Estimation and Inference in High Dimensional Networks, with Applications to Biological Systems

by

Ali Shojaie

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Statistics)
in The University of Michigan
2010

Doctoral Committee:

Professor George Michailidis, Chair
Professor Michael Boehnke
Professor Susan Murphy
Professor Jeremy Taylor
Professor Naysyin Wang
Associate Professor Kerby Shedden

In the memory of my father ...

ACKNOWLEDGEMENTS

I would like to thank the members of my doctoral committee, for their invaluable support and encouragement throughout the dissertation process. Special thank goes to my advisor, Professor George Michailidis; this dissertation would not have been completed without his help and guidance. Finally, I would also like to thank the faculty and staff in the Department of Statistics, and all my friends at the University of Michigan, who made the process of completing my PhD enjoyable.

I am always indebted to my family; in particular my mom, who has devoted her life to the success and well-being of her children. During the past eight years, I have had the blessing of being able to share all my success and disappointments with my wife and best friend, Sahar; I can never thank her enough for her ever lasting love and support.

TABLE OF CONTENTS

CHAPTER

LIST OF FIGURES

Figure

LIST OF TABLES

Table

ABSTRACT

This dissertation discusses several aspects of estimation and inference for high dimensional networks, and is divided into three main parts. First, to assess the significance of arbitrary subnetworks (e.g. pathways), I propose a latent variable model that directly incorporates the network information. By formulating the problem as a (generalized) mixed linear model, I introduce a general inference procedure for testing the significance of subnetworks, that can be used to test for changes in both expression levels of the corresponding nodes (e.g. genes), as well as the structure of the network. The framework is then extended for analysis of data obtained from complex experimental designs. We also study the effect of noise in the network information, both theoretically and empirically, and show that the proposed inference procedure is robust to the presence of random noise.

In the second part, we consider the problem of estimating directed graphs from observed data. The general problem of estimation of directed graphs is computationally NP-hard and direction of interactions may not be distinguishable from observations. I consider a special case of this problem, where the nodes (i.e. variables) inherit a natural ordering, and propose efficient penalized likelihood methods for estimating the underlying network structure. Consistency of the estimators in the high dimensional setting (more variables than observations) is established. I also propose an extension of the lasso penalty that results in improved estimation of graphical

Granger causality from time-course observations.

The last part of the dissertation is devoted to issues of dimension reduction and efficient computation in networks. I propose a dimension reduction algorithm for networks using Laplacian eigenmaps, discuss the connections of this method to principal component analysis, and formulate the inference problem using a group lasso penalty. I also address computational aspects of estimation in networks, by proposing a distributed algorithm based on block-relaxation and derive conditions required for convergence of the algorithm to the maximum likelihood estimates. Finally, I present an extension of the block-relaxation algorithm, called approximate block-relaxation, that facilitates the use of iterative algorithms in optimization problems with complex objective functions.

CHAPTER I

Introduction

A number of emerging statistical applications involve analyzing observations from complex systems. Biological, physical and social systems are few examples of such applications. Although these systems are inherently different, and have varying levels of complexity, some of the goals and concerns in their analysis are common. Systems, by definition, involve components that interact with each other in an orchestrated fashion to carry out the main function(s). It is therefore crucial to account for the interactions among components in the analysis of complex systems. From a statistical point of view, interactions among components of a system correspond to correlations among the related random variables, and these correlations can be incorporated into models in order to improve the efficiency in estimation and inference.

Many statistical methods have been developed to accommodate and take advantage of the correlations among observations. Models for analysis of temporal correlation among observations, and spatial correlations resulting from physical distances between variables are classic examples. In this dissertation, I consider the analysis of biological systems, represented by large networks. Although throughout this dissertation we often refer to gene networks, the methodology developed here is to a large extent applicable to other biological components (e.g. proteins and metabolites), and more generally can be used for analysis of high dimensional networks.

The goal of this research is to develop models that incorporate the network information in order to determine which components of the biological system (i.e. sub-networks) respond to specific environmental factors, or are involved in genetic contributions to complex diseases. To that end, two types of problems are considered: (i) incorporating the available information about the associations among genes in order to improve estimation and inference and (ii) identifying the association among genes, in case where such information is unavailable (network discovery).

Many researchers have recently been attracted to analysis of biological networks and significant contributions have been made in both of the above areas. To address the first problem above, I propose a latent variable model that directly incorporates the existing network information (e.g. information available in public repositories and genetic information websites) and show that this model can be represented as a (generalized) mixed linear model. We then use the framework of mixed models to present a general inference procedure for testing the significance of arbitrary subnetworks based on estimable linear combinations of fixed effect parameters. We discuss several possible test procedures and present an optimal method that allows us to isolate the effect of every subnetwork and to test for changes in both gene expression levels and the structure of the network. Extensions of the original model for analysis of general networks, including both directed and undirected edges, and generalizations for analysis of complex experiments are also discussed. We study the effect of noise in the network information and show that the proposed methodology is robust to random noise in the network information. This implies that, in case where the network information is unavailable, estimated network structures can be incorporated, as long as the estimation error is small. I also propose a dimension reduction method for networks, based on Laplacian eigenmaps with isolated (Neumann) boundary conditions, which can be used to extract informative features from arbitrary subnetworks. Finally, we address the computational aspect of the estimation problem on networks by proposing an iterative algorithm based on block-relaxation (BR). Conditions required for convergence of the algorithm to the maximum likelihood estimates and extensions of the algorithm for distributed computing are discussed.

To address the second problem (network reconstruction), we consider construction of directed graphs from observation. This is an computationally NP-hard problem and the direction of the edges may not be distinguishable from observations alone. We consider a special case of the problem, where the variables inherit a natural ordering, and propose penalized likelihood methods for estimation of high-dimensional networks. The asymptotic properties of the estimators are studied in high dimensional, sparse settings, and an efficient algorithm is proposed for estimation of model parameters. I also propose an extension of this model for estimation of directed edges from time-course observations (using the concept of Granger causality) and propose a novel penalty, called truncating lasso, for simultaneous estimation of model parameters and the order of the underlying vector autoregressive (VAR) model.

This dissertation consists of three main parts, each including two chapters. The first part is devoted to developing a general inference framework for analysis of arbitrary subnetworks. In Chapter II, we introduce the latent variable model for incorporating the network information and its mixed linear model representation. Basic properties of the model and the proposed inference procedure are discussed, and the network contrast vector for assessing the significance of subnetworks is introduced. Chapter III consists of an extension of the model of the previous chapter, which incorporates more complex experimental conditions and considers temporal correlations among observations. Robustness of the proposed methodology to random noise in network information is also discussed.

In the second part, we discuss the problem of network reconstruction. Chapter IV includes the penalized likelihood method of estimating directed acyclic graphs (DAGs). Asymptotic properties of the proposed estimators and efficient computation in high dimensional settings are also discussed. The material in Chapter V extends the model of the previous chapter, in order to estimate the so-called graphical Granger model from time-course observations. The truncating lasso penalty for simultaneous estimation of model parameters and the order of the underlying VAR is introduced in this chapter and computational, as well as theoretical, issues related to this new penalty are discussed.

Issues of dimension reduction, and efficient computation for high dimensional networks are discussed in the last part of the dissertation. Chapter VI, includes the proposed dimension reduction method for analysis of significance of subnetwork. The problem of inference for subnetworks is formulated as a group-penalized regression problem on graphs, and the properties of the resulting estimates are discussed. Finally, in Chapter VII, we discuss the proposed iterative algorithm, for distributed estimation of parameters of mixed linear models on networks, and discuss a generalization of the BR algorithm for the cases where the likelihood function is not readily available (e.g. the generalized mixed linear model). Possible extensions and directions for future research are discussed in Chapter VIII.

CHAPTER II

Network-Based Gene Set Analysis

2.1 Introduction

In standard analysis of differential expression, statistical significance of each gene is assessed independently and some method of multiple testing correction is used to adjust the p-values. Such methods are usually less sensitive in detecting genes that have smaller differences in mRNA abundance between different experimental conditions and may therefore be less powerful than desired. Furthermore, analyzing individual genes (*single-gene analysis*) often generates results that are not reproducible and lack meaningful biological interpretations. The focus of current research has thus shifted to analyzing *a priori* defined sets of genes (*gene set analysis*) and using external information to strengthen the analysis of differential expression. Analysis of gene sets may result in increased power compared to single gene analysis, when individual genes are not significant, but the combined effect of genes in the gene set indicates a significant association. Furthermore, methods of gene set analysis can preserve the correlation among genes which may lead to more reliable inference. These methods however, do not directly incorporate the external information about the interactions among genes represented by the gene network. In this chapter, I develop a model that directly incorporates the network information, and propose a general inference framework for testing the significance of genetic pathways.

2.1.1 A Motivating Example

In an interesting approach, Ideker et al. (2001) integrated gene expression and protein level data to study significant signaling and metabolic pathways in yeast *Saccharomyces cerevisiae.* They reported interactions among genes and proteins in different pathways along with information on the estimated correlation among genes in the network. The authors also grouped the genes into subnetworks (pathways) based on their biological functions. Figure 2.1, which was originally presented in Ideker et al. (2001), illustrates the network of genes under consideration. We have updated their network based on newly defined interactions reported in Bader et al. (2004). This results in a network of 343 genes with 419 interactions for which estimates of correlations among genes are also available (these data are referred to as the *Ideker data* henceforth).

The mRNA expression levels of genes in the Ideker data are measured in 9 different perturbations of GAL genes along with the wild type yeast. For each perturbation, two samples of data are available. The first set of samples represents the expression levels of genes in cells grown in presence of galactose (gal+), while the second set includes expression levels for cells grown in absence of galactose (gal–), where the main source of carbon is raffinose. Our primary goal is to determine the pathways that are *involved* (either induced or suppressed) in cell growth in gal+ compared to gal– environments. In other words, we would like to test whether each of 15 gene sets defined by yeast pathways in Ideker et al. (2001) is differentially expressed in gal+ compared to gal– medium. In Section 2.3, we study different aspects of analysis of gene sets, and discuss the need for including changes in both expression levels, as well as associations among genes in different experimental conditions.

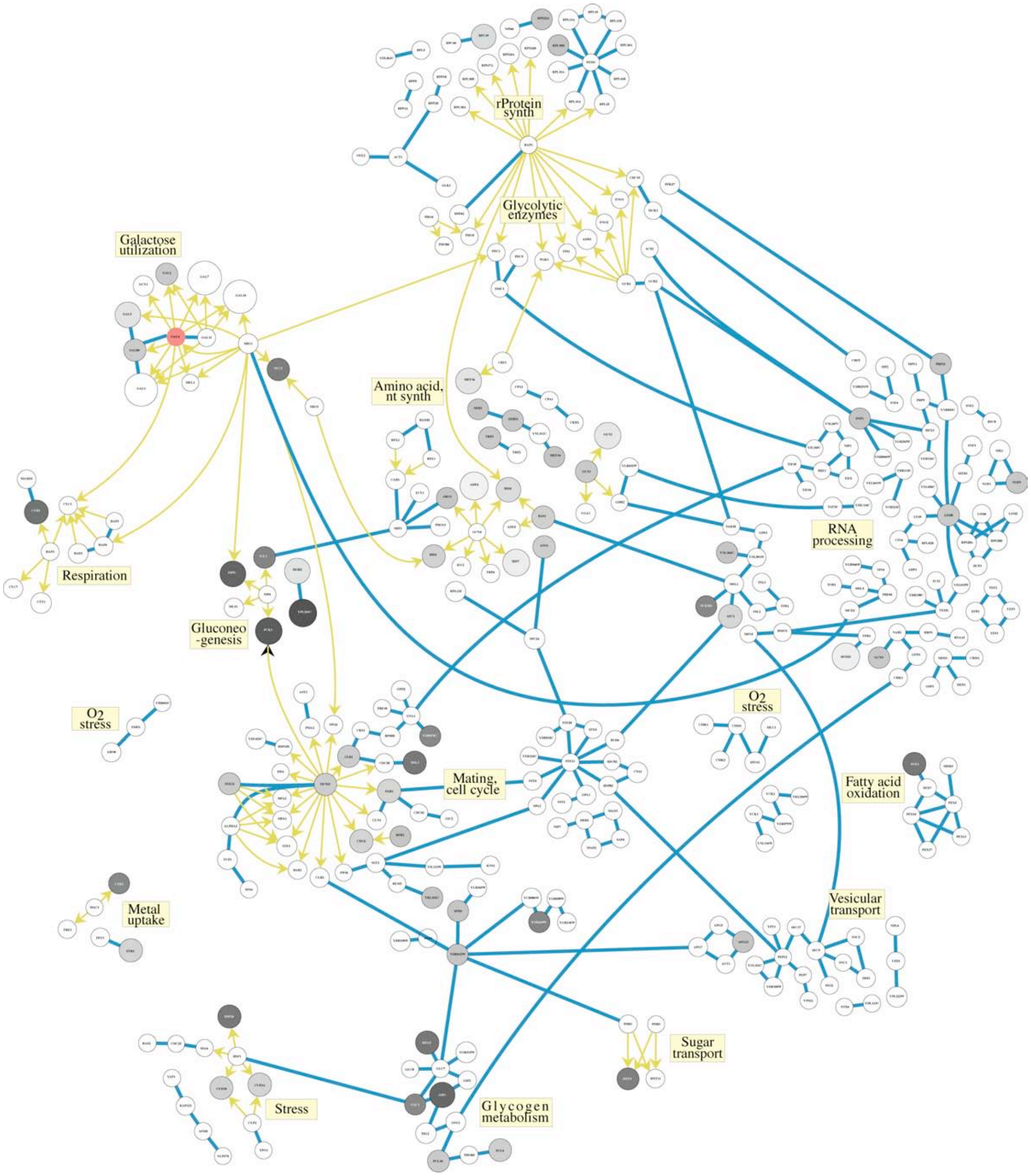


Figure 2.1: Yeast Galactose Utilization pathway published in Ideker et al. (2001), printed with permission from Science and the American Association for the Advancement of Science

We start by analyzing the Ideker data using the *Gene Set Enrichment Analysis* (GSEA) method of Subramanian et al. (2005). This method uses a permutation-based test to determine whether genes in *a priori* defined gene sets have non-random

Table 2.1: Analysis of Galactose utilization pathway using GSEA. The first two columns illustrate the pathway considered and the number of genes in the pathway. For each gene set, the Nominal p-value, FDR q-value and FWER p-value are reported along with the involvement of the pathway in gal+/gal– conditions.

	Pathway	Nominal	FDR	FWER	Involved
Pathway	size	p-val	q-val	p-val	in GAL(+/–)
Galactose Utilization	12	0.0020	0.00114	0.003	+
Amino Acid Synthesis	30	0.1853	0.21562	0.676	+
rProtein Synthesis	28	0.5261	0.44938	0.972	+
Stress	12	0.02004	0.19283	0.108	–
Vesicular Transport	19	0.07243	0.54138	0.489	–
Glycogen Metabolism	12	0.1321	0.41115	0.538	–
Respiration	9	0.1878	0.39508	0.637	–
O2 Stress	13	0.2384	0.6601	0.906	–
Fatty Acid Oxidation	7	0.4694	0.82373	0.963	–
Mating, Cell Cycle	58	0.3583	0.71842	0.968	–
Sugar Transport	2	0.7358	1	0.993	–
Metal Uptake	4	0.8374	1	0.997	–
Gluconeogenesis	7	0.8455	0.98853	0.997	–
RNA Processing	75	0.9879	1	1	–
Glycolytic Enzymes	16	0.9683	0.98189	1	–

associations with the phenotype. To that end, we first normalize the data so that the expression levels only represent the effect of the growth environment[1] . The results of the analysis are displayed in Table 2.1.

The first line of the table presents an expected outcome; the expression levels of genes in the Galactose Utilization pathway is expected to change in response to perturbations of GAL genes in the gal+ environment. On the other hand, although some of the pathways seem to have differential expression when cells lack galactose (e.g. Stress and Vesicular Transport), no other pathway appears significant after adjusting for multiple testing using the False Discovery Rate (FDR) controlling procedure of Benjamini and Hochberg (1995) with a q-value of 0.05. In Section 2.5, we revisit the analysis of the Ideker data based on the method proposed in this chapter, which directly incorporates the network information represented by the gene network in Figure 2.1.

[1] The mean expression levels of the two samples corresponding to each perturbation is subtracted from the two columns of data.

2.1.2 Background

Recent research on gene set analysis can be broadly classified into permutation-based methods motivated by the GSEA paper (Subramanian et al., 2005) and model-based approaches that make specific distributional assumptions about the gene expression data. The literature can be further categorized on whether direct or indirect external information on the gene network is employed. Tian et al. (2005) considered the problem of gene set analysis and described two hypotheses that should be considered when studying the significance of sets of genes. One of these hypotheses, which is the same as the hypothesis considered in GSEA, focuses on non-random association of genes in the gene set with the phenotype. The other hypothesis, considers non-random correlations between genes in a gene set. The test method proposed for the first hypothesis is based on permuting the class labels (column permutation) and the second hypothesis is tested by permuting genes (row permutation). Efron and Tibshirani (2007) formalized the idea of gene set analysis in a coherent statistical framework and examined the hypotheses presented in Tian et al. (2005). They also proposed an alternative test statistic with superior power properties and analyzed the effects of row and column permutations. Goeman and Bühlmann (2007) reviewed different methods proposed for testing significance of gene sets and highlighted important issues in selecting appropriate methods.

Although the above permutation-based methods are computationally intensive, they include minimum assumptions about the underlying biological model and are therefore robust to model misspecification. An alternative approach is based on model-based tests procedures, where specific distributions for the expression data are assumed. In one such approach, Jiang and Gentleman (2007) extended the idea of gene set analysis by adapting a linear model approach and adjusting for other covariates. They presented the gene sets in the form of an index matrix and offered a heuristic argument for using a normal approximation for testing sum of the expression levels of genes in each gene set. One major difficulty regarding model-based methods is the large number of variables (genes) compared to the small number of samples (the large p, small n problem, West, 2000). In such situations, estimation of model parameters becomes a challenging task and may result in unstable outcomes. However, additional sources of information besides the expression levels of genes could be used to improve the estimation. One possible source of external information is the underlying relationship between genes which itself is of independent interest. It is known that genes interact with each other through their protein products and form gene regulatory networks. Also, the protein products of groups of genes are involved in controlling specific functions in cells through genetic pathways. Increasing amount of information about these relationships is becoming available in public repositories, such as KEGG (Kyoto Encyclopedia of Genes and Genomes) (Kanehisa and Goto, 2000) and Gene Ontology (GO) (Ashburner et al., 2000) and can be used to improve the estimates of model parameters.

A number of researchers have recently used external information about gene networks to improve the analysis of gene sets. Rahnenführer et al. (2004) demonstrated that the sensitivity of detecting relevant pathways can be improved by integrating information about pathway topology. Barry et al. (2005) presented a permutation based procedure, called SAFE, that considers the underlying network structure. More recently, Wei and Li (2007) proposed a Markov Random Field model to incorporate the information on the gene network in the analysis. In a related approach, Wei and Pan (2008) have modeled the network information via latent variables into a spatially correlated mixture model. The latter two methods consider the problem of analysis of *single genes* on the network.

The above methods either (a) assume that the underlying gene network does not change as the experimental conditions change or (b) do not incorporate this change directly. However, changes in the underlying network structure can amplify the change in expression patterns and should be included in the analysis. For instance, Li (2002) demonstrated that the correlation patterns among ARG2 and other members of the urea-cycle pathway can change drastically as the expression level of ARG2 changes. Another concern in analyzing network data is to decorrelate subnetworks from the effects of other nodes in the network and to deal with nodes that belong to multiple networks. Alexa et al. (2006) present one such method which is an attempt to decorrelate GO graph structures. Their method focuses on decorrelating nodes at lower levels (children) from upper level nodes (parents).

In this chapter, I propose a latent variable model to directly incorporate the underlying gene network and present test statistics for testing the significance of arbitrary sub-networks based on the theory of mixed linear models. One major advantage of this method is that it not only considers the change in the expression levels of the genes in different conditions, but also reflects the change in network structures and correlations among genes. We also present a systematic approach that decorrelates each subnetwork from the other nodes while maintaining the interactions among genes in the subnetwork.

The rest of the chapter is organized as follows. In the next section, we introduce the proposed latent variable model and discuss some of its basic graph theoretical properties. In Section 2.3, I represent the latent variable model using the framework of mixed linear models and propose a general testing scheme based on the theory of mixed linear models. Section 2.3 ends with a result that is used to test the *pure* effect of each subnetwork. This result prevents tests of significance of subnetworks from being confounded with the effects of other subnetworks and also allows testing the effect of genes that belong to multiple networks. Section 2.4 includes three simulation studies for evaluating the performance of the new model under different testing conditions as well as studying the effect of noise in the network information on the proposed inference procedure. In Section 2.5, we revisit the Ideker data, and test the significance of pathways using the proposed model. Section 2.6 includes a discussion on limitations of the proposed model and future extensions.

2.2 The Latent Variable Model

Consider gene expression data $\mathcal{D}$ organized as a $p \times n$ matrix comprised of the expression levels of p genes for n samples, and let Y be the kth sample in the expression data (kth column of $\mathcal{D}$).

To model the correlation structure caused by the gene network, we represent the network as a directed graph $G = (V, E)$ with vertex set V, and edge set E, where E is represented by the $p \times p$ adjacency matrix A. Each nonzero element of the adjacency matrix, A_{ij}, represents a directed edge in the network. Elements of the adjacency matrix correspond to the strength of association among genes in the graph and are real values in $(-1, 1)$.

Consider the simple network of Figure 2.2:

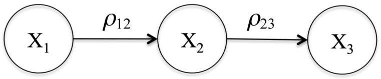


Figure 2.2: A simple gene network

Suppose $Y = X + \varepsilon$, where X represents the *signal* and $\varepsilon \sim N_p(0, \sigma_\varepsilon^2 I_p)$ the *noise.* Consider two adjacent genes i and j, where i affects j. One can represent the relationship between i and j using a simple linear model $X_j = \rho_{ij} X_i$. However, to account for unknown associations among genes and/or errors in the association weights, ρ_{ij}, we also add *latent variables* $\gamma_j \sim N_p(\mu_j, \sigma_\gamma^2)$ to represent the baseline expression level of gene j. For instance, γ_2 represents the expression level of gene 2 without the effect from gene 1. Thus for the simple gene network of Figure 2.2 we obtain

$$\begin{aligned} X_1 &= \gamma_1 \\ X_2 &= \rho_{12} X_1 + \gamma_2 = \rho_{12}\gamma_1 + \gamma_2 \\ X_3 &= \rho_{23} X_2 + \gamma_3 = \rho_{23}\rho_{12}\gamma_1 + \rho_{23}\gamma_2 + \gamma_3 \end{aligned}$$

These equations can be summarized in vector notation as:

$$Y = \Lambda\gamma + \varepsilon, \qquad \gamma \sim N_p(\mu, \sigma_\gamma^2 I_p), \qquad \varepsilon \sim N_p(0, \sigma_\varepsilon^2 I_p) \tag{2.1}$$

where Λ is called the *influence matrix* of the graph. In the simple example above, we have

$$\Lambda = \begin{pmatrix} 1 & 0 & 0 \\ \rho_{12} & 1 & 0 \\ \rho_{12}\rho_{23} & \rho_{23} & 1 \end{pmatrix}$$

Under such a model, Y is a normal random variable with mean $\mathbb{E}[Y] = \Lambda\mu$ and variance $\mathrm{Var}(Y_i) = \sigma_\gamma^2 \Lambda\Lambda' + \sigma_\varepsilon^2 I_p$, where Λ' denotes the transpose of matrix Λ.

In the remainder of this section, we study the relationship between the influence matrix, Λ, and the adjacency matrix of the graph, A. We provide a general result for the relationship between Λ and A as well as a compact expression that can be used to efficiently evaluate Λ for specific classes of graphs. We also discuss conditions under which the matrix Λ has full rank, which will be used in the analysis of the proposed inference procedure in Section 2.3.

Lemma II.1. *For any graph $G = (V, A)$ we have $\Lambda = A^0 + A^1 + A^2 + \cdots = \sum_{r=0}^{\infty} A^r$ (here A^0 is defined to be the identity matrix).*

Proof. From the matrix representation of the latent variable model in (2.1)

$$Y_i = \sum_{j=0}^{p} \Lambda_{ij}\gamma_j + \varepsilon_i, \qquad i = 1, \cdots, p$$

where $\Lambda_{ii} = 1$ and $\Lambda_{ij} \neq 0$ only if there is a path (of some length) on the graph from node i to node j. But for any graph G, the number of paths of length r ($r \in \mathbb{N}$) from v_i to v_j is given by the (i,j) element of A^r (Diestel, 2006). Therefore, $\Lambda_{ij} \neq 0$ whenever there exists r such that $[A^r]_{ij} > 0$. Hence, all possible paths from i to j are given by $[\sum_{r=0}^{\infty} A^r]_{ij}$. This implies that $\Lambda = \sum_{r=0}^{\infty} A^r$. □

Corollary II.2. *For any Directed Acyclic Graph (DAG), $\Lambda = A^0 + A^1 + A^2 + \cdots + A^p$.*

Proof. This follows immediately from Lemma II.1 by noting that since there are no loops in DAGs, the maximum length of paths equals p. □

The following results provide sufficient conditions for the matrix Λ to be of full rank. Although this guarantees validity of the model for at least some classes of directed graphs, it does not provide a necessary condition. Based on experiments with randomly generated adjacency matrices, there are in fact larger classes of graphs satisfying this property.

Lemma II.3. *For any Directed Acyclic Graph (DAG), the matrix Λ has full rank.*

Proof. The full rankness of Λ is proved by showing that Λ can be re-arranged into a lower triangular matrix with 1's on the diagonal.

First observe that $\Lambda_{ij} \times \Lambda_{ji} = 0$, since otherwise there will be a cycle in the graph. Also, from II.1 we have $\Lambda_{ii} = 1$.

Consider a reordering of rows (and correspondingly of columns) of the matrix in decreasing number of zeros. Every DAG has at least one root (a node that is not affected by any other node). This means that there is at least one row with $\Lambda_{kk} = 1$ and $\Lambda_{kj} = 0$ for all j. Permute Λ so that row k is the first row of the matrix and continue in the same way. Denote the number of zero elements of row i by ϕ_i and number of zeros in column j as ϕ_{Cj}. Then by the above observation, $\phi_{Ri} \geq p - \phi_{Ci}$ (here $p - \phi_{Ci}$ is the number of nonzero elements in column i).

To complete the proof, we need to show that the rearranged matrix Λ can be further permuted to result in a lower diagonal matrix. Suppose there exists $j > i$ such that $\Lambda_{ij} > 0$ and therefore $\Lambda_{ji} = 0$. If $\phi_{Rj} = \phi_{Ri}$ switch i and j to get a lower triangular matrix. However, if $\phi_{Rj} < \phi_{Ri}$ (i.e. if i is affected by a row with less number of zeros) there exists l such that $\Lambda_{jl} > 0$ but $\Lambda_{il} = 0$. However, $\Lambda_{jl} > 0$ means there exists a path from l to j and $\Lambda_{ij} > 0$ means that there exists a path from j to i. Thus there exists a path from l to i, i.e. $\Lambda_{il} > 0$, a contradiction. Therefore Λ must be a lower triangular matrix with $\Lambda_{ii} = 1$. □

Lemma II.4. *Consider a graph $G = (V,A)$ with influence matrix Λ*

a) If G is a Directed Acyclic Graph (DAG), then $A = I - \Lambda^{-1}$.

b) If the sum of absolute values of weights of edges ending at every node of the graph G is less than 1 (i.e. A is sub-stochastic), then $A = I - \Lambda^{-1}$ and Λ has full rank.

Proof. a) From Corollary II.2, $\Lambda = \sum_{r=0}^{p} A^r$ and hence

$$A\Lambda = \sum_{r=0}^{p} A^{r+1} = \Lambda + A^{p+1} - I$$

But when G is a DAG, $A^{p+1} = 0$ hence $A\Lambda = \Lambda - I$. By full rankness of Λ,

$A = I - \Lambda^{-1}$.

b) The condition in (b) implies that the sum of the absolute values of off-diagonal elements of A is less than 1. Let s_i be the sum of absolute values of off-diagonal elements of the ith row of A. Since the diagonal elements of A are 0, by the Gershgorin's Ring Theorem (Friedberg et al., 1996) if λ is an eigenvalue of A, we have $|\lambda| \leq s_i \leq 1$. Now let $\Lambda_m = \sum_{r=0}^{m} A^r$. Then $\Lambda = \lim_{m\to\infty} \Lambda_m$ and using an argument similar to part (a),

$$A\Lambda_m = \Lambda_m - I + A^{m+1}$$

Since eigenvalues of A are less than 1 in magnitude, $\lim_{m\to\infty} \Lambda_m$ exists (Friedberg et al., 1996) and by the eigen-decomposition of A, $A^{m+1} \to 0$ as $m \to \infty$. Hence taking the limit, we get $A\Lambda = \Lambda - I + A$. On the other hand, the established bound on the eigenvalues of A implies that all eigenvalues of $I - A$ are nonzero, which means that $I - A$ and therefore, Λ are full rank. Thus $A = I - \Lambda^{-1}$. □

Lemma II.4 establishes an alternative relationship between Λ and A and determines two classes of graphs for which such a relationship is valid. As noted before, conditions presented in this result are only sufficient. For the general graph $G = (V, A)$, if the spectral radius of A is less than 1, Λ has full rank and the relationship between A and Λ established in Lemma II.4 holds. On the other hand, in special cases where Λ is not of full rank, it may be possible to modify the graph and therefore apply the model presented here. For instance, one large class of graphs where Λ is not full rank consists of *cyclic* graphs. The cycles in biological networks are often representatives of feedback loops which are common features of cell cycle related networks. However, the feedback is usually effective after a time delay and therefore, when time series data are used to study these networks, the cycles can be broken down by distinguishing between nodes at the beginning and end of each cycle. Undirected edges (e.g. protein-protein interactions) can also be transformed into two directed edges using a common latent variable affecting both nodes. More generally, it is often possible to transform the graph by introducing dummy nodes and can hence apply the model presented here.

2.3 Inference

2.3.1 Preliminaries

In this section, we study the inference procedure for the proposed model. Although this method can be used to test a variety of hypotheses, in order to simplify the presentation, we focus on testing the equality of means of two experimental conditions. The extension to more complicated settings is discussed at the end of the section. As before, let Y be a given sample in the expression data (kth column of data matrix $\mathcal{D}$) and let Y^C and Y^T represent *control* and *treatment* conditions, with n_1 columns of $\mathcal{D}$ corresponding to control samples and $n_2 = n - n_1$ columns to treatment samples. Also let two sets of parameters (μ^C, Λ^C) and (μ^T, Λ^T) represent mean vectors and influence matrices under control and treatment conditions, respectively.

Let $\mathbf{b}$ be an indicator vector determining genes that belong to a specific gene set (pathway). In other words, $\mathbf{b}_j = 1$ if gene j is in gene set and 0 otherwise. We can test the significance of the gene sets by defining the test statistic $\mathbf{V} = \mathbf{b}Y^T - \mathbf{b}Y^C$ and testing:

$$H_0 : \mathbb{E}[\mathbf{V}] = 0 \quad vs. \quad H_1 : \mathbb{E}[\mathbf{V}] \neq 0 \tag{2.2}$$

Then under H_0:

$$\mathbb{E}_0[\mathbf{V}] = 0$$

and

$$\mathrm{Var}_0(\mathbf{V}) = (1/n^2)[n_2(\mathbf{b}\Lambda^T)(\mathbf{b}\Lambda^T)' + n_1(\mathbf{b}\Lambda^C)(\mathbf{b}\Lambda^C)']$$

Although the hypothesis in (2.2) can be tested using a generalized likelihood ratio test, it turns out that the latent variable model of Section 2.2 can be represented as a *Mixed Linear Model* (MLM). Using this framework, we can study a variety of spatio-temporal models and consider more general hypothesis testing problems.

2.3.2 Mixed Linear Model representation

Let $\mathbf{Y}$, $\boldsymbol{\gamma}$ and $\boldsymbol{\varepsilon}$ represent the rearrangement of vectors Y ,γ and ε into $np \times 1$ column vectors. Then $\mathbf{Y} = \mathbf{\Psi}\beta + \mathbf{\Pi}\boldsymbol{\gamma} + \boldsymbol{\varepsilon}$ where:

$$\beta = (\mu_1^C, \dots, \mu_p^C, \mu_1^T, \dots, \mu_p^T)'$$

$$\mathbf{\Psi} = \begin{pmatrix} \Lambda^C & \cdots & \Lambda^C & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \Lambda^T & \cdots & \Lambda^T \end{pmatrix}'$$

$$\mathbf{\Pi} = \mathrm{diag}(\Lambda^C, \dots, \Lambda^C, \Lambda^T, \dots, \Lambda^T)'$$

In this model, $\boldsymbol{\gamma}$ is the vector of (unknown) *random effects* and $\boldsymbol{\gamma}$ and $\boldsymbol{\varepsilon}$ are normally distributed random vectors with:

$$\mathbb{E}\begin{bmatrix} \gamma \\ \varepsilon \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

and

$$\mathrm{Var}\begin{bmatrix} \gamma \\ \varepsilon \end{bmatrix} = \begin{bmatrix} \Sigma_\gamma & 0 \\ 0 & \Sigma_\varepsilon \end{bmatrix}$$

For the latent variable model presented in the previous section, $\Sigma_\gamma = \sigma_\gamma^2 I$ and $\Sigma_\varepsilon = \sigma_\varepsilon^2 I$ and the variance of $Y^j, j \in \{C, T\}$ is given by $\sigma_\gamma^2 \Lambda^j (\Lambda^j)' + \sigma_\varepsilon^2 I$.

The estimate of β in the mixed linear model is given by (Searle, 1971):

$$\hat{\beta} = \left(\mathbf{\Psi}'\hat{\mathbf{W}}^{-1}\mathbf{\Psi}\right)^{-1}\mathbf{\Psi}'\hat{\mathbf{W}}^{-1}\mathbf{Y}$$

where $\mathbf{W} = (\sigma_\gamma^2\mathbf{\Pi}\mathbf{\Pi}' + \sigma_\varepsilon^2\mathbf{I}_{np})$. The estimate of β depends on estimates of σ_γ^2 and σ_ε^2 which can be estimated via *Restricted Maximum Likelihood* procedure (REML).

The framework of mixed linear models allows us to test a variety of hypotheses about β by considering tests of the form:

$$H_0 : l\beta = 0 \quad vs. \quad H_1 : l\beta \neq 0 \tag{2.3}$$

Here l is in general any *estimable* linear combination of β's (Searle, 1971). An example of such a vector is a *contrast vector*, which satisfies the constraint $\mathbf{1}'l = 0$. In the ensuing discussion, any linear combination of β's satisfying the estimability requirement is referred to as a *contrast vector*.

Based on the theory of mixed linear models, we can test (2.3) using the test statistic:

$$T = \frac{l\hat{\beta}}{\sqrt{l\hat{C}l'}} \tag{2.4}$$

where $C = \left(\mathbf{\Psi}'\mathbf{W}^{-1}\mathbf{\Psi}\right)^{-1}$.

Under the null hypothesis in (2.3), T has approximately a t distribution with ν degrees of freedom, where the degrees of freedom is estimated using the Satterthwaite approximation method (McLean and Sanders, 1988):

$$\nu = \frac{2\left(l\hat{C}l'\right)^2}{\tau' K \tau}$$

with $\tau = (\frac{\partial}{\partial\sigma_\gamma^2}lCl', \frac{\partial}{\partial\sigma_\varepsilon^2}lCl')'$ and K is the empirical covariance matrix of $(\sigma_\gamma^2, \sigma_\varepsilon^2)'$.

2.3.3 Computational issues and the use of the Mixed Linear Model

The mixed linear model facilitates the representation of the latent variable introduced in Section 2.2. However, estimation and inference in this framework involves forming the matrices $\boldsymbol{\Psi}$ and $\boldsymbol{\Pi}$, and performing operations involving products and inverses of these matrices. In the context of analysis of genetic data the dimensions of these matrices ($np \times 2p$ and $np \times np$) can cause serious difficulties in terms of computation time, memory requirement and numerical stability of the estimation algorithms. It is therefore necessary to derive alternative methods for estimation of parameters in the model. It turns out that due to the special structure of the model presented in Section 2.2, and the sparsity pattern of matrices $\boldsymbol{\Psi}$ and $\boldsymbol{\Pi}$, the formulas presented in the previous section can be substantially simplified. More specifically, for the problem stated in Section 2.3.2 we have:

$$\hat{\beta} = \begin{pmatrix} \hat{\beta}^C \\ \hat{\beta}^T \end{pmatrix} = \begin{pmatrix} \bar{Y}^C \\ \bar{Y}^T \end{pmatrix}$$

and

$$C = \begin{bmatrix} \frac{1}{n_1}\left(\sigma_\gamma^2 I_p + \sigma_\varepsilon^2(\Lambda^{C\prime}\Lambda^C)\right) & 0 \\ 0 & \frac{1}{n_2}\left(\sigma_\gamma^2 I_p + \sigma_\varepsilon^2(\Lambda^{T\prime}\Lambda^T)\right) \end{bmatrix}$$

In the particular case considered here, the REML estimates of the variance components can be directly computed as the maximizers of the REML equation without any need for iterative methods. However, profiling out one of the variance components may result in more stable solutions.

2.3.4 Role of the contrast vector

The estimates of β based on the mixed linear model represent the individual expression level of each gene in the network. Thus, in order to evaluate the combined effect of each gene set using the test statistic T, the choice of contrast vector l proves fairly crucial. More specifically, the choice of l determines the null and alternative hypotheses of the test in (2.3), which in turn affects its significance level and power. In this section, we present different choices of contrast vectors and study their properties and effects on the power of tests.

A simple choice for the contrast vector l is to use the indicator vector of the gene set. In other words,

$$l^{(1)} = (-\mathbf{b}, \mathbf{b}) \tag{2.5}$$

This simple choice of l corresponds to testing the following hypothesis:

$$H_0^{(1)} : \mathbf{b}\left(\mu^T - \mu^C\right) = 0 \quad vs. \quad H_1^{(1)} : \mathbf{b}\left(\mu^T - \mu^C\right) \neq 0 \tag{2.6}$$

which for each gene set g is equivalent to

$$H_0^{(1)} : \sum_{i \in g} \mu_i^T - \mu_i^C = 0 \quad vs. \quad H_1^{(1)} : \sum_{i \in g} \mu_i^T - \mu_i^C \neq 0 \tag{2.7}$$

Such a contrast vector however, only considers the mean expression levels of genes and does not reflect the combined effect of the set of genes in $\mathbf{b}$, which is affected by interactions among genes in the network.

When the underlying network structure and therefore the correlation among genes is known, a natural alternative to $l^{(1)}$ is to also include the influence matrices Λ^C and Λ^T. This leads to the following choice of contrast vector:

$$l^{(2)} = (-\mathbf{b}\Lambda^C, \mathbf{b}\Lambda^T) \tag{2.8}$$

which corresponds to testing the following hypotheses:

$$H_0^{(2)} : \mathbf{b}\left(\Lambda^T\mu^T - \Lambda^C\mu^C\right) = 0 \quad vs. \quad H_1^{(2)} : \mathbf{b}\left(\Lambda^T\mu^T - \Lambda^C\mu^C\right) \neq 0 \tag{2.9}$$

The null hypothesis presented in (2.9) may first seem less intuitive and the choice of $l^{(2)}$ rather arbitrary. However, the rationale behind the latter choice of contrast vector becomes clearer when we examine the test statistics corresponding to each one of the two null hypotheses in (2.6) and (2.9). In the case of the two-population test considered here, the above choices of contrast vectors lead to (after some algebra) the following test statistics:

$$T_1 = \frac{\mathbf{b}\left(\left(\Lambda^T\right)^{-1}\bar{Y}^T - \left(\Lambda^C\right)^{-1}\bar{Y}^C\right)}{\sqrt{\hat{\sigma}_\gamma^2\left(\frac{1}{n_1}+\frac{1}{n_2}\right)\mathbf{b}\mathbf{b}' + \hat{\sigma}_\varepsilon^2\left[\mathbf{b}\left(\frac{1}{n_2}\left(\Lambda^{T\prime}\Lambda^T\right)^{-1} + \frac{1}{n_1}\left(\Lambda^{C\prime}\Lambda^C\right)^{-1}\right)\mathbf{b}'\right]}} \tag{2.10}$$

and

$$T_2 = \frac{\mathbf{b}\left(\bar{Y}^T - \bar{Y}^C\right)}{\sqrt{\hat{\sigma}_\gamma^2\left[\mathbf{b}\left(\frac{1}{n_2}\Lambda^T\Lambda^{T\prime} + \frac{1}{n_1}\Lambda^C\Lambda^{C\prime}\right)\mathbf{b}'\right] + \hat{\sigma}_\varepsilon^2\left(\frac{1}{n_1}+\frac{1}{n_2}\right)\mathbf{b}\mathbf{b}'}} \tag{2.11}$$

From the above two equations it becomes clear than choosing $l^{(2)}$ as the contrast vector leads to a very familiar test statistic. The numerator of test statistic T_2 considers the difference in average observed values of expression levels and its denominator represents the variance of $\bar{Y}^T - \bar{Y}^C$ based on the mixed linear model.

