Comprehensive molecular portraits of human breast tumours, *Nature*, 490(7418), 61.

[66] Network, C. G. A. R., et al. (2013), Integrated genomic characterization of endometrial carcinoma, *Nature*, 497(7447), 67–73.

[67] Newman, M. E. (2005), Power laws, pareto distributions and zipf's law, *Contemporary physics*, 46(5), 323–351.

[68] Nica, A. C., and E. T. Dermitzakis (2008), Using gene expression to investigate the genetic basis of complex disorders, *Human molecular genetics*, 17(R2), R129–R134.

[69] Obata, K., S. J. Morland, R. H. Watson, A. Hitchcock, G. Chenevix-Trench, E. J. Thomas, and I. G. Campbell (1998), Frequent pten/mmac mutations in endometrioid but not serous or mucinous epithelial ovarian tumors, *Cancer research*, 58(10), 2095–2097.

[70] Petrochilos, D., A. Shojaie, J. Gennari, and N. Abernethy (2013), Using random walks to identify cancer-associated modules in expression data, *BioData mining*, 6(1), 17.

[71] Pino, M. S., M. Mino-Kenudson, B. M. Wildemore, A. Ganguly, J. Batten, I. Sperduti, A. J. Iafrate, and D. C. Chung (2009), Deficient dna mismatch repair is common in lynch syndrome-associated colorectal adenomas, *The Journal of Molecular Diagnostics*, 11(3), 238–247.

[72] Piotrowski, A., et al. (2014), Germline loss-of-function mutations in lztr1 predispose to an inherited disorder of multiple schwannomas, *Nature genetics*, 46(2), 182.

[73] Qian, Z., et al. (2017), Overexpression of foxo3a is associated with glioblastoma progression and predicts poor patient prognosis, *International journal of cancer*, 140(12), 2792–2804.

[74] Re, M., and G. Valentini (2012), Cancer module genes ranking using kernelized score functions, *BMC bioinformatics*, 13(14), S3.

[75] Re, M., and G. Valentini (2012), Random walking on functional interaction networks to rank genes involved in cancer, *Artificial Intelligence Applications and Innovations*, pp. 66–75.

[76] Reis, G. F., et al. (2015), Cdkn2a loss is associated with shortened overall survival in lower-grade (world health organization grades ii–iii) astrocytomas, *Journal of Neuropathology & Experimental Neurology*, 74(5), 442–452.

[77] Rodriguez, M. G., D. Balduzzi, and B. Schölkopf (2011), Uncovering the temporal dynamics of diffusion networks, *arXiv preprint arXiv:1105.0697*.

[78] Samimi, G., D. Fink, N. M. Varki, A. Husain, W. J. Hoskins, D. S. Alberts, and S. B. Howell (2000), Analysis of mlh1 and msh2 expression in ovarian cancer before and after platinum drug-based chemotherapy, *Clinical cancer research*, 6(4), 1415–1421.

[79] Scott, M., W. G. McCluggage, K. J. Hillan, P. A. Hall, and S. Russell (2006), Altered patterns of transcription of the septin gene, sept9, in ovarian tumorigenesis, *International journal of cancer*, 118(5), 1325–1329.

[80] Semba, S., N. Itoh, M. Ito, E. M. Youssef, M. Harada, T. Moriya, W. Kimura, and M. Yamakawa (2002), Down-regulation of pik3cg, a catalytic subunit of phosphatidylinositol 3-oh kinase, by cpg hypermethylation in human colorectal carcinoma, *Clinical cancer research*, 8(12), 3824–3831.

[81] Sharan, R., I. Ulitsky, and R. Shamir (2007), Network-based prediction of protein function, *Molecular systems biology*, 3(1), 88.

[82] Shepherd, T. G., and M. W. Nachtigal (2003), Identification of a putative autocrine bone morphogenetic protein-signaling pathway in human ovarian surface epithelium and ovarian cancer cells, *Endocrinology*, 144(8), 3306–3314.