It is also important to study the effect of the contrast vector on the power of tests. The two null hypotheses presented in (2.6) and (2.9) are different and therefore the usual power analysis cannot be applied to choose the right test. However, when $\Lambda^C = \Lambda^T = \Lambda$, the hypothesis presented in (2.6) is a special case of (2.9) (assuming that Λ has full rank) and it is possible to compare the powers of the two tests in this special case. When $\Lambda^C = \Lambda^T = \Lambda$, the null and alternative hypotheses are given in (2.6) and the test statistics T_1 and T_2 have the following simplified forms:

$$T_1 = \frac{\mathbf{b}\left(\Lambda^{-1}\left(\bar{Y}^T - \bar{Y}^C\right)\right)}{\sqrt{\mathbf{b}\left(\frac{1}{n_1}+\frac{1}{n_2}\right)\left(\hat{\sigma}_\gamma^2 I + \hat{\sigma}_\varepsilon^2\left(\Lambda'\Lambda\right)^{-1}\right)\mathbf{b}'}} \tag{2.12}$$

$$T_2 = \frac{\mathbf{b}\left(\bar{Y}^T - \bar{Y}^C\right)}{\sqrt{\mathbf{b}\left(\frac{1}{n_1}+\frac{1}{n_2}\right)\left(\hat{\sigma}_\gamma^2\left(\Lambda\Lambda'\right) + \hat{\sigma}_\varepsilon^2 I\right)\mathbf{b}'}} \tag{2.13}$$

From these equations, we can see that when no underlying network structure is taken into account, $(\Lambda = I)$ the two test statistics are the same. However, if there is an underlying network structure $(\Lambda \neq I)$, the test statistic in (2.13) represents the likelihood ratio test for testing the null hypothesis in (2.6), which is asymptotically most powerful. On the other hand, as $\left\|\Lambda^T - \Lambda^C\right\|$ increases, the test presented by $l^{(1)}$ will no longer be appropriate and we could expect $l^{(2)}$ to have a better performance.

In the more general case, where $\Lambda^C \neq \Lambda^T$, it is desirable for the test statistic to account for all of the interactions between genes in the specific subnetwork and to not include any effects from genes outside the subnetwork. Consider again the simple network of Figure 2.3, where the subnetwork of interest consists of X_2 and X_3 i.e. $b = (0, 1, 1)$. s

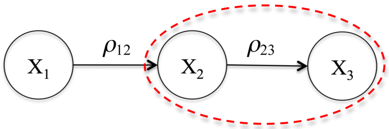


Figure 2.3: Illustration of the Network contrast vector on a simple network. Red dashed line indicates the interactions that are included in the contrast vector

It is then easy to see that

$$(\mathbf{b}\Lambda) = (\rho_{12} + \rho_{12}\rho_{23}, 1 + \rho_{23}, 1)$$

includes all the interactions among nodes connected to the subnetwork, while the proposed network contrast vector

$$(\mathbf{b}\Lambda \cdot \mathbf{b}) = (0, 1 + \rho_{23}, 1)$$

corresponds to the desired interactions. The following result describes a choice of a contrast vector that achieves this goal.

Lemma II.5. *Consider a $1 \times p$ indicator vector $\mathbf{b}$ and let $x \cdot y$ represent the element-wise product of x and y.*

Then $(\mathbf{b}\Lambda \cdot \mathbf{b})\gamma$ includes the effects of all the nodes in $\mathbf{b}$ on each other, but it is not affected by any node outside of the set of nodes indexed by $\mathbf{b}$.

Proof. Let $I_{\mathbf{b}} = \{i : \mathbf{b}_i = 1\}$. Based on the latent variable model, the jth column of Λ includes the influences of node j on all other nodes in the network. Therefore, $(\mathbf{b}\Lambda)_j$ is the influence of the jth node on all nodes in $\mathbf{b}$. Also, note that $\Lambda_{ii} = 1$ for all i and Λ_{ji} is non-zero only if there is a path from j to i.

Thus,

$$(\mathbf{b}\Lambda)_j = \begin{cases} \sum_{i \in I_{\mathbf{b}}} \Lambda_{ji} & j \notin I_{\mathbf{b}} \\ 1 + \sum_{i \in I_{\mathbf{b}}, i \neq j} \Lambda_{ji} & j \in I_{\mathbf{b}} \end{cases}$$

But $(\mathbf{b}\Lambda \cdot \mathbf{b})_j$ is non-zero only if $j \in I_{\mathbf{b}}$ and therefore

$$(\mathbf{b}\Lambda \cdot \mathbf{b})\gamma = \sum_{j \in I_{\mathbf{b}}} \gamma_j + \sum_{j \in I_{\mathbf{b}}} \sum_{i \in I_{\mathbf{b}}, i \neq j} \Lambda_{ji}\gamma_j$$

which means that $(\mathbf{b}\Lambda \cdot \mathbf{b})\gamma$ only includes the effects of elements of $\mathbf{b}$ on each other. □

The estimated β's in the latent variable model reflect the individual effect of each gene and therefore, can be thought of as the “pure signals”. Based on Lemma II.5, in order to include interactions among genes in each subnetwork and prevent any confounding effects, we define the *network contrast vector* by

$$l^{(N)} = \left(-\mathbf{b} \cdot \mathbf{b}\Lambda^{C}, \mathbf{b} \cdot \mathbf{b}\Lambda^{T}\right)$$

2.3.5 Comparison with other Gene Set Analysis techniques

In this section, we discuss the main differences between the approach proposed in this chapter, and the idea of gene set enrichment analysis (GSEA) presented in Subramanian et al. (2005) and generalized by Efron and Tibshirani (2007).

Permutation based methods of gene set analysis, including GSEA, first compute an association measure relating the expression levels of each gene in the list to the phenotype (e.g. the p-value from the two sample t-test). The individual association measures are then combined into an *enrichment score* for each gene set (GSEA uses a version of Kolmogorov-Smirnov test statistic, while a maxmin function is used in GSA). The main strength of the GSEA method, that is also inherited by its extensions, is that the correlation structure of genes in the gene set is preserved, and the permutation based distribution of the enrichment score also represents the correlation among genes. However, these methods compute the individual association measures of each gene separately and do not directly include the correlation among genes when calculating the enrichment score.

Alternatively, if efficient estimation of the covariance matrix is possible, parametric test statistics may be used to test the difference between the expression levels of the two treatment groups. This is not usually possible since in most microarray analysis applications the number of parameters needed to be estimated is considerably larger than the number of samples available ($n \ll p$). However, the external information about the underlying gene network can make this estimation problem tractable. For instance, in the mixed linear model proposed in this chapter, the covariance matrix is modeled as a function of few parameters which can be efficiently estimated from the data. Thus, it is possible to test the significance of each gene set using tests that include the expression levels of *all* genes in the gene set and also directly incorporates the covariance structure of the genes in each subnetwork. An example of such a test statistic is the T_2 test statistic discussed in Section 2.3.4, which is a version of the two-sample t-test. If the model is correctly specified, one could expect such a test statistic to be sensitive to changes in both the expression levels and also in the covariance structure. However, in the absence of external information about the network, estimation of the covariance matrix may be impractical and non-parametric methods like GSEA, may offer better inference properties.

In the next section, we carry out simulation studies to illustrate the difference between the proposed model and the GSEA method. We will also examine the effect of the choice of the contrast vector on the performance of the proposed test statistic.

2.4 Performance Analysis

Three sets of simulation studies are considered in this section. In the first simulation, we study different choices of contrast vectors and compare their performance with GSEA in a simple network. The second simulation study is designed to analyze the combined effect of change in mean and covariance between control and treatment conditions. In the last simulation, we evaluate the sensitivity of the proposed inference procedure to the presence of noise in the association weights. Note that in simulation studies of this section, it is assumed that the effect of the gene network is appropriately modeled using the latent variable model of Section 2.2 and that the topology of the network is correctly specified.

Table 2.2: Settings of mean parameters for the first simulation study.

Scenario	Mean Parameters
1	$\mu^T = \mu^C = 0$
2	$\mu^T = 2$ for top one-third levels of the tree, $\mu^T = \mu^C = 0$ for rest
3	$\mu^T = 2$ for top two-third levels of the tree, $\mu^T = \mu^C = 0$ for rest
4	$\mu^T = 2$ in the left branch of the tree (including the root node), $\mu^T = \mu^C = 0$ in the right branch

2.4.1 Simulation 1: Different Choices of Contrast Vector

In the first setting, a simple network structure consisting of an 8-level binary tree with 255 nodes is used. It is assumed that there are no interactions in the network under the control condition $(\Lambda^C = I)$. Under the treatment condition, genes on the network are assumed to be positively correlated with different association strengths: The association for the first 3 level of the genes in the network (top 7 genes in the tree) is assumed to be 0.8, genes in the next three levels (56 genes) have association equal to 0.5 and the remainder of the genes are weakly associated with $\rho = 0.2$. Under control, the mean vector for mRNA expression levels of genes is set to zero $(\mu^C = 0)$. Scenarios for mean expression levels under treatment are presented in Table 2.2 and Gene sets considered in this simulation are given in Table 2.3. The gene sets are chosen so that for each mean scenario there exists gene sets with highly expressed genes and also gene sets that represent non-differentially expressed genes.

Table 2.4 presents the estimated powers of the GSEA method and tests based on the three contrast vectors, $l^{(1)}$, $l^{(2)}$ and $l^{(N)}$, introduced in Section 3.3 based on 1000 simulations. The powers are calculated based on the FDR controlling procedure of Benjamini and Hochberg (1995) with a q-value of 0.05.

The positive correlation structure of the network affects the significance of the subnetworks selected for this comparison. When a specific gene in the network becomes differentially expressed, the other genes in the network that are influenced

Table 2.3: Gene sets considered in the first simulation study[2].

Gene Set	Genes considered
1	All genes in the network
2	top one-third levels of the tree
3	first two-third levels of the tree
4	the last level of the tree
5	left branch of the tree (including the root)
6	right branch of the tree (excluding the root)
7	20 percent of genes in the network selected randomly

by that gene will also have modified expression levels in the same direction and the combined subnetwork becomes strongly significant. This propagation mechanism explains the abundance of powers of 1 in the table. The first mean scenario in this study corresponds to the case that $\Lambda^C \mu^C = \Lambda^T \mu^T$. All the methods have nominal significance level of 0.05 for this test. On the other hand, there are some differences between the tests based on different contrast vectors and the GSEA method. As one expects from the discussion in Section 2.3.4, the test based on $l^{(2)}$ has higher power than the test based on $l^{(1)}$. It can also be seen that in all but one case, the power resulted from test based on $l^{(2)}$ is higher than the power for the GSEA method verifying the discussion of Section 2.3.5. There are few cases that deserve special attention. The GSEA method indicates no power for testing all the genes in the network under scenario 2. However, in this case the top 1/3 levels of the tree are significant and therefore it is natural to expect significant differences in overall expression levels. The same pattern can be observed when comparing the two methods for testing the right branch of the tree under the second scenario and the top 1/3 of genes under the third scenario. On the other hand, the test based on $l^{(2)}$ has a high false positive rate for testing the right branch of the tree in the situation where only the left branch is up-regulated (scenario 4) while the GSEA method correctly shows no deviation from the null hypothesis. The same phenomenon can be seen for the results of testing the last level of the tree in the case where the top 2/3 levels of

Table 2.4: Results of the first simulation study: Powers of tests based on GSEA and three contrast vectors for different mean scenarios. Multiple testing adjustment is based on FDR with $q^* = 0.05$. *Italic RED* entries indicate powers that are lower or higher than expected and numbers in **Bold GREEN** show powers close to values expected from the simulation design.

Scenario	Method	All	Top 1/3	Top 2/3	Last level	Left branch	Right branch	Random
1	GSEA	.000	.000	.000	.000	.000	.000	.000
	$l^{(1)}$	.024	.015	.014	.014	.019	.022	.018
	$l^{(2)}$	.023	.020	.015	.012	.011	.023	.019
	$l^{(N)}$	.022	.021	.015	.010	.011	.021	.018
2	GSEA	*.000*	1.000	1.000	.000	1.000	*.000*	**.000**
	$l^{(1)}$	*.119*	1.000	*.535*	.047	*.127*	*.056*	**.046**
	$l^{(2)}$	**1.000**	1.000	1.000	.090	.980	*.956*	*.523*
	$l^{(N)}$	**1.000**	1.000	1.000	.070	.979	**.562**	**.067**
3	GSEA	1.000	*.000*	1.000	.000	1.000	1.000	1.000
	$l^{(1)}$	1.000	**1.000**	1.000	.089	1.000	1.000	.999
	$l^{(2)}$	1.000	**1.000**	1.000	.568	1.000	1.000	1.000
	$l^{(N)}$	1.000	**1.000**	1.000	0.089	1.000	1.000	1.000
4	GSEA	1.000	*.000*	1.000	1.000	1.000	**.000**	1.000
	$l^{(1)}$	1.000	**.997**	1.000	1.000	1.000	**.089**	1.000
	$l^{(2)}$	1.000	**1.000**	1.000	1.000	1.000	*.476*	1.000
	$l^{(N)}$	1.000	**1.000**	1.000	1.000	1.000	**.089**	1.000

the tree are significant. The test based on $l^{(2)}$ is not able to isolate the significance of the genes under consideration from the effect of other genes in the network and can therefore result in high false positive rates. As expected based on Lemma II.5, the test based on $l^{(N)}$ resolves these shortcomings. The power of this test is close to the nominal significance level for testing the above two cases while it offers a high power in cases where the GSEA method fails to distinguish the significance of the subnetworks.

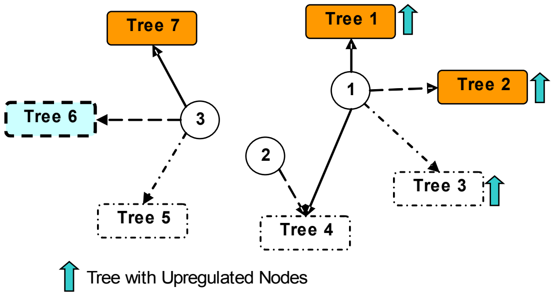


Figure 2.4: Design of the second simulation study. Solid arrows and boxes represent high positive association (0.6 here); dashed arrows and boxes represent high negative association (-0.6); dotted arrows and boxes indicate low positive association (0.1). The root genes 1 and 2 are up regulated while the expression level for gene 3 does not change.

Table 2.5: Results of the second simulation study. Estimated powers for the GSEA and the test based on $l^{(N)}$ for different mean scenarios and different subnetworks. In results for each subnetwork, the first row represents the power for the GSEA method and the second row displays the power for the test based on $l^{(N)}$. Settings of fonts and colors are similar to Table 2.4.

Mean Increase:		0	0.2	0.4	0.6	0.8	1.0
Tree 1	GSEA	1.000	1.000	1.000	1.000	1.000	1.000
	NetGSA	1.000	1.000	1.000	1.000	1.000	1.000
Tree 2	GSEA	1.000	1.000	1.000	1.000	1.000	1.000
	NetGSA	1.000	1.000	1.000	1.000	1.000	1.000
Tree 3	GSEA	.000	*.000*	*.000*	*.000*	*.000*	1.000
	NetGSA	.2500	**.9580**	**1.00**	**1.00**	**1.00**	1.000
Tree 4	GSEA	.000	.000	.000	.000	.000	.000
	NetGSA	.263	.277	.298	.278	.298	0.295
Tree 5	GSEA	.000	.000	.000	.000	.000	.000
	NetGSA	.281	.296	.290	0.297	0.305	0.281
Tree 6	GSEA	*.000*	*.000*	*.000*	*.000*	*.000*	*.000*
	NetGSA	**.982**	**.984**	**.986**	**.980**	**.978**	**.976**
Tree 7	GSEA	*.000*	*.000*	*.000*	*.000*	*.000*	*.000*
	NetGSA	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**

2.4.2 Simulation 2: Simultaneous Changes in Mean and Covariance

The second simulation study is designed to evaluate simultaneous changes in expressions levels as well as associations among genes. The network structure in this simulation consists of 3 root nodes and 7 five-level trees (220 genes total). The network consists of low and high association subnetworks and also includes both positive and negative correlations. Three of the subnetworks are considered to be differentially expressed (the level of expression increases in increments of 0.2) and the other subnetworks have equal values of mean in treatment and control conditions. Figure 2.4 illustrates the setting of parameters of this simulation study.

Table 2.5 presents the estimates of powers for the GSEA method and the test based on $l^{(N)}$ for testing different trees with increasing expression levels in a simulation with 1000 repetitions. It can be seen from the results that both of these methods reject the null hypothesis for tests related to trees with high positive correlation (subtrees 1, 2 and 7 in Figure 2.4). The GSEA method can only detect the significance of subtree 3 for large values of increase in the expression level while the test based on $l^{(N)}$, can detect this change for smaller values of increase. Subtrees 4 and 5 correspond to cases where the correlation among genes is minimal. Subtree 4 is affected by root genes 1 and 2 that are both up regulated but they have opposite correlations with genes in subtree 4. As one would expect, the powers for subtree 4 are similar to those of subtree 5 which suggests that the combined effect of genes 1 and 2 on subtree 4 is the same as the effect of gene 3 on subtree 5. Subtree 6 illustrates the fact that the test based on $l^{(N)}$ takes advantage of the known correlation structure even if the genes in the network are negatively correlated while the GSEA method cannot detect the change in the correlation structure between control and treatment conditions.

2.4.3 Simulation 3: Effect of Noise in Network Information

In the last simulation, we evaluate the sensitivity of the proposed inference procedure to presence of noise in association weights of the gene network. The network consists of 4 similar subnetworks, each with 40 genes. Under control, genes have mean $\mu^C = 1$ and the weights of the adjacency matrix are set to 0.2. The settings of the parameters under treatment are given in Table 2.6. The estimated powers of tests of significance of each subnetwork using a test based on $l^{(N)}$ are plotted in Figures 2.5 and 2.6. Figure 2.5 represents the case where the errors are introduced at random, that is, each weight in the adjacency matrix under treatment is perturbed by a uniform noise in the range $[-e, e]$ where e is a value between 0 and 0.4. On the other hand, Figure 2.6 represents the estimated powers of tests when a systematic bias is included in the weights of the adjacency matrix under treatment. It can be seen that if the underlying model is correctly specified, presence of random noise in weights of adjacency matrix will not significantly affect the power of the test. However, presence of systematic bias in the estimated weights can introduce both type I, as well as type II errors. This is illustrated by the increase of power of the test as

Table 2.6: Significant parameters of the third simulation study under the treatment condition. In all other cases $\mu^T = \mu^C = 1$ and $\rho^T = \rho^C = 0.2$.

Subnetwork	Mean	Association Weight
1	$\mu^T = 1.5$	$\rho^T = 0.6$
2	–	$\rho^T = 0.6$
3	–	–
4	$\mu^T = 1.5$	–

the difference between weights under treatment and control becomes more significant (Figure 2.6). It is important to note that the simulation considered here does not include errors in the topology of the network. These errors become more critical if the topology of the network, as well as the association weights, are estimated from expression data, which is beyond the scope of this article.

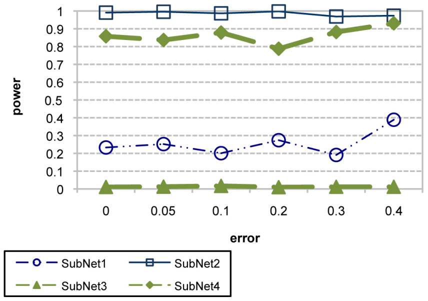


Figure 2.5: Estimated powers of test of significance of subnetworks in presence of *random noise* in weights of the adjacency matrix. Thick green plots represent the powers of subnetworks whose true adjacency matrices in control and treatment are the same.

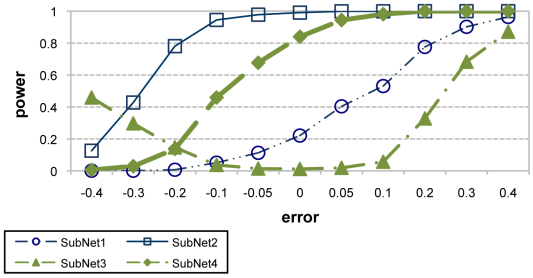


Figure 2.6: Estimated powers of test of significance of subnetworks in presence of *systematic bias* in weights of the adjacency matrix. Thick green plots represent the powers of subnetworks whose true adjacency matrices in control and treatment are the same.

2.5 Analysis of Yeast Galactose Utilization Pathway Data

In Section 2.1.1 we analyzed the yeast GAL pathway data (Ideker data) using the GSEA method, which revealed that the Galactose Utilization pathway is significantly activated in gal+ condition. In that analysis, the external information provided by the network was only used to determine the gene sets of interest. As discussed in Section 2.1.1, the Ideker data also includes strength of gene interactions in the network, Therefore, it is possible to directly incorporate the network information and use the proposed network based inference procedure. It is important to note that the Ideker data only includes one set of association weights for both gal+ and gal– conditions. In other words, in this section we assume $\Lambda^T = \Lambda^C = \Lambda$ and hence the proposed inference procedure can not test the change in the network structure. Assuming that the latent variable model correctly represents the effect of the underlying network, the increased power of the network based procedure is mainly due to directly incorporating the network information.

Table 2.7 compares results of analyzing the Ideker data using the GSEA method and the network based method presented in this chapter (using $l^{(N)}$). This table also includes results of analyzing this data using the GSA method of Efron and Tibshirani (2007) [3]. As one may expect, all three methods find the Galactose Utilization pathway to be statistically significant. Although the GSEA and the GSA methods agree on the significance of other subnetworks, it can be seen from Table 2.7 and Figure 2.7 that including the underlying network structure in the analysis reveals 4 additional significant pathways. Although additional experiments are needed to verify the result of Table 2.7, the biology of yeast cells may offer some insight to significance of newly detected pathways. These pathways can be categorized into

[3] The minmax criteria is used as the enrichment function in the GSA method.

Table 2.7: Analysis of yeast Galactose utilization pathways using GSEA, GSA and NetGSA. Results of NetGSA are based on the proposed network contrast vector $l^{(N)}$. For each method, the nominal p-value and whether the pathway is significant based on the FDR with $q^* = 0.05$ is reported.

		GSEA		GSA		$l^{(N)}$	
		NOM	FDR	NOM	FDR	NOM	FDR
PATHWAY	SIZE	p-val	signif	p-val	signif	p-val	signif
rProtein Synthesis	28	0.5261		0.278		0.0038	✓
Glycolytic Enzymes	16	0.9683		0.357		0.2825	
RNA Processing	75	0.9879		0.386		0.479	
Fatty Acid Oxidation	7	0.4694		0.299		0.0068	✓
O2 Stress	13	0.2384		0.285		0.4448	
Mating, Cell Cycle	58	0.3583		0.417		0.4317	
Vesicular Transport	19	0.07243		0.156		0.3693	
Sugar Transport	2	0.7358		0.458		0.3319	
Glycogen Metabolism	12	0.1321		0.034		0.3057	
Stress	12	0.02004		0.007		0.0000	✓
Metal Uptake	4	0.8374		0.326		0.0802	
Respiration	9	0.1878		0.091		0.0001	✓
Gluconeogenesis	7	0.8455		0.475		0.0383	
Galactose Utilization	12	0.002045	✓	0.001	✓	0.0000	✓
Amino Acid Synthesis	30	0.1853		0.054		0.0665	

two groups: Galactose Utilization and rProtein Synthesis pathways are involved in cell growth in gal+ environment while genes in the Stress, Respiration and Fatty Acid Oxidation pathways are induced in gal– environment. The Stress pathway has a low nominal p-value in both GSEA and GSA results, however, these methods do not consider this pathway significant. The significance of the Stress pathway is not surprising and can be explained by the fact that galactose is a more efficient source of carbon than raffinose. Thus in absence of galactose (gal–), the genes in the Environmental Stress Response (ESR) are induced Gasch et al. (2000); Gasch and Werner-Washburne (2002). The Fatty Acid Oxidation and Respiration pathways are also upregulated in gal– environment. The genes in the Respiration pathway are among the genes that are induced in the ESR.

Many of the stress defense mechanisms consume ATP and therefore, cellular stress could lead to the induced expression of respiration genes Hohmann and Mager (2003). Also, many genes involved in importing and exporting fatty acids are induced in ESR and the induction of these genes can increase the local concentration of fatty acids, which in turn may induce the expression of genes in Fatty Acid Oxidation pathway Hohmann and Mager (2003). The induction of Fatty Acid Oxidation and Respiration genes can be further explained by the coregulation of genes in these pathways. It should be noted that two of the genes in the Respiration pathway are directly affected by genes in GAL pathway (GAL4 regulates CYC1 and HAP4 is regulated by MIG1) and our proposed model can exploit such relationship in order to gain more statistical power. Finally, the significance of the rProtein Synthesis genes can be explained by growth dependent expression of these genes and the fact that ESR represses the expression of many protein synthesis genes Hohmann and Mager (2003).

2.6 Discussion

Finding significant subnetworks and pathways that are involved in certain biological phenomena has been the focus of many new studies. The main challenge is to formulate the null and alternative hypotheses that consider the change in the expression levels of the genes as well as the change in the network structure in response to environmental factors. In this chapter, we proposed a model-based approach for testing the significance of biological pathways using the underlying gene network and studied graph theoretic properties of the model. Our approach uses external information available about the underlying network and it hence depends on availability and quality of such data. The method proposed in this chapter, incorporates the weighted adjacency matrix of the network through a latent variable model and uses a flexible mixed linear representation. We discussed that the inference based on this method depends on the choice of the contrast vector and proposed a choice that offers improvement in power of the test compared to the GSEA method of Subramanian

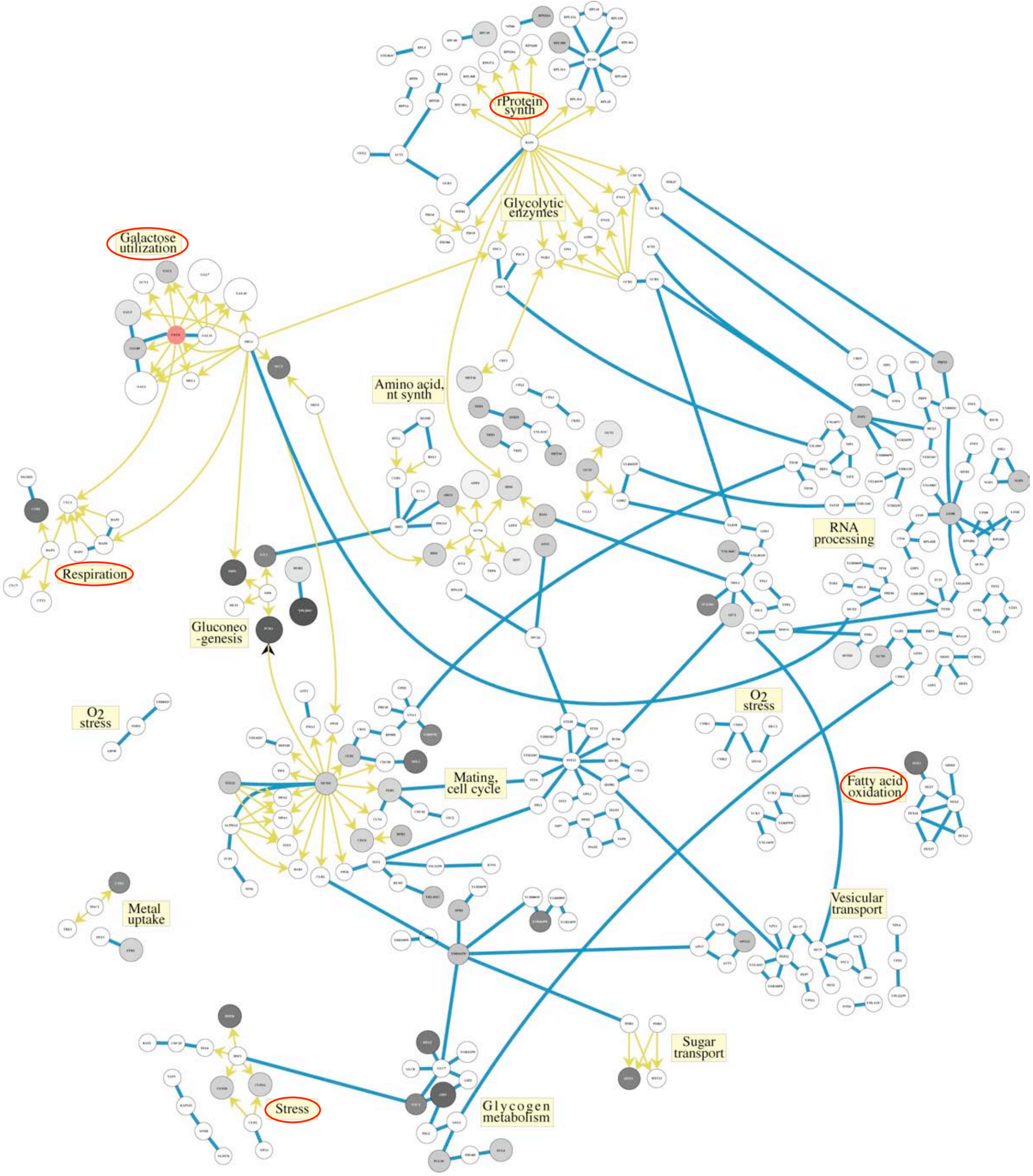


Figure 2.7: Significant pathways in Galactose utilization. Significant pathways have been marked with RED ovals.

et al. (2005). The simulation studies and the analysis of the yeast galactose utilization pathway reveal the ability of the proposed method in identifying significant pathways that are otherwise difficult to distinguish. Although the focus of this chapter was on testing the significance of subnetworks in the two population inference problem, the proposed method provides a general framework for studying a variety of phenotypes including analysis of time series mRNA data and the change in the network over time. More generally, different correlation structures among observations can be implemented in the mixed linear model and therefore, different types of data can be modeled using this framework. Considering parameters for environment factors and gene-gene and gene-environment interactions is also a straight forward extension of the proposed model.

The model presented in this chapter relies on two main assumptions: (a) The relationship between the expression levels of genes in the network can be represented linearly using the influence matrix of the network and (b) that the data follows a normal distribution. Although the first assumption is a crucial part of this analysis, the second assumption can be relaxed using the Generalized Mixed Linear Model (GMLM) framework. However, this would make the computational aspects of the problem more challenging.

The growth of information available on the underlying biological networks calls for effective methods that can utilize such information efficiently and requires extensions of statistical methods appropriate for studying of network structures. The model presented in this chapter requires external information on the weighted adjacency matrix of the network. Although more data is becoming available on gene and protein networks, many available network data only include the binary association among genes (network topology) and do not include information about the strength or direction of associations among genes. The problem of estimating the weighted adjacency matrix of the network, which is related to estimation of the covariance matrix, is of separate interest and is beyond the scope of this chapter. Chaudhuri et al. (2007) propose an efficient algorithm for estimating the association among genes when the topology of the network is known. The method proposed in this chapter can also be extended to the cases where only partial information about the network is available.

CHAPTER III

Network Enrichment Analysis in Complex Experiments

3.1 Introduction

Recent advances in high throughput technologies have facilitated the simultaneous study of components of complex biological systems. Microarray technologies provide information about the expression levels of virtually all genes in the genome of a given specie; the patterns of changes in these expressions over large groups of genes can determine how living organisms respond to their environment. However, genes interact with each other in an orchestrated fashion and analysis of individual genes without taking into account their interactions (*single gene analysis*) may result in reduced efficiency and bias. We provide next an overview of two classes of methods that aim to overcome this shortcoming and discuss advantages and disadvantages of each of the methods.

3.1.1 Background

Two classes of models have been investigated by researchers in order to account for interactions among components of biological systems in the differential analysis of genes and proteins. The first approach, known as *gene set analysis*, is to consider the joint effect of biologically related groups of genes. By performing gene set analysis, one can hope that when the combined effect of genes in a set is considered, the interactions among the genes are preserved and hence the resulting inference procedure implicitly includes such interactions. In addition, while individual genes may not show important changes of expression, the combined effect of changes in expressions of genes in a set (e.g. a genetic pathway) could unveil important changes in the state of the system. Hence, methods of gene set analysis offer improvements in both power, as well as interpretability of inference procedures. Examples of methods for gene set analysis include the Gene Set Enrichment Analysis (GSEA) of Subramanian et al. (2005) and its variants (Tian et al., 2005; Efron and Tibshirani, 2007), which use a permutation-based procedure in order to evaluate the significance of gene sets.

The second class of methods aims to directly incorporate available information about interactions among genes and proteins into differential analysis. Gene networks are efficient tools to represent and model interactions among genes (Rahnenführer et al., 2004) and have been used to improve the performance of differential analysis methods. Ideker et al. (2001) used integrated genomic and proteomic analysis of perturbed networks to discover interactions among genes. This was followed by proposing a method to test the significance of subnetworks through a permutation-based method (Ideker et al., 2002). Recently, Wei and Li (2007) and Wei and Pan (2008) have proposed Markov random field models to incorporate the network information in the differential analysis of genes. In these methods, connected genes in the networks are assumed to have “similar” expression levels and a Bayesian framework is developed using mixture models to evaluate whether each gene is differentially expressed.

A number of methods have recently been developed to combine the advantages of incorporating network information with strengths of enrichment analysis. Sanguinetti et al. (2008) considered a mixture model on graphs (MMG) to account for network information in proteomic data and used a simple percolation algorithm to define submodules. In Chapter II, we discussed a method that incorporates network information through a latent variable model and used the framework of mixed linear models (MLM) to test whether *a priori* defined gene sets are differentially expressed. They considered two special classes of networks, namely directed acyclic graphs (DAGs), as well as sub-stochastic graphs and proposed a test statistic for the two-class inference problem (e.g. treatment and control).

The above models can all be viewed as attempts to incorporate the spatial correlation caused by the gene network into the analysis of differentially expressed genes. Another important aspect of gene expression is the dynamic behavior of genes in response to environmental conditions (Gasch et al., 2000; Gasch and Werner-Washburne, 2002). The changes in gene expression levels over time may reveal unique features of biological systems that are not evident from studying gene expressions at a single time point. The temporal correlation among gene expressions can also be utilized to improve the efficiency of finding differentially expressed genes. Examples of models for time course gene expression data include Hong and Li (2006) and Yoneya and Mamitsuka (2007). *Spatio-temporal* models for gene expression analysis combine the advantages of both models. Wei and Li (2008) recently proposed a hidden spatio-temporal Markov random field model to account for both temporal correlation among expression levels, as well as spatial correlation among genes represented by the gene networks.

3.1.2 Outline

Currently available methods, reviewed above, focus either on incorporating network information for performing single gene analysis, or on gene set enrichment analysis for simple experimental conditions; e.g. treatment and control. Since methods of enrichment analysis are based on permutations tests (e.g. Ideker et al., 2002; Subramanian et al., 2005; Tian et al., 2005; Efron and Tibshirani, 2007), their extension to complex experimental settings, including the presence of temporal correlation among observations, is not straightforward.

In this chapter, I generalize the framework of Chapter II in order to develop a flexible framework for analysis of gene sets in complex experimental conditions, while incorporating the known network information. In particular, I

(a) propose a generalization of the network influence to analyze arbitrary networks with both directed, and undirected edges,

(b) exploit the flexibility of mixed linear models to develop a general inference procedure that can be used to analyze changes in biological pathways in complex experiments, including experiments with multiple factors together with time course data, and

(c) describe an inference framework for simultaneous tests of multiple hypotheses for analysis of pathways in complex experiments.

In addition, in order to estimate the parameters of the resulting mixed linear model, I propose an iterative algorithm based on the block-relaxation technique (de Leeuw, 1994), which will be the basis for developing a distributed algorithm for estimation of model parameters in Chapter VII. Finally, we study the effect of noise in the underlying network information, e.g. when interactions among genes or the associated weights are estimated, and establish conditions under which the proposed inference procedure is asymptotically insensitive to such noise. Through analysis of simulated, as well as real, data examples, we illustrate the small sample properties of the proposed inference procedure and show that the model performs well in the presence of limited samples (the application discussed in Section 3.4 has a single sample per experimental condition and time point, and includes 3 time points) and also exhibits good performance in the analysis of small gene sets.

The remainder of the chapter is organized as follows: in Section 3.2, the modeling framework is introduced and the mixed linear model representation is presented. The material in Sections 3.2.1 and 3.2.2 generalize the framework of Chapter II to analysis of general networks in complex experiments. Estimation and inference issues are discussed in Sections 3.2.3 and 3.2.4, respectively, and the asymptotic analysis of performance under noisy network information is presented in Section 3.2.5. The performance of the model is evaluated through simulation studies in Section 3.3. In particular, it is shown that while the performance of enrichment methods deteriorates in presence of temporal correlation, the proposed model can effectively handle the additional correlation. Finally, in Section 3.4, data from yeast environmental stress response (ESR) experiment of Gasch et al. (2000) are used to discover pathways that are differentially expressed in response to these stress factors. Section 3.5 summarizes the main findings and discusses some future research directions.

3.2 Model and Methods

Consider p genes whose expression data $\mathcal{D}$ is organized in a $p \times n$ matrix, where each column of $\mathcal{D}$ represents a realization of the expression levels of genes in the study. In general, assume that there are K different experimental conditions and each of $k = 1, \cdots, K$ conditions are studied in J_k time points. Further, assume that for each combination of experimental condition and time, there exists n_{jk} samples. Let $n = \sum_{k=1}^{K} \sum_{j=1}^{J_k} n_{jk}$ and denote by Y an arbitrary column of the expression matrix $\mathcal{D}$. In other words, Y consists of the expression levels of genes in the study for a given time point of a specific experimental condition.

3.2.1 The Latent Variable Model

In order to incorporate the network structure into the model, we represent the gene network by a directed graph $G = (V, E)$ with vertex set V, and edge set E. The edge set is captured in the $p \times p$ weighted adjacency matrix of the graph A, with positive and negative entries. Each nonzero element in the adjacency matrix, A_{ij}, represents a directed edge whose weight corresponds to the strength of association between the two vertices i and j. Undirected graphs correspond to a special case, where $A_{ij} = A_{ji}$. As in Chapter II, we represent the gene network by a directed graph $G - (V, E)$ with vertex set V and edge set E, and adapt the latent variable model of Chapter II.

In most applications, the available network information comes in the form of the adjacency matrix, which determines the association between each gene and its immediate neighbors in the graph. On the other hand, the influence matrix represents the effect of each gene on all the other genes in the network and is given by $\Lambda = \sum_{r=0}^{\infty} A^r$, where $A^0 = I_p$. In Chapter II, we showed that for the case of directed acyclic graphs, the relationship between Λ and A is given by $\Lambda = (I - A)^{-1}$. We also show that if the adjacency matrix of the network is sub-stochastic, i.e. its eigenvalues are smaller than 1 in magnitude, the above relationship between A and Λ still holds. This approach can also be adapted to define a latent variable model for chain graphs, where the network consists of undirected subgraphs that are connected by directed edges having no directed cycles (see e.g. Lauritzen, 1996). However, general gene networks, with both directed and undirected edges, may not satisfy the requirements of any of the above special classes of graphs. Therefore, an alternative approach is required to define the influence of the network for general graphs.

We start by normalizing the adjacency matrix A, by dividing its entries A_{ij} by the corresponding row sum of the absolute values $\sum_{j=1}^{p} |A_{ij}|$. Formally, let

$$\mathcal{L}(\zeta)_{ij} = \frac{A_{ij}}{(\sum_{j=1}^{p} |A_{ij}|) + \zeta}, \qquad \text{for some } \zeta > 0. \tag{3.1}$$

Then by Gershgorin's Disk Theorem (see e.g. Friedberg et al., 1996), the matrix $\mathcal{L}$ is sub-stochastic, and therefore, using the results in Lemma II.1, for each $\zeta > 0$ we get $\Lambda = (I - \mathcal{L}(\zeta))^{-1}$. Taking the limit, we get

$$\Lambda = \lim_{\zeta \to 0} (I - \mathcal{L}(\zeta))^{-1}.$$

This implies that, for general networks, the influence matrix of the graph can be defined as $\Lambda = (I - \mathcal{L})^{+}$, where $\mathcal{L}$ represents the normalized adjacency matrix with $\zeta = 0$ and $(I - \mathcal{L})^{+}$ denotes the Moore-Penrose pseudo-inverse of $I - \mathcal{L}$.

The normalization in (3.1) is motivated by the definition of the covariance matrix in Markov random fields (see e.g. Rue and Held, 2005). For undirected graphs with only positive weights in A, the matrix $I - \mathcal{L}$ also corresponds to a version of the Laplacian matrix of the graph (see e.g. Chung, 1997). Simulation studies show that small values of ζ (e.g. $\zeta \approx 0.01$) do not affect the outcome of the analysis, and $\Lambda = (I - \mathcal{L}(\zeta))^{-1}$ can be used to define the influence matrix.

3.2.2 Mixed Linear Model Representation

As a generalization of the model in Chapter II, consider the gene expression matrix of the previous section with K experimental conditions, $J_k, k = 1 \cdots K$ time points and n_{jk} observations at each combination of condition and time point. Let $\mathbf{Y}$, $\boldsymbol{\gamma}$ and $\boldsymbol{\varepsilon}$ represent the rearrangement of vectors Y, γ and ε into $np \times 1$ column vectors.

Then, using the framework of mixed linear models, we can write

$$\begin{aligned} \mathbf{Y} &= \mathbf{\Psi}\boldsymbol{\beta} + \mathbf{\Pi}\boldsymbol{\gamma} + \boldsymbol{\varepsilon} \\ \boldsymbol{\varepsilon} &\sim N_{np}(\mathbf{0}, \mathbf{R}), \qquad \mathbf{R} = \operatorname{diag}(R) \\ \boldsymbol{\gamma} &\sim N_{np}(\mathbf{0}, \mathbf{G}), \qquad \mathbf{G} = \sigma_\gamma^2 \mathbf{I_{np}} \end{aligned} \tag{3.2}$$

Here $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ are fixed and random effect parameters, and $\mathbf{\Psi}$ and $\mathbf{\Pi}$ are the corresponding design matrices of dimensions $np \times Kp$ and $np \times np$, respectively.

The precise form of these matrices depends on whether the influence matrix Λ can change over time or over different experimental conditions (see Harbison et al. (2004) for examples of changes in regulatory networks in different experimental conditions). To account for such changes in interactions among genes, let $\Lambda^{(jk)}$ denote the influence matrix of the network in condition k and time point j. The changes in network influence over time or in different experimental conditions can then be directly incorporated in the model by replacing the matrix Λ by the corresponding matrix $\Lambda^{(jk)}$ in design matrices $\mathbf{\Psi}$ and $\mathbf{\Pi}$, for fixed and random effect components in the mixed linear model of equation (3.2). Using this notation, $\mathbf{\Pi}$ is a block diagonal matrix with $\Lambda^{(jk)}$ on the diagonal, while $\mathbf{\Psi}$ is defined based on the setting of the experiment. More specifically, suppose χ is the design matrix of the linear regression model for a single gene, corresponding to K experimental conditions and J time points. The matrix $\mathbf{\Psi}$ is then defined by replacing each χ_{jk} with $\chi_{jk}\Lambda^{(jk)}$. In the special case of $\Lambda^{(jk)} = \Lambda$, the matrices $\mathbf{\Psi}$ and $\mathbf{\Pi}$ are defined as

$$\begin{aligned} \mathbf{\Psi} &= \chi \otimes \Lambda \\ \mathbf{\Pi} &= I_n \otimes \Lambda \end{aligned} \tag{3.3}$$

where $\otimes$ denotes the Kronecker product of two matrices. Examples of the use of the mixed linear model for different experimental conditions are provided in Sections 3.3 and 3.4.

This model provides a general framework for evaluating changes in gene expressions in different experimental conditions over time. The structure of the experiment may be fairly complex, corresponding to a factorial design or a block design (Kerr and Churchill, 2001b,a; Yang and Speed, 2002). Examples of such designs arise in the experiments of Gasch et al. (2000), Causton et al. (2001) and Gasch and Werner-Washburne (2002). Further, the model facilitates the specification of correlations caused by both the gene network, as well as temporal dependence among gene expressions. In fact, using the covariance matrices $\mathbf{R}$ and $\mathbf{G}$, a variety of correlation structures can be modeled (in Section 3.4, we provide detailed definition of design and covariance matrices for the analysis of yeast ESR data). In addition, the proposed model allows researchers to investigate the patterns of changes of expressions in different experimental conditions, and to study the expression profiles of gene sets over time, which could provide additional cues to the behavior of biological systems. Such experiments are not easily analyzed using the permutation-based enrichment analysis methods.

3.2.3 Parameter Estimation

It is easy to see that for the mixed linear model of equation 3.2 $\mathbf{W} \equiv \mathrm{Var}(\mathbf{Y}) = \sigma_\gamma^2 \mathbf{\Pi}\mathbf{\Pi}' + \mathbf{R}$ and the maximum likelihood estimate of $\boldsymbol{\beta}$ is given by (Searle, 1971):

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{\Psi}'\hat{\mathbf{W}}^{-1}\mathbf{\Psi}\right)^{-1}\mathbf{\Psi}'\hat{\mathbf{W}}^{-1}\mathbf{Y} \tag{3.4}$$

These estimates depend on estimates of the variance components, σ_γ^2 and $\mathbf{R}$, which are usually estimated via *Restricted Maximum Likelihood* (REML).

Lindstrom and Bates (1988) provided details of the Newton-Raphson and EM algorithms for estimation of parameters of MLM and presented evidence in favor of the former method. They also presented a method of reducing the dimension of the matrices involved in the calculation by breaking down the matrices $\mathbf{\Psi}$ and $\mathbf{\Pi}$ into smaller sub-matrices in case of repeated measures data. In dealing with specific problems, it may be possible to further reduce the dimension of these matrices by taking advantage of their structure and sparsity patterns. However, the size of the parameter vector and dimensions of the matrices involved in the calculations increase with the number of genes p. As a result, traditional methods available for estimation of parameters of MLM prove inefficient in mixed linear models for large gene networks. Therefore, estimation of MLM parameters in (3.2) requires efficient estimation procedures. Algorithm 1, which is a block-relaxation type algorithm (de Leeuw, 1994), makes the estimation of parameters tractable by partitioning the parameter space into smaller subspaces. To simplify the notation, we denote by $\boldsymbol{\theta} = (\boldsymbol{\theta}_\varepsilon, \boldsymbol{\theta}_\gamma)$ the vector of all variance parameters used to define $\mathbf{R}$ and $\mathbf{G}$ i.e. $\mathbf{R} = \mathbf{R}(\boldsymbol{\theta}_\varepsilon)$ and $\mathbf{G} = \mathbf{G}(\boldsymbol{\theta}_\gamma)$. Oberhofer and Kmenta (1974) proved the convergence of this algorithm under certain assumption on the estimates of the variance components. In fact, using the strict convexity of the negative log-likelihood function for mixed linear models, and the general theory of iterative algorithms (de Leeuw, 1994), it can be shown that this algorithm converges to the maximum likelihood estimates of the MLM parameters, provided the estimates of the covariance components result in a positive definite covariance matrix, and $\mathbf{\Psi}$ has full column rank. For the model presented here this is achieved if the variance components are estimated using the REML estimation criterion.

To further speed up the estimation process, one can also partition the estimation further over the subnetworks, which results in partitioning over both parameter, as

Algorithm 1 Block-Relaxation Algorithm for MLM Parameters

```
1. Find an initial estimate of $\hat{\boldsymbol{\beta}}^{(0)}$ (e.g. using OLS)
2. Repeat until convergence $m = 1, 2, \cdots$
   2.1. $\mathbf{e} := \mathbf{e}^{(m+1)} = \mathbf{Y} - \boldsymbol{\Psi}\hat{\boldsymbol{\beta}}^{(m)}$
   2.2. $\hat{\boldsymbol{\theta}}^{(m+1)} = \text{argmax}_{\theta} -\frac{1}{2}\left\{\text{logdet}\left(\mathbf{W}(\boldsymbol{\theta})\right) + \mathbf{e}'\mathbf{W}^{-1}(\boldsymbol{\theta})\mathbf{e}\right\}$
   2.3. $\hat{\boldsymbol{\beta}}^{(m+1)} = \left(\boldsymbol{\Psi}'\mathbf{W}^{-1}(\hat{\boldsymbol{\theta}}^{(m+1)})\boldsymbol{\Psi}\right)^{-1}\boldsymbol{\Psi}'\mathbf{W}^{-1}(\hat{\boldsymbol{\theta}}^{(m+1)})\mathbf{Y}$
```

well as observation spaces. It can be shown that, under specific conditions, estimates from such partitioning converge to the maximum likelihood estimates of the model parameters. We discuss one such algorithm in Chapter VII.

3.2.4 Inference

A variety of hypotheses about fixed effect parameters of mixed linear models can be tested by considering tests of the form:

$$H_0 : l\boldsymbol{\beta} = 0 \quad vs. \quad H_1 : l\boldsymbol{\beta} \neq 0 \tag{3.5}$$

Here l is in general any linear combination of β's which meets the estimability requirement of Searle (1971). An example of such vectors is a *contrast vector*, which satisfies the constraint $\mathbf{1}'l = 0$. In the following discussion, any linear combination of β's satisfying the estimability requirement is referred to as a *contrast vector*. In the setting of multiple experimental conditions, this inference framework allows tests of hypotheses of significance of parameters for each experimental condition, as well as tests of significant changes in responses of gene sets over time. More generally, different combinations of parameters can be tested using this framework, which allow researchers to fully investigate the behavior of gene sets of particular interest.

In Chapter II, we showed that for any given $1 \times p$ indicator vector $\mathbf{b}$ determining a specific subnetwork or gene set, the vector $(\mathbf{b}\Lambda \cdot \mathbf{b})\boldsymbol{\beta}$ includes the effects of all the nodes in $\mathbf{b}$ on each other, but it is not affected by any node outside the set of nodes indexed by $\mathbf{b}$ (here $\cdot$ denotes the Hadamard or componentwise product of two vectors). In words, $\mathbf{b}\Lambda$ introduces the influence of genes indexed by $\mathbf{b}$ on each other, while the componentwise product with $\mathbf{b}$ excludes the effects of nodes not in $\mathbf{b}$. The change in Λ in response to different experimental conditions or over time can be incorporated into this contrast vector by substituting Λ by the influence matrix of the specific time and experimental conditions, $\Lambda^{(jk)}$. Hence the contrast vector l is formed by replacing in the general formula the influence matrix of the network under the specific conditions. As an example, suppose $\Lambda^{(j)}$ represents the influence matrix of the network at time j, $j = 1, \cdots, J$ and $\boldsymbol{\beta} = (\boldsymbol{\beta}^{(1)'}, \cdots, \boldsymbol{\beta}^{(J)'})'$. Then the change in the expression levels of genes in the subnetwork indexed by $\mathbf{b}$ from time j to $j+1$ can be tested using

$$l = (0, \cdots, 0, -\mathbf{b}\Lambda^{(j)} \cdot \mathbf{b},\ \mathbf{b}\Lambda^{(j+1)} \cdot \mathbf{b}, 0, \cdots, 0)$$

Letting $\mathbf{C} = (\boldsymbol{\Psi}'\mathbf{W}^{-1}\boldsymbol{\Psi})^{-1}$, the significance of individual contrast vectors in (3.5) can be tested using the following Wald test statistic:

$$T = \frac{l\hat{\boldsymbol{\beta}}}{\sqrt{l\hat{C}l'}} \tag{3.6}$$

Under the null hypothesis, T follows approximately a t-distribution whose degrees of freedom ν can be estimated using the Satterthwaite approximation method (McLean and Sanders, 1988)

$$\nu = \frac{2(l\hat{C}l')^2}{\tau' V \tau}$$

where $\tau = \frac{\partial}{\partial\theta} lCl'$, and V is the empirical covariance matrix of θ.

When analyzing complex experiments, often multiple contrast vectors of interest are considered for a specific subnetwork. In such situations, (3.6) can be used to test the significance of the contrast vector corresponding to each hypothesis of interest.

The resulting p-values should then be adjusted for the total number of hypotheses tested amongst different subnetworks. Alternatively, one can combine these contrast vectors into a *contrast matrix* L, where each row of L includes one of the contrast vectors. The significance of the subnetwork can then be tested using the following test statistic:

$$F = \frac{\hat{\boldsymbol{\beta}}' L' (L\hat{C}L')^{-1} L\hat{\boldsymbol{\beta}}}{q} \tag{3.7}$$

where q is the rank of L. Under the null hypothesis of $L\boldsymbol{\beta} = 0$, F has an F-distribution with q and η degrees of freedom. To estimate η using the Satterthwaite approximation method, one first needs to find matrices P and D such that $LCL' = PDP'$ (the eigen-decomposition of LCL'). Then, denoting the mth row of L by l_m, η is calculated using:

$$\eta = \begin{cases} \frac{2E}{E-q} & E > q \\ 0 & o.w. \end{cases}$$

where

$$E = \sum_{m=1}^{q} \frac{\nu_m}{\nu_m - 2} I_{\{\nu_m > 2\}}, \qquad \nu_m = \frac{2D_m^2}{\tau_m' K \tau_m}.$$

The proposed F-test for the analysis of complex experiments reduces the number of hypotheses tested and offers a hierarchical testing approach. In particular, although some subnetworks may not show significant change with regard to individual hypotheses, the combined significance of the subnetwork due to multiple sources of differential expression may result in overall significance of the subnetwork. It is then possible to test the significance of individual hypotheses, in case the overall F-test for the subnetwork is significant. We illustrate this hierarchical testing procedure in Sections 3.3 and 3.4.

3.2.5 Uncertainty in Network Information

The method for network-based analysis of gene sets proposed here requires knowledge of interactions among genes and proteins, as well as the corresponding association weights. In addition, to fully exploit the strength of the proposed methodology in testing the changes in the network structure, as well as the expression levels of genes, the adjacency matrix of the network should be available for different experimental conditions and time points. However, available network information may be noisy, and available resources often only determine the presence of interactions among genes, and do not provide information on the strength of associations. Therefore, it may be necessary to estimate the network information. Estimation of gene networks from high throughput observations is an important problem in systems biology and of independent interest (see Chapter IV for a review of related literature). It is important to note that since the network information is used in both estimation of parameters, as well as inference, to prevent unidentifiability and bias, the observations used for estimation of the underlying network should be independent from those used for analysis of differential expression.

In this section, we analyze the effect of uncertainty in the network information, by studying the asymptotic properties of the proposed test statistic. Our main result concerns the general case of error in network information in the case of a two-population test, described in Chapter II. We also discuss the special case of estimating association weights, when the structure of the network is known.

In the following, we denote the available adjacency matrix of the network by $\tilde{A}$ and use the notation $\|A\|$ and $\|A\|_F$ to represent the matrix norm and Frobenius norm of A, respectively. Also, let d_i^A denote the weighted *in-degree* of node i according to the adjacency matrix A: $d_i^A = \sum_j |A_{ij}|$.

Theorem III.1. *Suppose* $\tilde{A} = A + \Delta_A$*, where* $\|\Delta_A\| = o_P(1)$*, and assume that* $\min_i d_i^{\tilde{A}} \geq 1$.[1] *Then,* (3.6) *is an asymptotically most powerful unbiased test for* (3.5)*.*

Proof. We consider here the special case where $\Lambda^C = \Lambda^T = \Lambda$ and only one gene set, the whole network, is tested. This implies that $\mathbf{b} = \mathbf{1}'$ and the proposed network contrast vector $\mathbf{b}\Lambda \cdot \mathbf{b}$ reduces to $\mathbf{b}\Lambda$ (the general case of $\Lambda^C \neq \Lambda^T$ and $\mathbf{b} \neq \mathbf{1}'$ follows from a similar argument).

First, recall that for directed acyclic graphs (DAGs), $\Lambda = \sum_{r=0}^{\infty} A^r$, and for general graphs, $\Lambda = \sum_{r=0}^{\infty} \mathcal{L}^r$, where $\mathcal{L} = D_A^{-1}A$ and $D_A = \mathrm{diag}(d_i^A)$. Then $\tilde{A} = A + \Delta_A$ implies that for DAGs

$$\tilde{\Lambda} = \sum_{r=0}^{\infty} \tilde{A}^r = \sum_{r=0}^{\infty} A^r + \sum_{r=0}^{\infty}\sum_{s=1}^{\infty} A^r \Delta_A^s \equiv \Lambda + \Delta_\Lambda, \qquad \|\Delta_\Lambda\| = o_P(1). \tag{3.8}$$

Similarly, for general graphs, we have

$$\tilde{\mathcal{L}} = D_{\tilde{A}}^{-1}\tilde{A} = D_{\tilde{A}}^{-1}(A + \Delta_A) \equiv \mathcal{L} + \Delta_{\mathcal{L}}$$

where

$$\|\Delta_{\mathcal{L}}\| \leq \|D_{\tilde{A}}^{-1}\|\|\Delta_A\| = 1/(\min_i d_i^{\tilde{A}})\|\Delta_A\| = o_P(1)$$

ex hypothesis. An argument similar to (3.8) implies that the following expression also holds for general graphs

$$\tilde{\Lambda} = \Lambda + \Delta_\Lambda, \qquad \|\Delta_\Lambda\| = o_P(1) \tag{3.9}$$

Now, using the results in Chapter II, the test statistic in (3.6) can be written as

$$T = \frac{\mathbf{b}(\bar{Y}^T - \bar{Y}^C)}{\sqrt{\mathbf{b}(n_1^{-1} + n_2^{-1})(\hat{\sigma}_\gamma^2 \tilde{\Lambda}\tilde{\Lambda}' + \hat{\sigma}_\varepsilon^2 I_p)\mathbf{b}'}} \tag{3.10}$$

where $\bar{Y}^T$ and $\bar{Y}^C$ represent the average expression of genes in the two experimental conditions and n_1 and n_2 represent the corresponding sample sizes. The test

[1] Note that $\min_i d_i^{\tilde{A}} \geq 1$ implies that the network is connected. However, the case of disconnected networks is an straightforward extension, as the networks can be analyzed separately.

statistic in (3.10) represents the likelihood ratio test for testing the null hypothesis in (3.5), which is asymptotically most powerful unbiased, provided correct network information is given. Therefore, to establish the result, it suffices to show that the effect of error in the network information is asymptotically negligible. However, since the numerator of the test in (3.10) does not depend on the network information, it suffices to show that the denominator is a consistent estimator.

To establish the consistency of estimates of the variance components, note that the negative log-likelihood function (up to a constant) for the two-population problem is given by

$$\ell(\theta) = n^{-1}\sum_{i=1}^{n} \operatorname{logdet}(W_i) + n^{-1}\sum_{i=1}^{n} r_i' W_i^{-1} r_i, \tag{3.11}$$

where $r_i = Y_i - \bar{Y}^C, i = 1, \ldots, n_1$, $r_i = Y_i - \bar{Y}^T, i = n_1 + 1, \ldots, n$ and θ is the vector of variance components. Then, using the fact that for the two-class problem with constant, but noisy network information $W_i = \operatorname{Var}(Y_i) = \sigma_\gamma^2 \tilde{\Lambda}\tilde{\Lambda}' + \sigma_\varepsilon^2 I_p$, we get

$$\ell(\theta; \tilde{\Lambda}) = \operatorname{logdet}(\sigma_\gamma^2 \tilde{\Lambda}\tilde{\Lambda}' + \sigma_\varepsilon^2 I_p) + n^{-1}\sum_{i=1}^{n} r_i' (\sigma_\gamma^2 \tilde{\Lambda}\tilde{\Lambda}' + \sigma_\varepsilon^2 I_p)^{-1} r_i. \tag{3.12}$$

Using (3.9) we can then approximate $\ell(\theta; \tilde{\Lambda})$ with its one-term Taylor expansion around Λ

$$\ell(\theta; \tilde{\Lambda}) = \ell(\theta; \Lambda) + \|\Delta_\Lambda\| \operatorname{trace}\left[(\nabla_\Lambda \ell(\theta; \Lambda))' \Delta_\Lambda / \|\Delta_\Lambda\|\right] + o(\|\Delta_\Lambda\|^2) \tag{3.13}$$

where $\nabla_\Lambda \ell(\theta; \Lambda)$ is the gradient of ℓ with respect to Λ (see e.g. Dattorro (2005)-Appendix D for details on directional derivatives and approximations for functions of matrices).

But, for square positive definite matrices we have $\nabla \operatorname{logdet}(X) = X^{-1}\nabla X$ and $\nabla X^{-1} = -X^{-1}\nabla X X^{-1}$. Hence, noting that $\nabla_\Lambda \Lambda\Lambda' = (\Lambda' + \Lambda)$, by the chain rule and after some algebra, we can write

$$\begin{aligned}\nabla_\Lambda \ell(\theta;\Lambda) &= \sigma_\gamma^2(\sigma_\gamma^2\Lambda\Lambda' + \sigma_\varepsilon^2 I_p)^{-1}(\Lambda'+\Lambda)\\ &\quad -n^{-1}\sigma_\gamma^2\sum_{i=1}^n r_i'(\sigma_\gamma^2\Lambda\Lambda'+\sigma_\varepsilon^2 I_p)^{-1}(\Lambda'+\Lambda)(\sigma_\gamma^2\Lambda\Lambda'+\sigma_\varepsilon^2 I_p)^{-1}r_i.\end{aligned}$$

Denote by $\tau^2 = \sigma_\varepsilon^2/\sigma_\gamma^2$ and $\Gamma = \Delta_\Lambda/\|\Delta_\Lambda\|$, and let

$$\begin{aligned}g(\theta) &= \operatorname{trace}\left[\Gamma'(\Lambda\Lambda'+\tau^2 I_p)^{-1}(\Lambda'+\Lambda)\right]\\ &\quad -n^{-1}\sigma_\gamma^{-2}\sum_{i=1}^n r_i'(\Lambda\Lambda'+\tau^2 I_p)^{-1}\Gamma'(\Lambda'+\Lambda)(\Lambda\Lambda'+\tau^2 I_p)^{-1}r_i.\end{aligned} \tag{3.14}$$

Replacing (3.14) in (3.13) then gives

$$\ell(\theta;\tilde\Lambda) = \ell(\theta;\Lambda) + \|\Delta_\Lambda\| g(\theta) + o(\|\Delta_\Lambda\|^2).$$

However,

$$\begin{aligned}|g(\theta)| &\le |\operatorname{trace}\left(\Gamma'(\Lambda\Lambda'+\tau^2 I_p)^{-1}(\Lambda'+\Lambda)\right)|\\ &\quad +n^{-1}\sigma_\gamma^{-2}\sum_{i=1}^n r_i'(\Lambda\Lambda'+\tau^2 I_p)^{-1}\Gamma'(\Lambda'+\Lambda)(\Lambda\Lambda'+\tau^2 I_p)^{-1}r_i\\ &\equiv \text{I} + \text{II}.\end{aligned}$$

Using von Neumann's inequality for the matrix trace (see e.g. Mirsky, 1975), and the relationship between singular values and matrix norms, we get

$$\begin{aligned}\text{I} &\le \sum_{j=1}^p \kappa_{[j]}([\Lambda'+\Lambda]\Gamma')\kappa_{[j]}([\Lambda\Lambda'+\tau^2 I_p]^{-1})\\ &\le p\kappa_{[1]}([\Lambda'+\Lambda]\Gamma')\kappa_{[1]}([\Lambda\Lambda'+\tau^2 I_p]^{-1})\\ &\le p\kappa_{[1]}(\Lambda'+\Lambda)\kappa_{[1]}(\Gamma)\kappa_{[1]}([\Lambda\Lambda'+\tau^2 I_p]^{-1})\end{aligned}$$

where $\kappa_{[j]}(A)$ denotes the j-th largest singular value of A. But, by definition, $\kappa_{[1]}(\Gamma) = 1$. Moreover, by construction, $\kappa_{[1]}(\Lambda'+\Lambda)$ is bounded by say M, and $\kappa_{[1]}([\Lambda\Lambda'+\tau^2 I_p]^{-1}) = 1/(\lambda_{[p]}(\Lambda\Lambda')+\tau^2)$, where $\lambda_{[p]}(\Lambda\Lambda')$ is the smallest eigenvalue of $\Lambda\Lambda'$ and hence is positive (by definition of Λ). This implies that $\text{I} < 2pM/\tau^2$.

On the other hand,

$$\begin{aligned}
\text{II} &\leq \sigma_\gamma^{-2} \|(\Lambda\Lambda' + \tau^2 I_p)^{-1} \Gamma'(\Lambda' + \Lambda)(\Lambda\Lambda' + \tau^2 I_p)^{-1}\| n^{-1} \sum_{i=1}^{n} r_i' r_i \\
&\leq \sigma_\gamma^{-2} \|(\Lambda\Lambda' + \tau^2 I_p)^{-1}\|^2 \|\Gamma'\| \|(\Lambda' + \Lambda)\| \| n^{-1} \sum_{i=1}^{n} r_i' r_i \\
&< \sigma_\gamma^{-2} 2\tau^{-4} M n^{-1} \sum_{i=1}^{n} r_i' r_i = 2\tau^{-2} \sigma_\varepsilon^{-2} M \mathbb{E}(\|r_i\|^2) \quad \text{w.p.1},
\end{aligned}$$

where the last step follows from the strong law of large numbers. This implies that provided the variance components are non-zero, with probability one $g(\theta)$ is bounded, and hence $\|\Delta_\Lambda\| g(\theta) = o_P(1)$. This in turn implies that $\ell(\theta; \tilde{\Lambda}) = \ell(\theta; \Lambda) + o_P(1)$.

Denote by $\mathcal{E}$ the event $[\ell(\theta; \tilde{\Lambda}) = \ell(\theta; \Lambda)]$. Then conditioning on $\mathcal{E}$, the estimates of the variance components are found by minimizing the negative log-likelihood function with true network information, which is a convex function of variance components. M-estimation results in Haberman (1989) imply that $\mathbb{P}(\hat{\theta} = \theta | \mathcal{E}) = 1$ and hence, $\hat{\theta} \rightarrow_P \theta$ as $\tilde{A} \rightarrow_P A$. However, this further implies that as $\tilde{A} \rightarrow_P A$, the denominator of the test statistic in (3.6) converges to the true value, and the result follows. □

Remark III.2. In the general case of complex experiments, the estimates of the fixed effects are also dependent on the network information. A similar result will then follow upon deriving the asymptotic distribution of the numerator of the test statistic in (3.6). In Section 3.3.3, we provide empirical evidence in support of the insensitivity of the proposed inference framework to the presence of noisy network information.

The above theorem guarantees that as long as the error is small in magnitude, the network-based inference procedure correctly determines the significance of the gene sets. In other words, a necessary condition for the proposed method to work in presence of noise in the network information is that $\|\Delta_A\| = o_P(1)$. As mentioned earlier, the problem of estimation of network structure for directed, as well as undirected, networks is an important problem in multivariate statistics and researchers have studied asymptotic properties of network estimation for different classes of problems. Here, we consider a special case of the problem of estimating high dimensional networks, where the structure of the network is known, and the problem is reduced to estimating association weights among genes. The following corollary shows that the proposed network-based gene set analysis procedure is not sensitive to the estimation noise in this setting. It is important to note that the conditions of this result only limit the degree of nodes in the graph and no constraint is required on the total number of nodes in the graph. In the following, d_i represents the unweighted in-degree of node i: the number of neighbors of i in undirected graphs and the number of parents of i in directed graphs.

Corollary III.3. *Let $\mathcal{G}$ be a DAG or an undirected graph, with p nodes and adjacency matrix A. Assume that $\max_i(d_i) = n^b$ for some $0 < b < 1$ and $\sum_{i\in\mathcal{G}}(d_i) = n^a$ for some $a > 0$. Further, assume that the structure (or skeleton) of the network is known, but the network information is obtained by estimating the association weights from an independent sample of size n. Then, the test statistics in (3.6) is an asymptotically unbiased most powerful test for (3.5).*

Proof. By Theorem III.1, it suffices to show that $\|\hat{A}-A\| = o_P(1)$. First, assume that $\mathcal{G}$ is a DAG. Then, by the results in Chapter IV, to find the association weights one needs to regress each node on the set of the parents of that node. Since $\max_i(d_i) = o(n)$, without loss of generality, we can assume that $\max_i(d_i) < n$, and therefore regular regression can be used to estimate the weights. The asymptotic normality of regression estimators then implies that each non-zero entry of the adjacency matrix converges with an exponential rate to the true value. Bonferroni's inequality and the fact that the total number of edges in the graph is a polynomial function of the sample size imply that $\|\hat{A} - A\| = o_P(1)$.

For undirected graphs, we note that partial correlations between each node i and its neighbors ne_i can be recursively estimated using the following formula:

$$\rho_{i,j|\mathrm{ne}_i} = \frac{\rho_{i,j|\mathrm{ne}_i\backslash h} - \rho_{i,h|\mathrm{ne}_i\backslash h}\rho_{j,h|\mathrm{ne}_i\backslash h}}{\sqrt{(1-\rho^2_{i,h|\mathrm{ne}_i\backslash h})(1-\rho^2_{j,h|\mathrm{ne}_i\backslash h})}}$$

However, Corollary 1 of Kalisch and Bühlmann (2007) implies that if $\max_i(d_i) < n-4$ estimated partial correlations converge to true values with an exponential rate. An argument similar to the case of DAGs then implies that $\|\hat{A} - A\| = o_P(1)$ and the result follows. □

3.3 Performance Analysis

In this section, we evaluate the small sample properties of the proposed inference procedure, through several simulation studies. In all settings, data are generated from a mixed linear model, where the Gaussian noise has an AR(1) correlation structure. We consider different combination of mean and network information, and investigate the effects of temporal correlation, as well as noise in the network information.

3.3.1 Multiple Experimental Conditions

The first simulation depicts the real data example of Section 3.4, which corresponds to analysis of responses of yeast cells to environmental stress factors. The network consists of a directed graph with 7 subnetworks and a total of 220 nodes. Each subnetwork in turn consists of a 4-level binary tree and a “hub” node. There are also 3 gateway genes that connect the subnetworks together. The adjacency matrix of the graph is considered to remain constant in different experimental conditions and different time points. The model includes changes in gene expressions under

Table 3.1: Parameter settings for the first simulation study.

Subnetwork	Non-zero Mean Parameters			
1	–			
2	$\alpha_2 = 2$			
3	$\alpha_2 = 1,$	$\delta_2 = 1$		
4	$\alpha_2 = 1,$	$\alpha_3 = 1$		
5	$\alpha_2 = 1,$	$\delta_3 = 1$		
6	$\alpha_2 = 1,$	$\alpha_3 = 1,$	$\delta_2 = 1$	
7	$\alpha_2 = 1,$	$\alpha_3 = 1,$	$\delta_2 = 1,$	$\delta_3 = 1$

different experimental conditions and different time points. Specifically,

$$\begin{aligned} \mathbb{E}Y_{11} &= \Lambda\mu \\ \mathbb{E}Y_{1k} &= \Lambda(\mu + \delta_k), & k &= 2,3 \\ \mathbb{E}Y_{jk} &= \Lambda(\mu + \alpha_j + \delta_k), & j,k &= 2,3 \end{aligned} \tag{3.15}$$

The settings of parameters in the first simulation are given in Table 3.1. Table 3.2 includes the estimated powers of the t-tests for different mean parameters, as well as powers of the F-test, for the overall significance of the subnetwork, estimated from 100 replications [2] with $n = 1$ observations at each combination of experimental condition and time point. To prevent redundancy, the contrast matrix L (see Section 3.2.4) consists only of contrast vectors used for the main effects (the parameters in the first 4 columns of the Table 3.2).

It can be seen from these results that when the model is correctly specified, the proposed inference procedure offers high power for detecting non-zero parameters, while maintaining close to nominal significance levels for non-significant parameters.

3.3.2 Effect of Temporal Correlation

The second simulation setting aims to illustrate the effects of temporal correlation, as well as changes in the network structure, in different experimental conditions.

[2] Simulation replicates are obtained by generating data sets according to the same model with different realizations of the random vectors $\boldsymbol{\varepsilon}$ and $\boldsymbol{\gamma}$)

Table 3.2: Estimated powers of t-test and F-test for the first simulation study. The first four columns of the table represent the powers for testing the significance of the mean parameters (α_2, δ_2, α_3 and δ_3 respectively). The powers for testing equality of main effects ($\alpha_2 = \alpha_3$ and $\delta_2 = \delta_3$) are given in the next two columns of the table. Entries in bold represent result of potential interest.

	Individual Parameters (t-test)						Subnetwork
Subnetwork	α_2	δ_2	α_3	δ_3	$\alpha_2 - \alpha_3$	$\delta_2 - \delta_3$	(F-test)
1	0.006	0.06	0.03	0.14	0.01	0.10	0.12
2	**1.00**	0.10	0.02	0.09	**1.00**	0.13	**1.00**
3	**0.99**	**1.00**	0.03	0.05	**0.99**	**1.00**	**1.00**
4	**0.98**	0.09	**1.00**	0.07	0.02	0.08	**1.00**
5	**0.99**	0.08	0.02	**1.00**	**0.99**	**1.00**	**1.00**
6	**0.99**	**1.00**	**1.00**	0.05	0.01	**1.00**	**1.00**
7	**1.00**	**0.99**	**1.00**	**1.00**	0.00	0.01	**1.00**

Since gene set enrichment analysis methods do not directly incorporate complex experiments, we consider a simple experimental design, including two experimental conditions. However, to illustrate the effect of temporal correlation, we consider the case where data are generated over 5 time points with no replicates. The temporal correlation among observations is generated using an AR(1) process with autocorrelation parameter ϕ. We consider a network consisting of 4 non-overlapping subnetworks (as described in the first simulation) regulated by 3 hub genes. The correlation among genes in each subnetwork is controlled by a single parameter ρ, with different values in distinct subnetworks and experimental conditions. The parameter settings for this simulation are given in Table 3.3.

Table 3.3: Parameter settings for the second simulation study. α_i and $\rho_i, i = 1, 2$ correspond to the ith experimental condition.

Subnet	Mean Parameters	Correlation Parameters
1	$\alpha_1 = \alpha_2 = 1$	$\rho_1 = \rho_2 = 0.2$
2	$\alpha_1 = 1, \alpha_2 = 2$	$\rho_1 = \rho_2 = 0.2$
3	$\alpha_1 = \alpha_2 = 1$	$\rho_1 = 0.2, \rho_2 = 0.7$
4	$\alpha_1 = 1, \alpha_2 = 2$	$\rho_1 = 0.2, \rho_2 = 0.7$

Given the true values of the parameters, the test statistic in (3.6) has a normal distribution, with means 0 and $l\beta$ under the null and alternative hypotheses, respectively. Hence, it is possible to calculate the true asymptotic powers of rejecting the null hypotheses for each of the subnetworks in this simple setting. Figure 3.1 includes

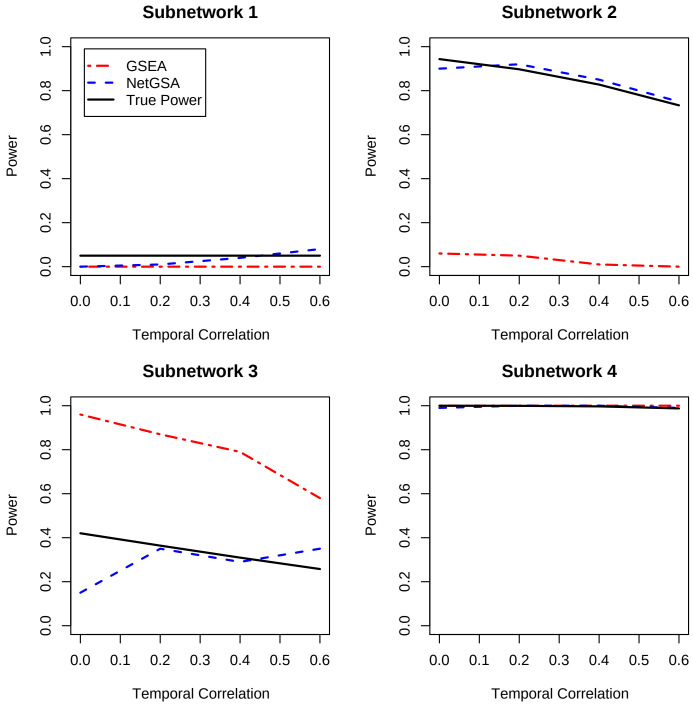


Figure 3.1: Estimated and true powers for tests of subnetworks in Simulation 2.

the estimated powers of tests using GSEA and the proposed network-based method (NetGSA), based on 100 replications, along with the true asymptotic powers of the corresponding tests. It can be seen that when the parameters are clearly insignificant or demonstrate strong significance (Subnetworks 1 and 4), both methods correctly determine the significance of the test. However, in less extreme scenarios (e.g. Subnetworks 2 and 3), the presence of temporal correlation along with the small sample size ($n = 1$) prevent GSEA from correctly determining the statistical significance of subnetworks. On the other hand, by accounting for the temporal correlation, NetGSA offers considerable improvement over GSEA.

Table 3.4 includes the details of estimated and true powers of tests of significance of subnetworks considered in Simulation 2. In order to investigate the effect of the sample size n on the power of the tests, we also consider the case of 10 independent samples for each experimental condition ($n = 10$). Powers of the tests with $n = 10$

are presented in Table 3.5. This table indicates that estimated powers of the proposed NetGSA method are more consistent with the values of the true powers for larger sample sizes. In addition, the presence of temporal correlation prevents the GSEA method from distinguishing the significance of subnetwork 2 even with larger sample sizes.

Table 3.4: Powers of second simulation study with $n = 1$. Entries in bold represent result of potential interest.

		ϕ			
		0	0.2	0.4	0.6
Subnetwork 1	GSEA	0.00	0.00	0.00	0.00
	NetGSA	0.00	0.01	0.04	0.08
	True Power	0.05	0.05	0.05	0.05
Subnetwork 2	GSEA	**0.06**	**0.05**	**0.01**	**0.00**
	NetGSA	**0.90**	**0.92**	**0.85**	**0.75**
	True Power	**0.94**	**0.90**	**0.83**	**0.73**
Subnetwork 3	GSEA	**0.96**	**0.87**	**0.79**	**0.58**
	NetGSA	**0.15**	**0.35**	**0.29**	**0.35**
	True Power	**0.42**	**0.36**	**0.31**	**0.26**
Subnetwork 4	GSEA	1.00	1.00	1.00	1.00
	NetGSA	0.99	1.00	1.00	0.99
	True Power	1.00	1.00	0.99	0.99

Table 3.5: Powers of second simulation study with $n = 10$. Entries in bold represent result of potential interest.

		ϕ			
		0	0.2	0.4	0.6
Subnetwork 1	GSEA	0.00	0.00	0.00	0.00
	NetGSA	0.00	0.02	0.00	0.06
	True Power	0.05	0.05	0.05	0.05
Subnetwork 2	GSEA	**0.06**	**0.04**	**0.03**	**0.05**
	NetGSA	**1.00**	**1.00**	**1.00**	**1.00**
	True Power	**1.00**	**1.00**	**1.00**	**1.00**
Subnetwork 3	GSEA	0.95	0.96	0.87	**0.61**
	NetGSA	1.00	1.00	0.98	**0.96**
	True Power	0.99	0.99	0.99	**0.98**
Subnetwork 4	GSEA	1.00	1.00	1.00	1.00
	NetGSA	1.00	1.00	1.00	1.00
	True Power	1.00	1.00	1.00	1.00

3.3.3 Uncertainty in Network Information

In Section 3.2.5, we showed that the proposed inference procedure is asymptotically insensitive to small noise in the network information, in case of the simple two-class problems. We also argued that similar results can be expected in more complex experiments. Here we provide empirical evidence for the robustness of the proposed method to noise in the network information in presence of temporal correlation, by considering the simulation settings of Section 3.3.2, with $n = 1$. The settings of mean and correlation parameters are identical to those in Table 3.3. In addition, the temporal correlation is fixed at $\phi = 0.4$. In each case, the data are generated according to the mixed linear model with the true network information, and estimation and inference is carried out using a perturbed version of the network information. The network information is perturbed by adding an i.i.d. uniform random variable $U \sim \text{Uni}[-\nu, \nu]$ to each non-zero entry of the adjacency matrix; ν is hence the level of random noise in the network. Figure 3.2 illustrates the estimated and the true powers for different levels of noise ν. It can be seen that even with small sample sizes ($n = 1$), the estimated powers are similar to the expected ones, and the powers with noisy network information are similar to those obtained in the absence of noise ($\nu = 0$). The results of this simulation indicate that the proposed method is robust to small levels of noise (e.g. up to $\sim 30\%$). In addition, the effect of noise is mainly significant in the case of Subnetwork 3, where the difference in the two populations is mainly due to the changes in the network information.

3.3.4 Changes in the Network in Complex Experiments

Our final simulation setting aims to further illustrate the effect of change in the weighted adjacency matrix of the graph in complex experiments. We consider a model with separate intercept and slope parameters, for each of the three treatment conditions. In other words,

$$\mathbb{E}Y_{kj} = \Lambda^{(k)}\alpha_k + \Lambda^{(k)}\delta_k t_j, \quad j, k = 1, 2, 3, \quad t = (5, 15, 30).$$

We consider the directed graph of the first simulation setting, but here we allow for

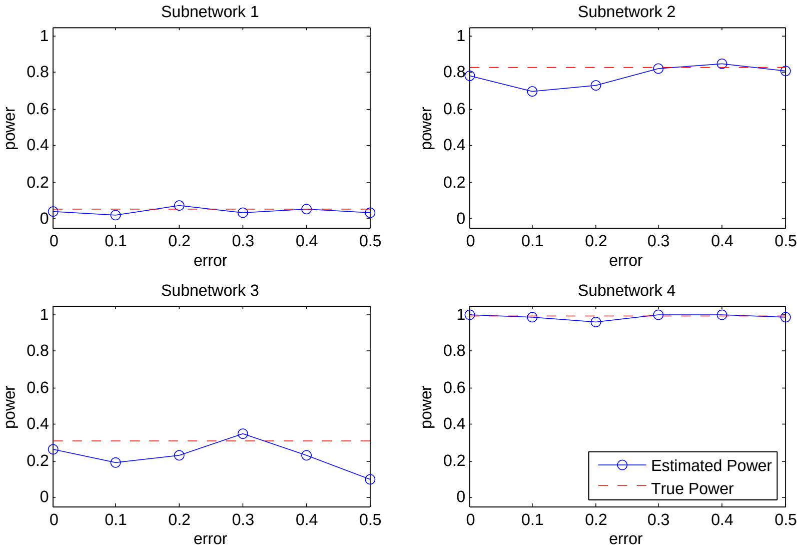


Figure 3.2: Estimated and true powers for tests of subnetworks in Simulation 3 for different values of random noise.

changes in both the adjacency matrix of subnetworks, as well as the mean parameters. For illustration purposes, the adjacency matrix of each subnetwork (and hence its influence matrix) is considered to be a function of a single parameter ρ with values in $(-1, 1)$, and entries of the adjacency matrix may attain different values in each of the three treatment conditions. Based on the latent variable model, as ρ increases, genes in the network would have higher effects on their neighbors. In this setting, subnetworks 2 and 6 only include changes in the fixed effect parameters. Subnetworks 1 and 7 have moderate changes in the fixed effect parameters coupled with changes in associations among genes. In subnetwork 3, the association among genes is the only source of change. Finally, the parameters of subnetwork 5 are designed so that the individual change in the parameters is not significant; however, the combined effect of changes in intercept and slope parameters is expected to be significant. Table 3.6 shows the settings of the parameters for this simulation.