[83] Shu, Q., J. Liu, X. Liu, S. Zhao, H. Li, Y. Tan, and J. Xu (2016), Gababr/gsk-3β/nf-κb signaling pathway regulates the proliferation of colorectal cancer cells, *Cancer medicine*, 5(6), 1259–1267.

[84] Slattery, M. L., A. R. Folsom, R. Wolff, J. Herrick, B. J. Caan, and J. D. Potter (2008), Transcription factor 7–like 2 polymorphism and colon cancer, *Cancer Epidemiology and Prevention Biomarkers*, 17(4), 978–982.

[85] Song, H., et al. (2007), Tagging single nucleotide polymorphisms in the brip1 gene and susceptibility to breast and ovarian cancer, *PLoS One*, 2(3), e268.

[86] Turner, F. S., D. R. Clutterbuck, and C. A. Semple (2003), Pocus: mining genomic sequence annotation to predict disease genes, *Genome biology*, 4(11), R75.

[87] Verhaak, R. G., et al. (2010), Integrated genomic analysis identifies clinically relevant subtypes of glioblastoma characterized by abnormalities in pdgfra, idh1, egfr, and nf1, *Cancer cell*, 17(1), 98–110.

[88] Vignati, S., et al. (2006), Cellular, molecular consequences of peroxisome proliferator-activated receptor delta activation in ovarian cancer cells, *Neoplasia*, 8(10), 851IN2–861IN12.

[89] Wallinga, J., and P. Teunis (2004), Different epidemic curves for severe acute respiratory syndrome reveal similar impacts of control measures, *American Journal of epidemiology*, 160(6), 509–516.

[90] Walsh, A. M., et al. (2015), Sprouty2 drives drug resistance and proliferation in glioblastoma, *Molecular Cancer Research*, 13(8), 1227–1237.

[91] Wang, S., H. Zhang, J. Zhang, X. Zhang, S. Y. Philip, and Z. Li (2015), Inferring diffusion networks with sparse cascades by structure transfer, in *International Conference on Database Systems for Advanced Applications*, pp. 405–421, Springer.

[92] Watts, D. J., and P. S. Dodds (2007), Influentials, networks, and public opinion formation, *Journal of consumer research*, 34(4), 441–458.

[93] Xu, G., and J. Y. Li (2016), Differential expression of pdgfrb and egfr in microvascular proliferation in glioblastoma, *Tumor Biology*, 37(8), 10,577–10,586.

[94] Xu, K., Z. Zhang, H. Pei, H. Wang, L. Li, and Q. Xia (1899), Foxo3a induces temozolomide resistance in glioblastoma cells via the regulation of β-catenin nuclear accumulation, *Oncology reports*, 37(4), 2391–2397.

[95] Xue, L., H. Zou, et al. (2012), Regularized rank-based estimation of high-dimensional nonparanormal graphical models, *The Annals of Statistics*, 40(5), 2541–2571.

[96] Yan, P. (2008), Distribution theory, stochastic processes and infectious disease modelling, in *Mathematical Epidemiology*, pp. 229–293, Springer.

[97] Yuan, M., and Y. Lin (2006), Model selection and estimation in regression with grouped variables, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1), 49–67.

[98] Zhang, W., N. Johnson, B. Wu, and R. Kuang (2012), Signed network propagation for detecting differential gene expressions and dna copy number variations, in *Proceedings of the ACM Conference on Bioinformatics, Computational Biology and Biomedicine*, pp. 337–344, ACM.

VITA

Christopher Ma was born on April 9, 1985, in Hong Kong. His family moved to Canada in 1996 and he received his secondary education in Toronto. He then moved back to Hong Kong and attended University of Hong Kong in Hong Kong, graduating in 2010 with a B.S. degree in computer science and mathematics. He immediately enrolled in graduate studies at University of Hong Kong Department of Computer Science. He earned his Masters degree from HKU in 2012 and later embarked on his Ph.D. study at the University of Mississippi in 2012. He gratefully acknowledges funding support from the National Science Foundation under Award Numbers EPS 0903787 and EPS 1006883 during his Ph.D studies.