Table 3.7 includes the estimated powers of F and t-tests. It can be seen that

Table 3.6: Significant parameters for the fourth simulation study. Unlisted parameters: $\alpha_{null} = 1$, $\delta_{null} = 0.02$ and $\rho_{null} = 0.2$).

	Significant Parameters	
Subnetwork	Mean	Influence Matrix
1	$\alpha_3 = 2$	$\rho_3 = 0.7$
2	$\alpha_3 = 3$	–
3	–	$\rho_3 = 0.7$
4	–	–
5	$\alpha_3 = 1.5, \delta_3 = 0.04$	–
6	$\delta_3 = 0.10$	–
7	$\delta_3 = 0.06$	$\rho_3 = 0.7$

Table 3.7: Estimated powers of F-test and t-test for the fourth simulation setting. Entries in bold represent results of potential interest.

	Individual Parameters (t-test)						Subnetwork
Subnetwork	$\alpha_1 - \alpha_2$	$\delta_1 - \delta_2$	$\alpha_1 - \alpha_3$	$\delta_1 - \delta_3$	$\alpha_2 - \alpha_3$	$\delta_2 - \delta_3$	(F-test)
1	0.102	0.094	**0.991**	0.066	**0.975**	0.098	**0.982**
2	0.099	0.081	**0.983**	0.073	**0.988**	0.091	**0.991**
3	0.091	0.085	**0.343**	0.052	**0.355**	0.102	**0.409**
4	0.103	0.082	0.121	0.080	0.122	0.100	0.029
5	0.122	0.138	**0.467**	**0.213**	**0.447**	**0.253**	**0.900**
6	0.131	0.112	0.100	**0.989**	0.161	**0.958**	**0.961**
7	0.121	0.150	0.365	**0.900**	0.364	**0.856**	**0.992**

powers of tests are higher than the significance level of 0.05 in cases where no changes are present. This may be attributed to the small sample size ($n = 1$). In such cases, family-wise error rates could provide more conservative inference. It can also be seen that the tests are sensitive to changes in the fixed effect parameters, as well as associations among genes, and the (positive) change in associations magnifies the change in the fixed effect parameters. Estimated powers for subnetwork 5 indicate that even if the individual effects are not strongly significant (low estimated powers of t-test for intercept and slope), their combined effect, represented by the power of the F-test, can be significant.

3.4 Yeast Environmental Stress Response (ESR)

The ability to respond to environmental changes is important for competitive fitness and survival of living organisms; understanding the response of cells to environmental changes can provide clues to molecular mechanisms that regulate gene

Table 3.8: Setting of parameters in the yeast ESR experiment

Experiment	Time points
Mild Heat Shock (29C to 33C)	5, 15, 30 min after 33C
Mild Heat Shock with 1M sorbitol at 29C and 33C	5, 15, 30 min after 33C
Mild Heat Shock with 1M sorbitol at 29C	5, 15, 30 min after 33C

expression in response to these changes (Causton et al., 2001). Cells respond to environmental stress factors through a complicated process that is often observable at the expression levels of a large class of genes. Gasch et al. (2000) studied the response of yeast *Saccharomyces cerevisiae* to a wide range of environmental stress factors, and observed the expression levels of genes in the yeast genome over different time intervals. Experimental settings included responses to temperature shocks, toxic chemicals and osmotic changes.

To illustrate the proposed network-based model, we selected a subset of the data available from Gasch et al. (2000). This particular set of experiments studies the response of yeast cells to mild heat shock at different levels of osmolarities (different amounts of sorbitol in the environment). The gene expressions were obtained at three different time points after the cells were resuspended at the final temperature. Table 3.8 provides the detailed settings of the experiment.

In order to apply our proposed network-based method, we need external information on the weighted adjacency matrix of the underlying gene network. YeastNet is a publicly available database, which includes genes whose functional interactions are verified by integrating a large number of available genomic and proteomic data sets (Lee et al., 2007). The result of this integration is a network of $\sim 102,000$ interactions among $\sim 5,900$ genes, covering 95% of known yeast genes. However, YeastNet only provides information on the topology of the network (connections between genes) and does not include the strengths of association of gene interactions. Different methods can be used to efficiently calculate association strengths of gene interactions, when the topology of the network is known (see e.g. Chaudhuri et al., 2007). The gene expression data provided in Gasch et al. (2000) includes additional experiments independent of those studied in this section, which can be used to derive association weights. Following connections to graphical models, we estimate the association weight of each edge by the partial correlation coefficient of the corresponding pair of genes (after correcting for time dependence). However, since the additional data do not reflect the same experimental settings, it is not possible to estimate separate influence matrices for different combinations of time and experimental conditions, and hence we ignore this variability. Using additional samples, one could calculate the influence matrix of the network for each of the 9 combinations of experimental conditions and time points, and incorporate these matrices in the design matrices for fixed and random effect parameters.

We are interested in determining pathways that are perturbed in response to the combinations of heat shock and variable osmolarities, as well as those whose expression profiles exhibit significant changes over time. To determine biologically relevant pathways, we use information on gene functions provided in the data set from Gasch et al. (2000), derived from the Gene Ontology (Ashburner et al., 2000). We define genetic pathways of interest by combining genes with similar functions into gene sets. Pathways with at least 5 genes are considered, and a total of 73 pathways and 2784 genes ($p = 2784$) with known functions are included in our analysis.

Since there are no replicates available in this data set, it is not possible to include any interaction terms in the model. Hence, we use the model in (3.16) to analyze the variations in gene expressions over time, and in response to different levels of sorbitol in the environment.

$$\begin{aligned} \mathbb{E}Y_{11} &= \Lambda\mu & & \\ \mathbb{E}Y_{1k} &= \Lambda(\mu+\delta_k), & & k=2,3 \\ \mathbb{E}Y_{jk} &= \Lambda(\mu+\alpha_j+\delta_k), & & j,k=2,3 \end{aligned} \tag{3.16}$$

Here α_j and δ_k represent the change from the baseline condition for jth time and kth experimental conditions and the temporal correlation among gene expressions is taken into account via an AR(1) model.[3]

The design matrices $\mathbf{\Psi}$ and $\mathbf{\Pi}$ are $9p \times 5p$ and $9p \times 9p$ matrices and the covariance matrix of $\mathbf{Y}$ is also $9p \times 9p$. In particular, denoting by ϕ be the AR(1) parameter and by σ^2_ε and σ^2_γ the variance components for ε and γ, the vector of variance parameters is $\theta = (\sigma^2_\gamma, \sigma^2_\varepsilon, \phi)$. Then using the notation of Section **??**, $\mathbf{G} = \sigma^2_\gamma \mathbf{I}_{9p}$, $\mathbf{\Pi} = I_n \otimes \Lambda$, and $\mathbf{R} = \sigma^2_\varepsilon I_3 \otimes R$, where

$$R = \begin{bmatrix} I & \phi I & \phi^2 I \\ \phi I & I & \phi I \\ \phi^2 I & \phi I & I \end{bmatrix}.$$

Finally, the design matrix for the fixed effect parameters is set up using (3.3) with χ the design matrix for a single gene according to the model in (3.16), using $\mathbf{\Psi} = \chi \otimes \Lambda$. Specifically,

$$\mathbf{\Psi}' = \begin{pmatrix} \Lambda & \Lambda & \Lambda & \Lambda & \Lambda & \Lambda & \Lambda & \Lambda & \Lambda \\ & & & \Lambda & \Lambda & \Lambda & & & \\ & & & & & & \Lambda & \Lambda & \Lambda \\ & \Lambda & & & \Lambda & & & \Lambda & \\ & & \Lambda & & & \Lambda & & & \Lambda \end{pmatrix}$$

[3] The model in (3.16) is a simplification of $\mathbb{E}Y_{jk} = \Lambda(\alpha_j + \delta_k)$, $j,k = 1,2,3$, where to reduce the number of parameters, the baseline case of $j = k = 1$ is represented with a single parameter μ.

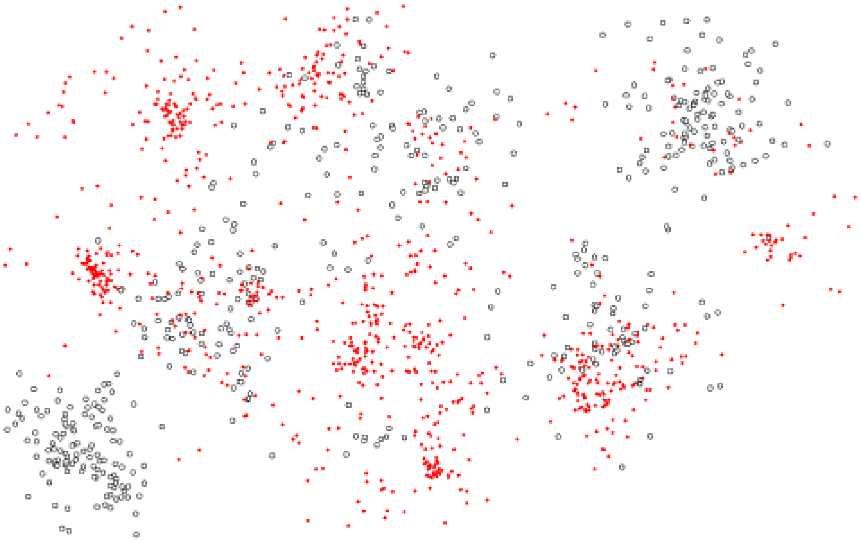


Figure 3.3: Network of yeast genes considered in the analysis of ESR. Red solid diamonds and empty circles represent genes in significant and nonsignificant pathways, respectively. The plot is drawn using cytoscape 2.6 (www.cytoscape.org).

Using the FDR procedure of Benjamini and Hochberg (1995) with $q^* = 0.05$, 47 pathways show significant changes in response to the experimental conditions and/or over time. Figures 3.3 and 3.4 depict the gene network of yeast and some of the significant pathways, respectively. Figure 3.3 provides a general overview of the whole network where the edges between the nodes are removed and the genes are classified into significant and nonsignificant in order to illustrate the pattern of differential expression throughout the network (clusters of significant and nonsignificant genes point to the corresponding pathways). Figure 3.4 looks more closely at some of the significant pathways with different degrees of connectivity, and both positive and negative associations among genes. Genes that appear to be isolated are in fact connected to the pathway through other genes that have been omitted when displaying each subnetwork separately.

Gasch et al. (2000) reported that about 900 genes showed significant changes of expression in response to environmental stress factors (over all experimental set-

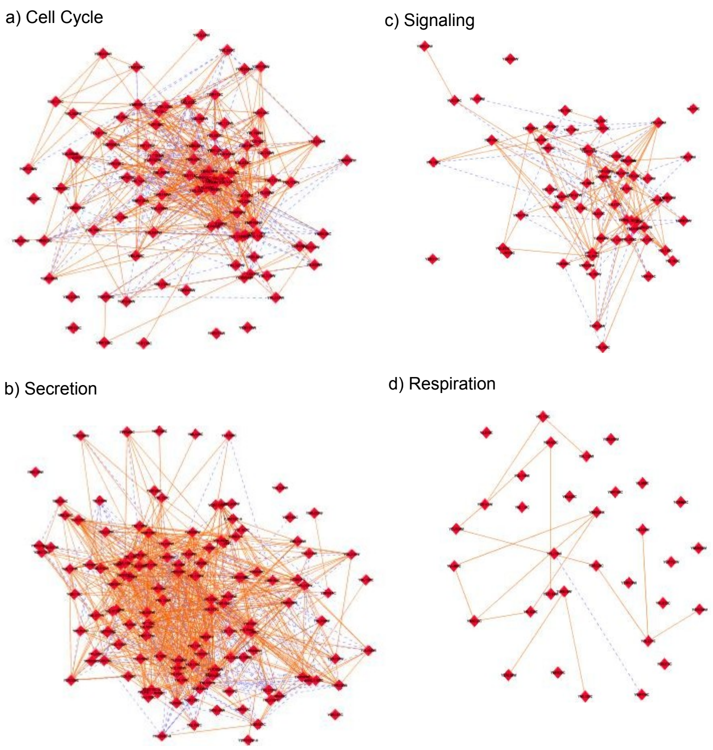


Figure 3.4: Selected significant pathways considered in analysis of yeast ESR. Solid orange edges indicate positive interactions and dashed blue edges represent negative associations among genes. Plots are drawn using cytoscape 2.6.

tings). They also classified the expression levels of these genes into two dominant patterns of expressions. The first set included about 600 genes, which were repressed in ESR, while the rest of genes were induced in ESR. Based on this analysis, genes repressed in ESR are involved in growth-related processes, various aspects of RNA metabolism, nucleotide biosynthesis, secretion, as well as the genes encoding ribosomal proteins. On the other hand, many genes induced in ESR are considered to offer cellular protection during stressful conditions, such as heat and osmotic shocks which were considered in our analysis. Some of these processes include Carbohydrate Metabolism, Cell Wall Modifications, Protein Folding And Degradation, DNA Damage Repair, Fatty Acid Metabolism, Metabolite Transport and Intracellular Signalling (see Gasch et al. (2000) for more details on the functions of genes repressed and induced in ESR).

Classification of genes by their functions is facilitated through our network-based enrichment analysis approach, and many of the processes reported in Gasch et al. (2000) are also found significant based on our proposed method. Moreover, examination of the estimated fixed effects allows us to study the pattern of expression of the significant pathways over time and under different levels of sorbitol.

Tables 3.9 includes the list of significant pathways in analysis of yeast ESR data, along with the p-values from the corresponding F-test. Table 3.10 provides a list of pathways that show changes of expression over time, as well as pathways that have different expression patterns in different experimental conditions (sorbitol levels). In this table, 24 pathways show significant changes of expression over time, 29 pathways correspond to the change in sorbitol level, and 12 pathways provide evidence for both type of changes. This analysis reveals new features of environmental stress response, by determining which pathways are activated in response to different changes in the cell's environment. Pathways whose expression levels do not change in response to sorbitol levels, are only activated in response to heat shock, an obvious example of such pathways being the Heat Shock Response. On the other hand, pathways that only demonstrate significant changes in response to sorbitol level are activated when the osmolarity level of the cell's environment is perturbed. Pathways that demonstrate changes in response to both types of changes include both induced and repressed pathways under ESR. Secretion, DNA Replication, rRNA Processing and Amino Acid Metabolism are examples of pathways that are repressed in the ESR, while different carbohydrate and fatty acid metabolism pathways as well as Oxidative Stress Response are induced under ESR.

Figure 3.5 provides an alternative view of the changes of expressions in response to environmental stress. In this plot, the average standardized expression levels of pathways, based on the value of the test statistics for each of the significant pathways, is displayed. The pathways are divided into induced and suppressed, based on their value of test statistic at time $t = 5$. As observed by Gasch et al. (2000), it can be seen that the change in the expression levels in response to environmental stress factors is transient. The average expression levels of experiments that include change in sorbitol level ($k = 2, 3$) are similar. However, these levels are different from the first experimental setting, where no sorbitol is present. Repressed pathways demonstrate a slight delay in the decline in transcription level. Gasch et al. (2000) characterized this as a feature of the second group of genes repressed in the ESR. Figure 3.5 also reveals that presence of sorbitol further reduces the expression level of genes. This is true for both induced and repressed pathways. It is important to note that, should the experiment included additional samples, more interesting analyses about interactions among heat shock and change of osmolarity would also be possible.

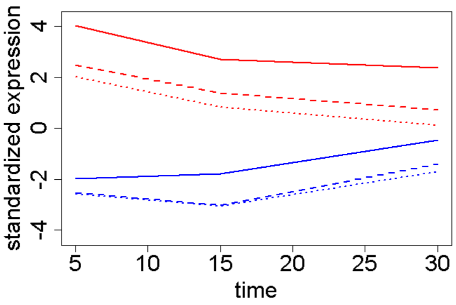


Figure 3.5: Average expression profile of significant pathways. Red and blue lines represent induced and suppressed pathways, respectively (positive and negative values at the first observation time), and solid, dashed and dotted lines indicate the first, second and third experimental conditions.

3.5 Conclusion

In this chapter, we extended the modeling framework of Chapter II to incorporate available information on general networks into the analysis of gene sets in complex

Table 3.9: Significant pathways in the analysis of yeast environmental stress response (ESR) data.

	Pathway Name	P-Value (F-test)	Pathway Size
1	PROTEIN SYNTHESIS	0	286
2	TRANSPORT	0	143
3	SECRETION	0	126
4	CELL CYCLE	0	97
5	CYTOSKELETON	0	83
6	LIPID METABOLISM	0	63
7	AMINO ACID BIOSYNTHESIS	0	60
8	DNA REPAIR	0	58
9	DNA REPLICATION	0	57
10	MEIOSIS	0	52
11	PROTEIN GLYCOSYLATION	0	51
12	PROTEIN FOLDING	0	40
13	RRNA PROCESSING	0	38
14	VACUOLAR PROTEIN TARGETING	0	38
15	GLYCOLYSIS	0	36
16	MATING	0	34
17	SUGAR METABOLISM	0	27
18	SPORULATION	0	22
19	AMINO ACID METABOLISM	0	21
20	AMINO ACID BIOSYNTHESIS	0	19
21	PYRIMIDINE BIOSYNTHESIS	0	12
22	STRESS RESPONSE	0	12
23	METHIONINE BIOSYNTHESIS	0	11
24	SALT TOLERANCE	0	8
25	GLYCEROL METABOLISM	0	6
26	HEAT SHOCK RESPONSE	0	6
27	TREHALOSE METABOLISM	0	6
28	AMINO ACID METABOLISM	0	5
29	B-VITAMIN BIOSYNTHESIS	0	5
30	HIGH OSMOLARITY	0	5
31	RESPIRATION	0.0001	30
32	PHOSPHOLIPID METABOLISM	0.0001	22
33	SPHINGOLIPID METABOLISM	0.0001	9
34	CHROMATIN STRUCTURE	0.0003	47
35	OXIDATIVE STRESS RESPONSE	0.0003	14
36	PURINE BIOSYNTHESIS	0.0006	18
37	CELL ORGANIZATION	0.0016	76
38	MRNA EXPORT	0.0028	9
39	RNA PROCESSING	0.0035	9
40	TRNA PROCESSING	0.0042	35
41	PYRIMIDINE METABOLISM	0.005	8
42	SIGNALING	0.0075	58
43	DRUG RESISTANCE	0.0078	11
44	TOXIN RESISTANCE	0.0122	26
45	ENDOCYTOSIS	0.0152	18
46	ATP SYNTHESIS	0.0163	20
47	PROTEIN TARGETING	0.017	66

experiments, including multiple factors and time course data. The framework utilizes mixed linear models and can handle changes in the network structure. Further, it can also be adapted to handle non-Gaussian data, using the framework of generalized mixed linear models (GLMM).

One of the challenges in analyzing gene expression data using the proposed model is the computational burden of the estimation process. Standard packages for solving mixed linear models cannot handle problems with large vectors/matrices of observations and parameters, without determining a specific independence structure. In this

Table 3.10: Analysis of ESR data: Pathways with significant changes over time and in response to sorbitol

Change over **time**			Change in response to **sorbitol**		
	Pathway Name	Pathway Size		Pathway Name	Pathway Size
1	PROTEIN SYNTHESIS	286	1	TRANSPORT	143
2	TRANSPORT	143	2	SECRETION	126
3	SECRETION	126	3	CELL CYCLE	97
4	LIPID METABOLISM	63	4	CELL ORGANIZATION	76
5	DNA REPAIR	58	5	LIPID METABOLISM	63
6	DNA REPLICATION	57	6	AMINO ACID BIOSYNTHESIS	60
7	RRNA PROCESSING	38	7	DNA REPLICATION	57
8	GLYCOLYSIS	36	8	MEIOSIS	52
9	MATING	34	9	PROTEIN GLYCOSYLATION	51
10	SUGAR METABOLISM	27	10	PROTEIN FOLDING	40
11	PHOSPHOLIPID METABOLISM	22	11	RRNA PROCESSING	38
12	AMINO ACID METABOLISM	21	12	GLYCOLYSIS	36
13	ATP SYNTHESIS	20	13	TRNA PROCESSING	35
14	ENDOCYTOSIS	18	14	RESPIRATION	30
15	PURINE BIOSYNTHESIS	18	15	SUGAR METABOLISM	27
16	OXIDATIVE STRESS RESPONSE	14	16	TOXIN RESISTANCE	26
17	STRESS RESPONSE	12	17	PHOSPHOLIPID METABOLISM	22
18	METHIONINE BIOSYNTHESIS	11	18	SPORULATION	22
19	PYRIMIDINE METABOLISM	8	19	AMINO ACID BIOSYNTHESIS	19
20	SALT TOLERANCE	8	20	PURINE BIOSYNTHESIS	18
21	GLYCEROL METABOLISM	6	21	OXIDATIVE STRESS RESPONSE	14
22	HEAT SHOCK RESPONSE	6	22	DRUG RESISTANCE	11
23	TREHALOSE METABOLISM	6	23	MRNA EXPORT	9
24	AMINO ACID METABOLISM	5	24	RNA PROCESSING	9
			25	SPHINGOLIPID METABOLISM	9
			26	GLYCEROL METABOLISM	6
			27	TREHALOSE METABOLISM	6
			28	AMINO ACID METABOLISM	5
			29	HIGH OSMOLARITY	5

chapter, we proposed an iterative algorithm based on block-relaxation for estimating the parameters of the model. This algorithm can be extended to further partition the parameter space and to also partition the set of observations over subnetworks (estimation over subnetworks).

The proposed methodology provides a flexible framework for studying the changes in expressions of sets of genes and allows systematic testing of such changes as the experimental conditions vary. It is important to note that this model requires external information about the underlying gene network, as well as information on the strength of association between genes. An increasing number of publicly available data sets offer information about the structure of the gene network (the 0-1 adjacency matrix) with different degrees of reliability. However, less information is available about the strength and direction of these connections. An attractive feature of the proposed network-based gene set analysis framework is its robustness to small noise in the network information. However, bias in the network information can result in both type I and II errors. The problem of estimation of (directed and undirected) gene networks is an important problem in systems biology, and of independent interest. It is however important to note that bias may result from using the same set of gene expression data in order to both estimate the underlying network, and test the significance of pathways.

Availability

`Matlab` codes for the proposed network-based gene set analysis (NetGSA) in the case of two-class inference problem are available at:

http://www.stat.lsa.umich.edu/∼shojaie/.

An R-package (`netGSA`) for the general problem is currently being developed and will be made available through R-CRAN upon completion.

CHAPTER IV

Penalized Likelihood Methods for Estimation of Directed Acyclic Graphs

4.1 Introduction

Graphical models are efficient tools for the study of statistical models through a compact representation of the joint probability distribution of the underlying random variables. The nodes of the graph represent the random variables, while the edges capture the relationships among them. Both directed and undirected edges are used to represent interactions among random variables. However, there is a conceptual difference between these two types of graphs: while undirected graphs are used to represent conditional independence, directed graphs often represent causal relationships (see Pearl, 2000). Directed acyclic graphs, also known as Bayesian networks, are a special class of directed graphs, where all the edges of the graph are directed and the graph has no cycles. Such graphs are the main focus of this chapter and unless otherwise specified, any reference to directed graphs in the sequel refers to directed acyclic graphs.

Directed graphs are used in graphical models and belief networks and have been the focus of research in the computer science literature (Pearl, 2000). Important applications involving directed graphs also arise in the study of biological systems, including cell signalling pathways and gene regulatory networks (Markowetz and

Spang, 2007).

Estimation of directed graphs is an NP-hard problem, and estimation of the direction of edges may not be possible due to observational equivalence (see Section 4.2). Most of the earlier estimation methods include greedy algorithms that search through the space of possible graphs. A number of methods are available for estimating directed graphs with small to moderate number of nodes. The max-min hill climbing algorithm (Tsamardinos et al., 2006), and the PC-algorithm (Spirtes et al., 2000) are two such examples. However, the space of directed graphs grows super-exponentially with the number of nodes (Robinson, 1977), and estimation using search-based methods, especially in high dimensional settings, becomes impractical. Bayesian methods (e.g Heckerman et al., 1995) are also computationally very intensive and therefore not particularly appropriate for high dimensional settings. Recently, Kalisch and Bühlmann (2007) proposed an implementation of the PC-algorithm with polynomial complexity in high dimensional sparse settings. When the variables inherit a natural ordering, estimation of directed graphs is reduced to estimating their structure or skeleton (see Section 4.3). Applications with a natural ordering of variables include estimation of causal relationships from temporal observations, estimation of transcriptional regulatory networks from gene expression data, and settings where additional experimental data can determine the ordering of variables. Examples of such applications are given in Section 4.6.

For Gaussian random variables, conditional independence relations among random variables are represented using an undirected graph, known as the conditional independence graph. The edges of this graph represent conditional dependencies among random variables, and correspond to non-zero elements of the inverse covariance matrix, also known as the precision matrix. Different penalization methods have been recently proposed to obtain sparse estimates of the precision matrix. Meinshausen and Bühlmann (2006) considered an approximation to the problem of sparse inverse covariance estimation using the lasso penalty. They showed under a set of assumptions, that their proposed method correctly determines the neighborhood of each node. Banerjee et al. (2008) and Friedman et al. (2008b) explored different aspects of the problem of estimating the precision matrix using the lasso penalty, while Yuan and Lin (2007) considered other choices for the penalty. Rothman et al. (2008) proved consistency in Frobenius norm, as well as in matrix norm, of the ℓ_1-regularized estimate of the precision matrix when $p \gg n$, while Lam and Fan (2009) extended their result and considered estimation of matrices related to the precision matrix, including the Cholesky factor of the inverse covariance matrix, using general penalties. Penalization of the Cholesky factor of the inverse covariance matrix has been also considered by Huang et al. (2006) and Levina et al. (2008), who used the lasso penalty in order to obtain a sparse estimate of the inverse covariance matrix. This method is based on the regression interpretation of the Cholesky factorization model and therefore requires the variables to be ordered a priori.

In this chapter, I consider the problem of estimating the skeleton of directed acyclic graphs, where the variables exhibit a natural ordering. The known ordering of variables is exploited to reformulate the likelihood as a function of the adjacency matrix of the graph, which results in efficient algorithm for estimation of structure of directed graphs using penalized likelihood methods. Although the results of this chapter are presented for the case of Gaussian observations, the proposed method can also be applied to non-Gaussian observations, provided the underlying causal mechanisms in the network are linear (see Section 4.2.2).

4.2 Representation of Directed Acyclic Graphs

4.2.1 Background and notation

Consider a graph $\mathcal{G} = (V, E)$, where V corresponds to the set of nodes with p elements and $E \subset V \times V$ to the edge set. The nodes of the graph represent random variables $X_1, \ldots, X_p$ and the edges capture the associations among them. An edge is called directed if $(j, i) \notin E$ whenever $(i, j) \in E$, and undirected when $(i, j) \in E$ if and only if $(j, i) \in E$. We denote by pa_i the set of parents of node i and for $j \in \mathrm{pa}_i$, we denote $j \to i$. The skeleton of a directed graph is the undirected graph that is obtained by replacing directed edges in E with undirected ones. Finally, throughout this chapter, we represent E using the adjacency matrix A of the graph; i.e. a $p \times p$ matrix whose (j, i)th entry is non-zero if there is an edge between nodes j and i.

The estimation of directed graphs is a challenging problem due to the so-called observational equivalence with respect to the same probability distribution. More specifically, regardless of the sample size, it may not be possible to infer the direction of causation among random variables from observational data. As an illustration, consider the simple graph in the right panel of Fig. 4.1. Reversing the direction of all edges of the graph results in a new graph, which is isomorphic to the original graph, and hence not distinguishable from observations alone.

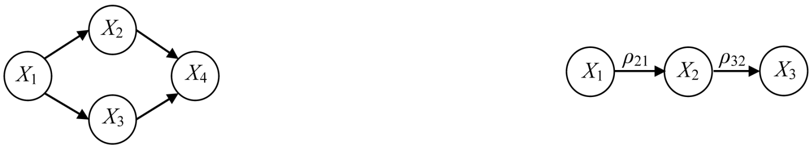


Figure 4.1: Simple directed acyclic graphs. Left: Illustration of observational equivalence in directed graphs, Right: A simple directed graph

The second challenge in estimating directed graphs is that conditional independence among random variables may not reveal the skeleton. The notion of conditional independence in directed graphs is represented using the concept of d-separation

(Pearl, 2000), or equivalently, the moral graph, obtained by removing the directions of the edges and marrying the parents of each node (Lauritzen, 1996). Therefore, estimation of the conditional independence structure reveals the structure of the moral graph, which includes additional edges between parents of each node. For instance, X_2 and X_3 are connected in the moral graph of the simple graph in the left panel of Fig. 4.1.

4.2.2 The latent variable model

The causal effect of random variables in a directed acyclic graph are often explained using structural equation models (Pearl, 2000). In particular,

$$X_i = f_i(\mathrm{pa}_i, Z_i), \quad i = 1, \ldots, p. \tag{4.1}$$

where the random variables Z_i are the latent variables representing the unexplained variation in each node. To model the association among the nodes, we consider a simplification of (4.1) with f_i being linear. More specifically, let ρ_{ij} represent the effect of node j on i for $j \in \mathrm{pa}_i$, then

$$X_i = \sum_{j \in \mathrm{pa}_i} \rho_{ij} X_j + Z_i, \quad i = 1, \ldots, p. \tag{4.2}$$

In the special case where the random variables are Gaussian, equations (4.1) and (4.2) are equivalent, in the sense that ρ_{ij} are the coefficients of the linear regression model of X_i on X_j, for $j \in \mathrm{pa}_i$. It is known in the normal case that $\rho_{ij} = 0$, if $j \notin \mathrm{pa}_i$.

Consider the simple graph in the right panel of Fig. 4.1; denoting the influence matrix of the graph by Λ, (4.2) can be written in compact form as $X = \Lambda Z$, where for the simple example above, we have

$$\Lambda = \begin{pmatrix} 1 & 0 & 0 \\ \rho_{12} & 1 & 0 \\ \rho_{12}\rho_{23} & \rho_{23} & 1 \end{pmatrix}.$$

Let the latent variables Z_i be independent with mean μ_i and variance σ_i^2. Then, $E(X) = \Lambda\mu$ and $\Sigma = \mathrm{Var}(X) = \Lambda D \Lambda^\intercal$, where $D = \mathrm{diag}\,(\sigma_i^2)$ and $\Lambda^\intercal$ denotes the transpose of the matrix Λ.

In Chapter II, we established the relationship between the influence matrix Λ, and the adjacency matrix A of the graph, and showed that in the case of directed acyclic graphs $\Lambda = (I - A)^{-1}$. In Section 4.3, we exploit this relationship to directly formulate the problem of estimating the skeleton of the graph.

Remark IV.1. The result of Lemma II.3 and the fact that $\Sigma = \Lambda D \Lambda^\intercal$ imply that for any directed acyclic graph, if $D_{ii} > 0$ for all i, then Σ is full rank. More specifically, let $\phi_j(\Sigma)$ denote the jth eigenvalue of matrix Σ. Then, $\phi_{\min}(\Sigma) > 0$, or $\phi_{\max}(\Sigma^{-1}) < \infty$. Similarly, since $\Sigma^{-1} = \Lambda^{-\intercal} D^{-1} \Lambda^{-1}$, full rankness of Λ implies that $\phi_{\min}(\Sigma^{-1}) > 0$, or equivalently $\phi_{\max}(\Sigma) < \infty$. This result also applies to all subgraphs of a graph.

The properties of the proposed latent variable model established in Lemma II.3 are independent of the choice of probability distribution. In fact, since the latent variables Z_i in (4.2) are assumed independent, given the entries of the adjacency matrix, the distribution of each random variable X_i in the graph only depends on the values of pa_i. Therefore, regardless of the choice of the probability distribution, the joint distribution of the random variables is compatible with $\mathcal{G}$ (Pearl, 2000, p. 16). Therefore, based on the equivalence of conditional independence and d-separation, if the joint probability distribution of random variables on a directed graph is generated according to the latent variable model (4.2), zeros of the adjacency matrix, A, determine conditional independence relations among random variables. As mentioned before, under the normality assumption, the latent variable model is equivalent to the general structural equation model. Although we focus on Gaussian random variables in the remainder of this chapter, the estimation procedure proposed in Section 4.3 can be applied to a variety of other distributions, if one is willing to assume the linear structure in (4.2).

4.3 Penalized Likelihood Estimation of Directed Graphs

4.3.1 Problem formulation

Consider the latent variable model of Section 4.2.2 and denote by $\mathcal{X}$ the $n \times p$ data matrix. We assume, without loss of generality, that the X_i's are centered and scaled, so that $\mu_i = 0$ and $\sigma_i^2 = 1, i = 1, \ldots, p$.

Denote by $\Omega \equiv \Sigma^{-1}$ the precision matrix of a p-vector of Gaussian random variables and consider a general penalty function $J(\Omega)$. The penalized estimate of Ω is then given by

$$\hat{\Omega} = \underset{\Omega \succ 0}{\operatorname{argmin}} \{-\operatorname{logdet}(\Omega) + \operatorname{trace}(\Omega S) + \lambda J(\Omega)\}, \tag{4.3}$$

where $S = n^{-1}\mathcal{X}^{\intercal}\mathcal{X}$ denotes the empirical covariance matrix and λ is the tuning parameter controlling the size of the penalty. Applications in biological and social networks often involve sparse networks. It is therefore desirable to find a sparse solution for (4.3). This becomes more important in the small n, large p setting, where the unpenalized solution is unreliable. The lasso penalty and the adaptive lasso penalty (Zou, 2006) are singular at zero and therefore result in sparse solutions. We consider these two penalties in order to find a sparse estimate of the adjacency matrix. However, the optimization algorithm proposed here can also be used with other choices of penalty function, if the penalty is applied to each individual component of the adjacency matrix.

Using the latent variable model of Section 4.2.2, and the relationship between the covariance matrix and the adjacency matrix of directed graphs established in Lemma II.1, the problem of estimating the adjacency matrix of the graph can be directly formulated as an optimization problem based on A. Specifically, if the underlying graph is directed, and the ordering of the variables is known, then A is a lower triangular matrix with zeros on the diagonal. Let $\mathcal{A} = \{A : A_{ij} = 0,\ j \geq i\}$. Then using the facts that $\det(A) = 1$ and $\sigma_i^2 = 1$, A can be estimated as the solution of the following optimization problem

$$\hat{A} = \operatorname*{argmin}_{A \in \mathcal{A}} \left[\operatorname{trace}\left\{(I - A)^{\mathsf{T}}(I - A)S\right\} + \lambda J(A)\right]. \tag{4.4}$$

In this chapter, we consider the general weighted lasso problem, where

$$J(A) = \sum_{i,j=1:p, j<i} w_{ij}|A_{ij}|. \tag{4.5}$$

The lasso and adaptive lasso problems are special cases of this general penalty. In the case of the lasso, $w_{ij} = 1$. In the original proposal of Zou (2006), the weights for the adaptive lasso are obtained by setting $w_{ij} = |\tilde{A}_{ij}|^{-\gamma}$, for some initial estimate of the adjacency matrix $\tilde{A}$ and some power γ. We consider the following modification of the original weights

$$w_{ij} = 1 \vee |\tilde{A}_{ij}|^{-\gamma}. \tag{4.6}$$

where the initial estimates $\tilde{A}$ are obtained from the regular lasso estimates, and $x \vee y$ represents the maximum of x and y. Aside from the truncation of weights from below, which is implemented to facilitate the study of asymptotic properties, the main difference between the adaptive lasso penalty using (4.6) and the proposal of Zou (2006) is the use of the lasso estimates to construct the weights. In Sections

4.4 and 4.5, we show that this modification, which could also be considered a two-stage or hybrid lasso penalty, results in improvements in terms of both asymptotic properties, as well as numerical performance, over the regular lasso penalty.

The objective function for both the lasso and adaptive lasso problems is convex. However, since the ℓ_1 penalty is non-differentiable, these problems can be reformulated using matrices $A_+ = \max(A, 0)$ and $A_- = -\min(A, 0)$. To that end, let W be the $p \times p$ matrix of weights for the adaptive lasso, or the matrix of ones for the lasso estimation problem. Problem (4.4) can then be formulated as:

(4.7)

$$\min_{A_+, A_- \succeq 0} \operatorname{trace}\left\{S(I - A_+ + A_-)^{\mathsf{T}}(I - A_+ + A_-) + \lambda(A_+ + A_-)W + \Delta(A_+ + A_-)1_{u^+}\right\},$$

where $\succeq 0$ is interpreted componentwise, Δ is a large positive number and 1_{u^+} is the indicator matrix for upper triangular elements of a $p \times p$ matrix, including the diagonal elements. The last term of the objective function, i.e. $\operatorname{trace}\{\Delta(A_+ + A_-)1_{u^+}\}$, prevents the upper triangular elements of the matrices A_+ and A_- to be nonzero.

Problem (4.7) corresponds to a quadratic optimization problem with non-negativity constraints and can be solved using standard interior point algorithms. However, such algorithms do not scale well with dimension and are only applicable if p ranges in the hundreds. In Section 4.3.2, we present an alternative formulation of the problem, which leads to considerably more efficient algorithms.

4.3.2 Optimization algorithm

Consider again the problem of estimating the adjacency matrix of directed graphs with the general weighted lasso penalty. Let $\mathfrak{a}_i$ be the ith row of matrix A, and denote by l^- the set of indices up to $l - 1$, i.e. $l^- = \{j : j = 1, \ldots, l - 1\}$. Then (4.4) can be written as

$$\hat{A} = \underset{A \in \mathcal{A}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{p} \left(\mathfrak{a}_i S \mathfrak{a}_i^{\top} - 2\mathfrak{a}_i S_i + \lambda |\mathfrak{a}_i| W_i \right) \right\}. \tag{4.8}$$

It can be seen that the objective function in (4.8) is separable and therefore it suffices to solve the optimization problem over each row of matrix A. Then, taking advantage of the lower triangular structure of A and noting that $A_{11} = 0$, solving (4.8) is equivalent to solving the following $p-1$ optimization problems

$$\hat{A}_{i,i^-} = \underset{\theta \in \mathbb{R}^{i-1}}{\operatorname{argmin}} \left\{ \theta^{\top} S_{i^-,i^-} \theta - 2S_{i,i^-}\theta + \lambda \sum_{j=1}^{i-1} |\theta_j| w_{ij} \right\}, \qquad i = 2, \ldots, p. \tag{4.9}$$

In addition, $S_{i^-,i^-} = n^{-1}(\mathcal{X}_{i^-})^{\top}\mathcal{X}_{i^-}$ and $S_{i,i^-} = n^{-1}(\mathcal{X}_i)^{\top}\mathcal{X}_{i^-}$, and hence the problem in (4.9) can be reformulated as the following ℓ_1-regularized least squares problems

$$\hat{A}_{i,i^-} = \underset{\theta \in \mathbb{R}^{i-1}}{\operatorname{argmin}} \left\{ n^{-1} \|\mathcal{X}_i - \mathcal{X}_{i^-}\theta\|_2^2 + \lambda \sum_{j=1}^{i-1} |\theta_j| w_{ij} \right\}, \qquad i = 2, \ldots, p. \tag{4.10}$$

The formulation in (4.10) indicates that the ith row of matrix A includes the coefficient of projection of X_i on $X_j, j = 1, \ldots, i-1$, which is in agreement with the discussion in Section 4.2.2. It also reveals a connection between estimation of the underlying graphs and the neighborhood selection approach of Meinshausen and Bühlmann (2006): when the underlying graph is directed, the approximate solution of the neighborhood selection problem is exact, if the regression model is fitted on the set of parents of each node instead of all other nodes in the graph.

Using (4.10), the problem of estimating directed graphs can be solved very efficiently. In fact, it suffices to solve $p-1$ lasso problems for estimation of least squares

Algorithm 2 Penalized likelihood estimation of directed graphs

```
1. Given the ordering $\mathcal{O}$, order the columns of observation matrix $\mathcal{X}$ in increasing order.
2. For $i = 2, 3, \ldots, p$,
    2.1. Denote $y = \mathcal{X}_i$, $X = \mathcal{X}_{i^-}$. Given the weight matrix $W$, let $w = W_{i,i^-}$, and solve
             $\hat{A}_{i,i^-} = \operatorname{argmin}\left\{ n^{-1}\|y - X\theta\|_2^2 + \lambda_i \sum_{j=1}^{i-1} |\theta_j| w_j \right\}$
```

coefficients, with dimensions ranging from 1 to $p-1$. To solve these problems, we use the efficient pathwise coordinate optimization algorithm of Friedman et al. (2007), implemented in the R-package `glmnet`. The proposed procedure is summarized in Algorithm 2.

4.3.3 Analysis of computational complexity

As mentioned in the introduction, the space of all possible directed graphs is super-exponential in the number of nodes and hence it is not surprising that the PC-algorithm, without any restriction on the space, has exponential complexity. Kalisch and Bühlmann (2007) recently proposed an efficient implementation of the PC-algorithm for sparse graphs; its complexity when the maximal neighborhood size q is small, is bounded with high probability by $O(p^q)$. Although this is a considerable improvement over the original algorithm, in many applications it can become fairly expensive. For instance, gene regulatory networks and signaling pathways include many hub genes, which lead to large values for q.

The reformulation of the directed graph estimation problem in (4.10) requires solving $p-1$ lasso regression problems. The cost of solving a lasso problem comprised of k covariates and n observations using the pathwise coordinate optimization algorithm is $O(nk)$; hence, the total cost of estimating the adjacency matrix of the graph is $O(np^2)$, which is the same as the cost of calculating the empirical covariance matrix S. Moreover, the formulation in (4.10) includes a set of non-overlapping sub-problems. Therefore, for problems with very large number of nodes and/or observations, the performance of the algorithm can be further improved by parallelizing the estimation of these sub-problems. The adaptive lasso version of the problem is similarly solved using the modification of the regular lasso problem proposed in Zou (2006), which results in the same computational cost as the regular lasso problem.

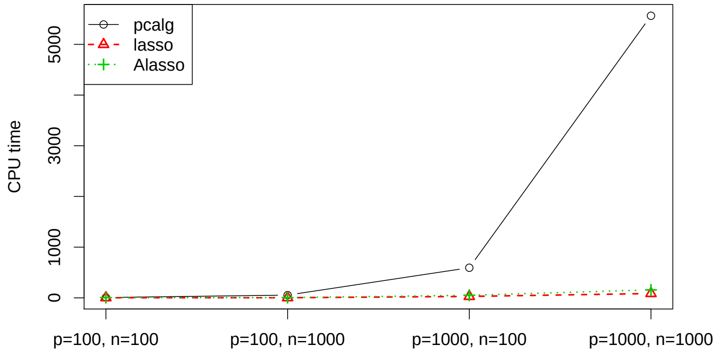


Figure 4.2: Comparison of CPU time for estimation of DAGs with different p and n. Results for PC-algorithm, lasso and adaptive lasso are presented.

To evaluate the performance of these algorithms, we compared the average CPU time, over 10 simulation runs, for estimation of directed graphs with different number of nodes, $p = 100, 1000$, and different sample sizes, $n = 100, 1000$. The computational time for the PC-algorithm increases with larger values of the average neighborhood size and the significance level α. Therefore, to control the computational complexity of the PC-algorithm, these parameters are set to 5 and $0 \cdot 01$, respectively. In addition, in order to compare equivalent quantities, we only consider the CPU time that the PC-algorithm requires for estimation of the graph skeleton. The results of this simulation study are given in Figure 4.2. It can be seen that, while for small to moderate values of p, i.e. $p = 100$, the computation time for the PC-algorithm is comparable to the time for the penalized likelihood algorithm, in a graph with $p = 1000$ and $n = 1000$, the average CPU time for the PC-algorithm could be up to two order of magnitudes larger than the equivalent time for Algorithm 2.

4.4 Asymptotic Properties

4.4.1 Preliminaries

Next, we establish theoretical properties of the lasso and adaptive lasso estimates of the adjacency matrix of directed graphs. Asymptotic properties of the lasso-type estimates with fixed design matrices have been studied by a number of researchers (Knight and Fu, 2000; Zou, 2006; Huang et al., 2008), while random design matrices have been considered by Meinshausen and Bühlmann (2006). On the other hand, Rothman et al. (2008) and Lam and Fan (2009) among others have studied asymptotic properties of estimates of covariance and precision matrices.

As discussed in Section 4.3.2, the problem of estimating the adjacency matrix of a directed graph is equivalent to solving $p-1$ non-overlapping penalized least square problems described in (4.10). In order to study the asymptotic properties of the proposed estimators, we focus on the asymptotic consistency of network estimation, i.e. the probability of correctly estimating the network structure, in terms of type I and type II errors. We allow the total number of nodes in the graph to grow as an arbitrary polynomial function of the sample size, while assuming that the true underlying network is sparse.

4.4.2 Assumptions

Let $X = (X_1, \dots, X_p)$ be a collection of p zero-mean Gaussian random variables with covariance matrix Σ, and let $\mathcal{X}$ and S be defined as in Section 4.3.1. To simplify the notation, denote by $\theta^i = A_{i,i^-}$ the entries of the ith row of A to the left of the diagonal. Further, let $\theta^{i,\mathcal{I}}$ be the estimate for the ith row, with values outside the set of indices $\mathcal{I}$ set to zero; i.e., $\theta^{i,\mathcal{I}} \equiv A_{i,i^-}$ and $A_{i,j} = 0, \ j \notin \mathcal{I}$.

The following assumptions are used in establishing the consistency of network estimation:

(A-0) For some $a > 0$, $p = p(n) = O(n^a)$ as $n \to \infty$, and there exists a $0 \le b < 1$ such that $\max_{i \in V} \operatorname{card}(\mathrm{pa}_i) = O(n^b)$ as $n \to \infty$. [1]

[1] When there are no ambiguities, we suppress the dependency of $p = p(n)$ and the set of nodes in graph ($V = V(n)$) on n.

(A-1) There exists $\nu > 0$ such that for all $n \in \mathbb{N}$ and all $i \in V$, $\mathrm{Var}\,(X_i \mid X_{i^-}) \geq \nu$.

(A-2) There exists $\delta > 0$ and some $\xi > b$ (with b defined above) such that for all $i \in V$ and for every $j \in \mathrm{pa}_i$, $|\pi_{ij}| \geq \delta n^{-(1-\xi)/2}$, where π_{ij} is the partial correlation between X_i and X_j after removing the effect of the remaining variables.

(A-3) There exists $\Psi < \infty$ such that for all $n \in \mathbb{N}$ and every $i \in V$ and $j \in \mathrm{pa}_i$, $\|\theta^{j,\mathrm{pa}_i}\|_2 \leq \Psi$.

(A-4) There exists $\kappa < 1$ such that for all $i \in V$ and $j \notin \mathrm{pa}_i$, $\left|\sum_{k\in\mathrm{pa}_i} \mathrm{sign}(\theta_k^{i,\mathrm{pa}_i})\theta_k^{j,\mathrm{pa}_i}\right| < \kappa$.

Assumption (A-0) determines the permissible rates of increase in the number of variables and the neighborhood size, as a function of n, (A-1) prevents singular or near singular covariance matrices, and (A-2) guarantees that true partial correlations are bounded away from 0.

Assumption (A-3) limits the magnitude of the shared ancestral effect between each node in the network and any of its parents. This is less restrictive than the equivalent assumption for the neighborhood selection problem, where the effects over all neighboring nodes are assumed to be bounded. In fact, in the case of gene regulatory networks, empirical data indicate that the average number of upstream-regulators per gene is less than 2 (Leclerc, 2008). Thus, the number of parents of each node is small, while each hub node can affect many downstream nodes.

Assumption (A-4) is referred to as neighborhood stability and is equivalent to the irrepresentability assumption of Huang et al. (2008). It has been shown that the lasso estimates are not in general variable selection consistent if this assumption is violated. Huang et al. (2008) considered the adaptive lasso estimates with general initial weights and proved their variable selection consistency under a weaker form of irrepresentability assumption, referred to as adaptive irrepresentability. We will show that when the initial weights for the adaptive lasso are derived from the regular lasso estimates as in (4.6), the assumption of neighborhood stability, as well as the less stringent assumption (A-3) are not required for establishing variable selection consistency of the adaptive lasso. This relaxation in assumptions required for variable selection consistency, is a result of the consistency of the regular lasso estimates, and the special structure of directed graphs. However, similar results can be obtained for the adaptive lasso estimates of the precision matrix, as well as regression models with fixed and random design matrices, under additional mild assumptions.

4.4.3 Asymptotic consistency of directed graph estimation

We start with a technical lemma that is a consequence of the Karush−Kuhn−Tucker conditions for the general weighted lasso problem and is used in the proof of Theorems IV.5 and IV.7.

Lemma IV.2. *Let $\hat{\theta}^{i,\mathcal{I}}$ be the general weighted lasso estimate of $\theta^{i,\mathcal{I}}$, i.e.*

$$\hat{\theta}^{i,\mathcal{I}} = \operatorname*{argmin}_{\theta:\theta_k=0,k\notin\mathcal{I}} \left\{ n^{-1}\|\mathcal{X}_i - \mathcal{X}\theta\|_2^2 + \lambda \sum_{k=1}^{p} |\theta_k| w_{ik} \right\}. \tag{4.11}$$

Define

$$G_j(\theta) = -2n^{-1}\mathcal{X}_j^{\intercal}(\mathcal{X}_i - \mathcal{X}\theta),$$

and let w_i be the vector of initial weights in the adaptive lasso estimation problem. Then a vector $\hat{\theta}$ with $\hat{\theta}_k = 0, \forall k \notin \mathcal{I}$ is a solution of (4.11) iff $\forall j \in \mathcal{I}, G_j(\theta) = -\operatorname{sign}(\hat{\theta}_j)w_{ij}\lambda$ if $\hat{\theta}_j \neq 0$ and $|G_j(\theta)| \leq w_{ij}\lambda$ if $\hat{\theta}_j = 0$. Moreover, if the solution is not unique and $|G_j(\theta)| < w_{ij}\lambda$ for some solution $\hat{\theta}$, then $\hat{\theta}_j = 0$ for all solutions of (4.11).

Proof. The proof of the lemma is identical to the proof of Lemma (A.1) in Meinshausen and Bühlmann (2006), except for inclusion of general weights w_{ij}, and is therefore omitted. □

Our first result studies the variable selection consistency of the lasso estimate. In this theorem, (i) corresponds to sign consistency, (ii) and (iii) establish control of type I type II errors, and (iv) addresses the consistency of network estimation. We denote by $\hat{E}$ the estimate of the edge set of the graph, and by $x \sim y$ the asymptotic equivalence between x and y.

Theorem IV.3. *Suppose that (A-0)-(A-4) hold and $\lambda \sim dn^{-(1-\zeta)/2}$ for some $b < \zeta < \xi$ and $d > 0$. Then for the lasso estimate there exist constants $c_{(i)}, \ldots, c_{(iv)} > 0$ such that for all $i \in V$, as $n \to \infty$*

(i) $\mathbb{P}\left\{\text{sign}\,(\hat{\theta}_j^{i,pa_i}) = \text{sign}\,(\theta_j^{i,pa_i}) \text{ for all } j \in pa_i\right\} = 1 - O\left\{\exp\left(-c_{(i)}n^{\zeta}\right)\right\}$,

(ii) $\mathbb{P}\left(\hat{pa}_i \subseteq pa_i\right) = 1 - O\left\{\exp\left(-c_{(ii)}n^{\zeta}\right)\right\}$,

(iii) $\mathbb{P}\left(pa_i \subseteq \hat{pa}_i\right) = 1 - O\left\{\exp\left(-c_{(iii)}n^{\zeta}\right)\right\}$, *and*

(iv) $\mathbb{P}(\hat{E} = E) = 1 - O\left\{\exp\left(-c_{(iv)}n^{\zeta}\right)\right\}$.

Proof. The proof of this theorem follows from arguments similar to those presented in Meinshausen and Bühlmann (2006) with minor modifications and replacing conditional independence for undirected graphs with d-separation for directed graphs, and is hence omitted. □

The next result establishes similar properties for the adaptive lasso estimates, without the assumptions of neighborhood stability. The proof of Theorem IV.5 makes use of the consistency of sparse estimates of the Cholesky factor of covariance matrices, established in Theorem 9 of Lam and Fan (2009). For completeness, we restate a simplified version of the theorem for our lasso problem, for which $\sigma_i = 1, i = 1, \ldots p$, and the eigenvalues of the covariance matrix are bounded (see Remark

IV.1). Throughout this section, we denote by s the total number of nonzero elements of the true adjacency matrix A.

Theorem IV.4 (Lam and Fan 2009)**.** *If* $n^{-1}(s+1)\log p = o(1)$ *and* $\lambda = O\left\{(\log p/n)^{1/2}\right\}$, *then* $\|\hat{A} - A\|_F = O_p\left\{(n^{-1}s\log p)^{1/2}\right\}$.

It can be seen from Theorem IV.4 that the lasso estimates are consistent as long as $n^{-1}(s+1)\log p = o(1)$. To take advantage of this result, we replace (A-0) with the following assumption

(A-0′) For some $a > 0$, $p = p(n) = O(n^a)$ as $n \to \infty$. Also, $\max_{i\in V} \text{card}\,(\text{pa}_i) = O(n^b)$ as $n \to \infty$, where $sn^{2b-1}\log n = o(1)$ as $n \to \infty$.

Assumption (A-0′) further restricts the number of parents of each node and also enforces a restriction on the total number of nonzero elements of the adjacency matrix. Condition $sn^{2b-1}\log n = o(1)$, implies that $b < 1/2$. Therefore, although the consistency of the adaptive lasso in Theorem IV.5 is established without making any further assumptions on the structure of the network, it is achieved at the price of requiring higher degree of sparsity in the network. We now state the main result regarding variable selection consistency of adaptive lasso. The theorem only requires assumptions (A-0′), (A-1) and (A-2), and assumptions (A-3) and (A-4) are no longer required.

Theorem IV.5. *Consider the adaptive lasso estimation problem, where the initial weights are calculated using regular the lasso estimates of the adjacency matrix of the graph in* (4.10)*. Suppose (A-0′), (A-1) and (A-2) hold and* $\lambda \sim dn^{-(1-\zeta)/2}$ *for some* $b < \zeta < \xi$ *and* $d > 0$*. Also suppose that the initial lasso estimates are found using a penalty parameter* λ^0 *that satisfies* $\lambda^0 = O\left\{(\log p/n)^{1/2}\right\}$*. Then, there exist constants* $c_{(i)}, \ldots, c_{(iv)} > 0$ *such that for all* $i \in V$*, as* $n \to \infty$*, (i)-(iv) in Theorem*

IV.3 hold.

Proof. To prove (i), note that by Bonferroni's inequality, and the fact that $\text{card}\,(\text{pa}_i) = o(n)$ as $n \to \infty$, it suffices to show that there exists some $c_{(i)} > 0$ such that for all $i \in V$ and for every $j \in \text{pa}_i$, $\mathbb{P}\left\{\text{sign}\,(\hat{\theta}_j^{i,\text{pa}_i}) = \text{sign}\,(\theta_j^{i,\text{pa}_i})\right\} = 1 - O\left\{\exp\left(-c_{(i)} n^{\zeta}\right)\right\}$ as $n \to \infty$.

Let $\hat{\theta}^{i,\text{pa}_i}(\beta)$ be the estimate of θ^{i,pa_i} in (4.11), with the jth component fixed at a constant value β,

$$\hat{\theta}^{i,\text{pa}_i}(\beta) = \underset{\theta \in \Theta_\beta}{\text{argmin}} \left\{ n^{-1} \|\mathcal{X}_i - \mathcal{X}\theta\|_2^2 + \lambda \sum_{k=1}^{p} |\theta_k| w_k \right\}, \tag{4.12}$$

where $\Theta_\beta \equiv \{\theta \in \mathbb{R}^p : \theta_j = \beta, \theta_k = 0, k \notin \text{pa}_i\}$. Note that for $\beta = \hat{\theta}_j^{i,\text{pa}_i}$, $\hat{\theta}^{i,\text{pa}_i}(\beta)$ is identical to $\hat{\theta}^{i,\text{pa}_i}$. Thus, if $\text{sign}\,(\hat{\theta}_j^{i,\text{pa}_i}) \neq \text{sign}\,(\theta_j^i)$, there would exist some β with $\text{sign}\,(\beta)\,\text{sign}\,(\theta_j^i) \leq 0$ such that $\hat{\theta}^{i,\text{pa}_i}(\beta)$ is a solution to (4.12). Since $\theta_j^i \neq 0, \forall j \in \text{pa}_i$, it suffices to show that for all β with $\text{sign}\,(\beta)\,\text{sign}\,(\theta_j^i) < 0$, with high probability, $\hat{\theta}^{i,\text{pa}_i}(\beta)$ can not be a solution to (4.12).

Without loss of generality, we consider the case where $\theta_j^i > 0$; $\theta_j^i < 0$ can be shown similarly. Then if $\beta \leq 0$, from Lemma IV.2, $\hat{\theta}^{i,\text{pa}_i}(\beta)$ can be a solution to (4.12) only if $G_j(\hat{\theta}^i(\beta)) \geq -\lambda w_{ij}$. Hence, it suffices to show that for some $c_{(i)} > 0$ and all $j \in \text{pa}_i$ with $\theta_j^i > 0$,

$$\mathbb{P}\left[\sup_{\beta \leq 0} \left\{G_j\left(\hat{\theta}^i(\beta)\right) < -\lambda w_{ij}\right\}\right] = 1 - O\left\{\exp\left(-c_{(i)} n^{\zeta}\right)\right\} \qquad \text{as } n \to \infty. \tag{4.13}$$

Define, $\mathcal{R}_i(\beta) = \mathcal{X}_i - \mathcal{X}\hat{\theta}^i(\beta)$. Then for every $j \in \text{pa}_i$ we can write

$$X_j = \sum_{k \in \text{pa}_i \backslash \{j\}} \theta_k^{j,\text{pa}_i \backslash \{j\}} X_k + Z_j, \tag{4.14}$$

where Z_j is independent of $\{X_k; k \in \text{pa}_i \backslash \{j\}\}$. Then by (4.14),

$$G_j\left(\hat{\theta}^i(\beta)\right) = -2n^{-1} \mathcal{Z}_j^{\intercal} \mathcal{R}_i(\beta) - \sum_{k \in \text{pa}_i \backslash \{j\}} \theta^{j,\text{pa}_i \backslash \{j\}} 2n^{-1} \mathcal{X}_k^{\intercal} \mathcal{R}_i(\beta).$$

By Lemma IV.2, it follows that for all $k \in \mathrm{pa}_i \backslash \{j\}$, $|G_k\left(\hat{\theta}^i(\beta)\right)| = |2n^{-1}\mathcal{X}_k{}^{\intercal}\mathcal{R}_i(\beta)| \leq \lambda w_{ik}$, and hence,

$$G_j\left(\hat{\theta}^i(\beta)\right) \leq -2n^{-1}\mathcal{Z}_j^{\intercal}\mathcal{R}_i(\beta) + \lambda \sum_{k \in \mathrm{pa}_i \backslash \{j\}} |\theta^{j,\mathrm{pa}_i\backslash\{j\}}| w_{ik}. \tag{4.15}$$

Using the fact that $|\theta^{j,\mathrm{pa}_i\backslash\{j\}}| \leq 1$, it suffices to show that

$$\mathbb{P}\left[\sup_{\beta \leq 0}\{-2n^{-1}\mathcal{Z}_j^{\intercal}\mathcal{R}_i(\beta)\} < -\lambda \sum_{k \in \mathrm{pa}_i} w_{ik}\right] = 1 - O\left\{\exp\left(-c_{(i)}n^{\zeta}\right)\right\} \qquad \text{as } n \to \infty, \tag{4.16}$$

or equivalently, $\mathbb{P}\left[\inf_{\beta \leq 0}\{2n^{-1}\mathcal{Z}_j^{\intercal}\mathcal{R}_i(\beta)\} < \lambda \sum_{k \in \mathrm{pa}_i} w_{ik}\right] = O\left\{\exp\left(-c_{(i)}n^{\zeta}\right)\right\}$ as $n \to \infty$.

It is shown in Lemma A.2. of Meinshausen and Bühlmann (2006) that for any $q > 0$, there exists $c_{(i)} > 0$ such that for all $j \in \mathrm{pa}_i$ with $\theta^i_j > 0$,

$$\mathbb{P}\left[\inf_{\beta \leq 0}\{2n^{-1}\mathcal{Z}_j^{\intercal}\mathcal{R}_i(\beta)\} \leq q\lambda\right] = O\left\{\exp\left(-c_{(i)}n^{\zeta}\right)\right\} \qquad \text{as } n \to \infty. \tag{4.17}$$

$$\mathbb{P}\left[\inf_{\beta \leq 0}\{2n^{-1}\mathcal{Z}_j^{\intercal}\mathcal{R}_i(\beta)\} > q\lambda\right] = 1 - O\left\{\exp\left(-c_{(i)}n^{\zeta}\right)\right\} \qquad \text{as } n \to \infty. \tag{4.18}$$

However, by definition $w_{ik} \geq 1$ and therefore, $\sum_{k \in \mathrm{pa}_i} w_{ik} \geq \mathrm{card}\left(\mathrm{pa}_i\right) \geq 1$, and (i) follows from (4.18).

To prove (ii), note that the event $\hat{\mathrm{pa}}_i \not\subseteq \mathrm{pa}_i$ is equivalent to the event that there exists a node $j \in i^{-}\backslash \mathrm{pa}_i$ such that $\hat{\theta}^i_j \neq 0$. In other words, denoting the latter event by $\mathcal{D}$, $\mathbb{P}\left(\hat{\mathrm{pa}}_i \subseteq \mathrm{pa}_i\right) = 1 - \mathbb{P}\left(\mathcal{D}\right)$.

$$\mathbb{P}\left(\hat{\mathrm{pa}}_i \subseteq \mathrm{pa}_i\right) = 1 - \mathbb{P}\left(\exists j \in i^{-}\backslash \mathrm{pa}_i : \hat{\theta}^i_j \neq 0\right). \tag{4.19}$$

Let $\mathrm{card}(E : \Xi)$ denote the number of elements in set E that satisfy the condition Ξ.

Let $\mathcal{E}$ denote the event $\left[\exists j \in i^{-}\backslash \mathrm{pa}_i : \hat{\theta}^i_j \neq 0\right]$. However, by Lemma IV.2, and since $w_{ij} \geq 1$,

$$\begin{aligned}\mathbb{P}\left(\mathcal{D}\right) &= \mathbb{P}\left(\exists j \in i^{-}\backslash \mathrm{pa}_i : |G_j(\hat{\theta}^{i,\mathrm{pa}_i})| \geq w_{ij}\lambda\right) \\ &\leq \mathbb{P}\left\{\exists j \in i^{-}\backslash \mathrm{pa}_i : |G_j(\hat{\theta}^{i,\mathrm{pa}_i})| \geq q\lambda \text{ and } w_{ij}\lambda \leq q\lambda \text{ for some } q \geq 1\right\} \\ &\leq \mathbb{P}\left(\exists j \in i^{-}\backslash \mathrm{pa}_i : w_{ij} \leq q \text{ for some } q \geq 1\right).\end{aligned}$$

But $w_{ij} = 1 \vee |\tilde{\theta}^i_j|^{-\gamma}$, with $\tilde{\theta}^i_j$ the lasso estimate of the adjacency matrix from (4.10). Hence, letting $\mathcal{F}$ be the event that there exists $j \in i^{-}\backslash \mathrm{pa}_i$ such that $w_{ij} \leq q$ for some $q > 0$, and using Lemma IV.2 we can write

$$\begin{aligned}\mathbb{P}\left(\mathcal{F}\right) &= \mathbb{P}\left(\exists j \in i^{-}\backslash \mathrm{pa}_i : |\tilde{\theta}^i_j| \geq q^{-1/\gamma} \text{ for some } q \geq 1\right) \\ &\leq \mathbb{P}\left(\exists j \in i^{-}\backslash \mathrm{pa}_i : |\tilde{\theta}^i_j| \geq q' \text{ for some } q' > 0\right) \\ &\leq \mathbb{P}\left(\exists j \in i^{-}\backslash \mathrm{pa}_i : \tilde{\theta}^i_j \neq 0\right) \\ &= \mathbb{P}\left(\exists j \in i^{-}\backslash \mathrm{pa}_i : |G_j(\tilde{\theta}^{i,\mathrm{pa}_i})| \geq \lambda^0\right).\end{aligned}$$

Since $\mathrm{card}\,(\mathrm{pa}_i) = o(n)$, we can assume, without loss of generality, that $\mathrm{card}\,(\mathrm{pa}_i) < n$, which implies that $\tilde{\theta}^{i,\mathrm{pa}_i}$ is an almost sure unique solution to (4.11) with $\mathcal{I} = \mathrm{pa}_i$. Let $\mathcal{E} = \left\{\max_{j \in i^{-}\backslash \mathrm{pa}_i} |G_j(\tilde{\theta}^{i,\mathrm{pa}_i})| < \lambda^0\right\}$. Then conditional on the event $\mathcal{E}$, it follows from the first part of Lemma IV.2 that $\tilde{\theta}^{i,\mathrm{pa}_i}$ is also a solution of the unrestricted weighted lasso problem (4.11) with $\mathcal{I} = i^{-}$. Since $\tilde{\theta}^{i,\mathrm{pa}_i}_j = 0, \forall j \in i^{-}\backslash \mathrm{pa}_i$, it follows from the second part of Lemma IV.2 that $\tilde{\theta}^i_j = 0, \forall j \in i^{-}\backslash \mathrm{pa}_i$. Hence,

$$\mathbb{P}\left(\exists j \in i^{-}\backslash \mathrm{pa}_i : \tilde{\theta}^i_j \neq 0\right) \leq 1 - \mathbb{P}(\mathcal{E}) = \mathbb{P}\left\{\max_{j \in i^{-}\backslash \mathrm{pa}_i} |G_j(\tilde{\theta}^{i,\mathrm{pa}_i})| \geq \lambda^0\right\}, \tag{4.20}$$

where $G_j(\tilde{\theta}^{i,\mathrm{pa}_i}) = -2n^{-1}\mathcal{X}_j^{\mathsf{T}}(\mathcal{X}_i - \mathcal{X}\tilde{\theta}^{i,\mathrm{pa}_i})$.

Since $\mathrm{card}\,(V) = O(n^a)$ for some $a > 0$, Bonferroni's inequality implies that to verify (ii) it suffices to show that there exists a constant $c_{(ii)} > 0$ such that for all $j \in i^{-}\backslash \mathrm{pa}_i$,

$$\mathbb{P}\left(|G_j(\tilde{\theta}^{i,\mathrm{pa}_i})| \geq \lambda^0\right) = O\left\{\exp\left(-c_{(ii)}n^{\zeta}\right)\right\} \quad \text{as } n \to \infty, \tag{4.21}$$

For $j \in i^{-} \backslash \mathrm{pa}_i$ one can write $X_j = \sum_{l \in \mathrm{pa}_i} \theta_l^{j,\mathrm{pa}_i} X_l + R_j$,

$$X_j = \sum_{l \in \mathrm{pa}_i} \theta_l^{j,\mathrm{pa}_i} X_l + R_j$$

where $R_j \sim N(0, \sigma_j^2),\ \sigma_j^2 \leq 1$ and R_j is independent from $X_l, l \in \mathrm{pa}_i$. Similarly, with R_i satisfying the same requirements as R_j, we get $X_i = \sum_{k \in \mathrm{pa}_i} \theta_k^{i,\mathrm{pa}_i} X_k + R_i$.

$$X_i = \sum_{k \in \mathrm{pa}_i} \theta_k^{i,\mathrm{pa}_i} X_k + R_i$$

Denote by $\mathcal{X}_{\mathrm{pa}_i}$ the columns of $\mathcal{X}$ corresponding to pa_i and let θ_{pa_i} be the column vector of coefficients with dimension $\mathrm{card}\,(\mathrm{pa}_i)$ corresponding to pa_i. Then,

$$\begin{aligned} \mathbb{P}\left\{|G_j(\tilde{\theta}^{i,\mathrm{pa}_i})| \geq \lambda^0\right\} &= \mathbb{P}\left\{|-2n^{-1}\mathcal{X}_j^{\intercal}(\mathcal{X}_i - \mathcal{X}\tilde{\theta}^{i,\mathrm{pa}_i})| \geq \lambda^0\right\} \\ &= \mathbb{P}\left[|-2n^{-1}\{\mathcal{X}_{\mathrm{pa}_i}\theta_{\mathrm{pa}_i}^{j,\mathrm{pa}_i} + \mathcal{R}_j\}^{\intercal}\{\mathcal{X}_{\mathrm{pa}_i}(\theta_{\mathrm{pa}_i}^{i,\mathrm{pa}_i} - \tilde{\theta}_{\mathrm{pa}_i}^{i,\mathrm{pa}_i}) + \mathcal{R}_i\}| \geq \lambda^0\right] \end{aligned}$$

Therefore,

$$\begin{aligned} \mathbb{P}\left\{|G_j(\tilde{\theta}^{i,\mathrm{pa}_i})| \geq \lambda^0\right\} &\leq \mathbb{P}\left\{|-2n^{-1}(\theta_{\mathrm{pa}_i}^{i,\mathrm{pa}_i} - \tilde{\theta}_{\mathrm{pa}_i}^{i,\mathrm{pa}_i})^{\intercal}\mathcal{X}_{\mathrm{pa}_i}^{\intercal}\mathcal{X}_{\mathrm{pa}_i}\theta_{\mathrm{pa}_i}^{j,\mathrm{pa}_i}| \geq \lambda^0/3\right\} + \\ &\quad \mathbb{P}\left\{|-2n^{-1}(\theta_{\mathrm{pa}_i}^{i,\mathrm{pa}_i} - \tilde{\theta}_{\mathrm{pa}_i}^{i,\mathrm{pa}_i})^{\intercal}\mathcal{X}_{\mathrm{pa}_i}^{\intercal}\mathcal{R}_j| \geq \lambda^0/3\right\} + \\ &\quad \mathbb{P}\left\{|-2n^{-1}(\mathcal{X}_{\mathrm{pa}_i}\theta_{\mathrm{pa}_i}^{j,\mathrm{pa}_i} + \mathcal{R}_j)^{\intercal}\mathcal{R}_i| \geq \lambda^0/3\right\} \\ &\equiv \mathrm{I} + \mathrm{II} + \mathrm{III}. \end{aligned}$$

Let 1_{pa_i} denote a vector of 1's of dimension $\mathrm{card}\,(\mathrm{pa}_i)$. Then using the fact that $|\theta_l^{j,\mathrm{pa}_i}| \leq 1,$for all $l \in \mathrm{pa}_i$, we can write

$$\mathrm{I} \leq \mathbb{P}\left\{2\|\theta_{\mathrm{pa}_i}^{i,\mathrm{pa}_i} - \tilde{\theta}_{\mathrm{pa}_i}^{i,\mathrm{pa}_i}\|_\infty n^{-1}(\mathcal{X}_{\mathrm{pa}_i}1_{\mathrm{pa}_i})^{\intercal}\mathcal{X}_{\mathrm{pa}_i}1_{\mathrm{pa}_i} \geq \lambda^0/3\right\}.$$

Then $\mathcal{X}_{\mathrm{pa}_i}^{\intercal}\mathcal{X}_{\mathrm{pa}_i} \sim \mathbb{W}_{\mathrm{card}\,(\mathrm{pa}_i)}(\Sigma_{\mathrm{pa}_i}, n)$ where $\mathbb{W}_m(\Sigma, n)$ denotes a Wishart distribution with mean $n\Sigma$. Hence, from properties of the Wishart distribution, we get $(\mathcal{X}_{\mathrm{pa}_i}1_{\mathrm{pa}_i})^{\intercal}\mathcal{X}_{\mathrm{pa}_i}1_{\mathrm{pa}_i} \sim \mathbb{W}_1(1_{\mathrm{pa}_i}^{\intercal}\Sigma_{\mathrm{pa}_i}1_{\mathrm{pa}_i}, n)$.

Since pa_i also forms a directed acyclic graph, the eigenvalues Σ_{pa_i} are bounded (see Remark IV.1), and hence

$$1_{\mathrm{pa}_i}^{\intercal} \Sigma_{\mathrm{pa}_i} 1_{\mathrm{pa}_i} \le \mathrm{card}\,(\mathrm{pa}_i) \phi_{\max}(\Sigma_{\mathrm{pa}_i}). \tag{4.22}$$

Therefore, if $Z \sim \chi_1^2$, then $n^{-1}(\mathcal{X}_{\mathrm{pa}_i} 1_{\mathrm{pa}_i})^{\intercal} \mathcal{X}_{\mathrm{pa}_i} 1_{\mathrm{pa}_i}$ is stochastically smaller than $\mathrm{card}\,(\mathrm{pa}_i)\phi_{\max}(\Sigma_{\mathrm{pa}_i})Z$. On the other hand, by Theorem IV.4,

$$\|A - \tilde{A}\|_F = O_p\left\{(n^{-1} s \log p)^{1/2}\right\},$$

and hence,

$$\|\theta_{\mathrm{pa}_i}^{i,\mathrm{pa}_i} - \tilde{\theta}_{\mathrm{pa}_i}^{i,\mathrm{pa}_i}\|_\infty = O_p\left\{(n^{-1} s \log p)^{1/2}\right\}. \tag{4.23}$$

Noting that $\mathrm{card}\,(\mathrm{pa}_i) = O(n^b)$, $b < 1/2$ and $p = O(n^a)$, $a > 0$, (4.22) and (4.23) imply that

$$\|\theta_{\mathrm{pa}_i}^{i,\mathrm{pa}_i} - \tilde{\theta}_{\mathrm{pa}_i}^{i,\mathrm{pa}_i}\|_\infty \,\mathrm{card}\,(\mathrm{pa}_i)\phi_{\max}(\Sigma_{\mathrm{pa}_i}) = O_p\left\{(sn^{2b-1} a \log n)^{1/2}\right\}.$$

By (A-0′), $sn^{2b-1}\log n = o(1)$ and hence by Slutsky's Theorem and properties of the χ^2-distribution, there exists $c_{(\mathrm{I})} > 0$ such that for all $j \in i^{-}\backslash\mathrm{pa}_i$, $\mathrm{I} = O\left\{\exp\left(-c_{(\mathrm{I})} n^{\zeta}\right)\right\}$ as $n \to \infty$.

Using a similar argument,

$$\mathrm{II} \le \mathbb{P}\left(2n^{-1}\|\theta_{\mathrm{pa}_i}^{i,\mathrm{pa}_i} - \tilde{\theta}_{\mathrm{pa}_i}^{i,\mathrm{pa}_i}\|_\infty |1_{\mathrm{pa}_i} \mathcal{X}_{\mathrm{pa}_i}^{\intercal} \mathcal{R}_j| \ge \lambda^0/3\right).$$

But columns of $\mathcal{X}_{\mathrm{pa}_i}$ have mean zero and are all independent of $\mathcal{R}_j$, hence it suffices to show that there exists $c_{(\mathrm{II})} > 0$ such that for all $j \in i^{-}\backslash\mathrm{pa}_i$ and for all $k \in \mathrm{pa}_i$,

$$\mathbb{P}\left\{2n^{-1}\|\theta_{\mathrm{pa}_i}^{i,\mathrm{pa}_i} - \tilde{\theta}_{\mathrm{pa}_i}^{i,\mathrm{pa}_i}\|_\infty \,\mathrm{card}\,(\mathrm{pa}_i)|\mathcal{X}_k{}^{\intercal}\mathcal{R}_j| \ge \lambda^0/3\right\} = O\left\{\exp\left(-c_{(\mathrm{II})} n^{\zeta}\right)\right\} \quad \text{as } n \to \infty. \tag{4.24}$$

By (4.23) and (A-0′), the random variable on the left hand side of (4.24) is stochastically smaller than $2n^{-1}|\mathcal{X}_k\mathcal{R}_j|$. By independence of X_k and R_j, $E(X_k R_j) =$

0. Also, using Gaussianity of both X_k and R_j, there exists $g < \infty$ such that $E\{\exp(|X_k R_j|)\} \leq g$. Since $\lambda^0 = O\{(\log p/n)^{1/2}\}$, by Bernstein's inequality (Van der Vaart and Wellner, 1996), $\mathbb{P}(2n^{-1}|\mathcal{X}_k \mathcal{R}_j| > \lambda^0/3) \leq \exp(-c_{(\mathrm{II})} n^{\zeta})$ for some $c_{(\mathrm{II})} > 0$ and hence (4.24) is satisfied.

Finally, for III we have

$$\mathbb{P}\left\{| - 2n^{-1}(\mathcal{X}_{\mathrm{pa}_i} \theta^{j,\mathrm{pa}_i}_{\mathrm{pa}_i} + \mathcal{R}_i)^{\mathsf{T}} \mathcal{R}_j| \geq \lambda^0/3\right\} = \mathbb{P}\left\{| - 2n^{-1} \mathcal{X}_i^{\mathsf{T}} \mathcal{R}_j| \geq \lambda^0/3\right\},$$

and using the Bernstein's inequality we conclude that there exists $c_{(\mathrm{III})} > 0$ such that for all $j \in i^{-} \backslash \mathrm{pa}_i$ and for all $k \in \mathrm{pa}_i$, $\mathrm{III} = O\left\{\exp\left(-c_{(\mathrm{III})} n^{\zeta}\right)\right\}$ as $n \to \infty$. The proof of (ii) is then complete by taking $c_{(ii)}$ to be the minimum of $c_{(\mathrm{I})}, \ldots, c_{(\mathrm{III})}$.

To prove (iii), note that $\mathbb{P}\left(\mathrm{pa}_i \subseteq \hat{\mathrm{pa}}_i\right) = 1 - \mathbb{P}\left(\exists j \in \mathrm{pa}_i : \hat{\theta}^i_j = 0\right)$, and let $\mathcal{E} = \left\{\max_{k \in i^{-} \backslash \mathrm{pa}_i} |G_j(\hat{\theta}^{i,\mathrm{pa}_i})| < \lambda w_{ij}\right\}$. It then follows from an argument similar to the proof of (ii) that conditional on $\mathcal{E}$, $\hat{\theta}^{i,\mathrm{pa}_i}$ is an almost sure unique solution of the unrestricted adaptive lasso problem (4.11) with $\mathcal{I} = i^{-}$. Therefore,

$$\mathbb{P}\left(\exists j \in \mathrm{pa}_i : \hat{\theta}^i_j = 0\right) \leq \mathbb{P}\left(\exists j \in \mathrm{pa}_i : \hat{\theta}^i_j = 0\right) + \mathbb{P}\left(\mathcal{E}^c\right).$$

From (i), there exists a $c_1 > 0$ such that $\mathbb{P}\left(\exists j \in \mathrm{pa}_i : \hat{\theta}^i_j = 0\right) = O\{\exp\left(-c_1 n^{\zeta}\right)\}$ and it was shown in (ii) that $\mathbb{P}\left(\mathcal{E}^c\right) = O\{\exp\left(-c_2 n^{\zeta}\right)\}$ for some $c_2 > 0$. Thus (iii) follows from Bonferroni's inequality.

The claim in (iv) follows from (ii) and (iii), and Bonferroni's inequality as $p = O(n^a)$. □

4.4.4 Choice of the tuning parameter

Both lasso and adaptive lasso estimates of the adjacency matrix, depend on the choice of the tuning parameter λ. Different methods have been proposed for selecting the value of the tuning parameter, including cross validation (Rothman et al., 2008) and the Bayesian information criterion (Yuan and Lin, 2007). However, choices of λ that result in the optimal classification error do not guarantee a small error for network reconstruction. We propose next a choice of λ for the general weighted lasso problem with weights w_{ij}. Let Z^*_q denote the $(1-q)$th quantile of standard normal distribution, and define

$$\lambda_i(\alpha) = 2n^{-1/2} Z^*_{\frac{\alpha}{2p(i-1)}}. \tag{4.25}$$

The following result establishes that such a choice controls the probability of falsely joining two distinct ancestral sets, defined next.

Definition IV.6. For every node $i \in V$, the ancestral set of node i, AN_i consists of all nodes j, such that j is an ancestor of i or i is an ancestor of j or i and j have a common ancestor k.

Theorem IV.7. *Under the assumptions of Theorems IV.3 and IV.5 above, for the lasso and adaptive lasso, respectively, for all $n \in \mathbb{N}$ the solution of the general weighted lasso estimation problem with tuning parameter determined in (4.25) satisfies*

$$\mathbb{P}(\text{there exists } i \in V : \hat{AN}_i \nsubseteq AN_i) \leq \alpha.$$

Proof. We first show that if $AN_i \cap AN_j = \emptyset$, then i and j are independent. Since $\Sigma = \Lambda\Lambda^{\mathsf{T}}$ and Λ is lower triangular,

$$\Sigma_{ij} = \sum_{k=1}^{\min(i,j)} \Lambda_{ik}\Lambda_{jk}. \tag{4.26}$$

We assume without loss of generality that $i < j$. The argument for $j > i$ is similar. Suppose for all $k = 1, \ldots, i$, that $\Lambda_{ik} = 0$ or $\Lambda_{jk} = 0$, then by (4.26) i and j are independent. However, by Lemma II.1, Λ_{jk} is the influence of kth node on j, and this is zero only if there is no path from k to j. If i is an ancestor of j, we have

$\Sigma_{ij} \neq 0$. On the other hand, if there is no node $k \in i^-$ such that k influences both i and j, i.e. k is a common ancestor of i and j, then for all $k = 1, \ldots, i$ we have $\Lambda_{ik}\Lambda_{jk} = 0$ and the claim follows.

Using Bonferroni's inequality twice and Lemma IV.2, we get

$$\begin{aligned}
\mathbb{P}(\exists i \in V : \hat{AN}_i \not\subseteq AN_i) &\leq p \max_{i \in V} \mathbb{P}\left(\exists j \in i^- \backslash AN_i : j \in \hat{\text{pa}}_i\right) \\
&\leq p(i-1) \max_{i \in V, j \in i^- \backslash AN_i} \mathbb{P}\left(j \in \hat{\text{pa}}_i\right) \\
&\leq p(i-1) \max_{i \in V, j \in i^- \backslash AN_i} \mathbb{P}\left\{|G_j(\hat{\theta}^{i,AN_i})| \geq \lambda w_{ij}\right\}.
\end{aligned}$$

However, by definition $w_{ij} \geq 1$, and hence it suffices to show that

$$(i-1)p \max_{i \in V, j \in i^- \backslash AN_i} \mathbb{P}\left\{|G_j(\hat{\theta}^{i,AN_i})| \geq \lambda\right\} \leq \alpha. \tag{4.27}$$

Note that $G_j(\hat{\theta}^{i,AN_i}) = -2n^{-1}\mathcal{X}_j^\top(\mathcal{X}_i - \mathcal{X}\hat{\theta}^{i,AN_i})$ and X_j is independent of X_k for all $k \in AN_i$. Therefore, conditional on $\mathcal{X}_{AN_i}$, $G_j(\hat{\theta}^{i,AN_i}) \sim (0, 4R^2/n)$, where $R^2 = n^{-1}\|\mathcal{X}_i - \mathcal{X}\hat{\theta}^{i,AN_i}\|_2^2 \leq n^{-1}\|\mathcal{X}_i\|_2^2 = 1$, by definition of $\hat{\theta}^{i,AN_i}$ and the fact that columns of the data matrix are scaled.

It follows that for all $j \in i^- \backslash AN_i$, $\mathbb{P}\left\{|G_j(\hat{\theta}^{i,AN_i})| \geq \lambda \mid \mathcal{X}_{AN_i}\right\} \leq 2\{1 - \Phi(n^{1/2}\lambda/2)\}$, where Φ is the cumulative distribution function for standard normal random variable. Using the choice of λ proposed in (4.25), we get $\mathbb{P}\left\{|G_j(\hat{\theta}^{i,AN_i})| \geq \lambda \mid \mathcal{X}_{AN_i}\right\} \leq \alpha\{(i-1)p\}^{-1}$, and the result follows. □

Theorem IV.7 is true for all values of p and n, but it does not provide any guarantee for the probability of false positive error for individual edges in the graph. We also need to determine the optimal choice of penalty parameter λ^0 for the first phase of the adaptive lasso, where the weights are estimated using the lasso. Since the goal of the first phase is to achieve prediction consistency, cross validation can be used to determine the optimal choice of λ^0. On the other hand, it is easy to see that the error-based proposal in (4.25) satisfies the requirement of Theorem IV.4 and can therefore be used to define λ^0. It is however recommended to use a higher value of significance level in estimating the initial weights, in order to prevent an over-sparse solution.

4.5 Performance Analysis

4.5.1 Preliminaries

In this section, we consider examples of estimating directed graphs of varying number of edges from randomly generated data. To randomly generate data, one needs to generate lower-triangular adjacency matrices with sparse nonzero elements, ρ_{ij}. However, in order to control the computational complexity of the PC-algorithm, we use the random directed graph generator in the R-package `pcalg` (Kalisch and Bühlmann, 2007), which generates graphs with given values of the average neighborhood size. The sparsity levels of graphs with different sizes are set according to the theoretical bounds in Section 4.4, as well as the recommendations of Kalisch and Bühlmann (2007) for the neighborhood size. More specifically, we use an average neighborhood size of 5, while limiting the total number of true edges to be equal to the sample size n.

Different measures of structural difference can be used to evaluate the performance of estimators. The structural Hamming distance represents the number of edges that are not in common between the estimated and true graphs, i.e. $\textsc{shd} = \text{card}\,(\hat{E}\backslash E) + \text{card}\,(E\backslash\hat{E})$, where $\hat{E}$ and E are defined as in Theorem IV.3. The main drawback of this measure is its dependency on the number of nodes, as well as the sparsity of the network. The second measure of goodness of estimation considered here is the Matthew's correlation coefficient, which is commonly used to assess the performance of binary classification methods (Baldi et al., 2000), and is defined as

$$\text{(4.28)} \qquad \textsc{mcc} = \frac{(\textsc{tp} \times \textsc{tn}) - (\textsc{fp} \times \textsc{fn})}{\{(\textsc{tp} + \textsc{fp})(\textsc{tp} + \textsc{fn})(\textsc{tn} + \textsc{fp})(\textsc{tn} + \textsc{fn})\}^{1/2}},$$

where TP, TN, FP and FN denote the total number of true positive, true negative, false positive and false negative edges, respectively. The value of (4.28) ranges from -1 to 1 with larger values corresponding to better fits, and -1 and 1 representing worst and best fits, respectively. Finally, in order to compare the performance of different estimation methods with theoretical bounds established in Section 4.4.3, we also report the values of true and false positive rates.

The performance of both the PC-algorithm, as well as our proposed estimators based on the choice of tuning parameter in (4.25), vary with different values of significance level α. In the following experiments, we first investigate the appropriate choice of α for each estimator. We then compare the performance of the estimators with an optimal choice of α. The results reported in this section are based on estimates obtained from 100 replications; further to offset the effect of numerical instability, we consider an edge present only if $|\hat{A}_{ij}| > 10^{-4}$.

4.5.2 Estimation of directed graphs from normally distributed observations

We begin with an example that illustrates the differences between estimation of directed graphs and conditional independence graphs. The first two images in Fig. 4.3 represent a randomly generated directed graph of size $p = 50$ along with the gray-scale image of the average precision matrix estimated based on a sample of size $n = 100$ using the graphical lasso algorithm (Friedman et al., 2008b). The image is obtained by calculating the proportion of times that a specific edge is present in 100 replications. To control the probability of falsely connecting two components of the graph, the value of the tuning parameter for the graphical lasso is defined

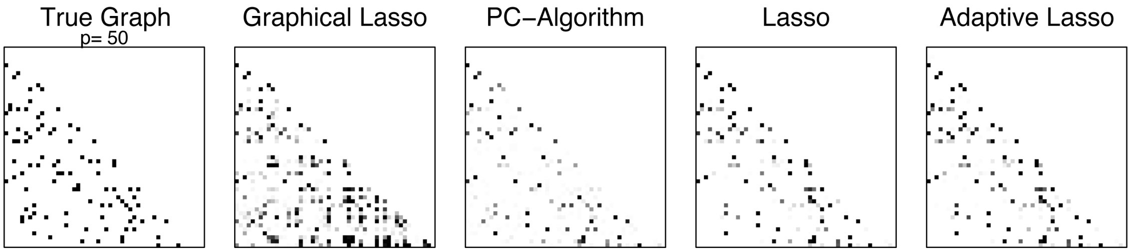


Figure 4.3: True directed graph along with estimates from Gaussian observations. The gray scale represents the percentage of inclusion of edges.

based on the error-based proposal in Banerjee et al. (2008, Theorem 2). It can be seen that the conditional independence graph has many more edges, 8% false positive rate compared to 1% for the lasso and adaptive lasso, and does not reveal the true structure of the underlying directed graph. Therefore, although methods of estimating conditional independence graphs are computationally efficient, they should not be used in applications, like estimation of gene regulatory networks, where the underlying graph is directed.

In simulations throughout this section, the sample size is fixed at $n = 100$, and estimators are evaluated for an increasing number of nodes, $p = 50, 100, 200$. Figure 4.4 shows the mean and standard deviation of the Hamming distances, expressed in base 10 logarithmic scale, for estimates based on the PC-algorithm, as well as the proposed lasso, and adaptive lasso methods for different values of the tuning parameter α and different network sizes. It can be seen that for all values of p and α, the adaptive lasso estimate produces the best results, and the proposed penalized likelihood methods outperform the PC-algorithm. This difference becomes more significant as the size of the network increases.

As mentioned in Section 4.2, it is not always possible to estimate the direction of the edges of a directed graph and therefore, the estimate from the PC-algorithm may include undirected edges. Since our penalized likelihood methods assume knowl-

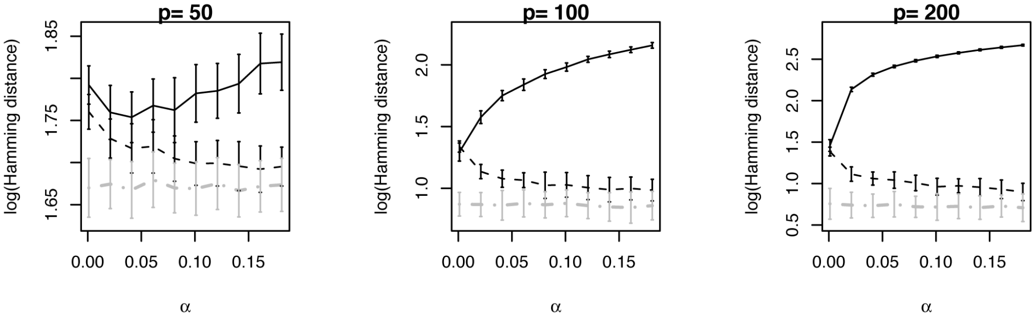


Figure 4.4: Logarithm, in base 10, of the Hamming distances for estimation of directed graph from normal observations. Results of the PC-algorithm (black solid), lasso (black dashes) and adaptive lasso (gray dot-dashes) are shown.

edge of the ordering of variables and estimate the structure of the network, in the simulations considered here, we only estimate the skeleton of the network using the PC-algorithm. We then use the ordering of the variables to determine the direction of the edges. The performance of the PC-algorithm for estimation of partially completed directed graphs may therefore be worse than the results reported here.

In our simulation results, observations are generated according to the linear structural equation model (4.2) with standard normal latent variables and $\rho_{ij} = \rho = 0 \cdot 8$. Additional simulation studies with different values of σ and ρ indicate that changes in σ do not have a considerable effect on the performance of the proposed models. On the other hand, as the magnitude of ρ decreases, the performance of the proposed methods, as well as the PC-algorithm deteriorates, but the findings of the above comparison remain unchanged.

The above simulation results suggest that the optimal performance of the PC-algorithm is achieved when $\alpha = 0 \cdot 01$. The performance of the lasso and adaptive lasso methods is less sensitive to the choice of α; however, a value of $\alpha = 0 \cdot 10$ seems to deliver more reliable estimates. In addition, our extended simulations indicate that the performance of the adaptive lasso does not vary significantly with the value of power γ and therefore we present the results for $\gamma = 1$.

Figure 4.3 represents images of estimated and true directed graphs created based on the above considerations for tuning parameters for $p = 50$. Similar results were also observed for larger values of p, $p = 100, 200$, and are excluded due to space considerations. Plots in Fig. 4.5 compare the performance of the three methods with the optimal settings of tuning parameters, over a range of values of p. It can be seen that the values of MCC confirm the above findings based on the Hamming distance. On the other hand, false positive and true positive rates only focus on one aspect of estimation at a time and do not provide a clear distinction between the methods.

As mentioned in Section 4.2, the representation of conditional independence in directed graphs adapted in our algorithm, is not restricted to normally distributed random variables; if the underlying structural equations are linear, the method proposed in this chapter can correctly estimate the underlying graph. In order to assess the sensitivity of the estimates to the underlying distribution, we performed two simulation studies with non-Normal observations. In both simulations, observations were generated according to a linear structural model. In the first simulation, the latent variables were generated from a mixture of a standard normal and a t-distribution with 3 degrees of freedom, while in the second simulation, a t-distribution with 4 degrees of freedom was used. The performance of the proposed algorithm for non-normal observations was similar to the case of Gaussian observations, with the adaptive lasso providing the best estimates, and the performance of penalized methods improving in sparse settings.

4.5.3 Sensitivity to perturbations in the ordering of the variables

Algorithm 2 assumes a known ordering of the variables. The superior performance of the proposed penalized likelihood methods in comparison to the PC-algorithm may

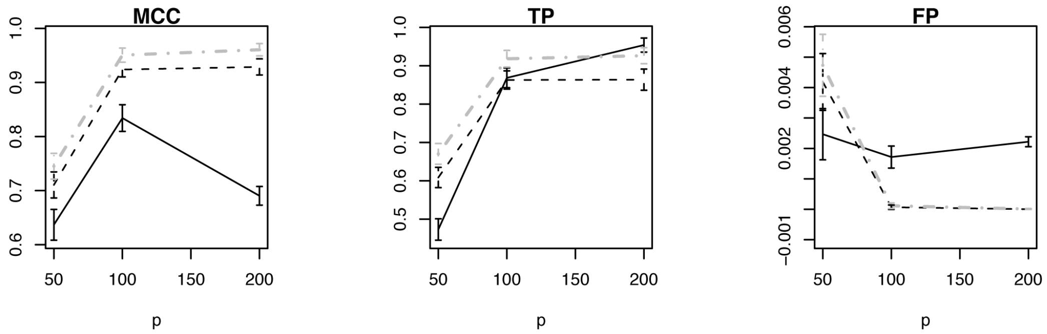


Figure 4.5: MCC, FP and TP for estimation of directed graph from normal observations. Results for the PC-algorithm (black solid), lasso (black dashes) and adaptive lasso (gray dot-dashes) are given.

be explained by the fact that additional information about the order of the variables significantly simplifies the problem of estimating directed graphs. Therefore, when such additional information is available, estimates using the PC-algorithm suffer from a natural disadvantage. However, as the underlying network becomes more sparse, the network includes fewer complex structures and it is expected that the ordering of variables should play a less significant role.

Next, we study the performance of the proposed methods, as well as the PC-algorithm in problems where the ordering of variables is unknown. To this end, we generate normally distributed observations from the latent variable model of Section 4.2.2. We then randomly permute the order of variables in the observation matrix and use the permuted matrix to estimate the original directed graph. Figure 4.6 illustrates the performance of the three methods for choices of α described in Section 4.5.2. It can be seen that for small, dense networks, the PC-algorithm outperforms the proposed methods. This is expected since the change in the order of variables causes the proposed algorithm to include unnecessary moral edges, while failing to recognize some of the existing associations. On the other hand, as the size of the network and correspondingly the degree of sparsity increase, the local structures become simpler and therefore the ordering of the variables becomes less crucial. Thus,

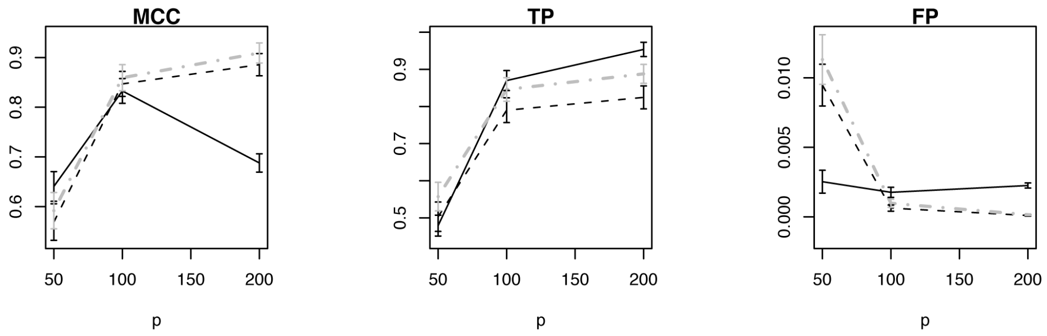


Figure 4.6: MCC, FP and TP for estimation of directed graph with random ordering. Results are given for the PC-algorithm (black solid), lasso (black dashes) and adaptive lasso (gray dot-dashes).

the performance of penalized likelihood algorithms is improved compared to that of the PC-algorithm. For the high dimensional sparse case, where the computational cost of the PC-algorithm becomes more significant, the penalized likelihood methods provide better estimates.

4.6 Real Data Application

4.6.1 Analysis of cell signalling pathway data

Sachs et al. (2003) carried out a set of flow cytometry experiments on signaling networks of human immune system cells. The ordering of the connections between pathway components were established based on perturbations in cells using molecular interventions and we consider the ordering to be known a priori. The data set includes $p = 11$ proteins and $n = 7466$ samples.

Friedman et al. (2008b) analyzed this data set using the graphical lasso algorithm. They estimated the graph for a range of values of the ℓ_1 penalty and reported moderate agreement, around 50% false positive and false negative rates, between one of their estimates and the findings of Sachs et al. (2003). True and estimated signaling networks using the PC-algorithm, with $\alpha = 0 \cdot 01$, and the lasso and adaptive lasso algorithms, with $\alpha = 0 \cdot 1$, along with the corresponding performance measures are given in Fig. 4.7. The estimated network using the PC-algorithm includes a number

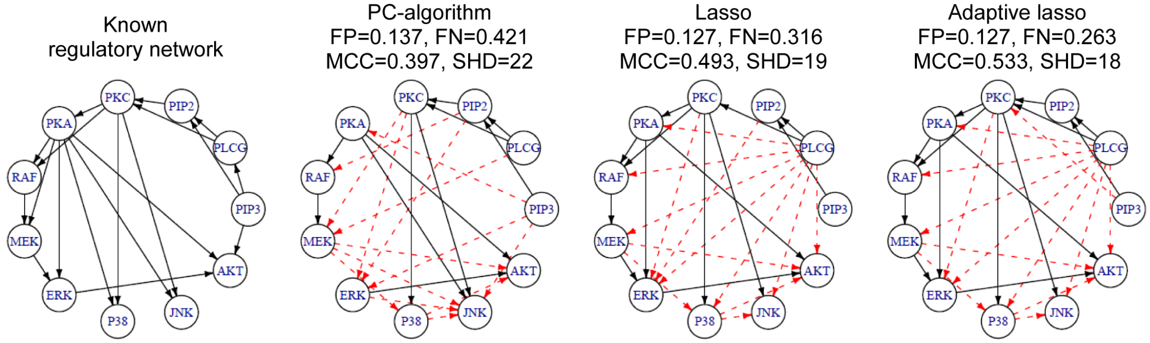


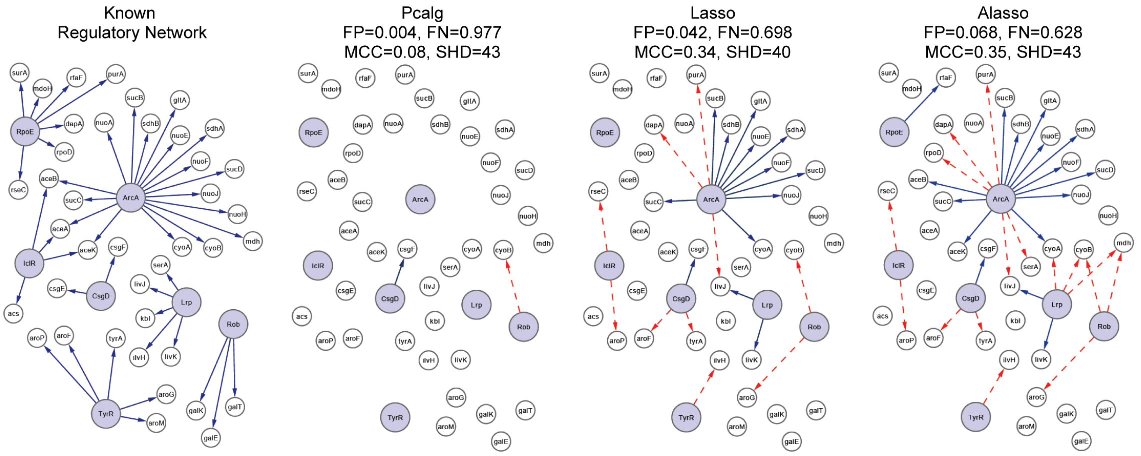
Figure 4.7: Known and estimated networks for human cell signalling data. True and false edges are marked with solid and dashed arrows, respectively.

of undirected edges. As in the simulation studies, we only estimate the structure of the network using the PC-algorithm and determine the direction of edges by enforcing the ordering of nodes. It can be seen that the adaptive lasso and lasso provide estimates that are closest to the true structure.

4.6.2 Transcription regulatory network of E-coli

As we discussed in Chapters II and III, transcriptional regulatory networks play an important role in controlling the gene expression in cells and incorporating the underlying regulatory network results in more efficient estimation and inference. Kao et al. (2004) proposed the network component analysis method to infer transcriptional regulatory network of Escherichia coli, E-coli. They also provided whole genome expression data over time, with $n = 24$, as well as information about the known regulatory network of E-coli.

In this application, the set of transcription factors are known a priori and the goal is to find connections among transcription factors and regulated genes through analysis of whole genome transcriptomic data. Therefore, the algorithm proposed in this chapter can be used by exploiting the natural hierarchy of transcription factors and regulated genes. Kao et al. (2004) provide gene expression data for 7 transcription factors and 40 regulated genes, i.e. $p = 47$. Figure 4.8 presents the known regu-


Known Regulatory Network

Pcalg FP=0.004, FN=0.977 MCC=0.08, SHD=43

Lasso FP=0.042, FN=0.698 MCC=0.34, SHD=40

Alasso FP=0.068, FN=0.628 MCC=0.35, SHD=43

Figure 4.8: Known and estimated transcription regulatory network of E-coli. Large gray nodes indicate the transcription factors, and true and false edges are marked with solid and dashed arrows, respectively.

latory network of E-coli along with the estimated networks and the corresponding performance measures using the PC-algorithm, lasso and adaptive lasso. The values of α are set as in Section 4.6.1. The relatively poor performance of the algorithms in this example can be partially attributed to the small sample size. However, it is also known that no single source of transcriptomic data is expected to successfully reveal the regulatory network and better estimates are obtained by combining different sources of data. It can be seen that the PC-algorithm can only detect one of the true regulatory connections, and both lasso and adaptive lasso offer significant improvements, mostly due to the considerable drop in the false negative rate, from 97% for the PC-algorithm to 63% for the adaptive lasso. In this case, lasso and adaptive lasso estimates are very similar, and the choice of the best estimate depends on the performance evaluation criterion.

4.7 Discussion

The penalized likelihood methods for estimation of the structure of directed acyclic graphs proposed in this chapter are derived based on the assumption that the variables inherit a natural ordering. These methods are applicable in a number of biological applications, where the ordering of the variables is known a priori, including estimation of transcriptional regulatory networks from gene expression data and reconstruction of causal networks from temporal observations. However, simulation studies indicate that the correct ordering of variables becomes less crucial for estimating high dimensional sparse graphs, and the methods proposed in this chapter may be an efficient alternative for search-based methods of estimating directed graphs.

CHAPTER V

Estimation of Graphical Granger Causality Using the Truncating Lasso Penalty

5.1 Introduction

A critical problem in systems biology is to discover causal relationships among components of biological systems. Gene regulatory networks, metabolic networks and cell signalling networks capture causal relationships in cells. Discovery of causal relationships may be only possible through carefully designed experiments, which can be challenging. However, gene regulation is carried out by binding of protein products of transcription factors to *cis*-regulatory elements of genes. Such regulatory mechanisms are evident if the expression levels of gene X is affected by changes in expression levels of gene Y. Therefore, time course gene expression data can be used to discover causal relationships among genes and construct the gene regulatory network.

Different methods have been developed to infer causal relationships from time series data, including dynamic Bayesian Networks (Murphy, 2002) and Granger causality (Granger, 1969). In dynamic Bayesian Networks (DBNs) the state space of Bayesian Networks is expanded by replicating the set of variables in the network by the number of time points. Cyclic networks are then transformed to DAGs by breaking down cycles into interactions between variables at two different time points. Ong et al. (2002) and Perrin et al. (2003) among others have applied Bayesian networks to infer causal relationships among components of biological systems.

On the other hand, the concept of Granger causality states that gene X is Granger-causal for gene Y if the autoregressive model of Y based on past values of both genes is significantly more accurate than the model based on Y alone. This implies that changes in expression levels of genes could be explained by expression levels of their transcription factors. Therefore, statistical methods can be applied to time-course gene expression observations to estimate Granger causality among genes.

Exploring Granger causality is closely related to analysis of multivariate vector autoregressive (VAR) models, which are widely used in econometrics. Yamaguchi et al. (2007) and Opgen-Rhein and Strimmer (2007) employed VAR models to learn gene regulatory networks, while Fujita et al. (2007) proposed a sparse VAR model for better performance in cases when the number of genes, p is large compared to the sample size, n. Similar sparse models have also been considered by Mukhopadhyay and Chatterjee (2007).

Zou and Feng (2009) compared the performance of DBNs and Granger causality methods for estimation of causal relationships and concluded that the performance of the two approaches depend on the length of the time series as well as the sample size. The findings of Zou and Feng (2009) emphasizes the need for sparse models in cases where the sample size is small. In particular, when $p \gg n$, penalized methods often provide better prediction accuracy. Arnold et al. (2007) applied the lasso (or ℓ_1) penalty to discover the structure of graphical models based on the concept of Granger causality and studied the relationship between different key performance indicators in analysis of stock prices.

Asymptotic and empirical performances of the lasso penalty for discovery of graphical models have been studied by many researchers and a number of extensions of the original penalty have been proposed (we refer to these variants of the lasso penalty as "lasso-type" penalties). Meinshausen and Bühlmann (2006) showed that the lasso penalty does not achieve variable selection consistency unless the so-called *irrepresentability* assumption holds. To overcome the shortcomings of the lasso penalty, Zou (2006) proposed the adaptive lasso penalty, and showed that for fixed p, if appropriate weights are used, the adaptive lasso penalty can achieve variable selection consistency even if the irrepresentability assumption is violated. In fact, as we showed in Chapter IV, if initial weights are derived from regular lasso estimates, the adaptive lasso penalty is also consistent for variable selection in high dimensional sparse settings.

In Chapter IV, we discussed the representation of the joint probability distribution of random variables in graphical models on directed acyclic graphs (DAGs). We also proposed a penalized likelihood approach, using both lasso and adaptive lasso penalties, for estimation of the skeleton of DAGs and discussed the asymptotic properties of the resulting estimators.

The lasso estimate of the graphical Granger model may result in a model in which X is considered to influence Y in a number of different time lags. Such a model is hard to interpret and inclusion of additional covariates in the model may result in poor model selection performance. Lozano et al. (2009) have recently proposed to use a group lasso penalty (Yuan and Lin, 2007) in order to obtain a simpler Granger graphical model. The group lasso penalty takes the average effect of X on Y over different time lags and considers X to be Granger-causal for Y if the average effect is significant. However, this results in significant loss of information, as the time difference between activation of X and its effect on Y is ignored. Moreover, due to the averaging effect, the sign and magnitude of effects of the variables on each other can not be determined from the group lasso estimate. Hence, whether X is an activator or a suppressor for Y and/or the magnitude of its effect remain unknown.

In this chapter, I propose a novel *truncating lasso* penalty for estimation of graphical Granger models. The proposed penalty has two main features: (i) it automatically determines the order of the VAR model, i.e. the number of effective time lags and (ii) it performs model simplification by reducing the number of covariates in the model. I propose an efficient iterative algorithm for estimation of model parameters, provide an error-based choice for the tuning parameter and prove the consistency of the resulting estimate, both in terms of sign and magnitude of the effects, as well as, variable selection properties. The proposed method is applied to simulated and real data examples, and is shown to provide better estimates than alternative penalization methods.

The remainder of the chapter is organized as follows. Section 5.2, starts with a discussion of review of the concept of graphical Granger causality. The proposed truncating lasso penalty and asymptotic properties of the estimator are discussed in section 5.2.2, while the optimization algorithm is presented in section 5.2.4. Results of simulation studies are presented in section 5.3.1 and applications of the proposed model to time course gene expression data on E-coli and human cancer cell line (HeLa cells) are illustrated in sections 5.3.2 and 5.3.3, respectively. A summary of findings and directions of future research are discussed in section 5.4.

5.2 Model and Methods

5.2.1 Graphical Granger Causality

Let $X^{1:T} = \{X\}_{t=1}^{T}$ and $Y^{1:T} = \{Y\}_{t=1}^{T}$, be trajectories of two stochastic processes X and Y up to time T and consider the following two regression models:

$$Y^T = AY^{1:T-1} + BX^{1:T-1} + \varepsilon^T \tag{5.1}$$

$$Y^T = AY^{1:T-1} + \varepsilon^T \tag{5.2}$$

Then X is said to be Granger-causal for Y if and only if the model 5.1 results in significant prediction improvement over model 5.2. Graphical Granger models extend the notion of Granger causality among two variables to p variables. More generally, let $X_1, \ldots, X_p$ be p stochastic processes and denote by $\mathbf{X}$ the rearrangement of these stochastic processes into a vector time series, i.e.

$$\mathbf{X}^t = (X_1^t, \ldots, X_p^t)^\intercal \tag{5.3}$$

We consider models of the form

$$\mathbf{X}^T = A^1\mathbf{X}^{T-1} + \ldots A^{T-1}\mathbf{X}^1 + \varepsilon^T. \tag{5.4}$$

In the graphical Granger model, X_j^t is said to be causal for X_i^T if the corresponding coefficient, $A_{i,j}^t$ is statistically significant. In that case, there exists an edge $X_j^t \rightarrow X_i^T$ in the graphical model with $T \times p$ nodes.

Such a model corresponds to a DAG with $T \times p$ variables, in which the ordering of the set of p-variate vectors $\mathbf{X}^1, \ldots, \mathbf{X}^T$ is determined by the temporal index and the ordering among the elements of each vector is arbitrary. Lasso-type estimates of DAGs can therefore be used in the context of graphical Granger models in order to select causal effects of variables on each other. The model in (5.4) is also equivalent to vector autoregressive (VAR) models, which have been used for estimation of graphical Granger causality by a number of researchers, including Arnold et al. (2007).

5.2.2 Truncating Lasso for Graphical Granger Models

Consider a graphical model with p variables, observed over T time points, and let d be the order of the VAR model or the effective number of time lags (in (5.4) $d = T-1$). As in section Chapter IV, let $\mathcal{X}^t$ denote the design matrix corresponding to t-th time point, and $\mathcal{X}_i^t$ be its i-th column.

The truncating lasso estimate of the graphical Granger causality is found by solving the following estimation problem for $i = 1, \ldots, p$:

$$\operatorname*{argmin}_{\theta^t \in \mathbb{R}^p} n^{-1} \|\mathcal{X}_i^T - \sum_{t=1}^{d} \mathcal{X}^{T-t}\theta^t\|_2^2 + \lambda \sum_{t=1}^{d} \Psi^t \sum_{j=1}^{p} |\theta_j^t| w_j^t \tag{5.5}$$

$$\Psi^1 = 1, \qquad \Psi^t = M^{I\{\|A^{(t-1)}\|_0 < p^2\beta/(T-t)\}},\ t \geq 2$$

where M is a large constant, and β is the allowed false negative rate, determined by the user. The choice of β and the properties of the resulting estimator are discussed in the remainder of this section.

To illustrate the main idea behind the truncating lasso penalty, we begin by examining the regular lasso estimate of the graphical Granger model. Using the above notation, the general weighted lasso estimate of the graphical Granger model is found by solving the following p non-overlapping ℓ_1-regularized least square problems for $i = 1, \ldots, p$:

$$\operatorname*{argmin}_{\theta^t \in \mathbb{R}^p} n^{-1} \|\mathcal{X}_i^T - \sum_{t=1}^{d} \mathcal{X}^{T-t}\theta^t\|_2^2 + \lambda \sum_{t=T-d}^{T-1} \sum_{j=1}^{p} |\theta_j^t| w_j^t \tag{5.6}$$

The weighted lasso penalty suffers from two limitations. Firstly, the order of the VAR model, d is often unknown and therefore is set to $T-1$. Hence there are $p(T-1)$ covariates in the weighted lasso estimation problem. Moreover, the weighted lasso estimate may potentially include edges from different time points of variable X_j to any given variable X_i. To overcome these problems, Lozano et al. (2009) proposed to use the group lasso estimate, in which the values of coefficients of each variable over the past time points are grouped. The drawback of group lasso penalty is that information on the time lag between activation of gene j and its effect on gene i is lost. Moreover, the resulting estimate does not provide consistent information about the magnitude and sign of the interaction. Thus, important questions including the activation or inhibition effect of X_j on X_i can not be answered.

The proposed truncating lasso penalty addresses the above shortcomings of the regular lasso penalty, while preventing the loss of information which occurs if the group lasso penalty is used. The truncating effect of the proposed penalty (imposed by Ψ^t) is motivated by the rationale that the number of effects (edges) in the graphical model decreases as the time lag increases. Consequently, if there are fewer than $p^2\beta/(T-t)$ edges in the $(t-1)$st estimate, all the later estimates are forced to zero. Hence, the truncating lasso penalty provides an estimate of the order of the underlying VAR model. In addition, by applying this penalty the number of covariates in the model is reduced as the coefficients for effects of genes on each other after the estimated time lag are forced to zero.

The following result indicates that the proposed truncating lasso penalty provides a desirable estimate of the graphical Granger causality. In particular, we show that the resulting estimate is consistent for variable selection (i.e. the correct edges are estimated with increasing probability, as the sample size increases). Moreover, the magnitude and signs of the effects are consistently estimated. A more detailed version of the theorem, including assumptions and a sketch of the proof is given in the Appendix.

Theorem V.1 (Consistency of Truncating Adaptive Lasso). *Let s be the total number of true edges in the graphical Granger model and suppose that for some $a > 0$, $p = p(n) = O(n^a)$ and $|\mathrm{pa_i}| = \mathrm{O(n^b)}$, where $sn^{2b-1}\log n = o(1)$ as $n \to \infty$. Moreover, suppose that there exists $\nu > 0$ such that for all $n \in \mathbb{N}$ and all $i \in V$, $\mathrm{Var}\left(X_i^T | X_{1:p}^{T-d:T-1}\right) \geq \nu$ and there exists $\delta > 0$ and some $\xi > b$ such that for every $i \in V$ and for every $j \in \mathrm{pa_i}$, $|\pi_{ij}| \geq \delta n^{-(1-\xi)/2}$, where π_{ij} is the partial correlation between X_i and X_j after removing the effect of the remaining variables.*

Assume that $\lambda \sim dn^{-(1-\zeta)/2}$ for some $b < \zeta < \xi$ and $d > 0$, and the initial weights are found using lasso estimates with a penalty parameter λ^0 that satisfies $\lambda^0 = O(\sqrt{\log p/n})$. Also, for some large positive number g, let

$$\Psi^t = g \exp\left(nI\{\|A^{(t-1)}\|_0 < p^2\beta/(T-t)\}\right)(\text{ i.e. } M = ge^n).$$

Then if true causal effects diminish over time, there exists a solution of the truncating adaptive lasso penalty and constants $c_{(i)} - c_{(iv)}$ that satisfy the following:

(i) Type I error: $\mathrm{pr}\left(\hat{\mathrm{pa}}_i \subseteq \mathrm{pa}_i\right) = 1 - \beta - O\left\{\exp\left(-c_{(i)}n^{\zeta}\right)\right\}$

(ii) Type II error: $\mathrm{pr}\left(\mathrm{pa}_i \subseteq \hat{\mathrm{pa}}_i\right) = 1 - O\left\{\exp\left(-c_{(ii)}n^{\zeta}\right)\right\}$

(iii) Order of VAR: $\mathrm{pr}\left\{\hat{d} = d\right\} = 1 - \beta - O\left\{\exp\left(-c_{(iii)}n^{\zeta}\right)\right\}$

(iv) Direction of influence: For $i, j = 1, \ldots, p, t = 1, \ldots T-1$,

$$\mathrm{pr}\left\{\mathrm{sign}\left(\hat{A}_{ij}^t\right) = \mathrm{sign}\left(A_{ij}^t\right)\right\} = 1 - O\left\{\exp\left(-c_{(iv)}n^{\zeta}\right)\right\}$$

Proof. If $\beta = 0$, inclusion of the true causal effect, exclusion of incorrect effects and consistency of signs and magnitudes of effects follow from Theorem IV.5. Since β has no effect on the probability of false positive, this proves (ii) and (iii). For any given $\beta > 0$, suppose t_0 is the smallest t for which $\|A^{(t-1)}\|_0 < p^2\beta/(T-t)$. Then for $t < t_0$ $\Psi^t = 1$ and has no effect on the estimate. Let $t \geq t_0$. Then using the KKT conditions, a coefficient is included in the weighted lasso estimate only if $|2n^{-1}(\mathcal{X}_j^t)^\intercal(\mathcal{X}_i^T - \mathcal{X}^t\theta^t)| > \Psi^t\lambda w_j^t$. However, $(\mathcal{X}_j^t)^\intercal(\mathcal{X}_i^T - \mathcal{X}^t\theta^t)$ is stochastically smaller than $(\mathcal{X}_j^t)^\intercal\mathcal{X}_i^T$, which is in turn a polynomial function of n. On the other hand, λ and w_j^t are also polynomial functions of n, whereas Ψ^t increases exponentially as $n \to \infty$. Hence, for all $j = 1, \ldots, p$ and $t \geq t_0$, there exists an n such that $|2n^{-1}(\mathcal{X}_j^t)^\intercal(\mathcal{X}_i^T - \mathcal{X}^t\theta^t)| < \Psi^t\lambda w_j^t$ and therefore, $A^t = 0, t \geq t_0$. However, since the number of true causal effects diminish over time, the total number of true edges in time lags $t \geq t_0$ is less than β. This proves the first part of (i).

Finally, to prove that the order of VAR is correctly estimated, i.e. $d = t_0 - 1$, we consider two complementary events: $d < t_0 - 1$ and $d > t_0 - 1$. Prior to t_0, false positives occur with exponentially small probability, hence, the probability that $d < t_0 - 1$, is negligible. On the other hand, $d > t_0 - 1$ only if true edges are not included in $\hat{A}_0^t$ and as a result $\|\hat{A}^{(t_0-1)}\|_0 < p^2\beta/(T-t_0)$. But false negatives occur if true edges vanish in the adaptive lasso estimate. However, adaptive lasso finds the true edges with exponentially large probability, hence, $\mathbb{P}(d < t_0 - 1) \geq 1 - \beta - O(\exp(-cn^d))$ for constants c and d. This completes the proof. □

5.2.3 Choice of the Tuning Parameter

Estimation of the graphical Granger model using the truncating lasso penalty requires selection of two parameters, λ and β. As mentioned in the previous section, β is the allowed rate of false negatives. Therefore, selection of β can be based on the cost of false negatives in the specific problem at hand, as well as the sample size; as with any other statistical test, as sample size increases, smaller values of β can be considered. A practical strategy for selecting β is to first find the lasso (or adaptive lasso) estimate and select β so that the desired false negative rate is achieved.

The second parameter, λ is common in all penalized estimation methods. We propose the following error-based choice for selection of λ. Let Z_q^* be the $(1-q)$-th percentile of the standard normal distribution and consider:

$$\lambda = 2n^{-1/2} Z^*_{\frac{\alpha}{2dp^2}} \tag{5.7}$$

it then follows from Theorem IV.7 that for any value of n, this choice of λ controls a version of false positive rate at the given level of α, provided that columns of the design matrix are scaled so that $n^{-1}\mathcal{X}_i^\intercal \mathcal{X}_i = 1$. In section 5.3.1, we evaluate the performance of the proposed method for a range of values of α, and show that the performance is not heavily influenced by that choice.

5.2.4 Algorithm and Computational Complexity

In the previous section, we discussed that the truncating lasso estimate of the graphical Granger model in (5.5) is found by solving p weighted lasso problems. However, the optimization problem in (5.5) is non-convex and can not be solved directly, especially since the truncating factor Ψ^t depends on the values of the coefficients at the previous time points. Here we propose an iterative Block-Relaxation algorithm (de Leeuw, 1994), which can be efficiently used to estimate the parameters of the model.

The main idea of the algorithm is to further break down each of the p sub-problems into d weighted lasso problems, starting with the observations at the most recent time lag, $T-1$. This iterative process is continued by calculating the truncating factor Ψ^t at each $t = 1, \ldots, d$ based on the values of the coefficients at the previous time points and solving a weighted lasso problem over p variables at each time point. Algorithm 3 outlines the above iterative procedure for finding the estimates of the graphical Granger model.

Unlike the (adaptive) lasso problem, the objective function of the truncating lasso problem is non-convex. Therefore, a global minimum for the resulting optimization problem may not exist. However, the following result shows that the proposed algorithm always converges, although the accumulation point may be a local minimum.

Lemma V.2. *Algorithm 3 converges to a (local) minimizer of the (adaptive) truncating lasso estimation problem.*

(Sketch of the Proof). Although the overall objective function is non-convex, each sub-problem is a weighted lasso problem and is therefore convex. It is also known that with high probability, the solution of the lasso problem is unique (see Osborne et al., 2000). On the other hand, each subproblem is obtained by a (continuous) projection mapping from $\mathbb{R}^{d \times p}$ to $\mathbb{R}^p$. The lemma follows from the results of de Leeuw (1994). □

Both lasso as well as adaptive lasso problems include $d \times p$ covariates in each penalized regression problem. Therefore, using the `shooting` Algorithm of Friedman et al. (2008a) (implemented in the R-package `glmnet`), estimation of the (adaptive) lasso problem requires $O(nd^2p^2)$ operations. On the other hand, partitioning over time points reduces the computational burden of each subproblem to $O(ndp^2)$. From the general theory of Block-Relaxation algorithms (see de Leeuw, 1994), it can be

Algorithm 3 Iterative Algorithm for Estimation of Truncation Lasso

```
Repeat for $k = 1, 2, \dots$ (until convergence)
1. For $t = 1, \dots, d$
    1.1. Calculate $\Psi^t$ based on estimates in $t' = 1, \dots, t-1$
    1.2. Using the most recent estimate $\hat{A}^{t'}$, find:
       $R^t = \mathcal{X}^T - \sum_{t'=1, t' \neq t}^{d} \hat{A}^{t'} \mathcal{X}^{T-t'}$
2. For $i = 1, \dots, p$, let $r := R_i^t$
    2.1. Using the shooting Algorithm, solve
argmin $\left\{ n^{-1} \|r - \sum_{t=1}^{d} \mathcal{X}^{T-t} \theta^t\|_2^2 + \lambda \sum_{t=1}^{d} \Psi^t \sum_{j=1}^{p} |\theta_j^t| w_j^t \right\}$
```

shown that Algorithm 3 has at least a linear convergence rate. However, in our extensive simulation studies, the algorithm often converges in less than 10 iterations, and for large values of d, may require less time than the lasso estimate.

5.3 Results

5.3.1 Simulation Studies

We evaluate the performance of the proposed truncating lasso penalty, as well as the lasso and adaptive lasso penalties in reconstructing the Granger graphical models from time series observations. Two sets of simulation studies are performed, in which 20 independent and identically distributed (*i.i.d.*) observations for 20 genes are generated according to a VAR model of order 2 ($d = 2$). In both simulations, a Gaussian noise with standard error of $\sigma = 0.2$ is added to the observations and the time series includes 10 time points. In the first simulation study, we enforce the autoregulatory interactions, by including an edge from X_i^{t-1} to X_i^t for all genes i. This implies that the expression level of each gene is influenced by its expression level at the previous time point. Autoregulatory interactions are included randomly in the second simulation study.

Figures 5.1, 5.2 and 5.3 illustrate the performance of lasso (`lasso`), adaptive lasso (`Alasso`), truncating lasso (`Tlasso`) and truncating adaptive lasso (`TAlasso`) estimators in the simulation studies. Figure 5.1 provides a graphical illustration of the estimated networks and the effect of the truncating penalty. As it can be seen, both lasso and adaptive lasso estimates include additional edges beyond the true order of the VAR model (indicated by small rectangles). On the other hand, since the number of covariates ($d \times p$) is much larger than the sample size n, these estimators can not correctly distinguish some of the true edges. However, by reducing the number of covariates through truncation, the truncating lasso penalty provides

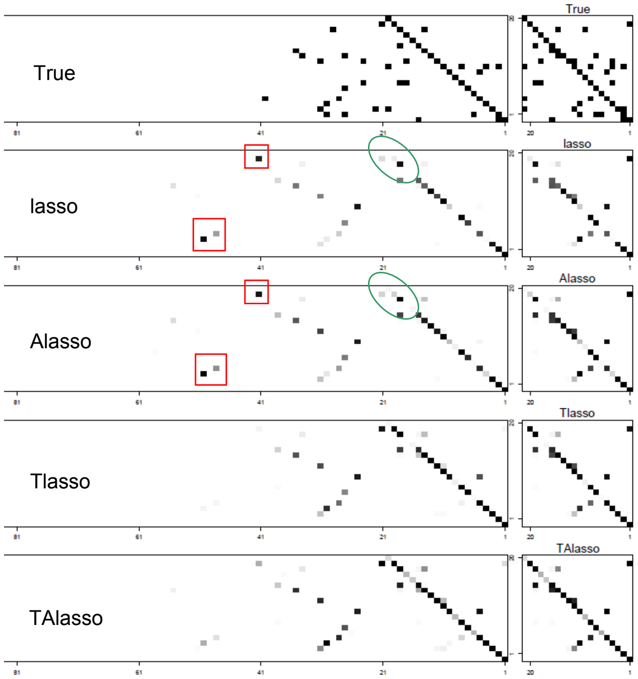


Figure 5.1: Images of the adjacency matrix of estimated graphs for `lasso`, `Alasso`, `Tlasso` and `TAlasso`. The gray-scale represent percentage of times where an edge is present in 50 simulations with $\alpha = 0.1$. Images on the left correspond to the performance of estimators over time, and images on the right represent the cumulative estimate of the graphical model. Some false positives and negatives are marked with rectangles and ovals.

improvements for both of these shortcomings. Similar results are obtained for the second simulation study, and are excluded, due to the space limitation.

To measure the performance of the estimators, we consider average values, as well as standard deviations, of three different performance criteria over 50 simulations:

(1) The Structural Hamming Distance (SHD) between the estimated graph and the true network, which measures the total number of differences in edges of the estimated graph compared to the true network,

(2) The F_1 measure, which is the harmonic mean of *precision* (P) and *recall* (R) for the estimated graphs, and

(3) The partial ROC curve, obtained by plotting the values of true positive rate against the values of false positive rate.

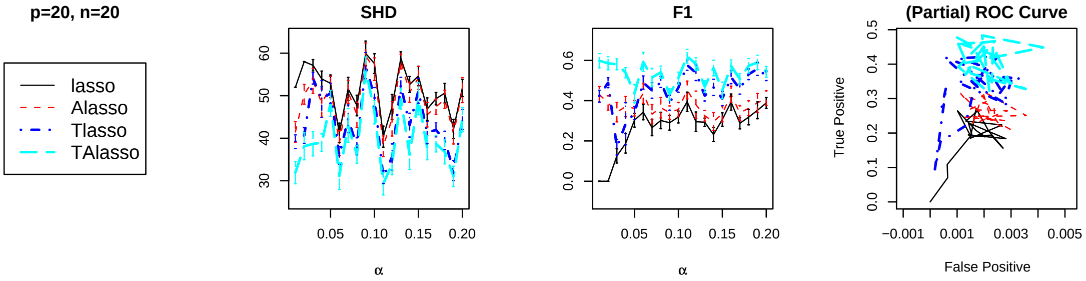


Figure 5.2: Mean and standard deviation of performance criteria for `lasso`, `Alasso`, `Tlasso` and `TAlasso` in estimation of graphical model over time.

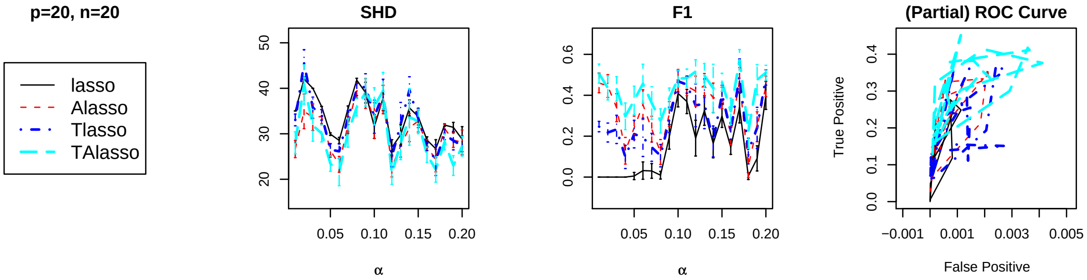


Figure 5.3: Mean and standard deviation of performance criteria for `lasso`, `Alasso`, `Tlasso` and `TAlasso` in estimation of graphical model, ignoring the time lag information.

In both simulations, the value of the tuning parameter for the penalty coefficient, α, is varied from 0.01 to 0.2, while the value of the second tuning parameter for the truncating lasso penalty, β is fixed at 0.1.

Figures 5.2 and 5.3 illustrate the performance of the above estimators in correctly estimating the graphical model over time, in the first and second simulation studies, respectively. It can be seen that in all cases, `TAlasso` provides the best estimate. On the other hand, the advantage of `Tlasso` over `Alasso` diminishes in the absence of autoregulatory interactions.

The above simulation studies provide additional evidence in favor of the adaptive lasso procedure, and that the proposed truncation mechanism offers additional improvement for estimation of Granger causality over the regular version of the lasso penalty. Additional simulation studies indicate that as the sample size increases, lasso and adaptive lasso estimates can overcome the curse of dimensionality and hence the truncating lasso estimates offer less significant improvements. Also, in simulations, where the time lag information is ignored, i.e. if only the structure of the graphical models is considered, the performance of the proposed methods is improved, but the comparative results obtained here remain unchanged.

5.3.2 Analysis of the Regulatory Network of E-coli

Kao et al. (2004) proposed to use Network Component Analysis to infer the transcriptional regulatory network of Escherichia coli (E-coli). They also provided whole genome expression data over 8 time points with different sample sizes, as well as information about the known regulatory network of E-coli. Figure 5.4 represents true and estimated regulatory networks along with performance measures of both `Alasso`, as well as `TAlasso` penalties. It can be seen that the rate of recall is improved in the `TAlasso` estimate, resulting in a higher F_1 measure. The improved performance of the `TAlasso` penalty in comparison to the `Alasso` penalty, as well as the overall performance of this estimator, further validate our numerical analysis.

For comparison, we also provide the estimated regulatory network using our implementation of the group lasso penalty of Lozano et al. (2009) (`grpLasso`). It can be seen that in comparison to `TAlasso`, `grpLasso` performs poorly in this example[1].

5.3.3 Analysis of BioGRID Network in HeLa Cells

The genome-wide expression of cell cycle genes in human cancer cell lines (HeLa) were analyzed by Whitfield et al. (2002). The authors performed different experiments resulting in multiple mRNA time-course samples. Sambo et al. (2008) extracted a subset of 9 genes from the human cell cycle genes for which the regulatory network is already determined in the BioGRID database (www.thebiogrid.org).

[1] This may be due to the fact that our implementation does not achieve the optimal setting of parameters, or other details not specified in Lozano et al. (2009).

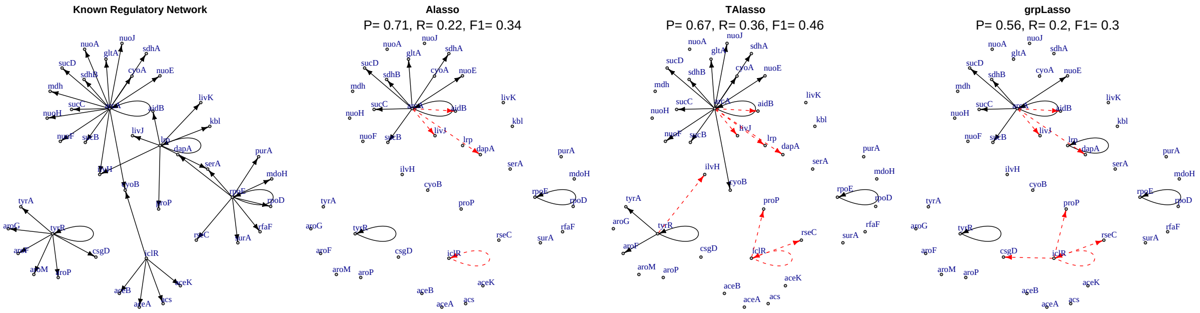


Figure 5.4: Known transcription regulatory network of E-coli along with estimates based on `Alasso`, `TAlasso` and `grpLasso`. True edges (True Positives in estimated networks) are marked with solid black arrows, while False Positives are indicated by dashed red arrows.

The authors developed an algorithm for reverse engineering causal gene networks, called `CNET`, and applied it to this data set. `CNET` is a search-based algorithm, which searches over the space of possible graphs, in order to find the candidate graph with the highest score.

This set of 9 genes was also analyzed by Lozano et al. (2009). Figure 5.5 represents the true regulatory network along with estimated networks using our proposed `TAlasso` estimate, as well as the estimates based on the group lasso and `CNET` methods. As with the other two groups, we used the third experiment of Whitfield et al. (2002), consisting of 47 time points and we considered a maximum time lag of $d = 3$. The estimates for group lasso and `CNET` were reconstructed based on the plots presented by authors, ignoring autoregulatory interactions in the group lasso estimate[2]. The best performance is achieved by the `CNET` algorithm and the authors point out that this result is in line with the best performance obtained in simulated data sets. The performance of the `TAlasso` method is slightly better than the group lasso estimate. It is important to note that although penalization methods (group lasso and truncating lasso) fail to perform as well as search-based algorithms like the

[2] There appears to be a typo in results of Lozano et al. (2009): The BioGRID network should be referred to as the network in Figure 5b (instead of 5a in the paper). Also, the precision, recall and F_1 measures based on the network in Figure 5 are different from the values reported in the paper.

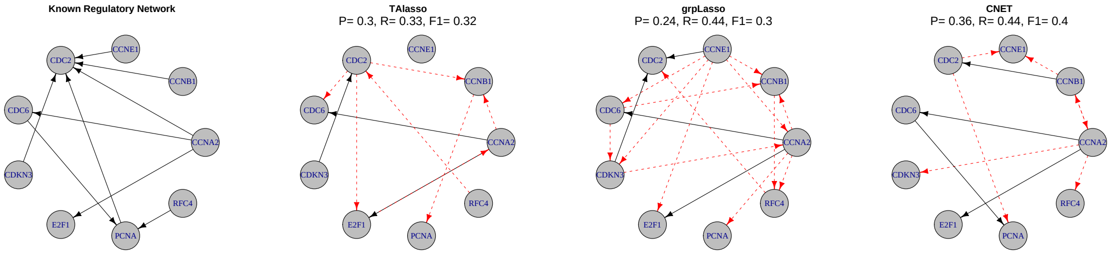


Figure 5.5: Known BioGRID network of human Hela Cell genes along with the estimates based on `TAlasso`, `grpLasso` and `CNET`. True edges (True Positives in estimated networks) are marked with solid black arrows, while False Positives are indicated by dashed red arrows.

`CNET` algorithm, they are computationally more efficient and can be used to analyzed large networks, whereas search-based algorithm become intractable for analysis of real-world biological networks.

It can be seen from Figure 5.5 that two of the correctly estimated edges, from CCNA2 to CDC6 and E2F1, are shared in all three estimates and that all true positives of `TAlasso` are also found by `grpLasso`. On the other hand, a number of estimated edges not present in the BioGRID network are found in two or more estimates. This may suggest that some of the estimated edges (e.g. the edge from CCNA2 to CCNB1) may represent valid regulatory links that are not included in the BioGRID data set. Validation of such hypotheses requires further investigations and/or experimental studies.

A main advantage of the truncating lasso estimate is that it also provides information on the time lag of regulatory effects of transcription factors on other genes. Table 5.1 provides details of information on effective time lags of effects of genes in the network. Such information provides valuable clues to the underlying regulatory mechanism but is overlooked in the other two methods.

5.4 Concluding Remarks

Estimation of gene regulatory networks is a crucial problem in computational biology. Information conveyed from these networks can be exploited to improve estimation and inference procedures, in particular to determine which pathways are involved in the cell's response to environmental factors or in disease progression (see Chapters II and III). Such information is also critical in drug development and medicine. In this chapter, we proposed a novel penalization method, called truncating lasso, for estimation of gene regulatory networks based on the concept of Granger causality. The proposed method can correctly determine the order of the underlying time series, and uses that information to reduce the number of covariates. Such reduction, in turn results in better false positive and false negative rates. Moreover, the proposed method provides information on the time lags of regulatory effects of genes on each other.

Granger causality is an intuitive concept and its underlying assumption (that expressions of genes at each time point are only affected by expression levels at previous times) can be justified in the study of biological systems. However, from a technical point of view, it may be possible to reformulate the resulting autoregressive model using different causal relationships. A more practical issue concerns the time lags between observations: When observations are observed on coarse time intervals, some of the underlying causal effects may not be distinguishable. The success of

Table 5.1: Time lag of regulatory effects of genes in the estimate of BioGRID network based on the `TAlasso` algorithm.

Interaction	Time lag	Interaction	Time lag
CCNA2 → CCNB1	1	CDC2 → CDC6	1
CDNK3 → CDC2	1	CDC2 → E2F1	2
CCNA2 → E2F1	1	CCNA2 → CDC6	2
CCNB1 → PCNA	1	E2F1 → CCNA1	2
CDC2 → CCNB1	1	RFC4 → CDC2	2

reverse engineering algorithms, in particular penalization methods, requires repeated time series observations over fine time grids.

The method proposed in this chapter offers significant improvements over both lasso and adaptive lasso estimates, especially for small to moderate sample sizes. This is achieve by excluding unnecessary covariates from the regression problem. Further improvements may be possible by exploiting the stationarity of the stochastic process in order to take advantage of full information provided in the time series, and should be considered in the future.

Availability

The proposed truncating lasso method, as well as the penalized likelihood estimation method of Chapter IV are implemented in the R-package 'grangerTlasso' and is freely available at www.stat.lsa.umich.edu/∼shojaie.

CHAPTER VI

Dimension Reduction and Inference in High Dimensional Networks Using Laplacian Eigenmaps

6.1 Introduction

As discussed in previous chapters, interactions among components of biological systems play a crucial role in determining the state of the system. Therefore, incorporating available information about such interactions provides more efficient methods of estimation and inference. In Chapter II, we reviewed two classes of approaches that attempt to account for interactions among biological components. The first approach, gene set analysis (Subramanian et al., 2005; Tian et al., 2005; Efron and Tibshirani, 2007), is based on the idea that by assessing the significance of sets rather than individual genes/proteins, interactions among them can be preserved. In addition, this method unveils the combined effect of sets of genes/proteins, which may not be evident from tests of individual components. On the other hand, the second approach, network-based analysis (Ideker et al., 2002; Wei and Li, 2007; Wei and Pan, 2008; Sanguinetti et al., 2008), directly incorporates the interactions among genes/proteins, in order to assess the significance of individual components. To combine the advantages of both of these approaches, in Chapter II, we introduced a model for incorporating the regulatory gene network, and developed an inference framework for analysis of biological pathways. In Chapter III, we extended this framework, called NetGSA, to analyze general gene/protein networks, in complex experimental conditions.

In the NetGSA framework, a global model is introduced with parameters for individual genes/proteins, and the parameters are then combined appropriately in order to assess the significant (or changes) in biological pathways. The test statistic proposed in NetGSA has desirable properties, including efficiency and interpretability. However, the main challenge in applying NetGSA to study the biological systems in eukaryotes, in particular mammalians, results from the complexity of the genome and the large number of genes and proteins that need to be simultaneously analyzed. This results in extensive computational time[1]. In addition, the number of parameters (and hence the complexity of the models incorporated) is limited by the available sample size. In this chapter, I propose a dimension reduction technique for networks, based on Laplacian eigenmaps, with the goal of providing an optimal low-dimensional projection for the space of random variables in each subnetwork.

We start by reviewing the Laplacian eigenmaps in Section 6.2, and establish their connection to principal component analysis (PCA) for random variables on the graph. Inference for significance of subnetworks (pathways) is discussed in Section 6.3, where we introduce Laplacian eigenmaps with boundary conditions and use the group-lasso penalty to formulate the inference problem as a *penalized principal regression problem on the graph*. Results of applying the new methodology to simulated, as well as real, data examples are presented in Section 6.4, and a summary and directions for future research are given in Section 6.5.

[1] In Chapter III, we proposed an iterative algorithm for estimation of mixed linear model parameters. In the next chapter, we will further extend this algorithm in order to develop a *distributed* estimation framework.

6.2 Laplacian Eigenmaps

6.2.1 Preliminaries

Consider p genes (proteins or metabolites), whose expression data is given in the form of an $n \times p$ matrix X, and let y denote the $n \times 1$ vector of treatment conditions. For simplicity, here we consider the case of a two-class inference problem, which implies that y only includes two classes (e.g. treatment and control). Extensions to more general experimental settings are discussed in Section 6.5. Throughout this chapter we assume that interactions among genes are given in the form of an undirected (weighted) graph $\mathcal{G} = (V, E)$, with V the set of nodes in the graph and $E \subseteq V \times V$ its edge set. The edge set and the strength of associations among genes is captured through the adjacency matrix of the graph A. Specifically, there is an edge between genes i and j if the A_{ij} (and hence A_{ji}) is non-zero. In addition, we assume that $A_{ij} \geq 0$. The gene sets of interest are defined based on common biological function, co-regulation or chromosomal location, and the goal of this chapter is to develop dimension reduction methods on networks, in order to assess the significance of *a priori* defined gene sets (subnetworks) with minimal information loss.

6.2.2 Graph Laplacian and Eigenmaps

Laplacian eigenmaps are defined using the eigenfunctions of the graph Laplacian, which is commonly used in spectral graph theory, computer science and image processing. Applications based on Laplacian eigenmaps include image segmentation and the normalized cut algorithm of Shi and Malik (2000), spectral clustering (Saerens et al., 2004) and collaborative filtering (Fouss et al., 2004).

The Laplacian matrix and its eigenvectors have also been used in biological applications. Li and Li (2008, 2010) used the Laplacian matrix in order to define a network-penalty for variable selection on graphs, and Rapaport et al. (2007) used the interpretation of Laplacian eigenmaps as a Fourier basis to propose supervised and unsupervised classification methods.

Different definitions and representations have been proposed for the spectrum of graph, and the results may vary depending on the definition of the Laplacian matrix (see Chung, 1997, for a review). Here, we follow the notation of Chung (1997), and consider the *normalized* Laplacian matrix of the graph. To that end, let D denote the diagonal degree matrix for A, i.e. $D_{ii} = \sum_j A_{ij} \equiv d_i$, and define the Laplacian matrix of the graph by

$$\mathcal{L} = D^{-1/2}(D - A)D^{-1/2}.$$

The Laplacian matrix can be alternatively defined as

$$\mathcal{L}_{ij} = \begin{cases} 1 - \frac{A_{jj}}{d_j} & j = i, d_j \neq 0 \\ -\frac{A_{ij}}{\sqrt{d_i d_j}} & j \sim i \\ 0 & o.w. \end{cases}$$

Chung (1997) shows that $\mathcal{L}$ is positive semidefinite (PSD) with eigenvalues $0 = \lambda_0 \leq \lambda_1 \leq \ldots \leq \lambda_{p-1} \leq 2$. The eigenvalues of $\mathcal{L}$ are known as the spectrum of $\mathcal{G}$, and optimize the Rayleigh quotient for $\mathcal{L}$

$$\frac{\langle g, \mathcal{L}g \rangle}{\langle g, g \rangle} = \frac{\sum_{i \sim j} (f(i) - f(j))^2}{\sum_j f(j)^2 d_j}, \tag{6.1}$$

It can be seen from (6.1), that the 0-eigenvalue of $\mathcal{L}$ is $g = D^{1/2}\mathbf{1}$, corresponding to the average over the graph $\mathcal{G}$. The first non-zero eigenvalue of $\mathcal{L}$, λ_1 is the harmonic eigenfunction of $\mathcal{L}$ and is given by

$$\lambda_1 = \inf_{f \perp D\mathbf{1}} \frac{\sum_{j \sim i} (f(i) - f(j))^2}{\sum_j f(j)^2 d_j}$$

The eigenfunction corresponding to λ_1 corresponds to the Laplace-Beltrami operator on Reimannian manifolds. More generally,

$$\lambda_k = \inf_{f \perp DC_{k-1}} \frac{\sum_{j \sim i} (f(i) - f(j))^2}{\sum_j f(j)^2 d_j}$$

where C_{k-1} is the projection to the subspace corresponding to the first $k-1$ eigenvalues.

6.2.3 Principal Component Analysis for Graphs

Previous applications of the graph Laplacian and its spectrum often focus on the properties of the graph; however, the connection to the probability distribution of the random variables on nodes of the graph have not been previously discussed. In graphical models the undirected graph $\mathcal{G}$ among random variables corresponds naturally to a Markov random field. In particular, when the expression data X is Gaussian, the following result establishes a relationship between the Laplacian eigenmaps and the principal components of the random variables defined on the nodes of the graph.

Lemma VI.1. *Assume that the random variables on the nodes of the graph are jointly normally distributed. Then, the eigenfunctions of $\mathcal{L}$ correspond to the principal components of X, with the eigenfunction corresponding to the smallest non-zero eigenvalue representing the leading principal component.*

Proof. First note that the inverse covariance (or precision) matrix has the same non-zero pattern as the adjacency matrix of the graph, with non-zero diagonal entries equal to the inverse partial variance of each variable (see e.g. Rue and Held, 2005). Then, using the conditional autoregression (CAR) model of Besag (1974, 1975), the precision matrix is obtained by appropriately normalizing the adjacency matrix to be positive definite (PD). The most common choice for normalization is to ensure that the precision matrix is diagonally dominant (see Rue and Held, 2005, for additional details). However, as mentioned in the previous section, the Laplacian matrix is PSD, which implies that for any $\zeta > 0$, $\tilde{\mathcal{L}} = \mathcal{L} + \zeta I_p$ is a positive definite matrix with the same non-zero pattern as the adjacency matrix of the graph. This in turn implies that, $\tilde{\mathcal{L}}$ and $\tilde{\mathcal{L}}^{-1}$ correspond to the precision and covariance matrix of the random variables $X_1, \ldots, X_p$ defined on the nodes of the graph. Let $\mathcal{L}^+$ denote the Moore-Penrose generalized inverse of $\mathcal{L}$. Then, taking limit as $\zeta \to 0$, it can be seen that $\mathcal{L}$ and $\mathcal{L}^+$ correspond to the precision and covariance matrix of X, respectively. □

An alternative justification for the above result, for general probability distributions defined on graphs, can be given based on the problem of finding the optimal embedding of graph $\mathcal{G}$ in a lower dimensional Euclidean space. For unweighted graphs, this justification was given by Belkin and Niyogi (2003), using the unnormlized version of the Laplacian matrix. We first consider the case of one dimensional embedding. In this case, the goal is to find an embedding $v = (v_1, \ldots, v_p)^\intercal$ that preserves the distances among the nodes of the graph. Although no explicit assumption about the joint probability distribution of the random variables is made, to justify the relationship in Lemma VI.1, one needs to assume that the graph captures the "similarity" among random variables, with nodes adjacent in the graph being similar and the distance of the shortest path between two non-adjacent nodes defining their similarity. The objective function of the embedding problem is then given by

$$Q = \sum_{i,j} (v_i - v_j)^2 A_{ij}.$$

However, Belkin and Niyogi (2003) show that $Q = 2v^\intercal(D - A)v$, and hence, the optimal embedding is found by solving the following optimization problem

$$\operatorname*{argmin}_{v^\intercal D v = 1} v^\intercal (D - A) v \tag{6.2}$$

Setting $u = D^{1/2}v$, (6.2) is equivalent to solving

$$\underset{\|u\|=1}{\operatorname{argmin}}\, u^{\intercal}\mathcal{L}u$$

whose solution corresponds to finding the eigenvector corresponding to the smallest eigenvalue of $\mathcal{L}$. The more general problem of embedding into an m-dimensional Euclidean space is solved similarly by finding the eigenvectors corresponding to the m smallest eigenvalues of $\mathcal{L}$.

Lemma VI.1 establishes the connection between the eigenvalues of the Laplacian matrix and the principal components of the covariance matrix of random variables defined on the graph. Based on this result, we can develop dimension reduction algorithms that summarize the information in the entire biological network into few feature vectors. Although the resulting dimension reduction method can be used efficiently in classification (as in Rapaport et al., 2007), the eigenvalues of $\mathcal{G}$ do not provide any information about significance of arbitrary subnetworks, and therefore cannot be used to analyze the changes in individual pathways. In the next section, we introduce a restricted version of Laplacian eigenmaps, and discuss the problem of analysis of biological pathways.

6.3 Analysis of Pathways and PCR on Graph (GPCR)

6.3.1 Pathway Analysis

In Chapter II we argued that in order to analyze the effect of each pathway, the test statistic needs to represent the “pure” effect of the pathway, without being influenced by effects from external nodes. Using the framework of mixed linear models, this was achieved through the network contrast vector defined in Lemma II.5. However, in order to achieve dimension reduction, we need a method for analysis of pathways that only incorporates *local* information at the level of each pathway, and possibly its neighbors. This is illustrated in the simple example of Figure 6.1.

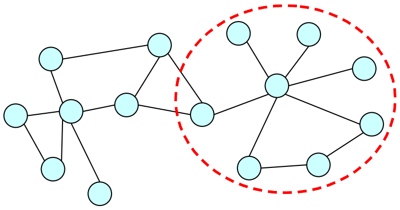


Figure 6.1: Illustration of the pathway effect. Pathway of interest is marked with the dotted circle.

Using the connection of the Laplace operator in Reimannian manifolds to heat flow (see e.g. Belkin and Niyogi, 2003), the problem of analysis of arbitrary subnetworks can be translated as a heat equation with boundary conditions. It then follows that in order to assess the "effect" of each subnetwork, the appropriate boundary conditions should block the flow of heat at the boundary of the set. This corresponds to insulating the boundary, also known as the Neumann boundary condition. For the general heat equation $\tau(v, x)$, this boundary condition is given by

$$\frac{\partial \tau}{\partial v}(x) = 0$$

at each boundary point x, where v is the normal direction orthogonal to the tangent hyperplane at x.

The problem of eigenvalues of subgraphs with boundary conditions is studied in Chung (1997). In particular, let S be any (connected) subnetwork of $\mathcal{G}$, and denote by δS the boundary of S in $\mathcal{G}$. The Neumann boundary condition states that for every $x \in \delta S$,

$$\sum_{y:\{x,y\} \in \delta S} (f(x) - f(y)) = 0 \tag{6.3}$$

The Neumann eigenfunctions of S are optimizers of the restricted Rayleigh quotient, given by

$$\lambda_{S,i} = \inf_{f} \sup_{g \in C_{i-1}} \frac{\sum_{\{t,u\} \in S \cup \delta S} (f(t) - f(u))^2}{\sum_{t \in S} (f(t) - g(t))^2 \, d_t}$$

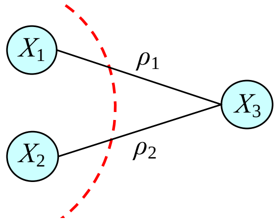


Figure 6.2: Illustration of the Neumann random walk. The dotted curve indicates the boundary of the subnetwork.

where C_{i-1} is the projection to the space of the first $i-1$ eigenfunctions.

Chung et al. (1996) establish a connection between the Neumann boundary conditions and a reflected random walk on the graph, and show that the Neumann eigenvectors can be alternatively calculated from the eigenvectors of the transition probability matrix of this reflected random walk, also known as the Neumann random walk (see Chung, 1997, for additional details).

Let $\tilde{P}$ and P denote the transition probability matrix of the reflected random walk, and the original random walk defined on $\mathcal{G}$, respectively. Noting that $P = D^{-1}A$, we can extend the results of Chung (1997) as follows. For the general case of weighted graphs, define the transition probability matrix of the reflected random walk by

$$\tilde{P}_{ij} = \begin{cases} P_{ij} & j \sim i, i, j \in S \\ P_{ij} + \frac{A_{ik}A_{kj}}{d_i d'_k} & j \sim k \sim i, k \notin S \\ 0 & o.w. \end{cases} \tag{6.4}$$

where

$$d'_k = \sum_{i \sim k, i \in S} A_{ki}$$

denotes the degree of the node k in S. Then, the Neumann eigenvalues are given by $\lambda_i = 1 - \kappa_i$, where κ_i is the ith eigenvalue of $\tilde{P}$.

The connection with the Neumann random walk also sheds light into the effect of the proposed boundary condition on the joint probability distribution of the random variables on the graph. To illustrate this, consider the simple graph of Figure 6.2. For the moment, suppose that the random variables X_1, X_2, X_3 are Gaussian, and the edges from X_1 and X_2 to X_3 are directed. As discussed in Chapter II, the joint probability distribution of the random variables on the graph is then given by linear structural equation models:

$$\begin{aligned} X_1 &= \gamma_1 \\ X_2 &= \gamma_2 \qquad\qquad \Rightarrow X = \Lambda\gamma, \quad \Lambda = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \rho_1 & \rho_2 & 1 \end{pmatrix} \\ X_3 &= \rho_1 X_1 + \rho_1 X_2 \end{aligned}$$

Then the conditional probability distribution of X_1 and X_2 given X_3, is then Gaussian, with the inverse covariance matrix given by

$$\begin{pmatrix} 1+\rho_1^2 & \rho_1\rho_2 \\ \rho_1\rho_2 & 1+\rho_2^2 \end{pmatrix} \tag{6.5}$$

A comparison between (6.4) and (6.5) reveals that the proposed Neumann random walk corresponds to conditioning on the boundary variables, if the edges going from the set S to its boundary are directed. The reflected random walk, for the original problem, therefore corresponds to first setting all the influences from other nodes in the graph to nodes in the set S to zero (resulting in directed edges) and then conditioning on the boundary variables.

6.3.2 Group-Penalized PCR on Graph

Using the Neumann eigenvectors of subnetworks, we now define a principal component regression on graphs, which can be used to analyze the significance of pathways. Let $\mathcal{N}_j$ denote the $|S_j| \times m_j$ matrix of the m_j smallest Neumann eigenfunctions for

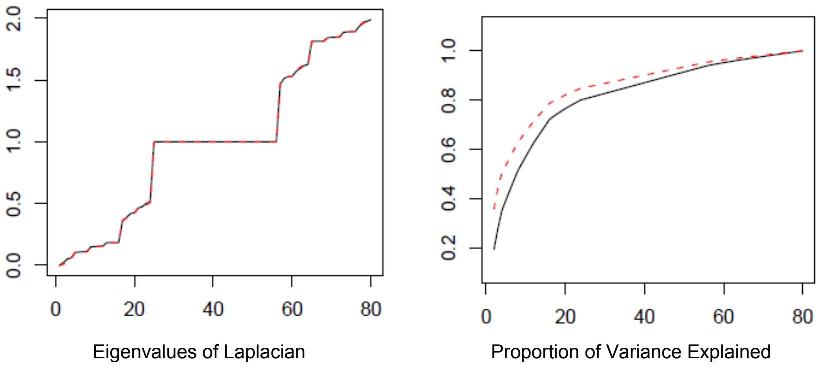


Figure 6.3: Laplacian eigenvalues. Left: eigenvalues of two simulated graphs. Right: proportion of variance explained by non-zero eigenfunctions.

subgraph S_j. Also, let X_j be the $n \times |S_j|$ matrix of observations for the j-th subnetwork. An m_j dimensional projection of the original data matrix X_j is then given by

$$\tilde{X}_j = X_j N_j.$$

Different methods can be used in order to determine the number of eigenfunctions m_j for each subnetwork. A simple procedure determines a predefined threshold for the proportion of variance explained by each eigenfunction. These proportions can be determined by considering the reciprocal of Neumann eigenvalues (ignoring the 0-eigenvalue). Figure 6.3 gives an example of eigenvalues for the two simulated graphs in Section 6.4.

The significance of subnetwork S_j is a function of the combined effect of all the nodes, captured by the transformed data matrix $\tilde{X}_j$. This combined effect can be evaluated by imposing a group penalty on the coefficient of the regression of the vector of experimental factors y on the transformed data matrices $\tilde{X}_j$. Using the group lasso penalty of Yuan and Lin (2006), we estimate the significance of the subnetwork by solving the following optimization problem

$$\underset{\beta}{\operatorname{argmin}} \|y - \sum_{j=1}^{J} \tilde{X}_j \beta_j\| + \gamma \sum_{j=1}^{J} \|\beta_j\|_2 \tag{6.6}$$

where J is the total number of subnetworks considered.

The problem in (6.6) is solved using the R-package `grplasso` (Meier et al., 2008), and the optimal value of the tuning parameter γ in (6.6) is determined by performing k-fold cross validation, with the goal of minimizing the Bayesian information criterion (BIC), with the degrees of freedom of the model estimated based on the number of subnetworks with nonzero effects.

As a final remark, we note the problem in (6.6) can be reformulated as a group-penalized regression problem in the original space. To see this, suppose that we include all of the eigenfunctions for each subnetwork S_j and assume that there are no overlaps amongst the subnetworks. Then, denoting the new vector of coefficients as $\beta_j = N_j\beta_j$, the problem in (6.6) can be written as

$$\underset{\beta}{\operatorname{argmin}} \|y - X\beta\| + \sum_{j=1}^{J} \|\beta_j\|_{\Theta_j} \tag{6.7}$$

where Θ_j is a diagonal matrix consisting of norms of Neumann eigenfunctions for S_j and $\|u\|_{\Lambda_j} = (\sum u_i^2 \theta_{j,i})^{1/2}$.

The formulation in (6.7) corresponds to a group-lasso penalized logistic regression problem on the graph, where the groups are defined by subnetworks and the penalty for each coefficient is given by the norm of the Neumann eigenfunction. Using this formulation, asymptotic properties of the resulting estimators follow directly from results established in the literature. An example of such properties includes the consistency of the estimators in terms of the logistic loss function established in Meier et al. (2008). In particular, suppose that β^0 is the minimizer of the true risk function, and $\hat{\beta}$ is the estimate based on the group lasso penalized empirical risk function, i.e.

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \left\{ n^{-1} \sum_{i=1}^{n} [-y_i \eta_\beta(X_i) + \log\{1 + \exp(\eta_\beta(X_i))\}] + \frac{\gamma}{n} \sum_{j=1}^{J} \|\beta_j\|_{\Theta_j} \right\}$$

where

$$\eta_\beta(x) = \beta_0 + \sum_{j=1}^{J} X_{i,j}{}^{\intercal}\beta_j$$

Following Meier et al. (2008), we make the following assumptions

(A1) For some constant $0 < \xi \leq 0.5$, $\xi \leq \mathbb{P}_{\beta^0}[Y = 1|x] \leq 1 - \xi$ for all x.

(A2) The matrix $\Sigma = \mathbb{E}[xx^{\intercal}]$ is non-singular

(A3) After normalizing the predictors so that $\mathbb{E}[x_j{}^{\intercal}x_j] = I_{|S_j|}$, we assume

$$\max_x \max_j \|x_j\|^2 \leq nL_n^2$$

The following result establishes the consistency of the group lasso estimate.

Theorem VI.2. *Let N_0 be the number of non-zero group effects. Then assuming (A1)-(A3), there exists constants C_1, C_2, C_3, C_4 and c_1, c_2 such that if the group sizes are fixed, but the total number of groups J is allowed to diverge, and if $C_1(1 + N_0^2)L_n^2 \log(J) \leq c_1$ and $C_1 \log(J) \leq \lambda \leq c_1(1 + N_0^2)L_n^2$ then*

$$\mathbb{P}\left\{\mathbb{E}[\eta_{\hat{\beta}} - \eta_{\beta^0}]^2 \geq c_2(1 + N_0\lambda)/n\right\} \leq C_2\left\{\log(n)\exp(-\lambda/C_3) + \exp(-(C_4L_n^2)^{-1})\right\}$$

Proof. The result follows directly from the material in Section 2.4 of Meier et al. (2008) by reformulating the problem as a group-lasso problem on a graph, with the original variables (i.e. columns of the original data matrix X). □

6.4 Data Analysis

6.4.1 Simulation Study

To evaluate the performance of the proposed method, we generate a small network of 80 genes, with 8 subnetwork. The expression levels of genes are generated according to a normal distribution with mean μ. Under the null hypothesis, all mean values are set to 1 and the association weight for all edges of the network equals 0.2

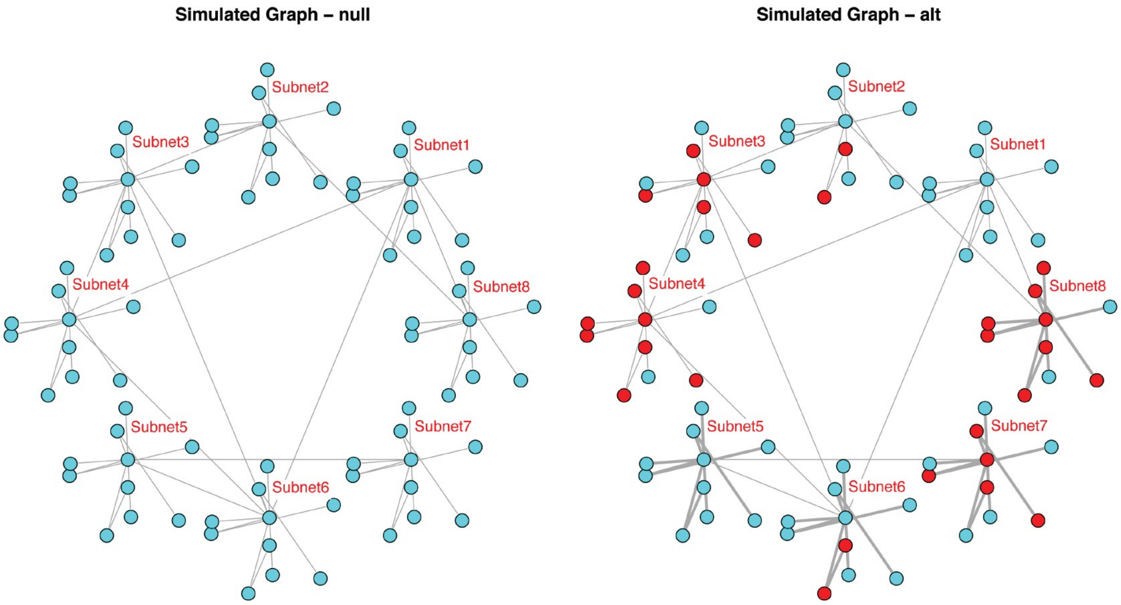


Figure 6.4: Simulated networks for analysis of graph PCR method. Left: setting of parameters under null, Right: setting of parameters under alternative.

(left panel of Figure 6.4). The setting of parameters under the alternative hypothesis are given in Table 6.1, with $\mu_{alt} = 3$.

Table 6.1: Setting of the parameters under the alternative hypothesis.

Subnetwork	% μ_{alt}	association weight (ρ)
1	0.05	0.2
2	0.20	0.2
3	0.50	0.2
4	0.80	0.2
5	0.05	0.6
6	0.20	0.6
7	0.50	0.6
8	0.80	0.6

The estimated powers of the tests for each of the subnetworks based on 200 simulation with $n = 50$ observations are given Figure in 6.5. It can be seen that while the proposed GPCR method offers improvements over GSEA, especially in case of subnetworks 3 and 6, it results in less accurate inference compared to NetGSA. This is expected as GPCR only incorporates the local information about the association among nodes in the graph, while NetGSA incorporates the global interaction information, and is expected to be more efficient. However, GPCR becomes an attractive

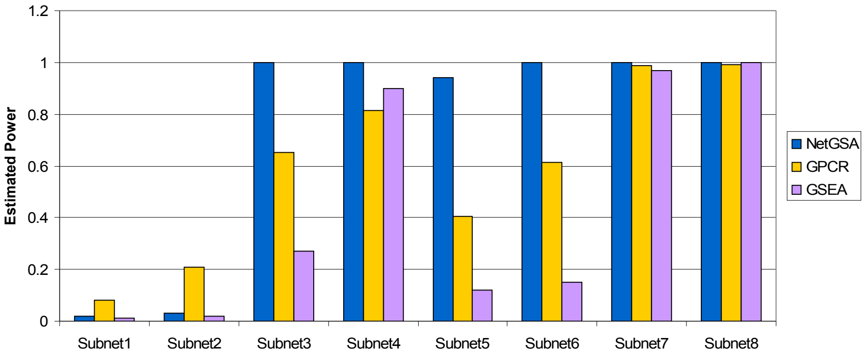


Figure 6.5: Results of simulation analysis for the graph PCR method.

alternative in high dimensional settings, and in particular in distributed analysis of pathway effects. In addition, when the sample size is small, GPCR can be used to reduce the dimension of the parameter space. This allows researchers to investigate more complex hypotheses by including additional parameters for the effect of each pathway.

6.4.2 Analysis of Yeast GAL Pathways

In Chapter II, we analyzed the pathways involved in Galactose utilization in yeast based on the data from Ideker et al. (2001), and compared the performance of the NetGSA and GSEA methods. The data includes a sample of 18 gene expression values under two experimental conditions, (gal+) and (gal–), and interactions among genes are also defined in Ideker et al. (2001). The interactions among genes, along with significance of individual genes (based on single gene analysis) are given in Figure 6.6, and the results of significance analysis based on NetGSA, GSEA and the proposed GPCR are given in Table 6.2. As in the simulated example, the results of this analysis indicate that GPCR results in improved efficiency over GSEA, while failing to detect the significance of some of the pathways detected by NetGSA.

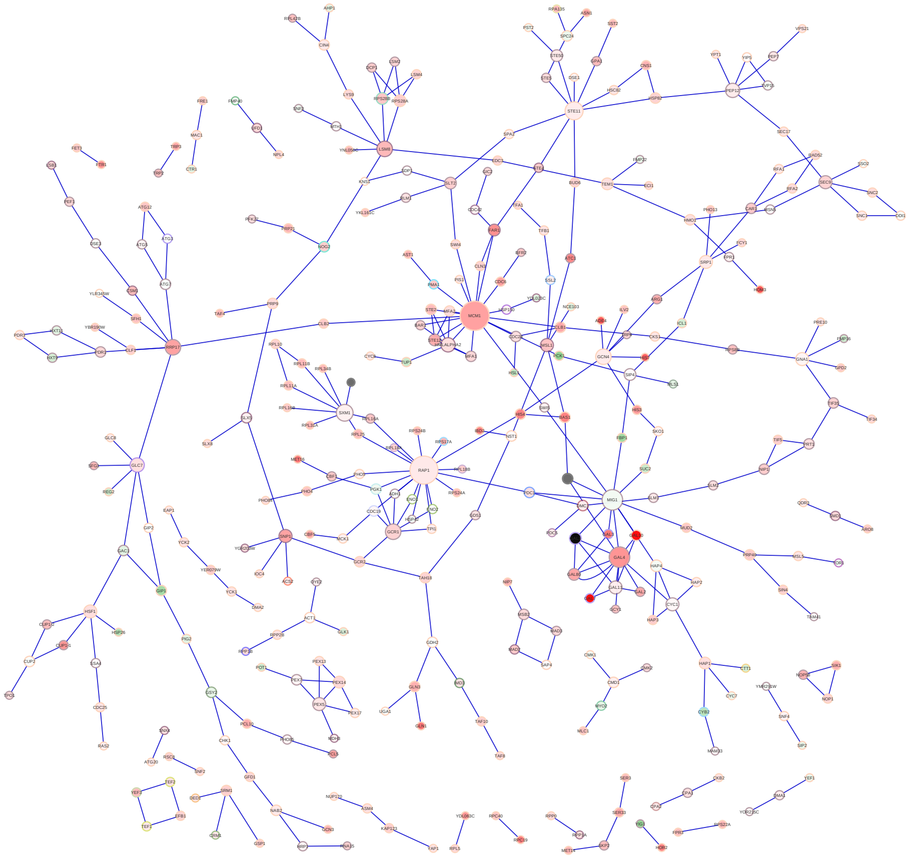


Figure 6.6: Network of yeast genes involved in Galactose utilization. The shades of red color indicate the significance of individual genes based on single gene analysis.

6.5 Conclusion

In this chapter, we proposed a principal component regression method for graphs, called GPCR, using Laplacian eigenmaps with Neumann boundary conditions. The proposed method offers a systematic approach for dimension reduction in networks, with *a priori* defined subnetworks of interest. It can also incorporate both weighted and unweighted adjacency matrices and can be easily extended for analysis of complex experimental conditions using the framework of generalized linear models (GLMs). This method can also be used to assess the effect of biological pathways in longitudinal and time-course studies.

Our simulation studies, and the real data example indicate that the proposed

Table 6.2: Significant pathways in Galactose utilization.

PATHWAY	Size	NetGSA	GPCR	GSEA
rProtein Synthesis	28	✓		
Glycolytic Enzymes	16			
RNA Processing	75			
Fatty Acid Oxidation	7	✓	✓	
O2 Stress	13			
Mating, Cell Cycle	58			
Vesicular Transport	19			
Sugar Transport	2			
Glycogen Metabolism	12			
Stress	12	✓	✓	
Metal Uptake	4			
Respiration	9	✓		
Gluconeogenesis	7			
Galactose Utilization	12	✓	✓	✓
Amino Acid Synthesis	30			

GPCR method offers significant improvements over the methods of gene set analysis. However, it does not achieve optimal powers in comparison to NetGSA. This difference in power may be attributable to the mechanism of incorporating the network information in the two methods: while NetGSA incorporates the full network information, GPCR only account for local network information, at the level of each subnetwork, and restricts the interactions with the rest of the network based on the Neumann boundary condition. However, the efficiency of GPCR depends on the performance of the group-lasso penalty, which is in turn governed by the choice of the tuning parameter. It is well known that the optimal tuning parameter for classification does not correspond to the optimal choice of tuning parameter for variable selection. Therefore, investigating other choices of tuning parameter, in particular determining the tuning parameter based on the probability of false positive should be a focus for future research.

Although our limited comparisons indicate that GPCR may not be as powerful in detecting the significance of pathways as the full modeling framework of NetGSA, GPCR could result in significant improvement in terms of computational time and memory requirements for analysis of high dimensional networks. In addition, the reduction in the dimension resulting from GPCR facilitates the analysis of complex experiments in the settings with small sample sizes.

Current analyses of asymptotic properties of group penalized estimators are often focused on consistency of the parameters, which is not of main interest in our context. Evaluating the performance of the proposed inference procedure requires establishing variable selection consistency of the estimator in high dimensional settings. Finally, it is crucial to investigate the effect of noise in the network information on the performance of the proposed method.

CHAPTER VII

Iterative Algorithms for Estimation and Inference in High Dimensional Networks

7.1 Introduction

Modern statistical applications often involve estimating a large number of parameters. Gene expression analysis, image analysis, and pattern recognition problems are examples of such applications. Statistical analysis in such applications often requires finding solutions to high dimensional optimization problems. This highlights the need for more efficient optimization algorithms that could take advantage of recent developments in computing technology, in particular multi-processor computes. In Chapter III an iterative algorithm based on block-relaxation was presented for the estimation of mean and variance parameters in mixed linear models. Although the convergence of this algorithm has been already established (Oberhofer and Kmenta, 1974), in this chapter we use the theory of block-relaxation (BR) algorithms (de Leeuw, 1994) to prove the convergence of this algorithm under less restrictive assumptions. In the remainder of this chapter, we start with a brief review of the theory of block-relaxation in section 7.2. In Section 7.3, I extend the algorithm of Chapter III, in order to propose a distributed version of the BR algorithm. Finally, in Section 7.4, we discuss two possible extensions of this framework that can be used to prove convergence of approximate iterative algorithms like quasi-likelihood methods for estimation of the parameters in generalized mixed linear models (GLMM).

7.2 Review of Block-Relaxation Algorithms

Following the notation in de Leeuw (1994), let f be a real-valued functional defined on the product set $\Omega = \Omega_1 \otimes \cdots \otimes \Omega_T$, with $\Omega_t \subseteq \mathbb{R}^{n_t}$. Thus, throughout this chapter we assume that the domain of f is a subset of $\mathbb{R}^n$, where $n = \sum_{t=1}^{T} n_t$, and its range is $\mathbb{R}$. The general form of block-relaxation algorithms is given in Algorithm 4. It has been pointed out in de Leeuw (1994) that many iterative optimization algorithms in statistics including the EM (Dempster et al., 1977) and the back-fitting or Gauss-Siedel algorithms (Buja et al., 1989), are special cases of this general framework.

Algorithm 4 Block-Relaxation Algorithm

```
1. Find an initial estimate $x^{(0)} \in \Omega$
2. Repeat until convergence $k = 1, 2, \cdots$
   2.1. $x_1^{(k+1)} = \text{argmin}_{x_1 \in \Omega_1} f(x_1, x_2^{(k)}, \cdots, x_T^{(k)})$
   2.2. $x_2^{(k+1)} = \text{argmin}_{x_2 \in \Omega_2} f(x_1^{(k+1)}, x_2, x_3^{(k)}, \cdots, x_T^{(k)})$
   ⋮
   2.T. $x_T^{(k+1)} = \text{argmin}_{x_T \in \Omega_T} f(x_1^{(k+1)}, x_2^{(k+1)}, \cdots, x_{T-1}^{(k+1)}, x_T)$
```

In general, it is possible to alternate through the subproblems in Algorithm 4 in different orders. It is also possible (with a higher degree of difficulty) to extend some of the results of this chapter accordingly, but for simplicity, we only consider the case of sequential updating over subspaces. Let $x^{(k)} \triangleq (x_1^{(k)}, \cdots, x_T^{(k)})$ and $f^{(k)} \triangleq f(x^{(k)})$. Also denote by Ω^0 the level set of f given by the initial estimate, i.e. $\Omega^0 = \{x \in \Omega | f(x) \leqslant f^{(0)}\}$. Let $\Delta_t, t = 1, \cdots, T$ be point to set maps from Ω to 2^Ω, and define

$$\Gamma_t^f(x) \triangleq \text{argmin}\{f(\bar{x}) | \bar{x} \in \Delta_t(x)\}$$

A number of results are available about the convergence of the sequence $\{f^{(k)}\}$ and accumulation points of the sequence $\{x^{(k)}\}$. In fact, if Ω_0 is compact and f is continuous on Ω, it is easy to show that the sequence $\{f^{(k)}\}$ converges, $\{x^{(k)}\}$ has a convergent subsequence, and the accumulation points of $\{x^{(k)}\}$ all have the same function value, f^{∞} (see Theorem 1 in de Leeuw, 1994). However, in order to characterize the sequence $\{x^{(k)}\}$ and determine whether this sequence converges to a unique minimizer of f, we need some additional conditions. The following theorem due to Fiorot and Huard (1979) states the conditions required for global convergence of the general block-relaxation algorithm.

Theorem VII.1. *(Fiorot and Huard, 1979) If*

- $x \in \Delta_t(x) \qquad \forall x, t$
- Δ_t *is continuous on* Ω *(i.e. both upper and lower semi-continuous)*
- f *has a unique minimum over* $\Delta_t(x)$ *for all* x *and* t
- Ω_0 *is compact*

then,

- *the sequence* $\{x^{(k)}\}$ *is asymptotically regular, i.e.* $\|x^{(k+1)} - x^{(k)}\| \to 0$
- *each accumulation point of the sequence is a fixed point of each of the* Γ_t^f

Algorithm 4 is based on a special class of block-relaxation methods, called *cyclic* methods, where the parameters are updated in a specific order, in other words $x^{(k+1)} \in \otimes_{t=1}^{T} \Gamma_t^f(x^{(k)})$. Also, for this algorithm we have:

(7.1)

$$\Delta_t(x_1, \cdots, x_{t-1}, x_{t+1}, \cdots, x_T) = \{x \in \mathbb{R}^{n_t} | (x_1, \cdots, x_{t-1}, x, x_{t+1}, \cdots, x_T) \in \Omega\}$$

Note that for this map,

$x_t \in \Delta_t(x_1, \cdots, x_{t-1}, x_{t+1}, \cdots, x_T)$ if and only if $(x_1, \cdots, x_{t-1}, x_t, x_{t+1}, \cdots, x_T) \in \Omega$.

Therefore, it is easy to see that the $\Delta_t(x)$ as given in (7.1) satisfies the first two conditions of Theorem VII.1.

7.3 Distributed Estimation of MLM parameters

As discussed in Chapter III, the maximum likelihood estimates of the fixed effect parameters in mixed linear models are given by:

$$\hat{\beta} = \left(\Psi'\hat{W}^{-1}\Psi\right)^{-1}\Psi'\hat{W}^{-1}\mathbf{Y}$$

where $W = (\sigma_\gamma^2 \Pi\Pi' + \mathbf{R}(\theta_\varepsilon))$.

It can be seen that $\hat{\beta}$ depends on estimates of σ_γ^2 and θ_ε^2, which are often estimated using restricted maximum likelihood (REML). Although this estimation problem is trivial for problems with relatively small number of parameters, application of mixed linear models in high dimensional networks requires estimating a large number of parameters and is often computationally intractable. The difficulty arises from the following challenges: first, the computational complexity of the algorithm is in general $O((Kp)^3)$, where K is the number of experimental conditions, and secondly, the memory required for storing the design matrices Ψ and Π in ultra-high dimensional problems surpasses available RAM on desktop computers. Thus, traditional methods for estimation of MLM parameters can only handle a few hundred parameters. To overcome these challenges, we present an iterative algorithm, based on an extension of block-relaxation which breaks down the problem into smaller estimation problems to achieve reasonable complexity and reduce the memory requirements. In addition, this algorithm provides a method for partitioning over spaces of parameters, as well as observations, by considering estimation over subnetworks and can be used as a basis for distributed estimation of mixed linear model (MLM) parameters in high dimensional networks.

The algorithm consists of three main steps:

(i) Partitioning the parameter space, to fixed effect parameters and variance components,

(ii) Partitioning the space of observations for distributed estimation of fixed effect parameters β, and

(iii) Augmenting the parameter space, by adding estimation of random effect parameters, in order to achieve separability in estimation of variance components.

Below we elaborate on each of these steps, and provide theoretical justification for convergence of the algorithm to maximum likelihood estimates.

7.3.1 Step 1: B-R Algorithm for estimation of MLM parameters

In Chapter III we presented a block-relaxation algorithm for estimation of MLM parameters that separates the estimation of fixed effect parameters from variance components. Here, we derive the conditions that are required to guarantee convergence. We first restate the algorithm.

Algorithm 5 Block-Relaxation Algorithm for Estimation of MLM Parameters

```
1. Find an initial estimate of $\hat{\beta}^{(0)}$ (e.g. using Ordinary Least Squares)
2. Repeat until convergence $k = 1, 2, \cdots$
    2.1. $\mathbf{e}^{(k+1)} = \mathbf{Y} - \mathbf{\Psi}\hat{\beta}^{(k)}$
    2.2. $\hat{\theta}^{(k+1)} = \text{argmax}_{\theta} -\frac{1}{2}\left(\log|\mathbf{W}(\theta)| + \mathbf{e}^{(k+1)\prime}\mathbf{W}^{-1}(\theta)\mathbf{e}^{(k+1)}\right)$
    2.3. $\hat{\beta}^{(k+1)} = \left(\mathbf{\Psi}'\hat{\mathbf{W}}^{-1}(\hat{\theta}^{(k)})\mathbf{\Psi}\right)^{-1}\mathbf{\Psi}'\hat{\mathbf{W}}^{-1}(\hat{\theta}^{(k)})\mathbf{Y}$
```

Lemma VII.2. *Suppose that the design matrix Ψ has full column rank and estimates of variance components θ in each step of Algorithm 5 result in a positive definite (PD) covariance matrix (i.e. $\exists c \geq 0$ such that $W(\hat{\theta}^{(k)}) \succeq cI$, where $A \succeq B$ implies that $A - B$ is PD.). Then the algorithm converges to the maximum likelihood estimate of the mean and variance parameters.*

Proof. We prove the result by verifying the assumptions of Theorem VII.1. First note that the map $\Delta_t(x)$ in the above algorithm is given by:

$$\Delta_\beta(\theta) = \{\beta | (\beta, \theta) \in \Omega\}$$

$$\Delta_\theta(\beta) = \{\theta | (\beta, \theta) \in \Omega\}$$

where $\Omega = \mathbb{R} \times \mathbb{R}^+$. This mapping is a special case of (7.1) and hence satisfies the first two conditions of Theorem VII.1. Also, note that the sequence of optimization problems in Algorithm 5 is equivalent to the maximum likelihood function for mixed linear models. We next find conditions required for the third assumption of Theorem VII.1. To show the uniqueness of the minimum in step 2.2 of the algorithm, let

$$f_{-\beta}^{(k+1)}(\theta) \triangleq -\log \left|\mathbf{W}^{-1}(\theta)\right| + \mathbf{e}^{(k+1)'}\mathbf{W}^{-1}(\theta)\mathbf{e}^{(k+1)}$$

Then $\mathbf{e}^{(k+1)'}\mathbf{W}^{-1}(\theta)\mathbf{e}^{(k+1)}$ is an affine function of $\mathbf{W}^{-1}$, and $-\log|\mathbf{W}^{-1}(\theta)|$ is a strongly convex function of $\mathbf{W}^{-1}$. Therefore, $f_{-\beta}^{(k+1)}$ is a strongly convex function of $\mathbf{W}^{-1}$ and as a result, $\hat{\mathbf{W}}^{(k+1)^{-1}}$ and hence $\hat{\mathbf{W}}^{(k+1)}$ are uniquely determined. Then, in order for θ to be the unique minimizer of $f_{-\beta}^{(k+1)}(\theta)$, $\mathbf{W}^{-1}(\theta)$ needs to be a bijection in θ; hence $\mathbf{G} = \mathbf{G}(\theta_\gamma)$ and $\mathbf{R} = \mathbf{R}(\theta_\varepsilon)$ should be bijections. Then observe that $\hat{\beta}^{(k+1)}$ in step 2.3 of the algorithm is the solution of the stationary condition for the following optimization problem:

$$\hat{\beta}^{(k+1)} = \underset{\beta}{\operatorname{argmin}} \left\{\beta'(\mathbf{\Psi}'\hat{\mathbf{W}}^{(k)^{-1}}\mathbf{\Psi})\beta - 2\beta'\mathbf{\Psi}'\hat{\mathbf{W}}^{(k)^{-1}}\mathbf{Y}\right\} \tag{7.2}$$

The objective function is a quadratic function of β which has a unique minimum if $\mathbf{\Psi}'\hat{\mathbf{W}}^{(k)^{-1}}\mathbf{\Psi}$ is positive definite (PD). However, if $\hat{\mathbf{W}}^{(k)}$ is PD, we can write $\mathbf{\Psi}'\hat{\mathbf{W}}^{(k)^{-1}}\mathbf{\Psi} = (L\mathbf{\Psi})'(L\mathbf{\Psi})$ where L is a upper-triangular matrix with positive diagonal elements. Thus, $\mathbf{\Psi}'\hat{\mathbf{W}}^{(k)^{-1}}\mathbf{\Psi}$ is PD if and only if $\mathbf{\Psi}$ is full rank and $\hat{\mathbf{W}}^{(k)}$ is positive definite. Therefore, the convergence of the algorithm requires the additional assumption that the estimates of θ in step 2.2 of the algorithm result in a PD estimate of $\mathbf{W}$, i.e.

there exists a $c > 0$ such that $\hat{\mathbf{W}}^{(k)} \succeq cI$[1]. In that case, $\mathbf{W}^{-1}(\theta)$ would also be a bijection.

Finally, to prove the last condition of the theorem, let $\tau = f(x)$. Then $\Omega^0 = \{x \in \Omega | f(x) \leqslant f^{(0)}\}$ is the inverse image of the set $\Xi^0 = \{\tau \in \mathbb{R} | \tau \leqslant \tau^{(0)}\}$. If $f(\cdot)$ assumes a minimum, τ^*, on Ω, then $\Xi^0 = [\tau^*, \tau^{(0)}]$ which is closed and bounded and hence compact. Continuity of the log-likelihood function implies Ω^0 is also compact and Theorem VII.1 is applicable. □

The positive definiteness of $\mathbf{W}$ in step 2.2 can be achieved by adding the constraint $\hat{\mathbf{W}}^{-1} \preceq CI$ for a large positive constant C. This results in a positive semi-definite programming problem in terms of $\hat{\mathbf{W}}^{-1}$. It is also possible to replace the objective function in 2.2 with the restricted maximum likelihood (REML) function, which often results in a positive definite estimate of $\mathbf{W}$. However, this is not further pursued here.

7.3.2 Step 2: Distributed Estimation of Fixed Effect Parameters β

It is easy to see that the estimation of fixed effect parameters β in Step 2.3 of Algorithm 5 can be transformed into a simple multiple regression problem, using the *weighted least squares* algorithm. In particular, if S is the Cholesky factor of $W^{(k)}$, β can be found by solving the following equation:

$$\hat{\beta}^{(k+1)} = \left(\tilde{\mathbf{\Psi}}'\tilde{\mathbf{\Psi}}\right)^{-1}\tilde{\mathbf{\Psi}}'\tilde{\mathbf{Y}}$$

where $\tilde{\mathbf{\Psi}} = S^{-1}\mathbf{\Psi}$ and $\tilde{\mathbf{Y}} = S^{-1}\mathbf{Y}$. It is important to note that, in case of directed acyclic graphs (DAGs), S is directly computed based on the influence matrix of the graph Λ.

[1] Oberhofer and Kmenta (1974) require the estimate of $\mathbf{W}$ to be also bounded above, i.e. $\hat{\mathbf{W}}^{(k)} \preceq CI$ for some $C > 0$.

The problem of estimation of fixed effect parameters in each step of the algorithm can therefore be solved using distributed methods for estimation of least squares parameters (see e.g. Lange, 2004).

7.3.3 Step 3: Augmented Estimation of Variance Components θ

To complete the distributed algorithm, and provide an efficient method for estimation of variance components, let $v = (Y', \beta', \gamma')'$. We then note that for mixed linear models

$$p(v \mid \theta) = p(Y \mid \beta, \gamma, \theta) p(\beta, \gamma \mid \theta)$$

On the other hand,

$$p(Y \mid \beta, \gamma, \theta) = p(Y \mid \beta, \gamma, \theta_\varepsilon) = p(\varepsilon \mid \theta_\varepsilon)$$

and

$$p(\beta, \gamma \mid \theta) = p(\gamma \mid \theta_\gamma)$$

Therefore, given the vector of random effects γ, we have

$$\ell(\theta; y, \beta, \gamma) = \ell(\theta_\varepsilon; \varepsilon) + \ell(\theta_\gamma; \gamma) + c \tag{7.3}$$

where ℓ represents the log-likelihood function. Equation (7.3) implies that given the vector of random effects, the problem of estimation of variance components is separable. This is particularly appealing as estimation of θ_γ and θ_ε from current estimates of γ and ϵ often involves simple optimization problems with closed-form solutions. To complete this step, we note that given the residuals at iteration k ($r^{(k)} = Y - \Psi\hat{\beta}^{(k)}$), the estimates of the random effects γ can be found by solving (see e.g. Demidenko, 2004):

$$\hat{\gamma}^{(k)} = \underset{\gamma}{\operatorname{argmin}} \sum_{i=1}^{n} \|r_i^{(k)} - \Lambda\gamma_i\|^2 + \hat{\sigma}_\gamma^{2,(k-1)} \gamma_i' \gamma_i \tag{7.4}$$

The penalized regression problem in equation (7.4) can be solved using distributed algorithms for regression, including the block-coordinate descent algorithm of Friedman et al. (2007). Given the estimates of the random effects, ε can be estimated using the relationship between observations, latent variables and noise, explained in Chapter II. Specifically,

$$\hat{e}^{(k)} = r^{(k)} - \Lambda\hat{\gamma}^{(k)}$$

Finally, the estimates of variance components are found by solving the following optimization problems:

$$\hat{\sigma}_{\gamma}^{2,(k)} = \operatorname*{argmin}_{\sigma^2} \ell(\sigma^2; \hat{\gamma}^{(k)}), \qquad \hat{\theta}_{\varepsilon}^{(k)} = \operatorname*{argmin}_{\theta_\varepsilon} \ell(\theta_\varepsilon; \hat{\varepsilon}^{(k)})$$

7.4 Approximate iterative algorithms

The algorithm in Section 7.3 provides an alternative method for optimizing complex objective functions and is especially useful for distributed computing. However, this algorithm requires the objective function to be readily available. When the calculation of the objective function is expensive or even impractical, it may be possible to estimate the function $f(x)$ by a new function $g(x)$ and obtain an estimate of the optimal points by optimizing g over Ω (obviously, the set of optimal points of f should be contained in the domain of g). An example of this method is estimation of the parameters of generalized mixed linear models using the quasi-likelihood approach. In this section, we discuss the use of iterative algorithms for optimization of the function g and study the conditions that are needed for convergence of solutions of the iterative algorithm to the optimal points of $f(x)$[2]. Let $f(x) = g(x) + r(x)$, where g is a smooth function that can be easily minimized, and r is the remainder function. The algorithm used in this section is the same as Algorithm 4 with $f(x)$

[2] The discussion here is mainly about the convergence of iterative algorithms and does not concern the asymptotic properties of the resulting estimators.

being replaced by $g(x)$. In other words, in order to find the optima of $f(x)$, the function $g(x)$ is minimized and the remainder function $r(x)$ is not included in the optimization. We first state our approximate iterative algorithm along with a special case of Theorem VII.1 which is focused on strongly convex functions.

Algorithm 6 Approximate Block-Relaxation Algorithm

```
1. Approximate the function $f(x)$ by $g(x)$ such that $f(x) = g(x) + r(x)$
2. Find an initial estimate $x^{(0)} \in \Omega$ and define $\Omega_g^0 = \{x \in \Omega | g(x) \leqslant g(x^{(0)})\}$
3. Repeat until convergence $k = 1, 2, \cdots$
    3.1. $x_1^{(k+1)} = \text{argmin}_{x_1 \in \Omega_1} g(x_1, x_2^{(k)}, \cdots, x_T^{(k)})$
    3.2. $x_2^{(k+1)} = \text{argmin}_{x_2 \in \Omega_2} g(x_1^{(k+1)}, x_2, x_3^{(k)}, \cdots, x_T^{(k)})$
    $\vdots$
    3.T. $x_T^{(k+1)} = \text{argmin}_{x_T \in \Omega_T} g(x_1^{(k+1)}, x_2^{(k+1)}, \cdots, x_{T-1}^{(k+1)}, x_T)$
```

Lemma VII.3. *Let $\Delta_t(x)$ be defined by (7.1). If*

- *f is strongly convex*
- *Ω^0 is closed*

then

- *the sequence $\{x^{(k)}\}$ is asymptotically regular*
- *each accumulation point of the sequence is a fixed point of each of the Γ_t^f*

Proof. Strong convexity of f implies uniqueness of the optimum points of each subproblem. Based on the previous argument about Δ_t, it suffices to prove the last condition of Theorem VII.1. Strong convexity implies that there exists an $m > 0$ such that

$$\nabla^2 f(x) \succeq mI, \qquad \forall x \in \Omega^0 \tag{7.5}$$

where $A \succeq B$ means that $A - B$ is positive semi-definite (PSD). But for $x, y \in \Omega^0$

we have

$$f(y) = f(x) + \nabla f(x)'(y-x) + \frac{1}{2}(y-x)'\nabla^2 f(z)(y-x) \tag{7.6}$$

for some z on the line segment $[x, y]$[3]. Then, (7.5) implies that

$$\frac{1}{2}(y-x)'\nabla^2 f(z)(y-x) \geq \frac{m}{2}\|y-x\|^2$$

and hence

$$f(y) \geq f(x) + \nabla f(x)'(y-x) + \frac{m}{2}\|y-x\|^2 \qquad x, y \in \Omega^0 \tag{7.7}$$

which implies that Ω^0 is bounded. Since $\Omega^0 \subseteq \mathbb{R}^n$, and is closed and bounded, the last condition is verified. □

Remark VII.4. The first assumption of Lemma VII.8 (closedness of Ω^0) is usually verified by checking that *all* sublevel sets of f are closed. This is true if the domain of f is $\mathbb{R}^n$ or if $f(x) \to \infty$ as x converges to the boundary of domain f.

Remark VII.5. Since the maximum eigenvalue of $\nabla^2 f(x)$ is a continuous function of x on Ω^0, boundedness of Ω^0 implies that $\nabla^2 f$ is bounded above on Ω^0 (Boyd and Vandenberghe, 2004), i.e. there exist a constant M such that

$$\nabla^2 f(x) \preceq MI, \qquad \forall x \in \Omega^0 \tag{7.8}$$

The following two theorems (stated in de Leeuw, 1994) provide additional convergence conditions, which are needed in the proof of our first convergence result. In this section, we consider general iterative algorithms. To that end, let $\mathcal{A}$ be the algorithmic map, i.e. $x^{(k+1)} = \mathcal{A}(x^{(k)})$ and consider the following definition of a general iterative *algorithm*

1. start at an arbitrary $x^{(0)} \in \Omega$

[3] The line segment $[x, y]$ is defined by all convex linear combinations of x and y.

2. if $x^{(k)}$ is a fixed point *stop*, otherwise, set $x^{(k+1)} = \mathcal{A}(x^{(k)})$ and continue.

Theorem VII.6. *(Meyer, 1976) If*

- *$\mathcal{A}$ is uniformly compact on Ω, i.e. there is a compact $\tilde{\Omega} \subseteq \Omega$ such that $\mathcal{A}(x) \subseteq \tilde{\Omega}$*
- *$\mathcal{A}$ is upper-semincontinuous or closed on Ω (with possible exception of the fixed points of $\mathcal{A}$). In other words, for any $x \in \Omega$, where x is not a fixed point of $\mathcal{A}$, if $y_i \in \mathcal{A}(x_i)$ and $y_i \to y$ and $x_i \to x$ then $y \in \mathcal{A}(x)$*
- *$\mathcal{A}$ is strictly monotonic on Ω, i.e. $y \in \mathcal{A}(x)$ implies $f(y) < f(x)$ if x is not a fixed point of $\mathcal{A}$.*

then all accumulation points of the sequence $\{x^{(k)}\}$ generated by the algorithm are fixed points of $\mathcal{A}$. Moreover, $\{x^{(k)}\}$ is asymptotically regular.

Theorem VII.7. *(Ostrowski, 1966) If*

- *the iterative algorithm converges to x^∞*
- *$\mathcal{A}$ is differentiable at x^∞*
- $0 < \rho = \|\mathcal{D}\mathcal{A}(x^\infty)\| < 1$

then the algorithm is linearly convergent with rate ρ.

For the algorithms considered in this section, the map $\mathcal{A}$ is defined by the solution of the stationary conditions for each of the subproblems. Therefore, in order to apply the above theorem we need to add the extra assumption that g is differentiable sufficiently many times.

To simplify the notation, we define the following variables for every two consecutive steps of the algorithm (k and $k+1$)

$$\begin{aligned}
x_t^{(+)} &= \left(x_1^{(k+1)}, \cdots, x_{t-1}^{(k+1)}, x_t^{(k+1)}, x_{t+1}^{(k)}, \cdots, x_T^{(k)}\right) \\
x_t^{(-)} &= \left(x_1^{(k+1)}, \cdots, x_{t-1}^{(k+1)}, x_t^{(k)}, x_{t+1}^{(k)}, \cdots, x_T^{(k)}\right) \\
x_t^{(\cdot)} &= \left(x_1^{(k+1)}, \cdots, x_{t-1}^{(k+1)}, x_t, x_{t+1}^{(k)}, \cdots, x_T^{(k)}\right)
\end{aligned}$$

Based on this notation,

$$x_t^{(+)} = \mathcal{A}(x_t^{(-)}) = \operatorname*{argmin}_{x_t \in \Omega_t} g\left(x_t^{(\cdot)}\right) \tag{7.9}$$

Lemma VII.8. *Consider the sequence $\{x^{(k)}\}$ defined in Algorithm 6 and let $\Delta_t(x)$ be defined by 7.1. Also, let ρ be the linear rate of convergence of the algorithm, given in Theorem VII.7. Denote by m and M the lower and upper bounds on the Hessian of g. If*

- *g is strongly convex*
- *Ω_0 is closed*
- *the relative change in r in every subspace Ω_t is less than $\frac{m^2}{8M}(1-\rho)\|x - x^*\|$ where x^* denotes the minimizer of g*

then

- *the sequence $\{x^{(k)}\}$ is asymptotically regular*
- *each accumulation point of the sequence is a fixed point of each of the Γ_t^f*

Proof. We need to show that the assumptions of Theorem VII.6 hold for Algorithm 6. The map $\mathcal{A}$ is defined by the solutions of the stationary conditions for each of the subproblems in Algorithm 6. Strong convexity of g implies that $\mathcal{A}$ is strictly monotone with respect to g, i.e. $g(x_t^{(+)}) < g(x_t^{(-)})$. Thus, for every x in Ω, $\mathcal{A}(x) \subseteq \Omega_g^0$.

On the other hand, strong convexity of g also implies that Ω_g^0 is bounded, which along with closedness of Ω_g^0 imply that $\mathcal{A}$ is uniformly compact with Ω_g^0 playing the role of $\tilde{\Omega}$. Upper-semicontinuouity of $\mathcal{A}$ follows from its definition as the solution to stationary conditions and the fact that this g is twice differentiable.

To show the third assumption of Theorem VII.6, we need conditions on g and r that make $\mathcal{A}$ strictly monotone with respect to f (i.e. $f(x_t^{(+)}) < f(x_t^{(-)})$). But since g is strongly convex, we have $g(x_t^{(+)}) < g(x_t^{(-)})$ unless x is a fixed point. Therefore, if the increase in r is less than the decrease in g in each step of the algorithm, f will be strictly monotone. In other words, it suffices to have

$$r(x_t^{(+)}) - r(x_t^{(-)}) < g(x_t^{(-)}) - g(x_t^{(+)}) \tag{7.10}$$

By strong convexity of g the assumptions of Theorem VII.7 are satisfied and therefore the algorithm converges linearly with rate ρ defined in Theorem VII.7, i.e.

$$g(x_t^{(+)}) - g(x^*) \leq \rho \left(g(x_t^{(-)}) - g(x^*) \right)$$

which implies

$$g(x_t^{(-)}) - g(x_t^{(+)}) \leq (1-\rho) \left[g(x_t^{(-)}) - g(x^*) \right] \tag{7.11}$$

However, (7.6) and (7.8) imply that

$$g(x_t^{(+)}) \leq g(x_t^{(-)}) + \nabla g(x_t^{(-)})'(x_t^{(+)} - x_t^{(-)}) + \frac{M}{2} \|x_t^{(+)} - x_t^{(-)}\|^2 \tag{7.12}$$

Minimizing both sides of 7.12 over $x_t^{(+)}$ gives

$$g(x_t^{(-)}) - g(x^*) \geq \frac{1}{2M} \|\nabla g(x)\|^2$$

From a similar argument using the lower bound on the Hessian of g it can be seen that (Boyd and Vandenberghe, 2004)

$$\|\nabla g(x)\| \geq \frac{m}{2} \|x - x^*\|$$

Therefore, to satisfy (7.10) it suffices to have

$$|r(x_t^{(+)}) - r(x_t^{(-)})| < \frac{m^2}{8M}(1-\rho)\|x - x^*\|^2$$

Since $\|x_t^{(-)} - x^*\| \geq \|x_t^{(-)} - x_t^{(+)}\|$, the required condition on r can be written as

$$\frac{|r(x_t^{(+)}) - r(x_t^{(-)})|}{\|x_t^{(+)} - x_t^{(-)}\|} < c\|x_t^{(-)} - x^*\| \tag{7.13}$$

and hence Theorem VII.6 is applicable. □

The result in Lemma VII.8 holds for general block-relaxation algorithms. However, the condition on r given in this result is not easily verified as it requires the knowledge of the minimizer of g. In order to improve this result, we consider here a special class of block-relaxation algorithms, namely *coordinate cyclic descent* algorithms, where every subproblem consists of optimization over a one-dimensional parameter space. In other words, in this version of algorithm, $\Omega_t \subseteq \mathbb{R}$ and $T = n$. We first state a result about the Lipschitz property of operators on lines.

Theorem VII.9. *(Ostrowski, 1973) Let $H(\omega)$ be an operator mapping the interval $[l, u]$ into a normed linear space and suppose there exist a constant $L \geq 0$ such that*

$$\limsup_{h\to 0} \frac{\|H(\omega + h) - H(\omega)\|}{|h|} \leq L$$

Then H is Lipschitz continuous with constant L.

Lemma VII.10. *Consider the sequence $\{x^{(k)}\}$ defined by the cyclic coordinate descent version of Algorithm 6 and suppose that g is strongly convex and Ω_0 is closed. If*

$$\frac{|r(x_i^{(+)}) - r(x_i^{(-)})|}{\|x_i^{(+)} - x_i^{(-)}\|} < \frac{m}{2M}|\nabla_i g(x_i^{(-)})| \tag{7.14}$$

then

- *the sequence* $\{x^{(k)}\}$ *is asymptotically regular*
- *each accumulation point of the sequence is a fixed point of each of the* Γ_t^f

Proof. The first two conditions of Theorem VII.6 are shown by the argument in Lemma VII.8. We show that (7.14) implies (7.10). To that end, observe that (7.7) implies that $\nabla_{ii}^2 g(x_i^{(-)}) \leq M$ and by and Theorem VII.9 we get

$$|\nabla_i g(x_i^{(+)}) - \nabla_i g(x_i^{(-)})| \leq M\|x_i^{(+)} - x_i^{(-)}\| \tag{7.15}$$

In each iteration of cyclic coordinate descent, the function is minimized with respect to the ith component and hence $\nabla_i g(x_i^{(+)}) = 0$. Therefore (7.14) and (7.15) imply that

$$|r(x_i^{(+)}) - r(x_i^{(-)})| < \frac{m}{2}\|x_i^{(+)} - x_i^{(-)}\|^2 \tag{7.16}$$

Applying (7.6) to $x_i^{(+)}$ and $x_i^{(-)}$ and using the fact that if y is obtained by updating the value of x along any *descent direction*, $(y-x)\nabla g(y) \leq 0$ we get

$$(m/2)\|x_i^{(-)} - x_i^{(+)}\|^2 \leq g(x_i^{(-)}) - g(x_i^{(+)})$$

which proves the lemma. □

7.5 Discussion

We discussed a special class of block-relaxation algorithms and provided conditions for the convergence of these algorithms. As an example, we considered the algorithm for estimating the parameters of mixed linear models used in Chapter III. We showed that the convergence of this algorithm can be easily verified if the estimated covariance matrix is positive definite (which is often guaranteed if REML equations are used for estimation of variance components). It is of special interest to study the convergence of the proposed algorithm for estimation of generalized mixed linear models (GLMM) parameters based on the quasi-likelihood methods. McCulloch (2003) provides a review of the algorithms used for estimation of GLMM parameters and points out that the quasi-likelihood method with a second-order Laplace approximation may not work well when the distribution of the data is far from normal. Researchers have therefore considered higher order Laplace approximations (Breslow and Lin, 1995; Lin and Breslow, 1996; Raudenbush et al., 2000). The theory provided here may provide a guide to the appropriate degree of approximation in such problems.

The application of the models discussed in Chapters II and III to human gene expression data sets requires algorithms that efficiently reduce the amount of memory required for computation. A potential extension of the algorithms considered in this section is estimating the parameters of the model over subgraphs. More specifically, the idea of cycling through the parameter space may be used to cycle through subgraphs which can result in considerable reduction in the required memory.

CHAPTER VIII

Concluding Remarks

In this work, I addressed several issues related to estimation and inference in high dimensional networks, with a specific focus on analysis of biological systems. The main goal of this dissertation was to develop a flexible and efficient inference procedure that incorporates the available network information, and can be used to test the significance of arbitrary subnetworks. Such a method can be used to assess which biological pathways respond to changes in environmental conditions, or are involved in disease initiation or progression.

The inference framework presented in the first part of this dissertation (Chapters II and III) directly incorporates the available network information and can be used to test the significance of arbitrary subnetworks. In addition, we showed that this framework is robust to presence of random noise in the network information, and hence is also able to incorporate estimated network information. The second part of the dissertation (Chapters IV and V) was therefore devoted to estimating directed graphs from observations using penalized likelihood methods. Finally, in the last part of the dissertation, I addressed additional issues regarding estimation and inference in high dimensional networks, including dimension reduction techniques for subnetworks, using Laplacian eigenmaps with Neumann boundary conditions (Chapters VI) and distributed estimation of mixed model parameters in networks (Chapter VII).

The inclusion of network information results in more efficient inference procedures compared to classical methods of analysis of biological components. This increase in efficiency can be specifically important in the study of complex diseases. In addition, complex diseases often result from changes in expression levels of many individual components, and although the effect of individual components may be negligible, the analysis of the combined effect of pathways using the methodology proposed in this research might reveal new clues about the underlying genetic factors. Finally, the proposed model also provides a general framework that can be used in problems that involve additional correlation structures, different distributional assumptions and/or interactions among genes and environments.

The modeling framework presented in this dissertation is a special case of a more general class of models, where the response is modeled as a function of the design variables X, and the underlying network structure G, i.e. $y = f(X) + g(G)$ (see Culp et al., 2009, for additional discussion). In this general model, the functions f and g represent the relationship between the response with mean effects and the correlation structure, respectively. Using this general framework, it is possible to extend the model presented here to other applications, including analysis of social networks and computer networks. A number of related problems may be of specific interest when general models are considered. In the applications considered in this work, we have focused on problems where the subnetworks of interest are defined *a priori*. An interesting extension (which may be of specific interest in *anomaly detection*) is to find the subnetworks that are perturbed, or in general different than the rest of the nodes.

Throughout this work, we considered networks presented by general weighted adjacency matrices. This is specifically important in the study of biological systems; gene regulatory networks include both inhibitors and suppressors which cause positive and negative associations among the nodes of the network. However, the nature of interactions among components of biological system often involve additional levels of complexity. For instance, genes, proteins and metabolites all play important roles in carrying out cell functions, and interactions among these components are different in nature. To handle these applications appropriately, it may be necessary to define new graph-theoretical concepts, and to consider new models with multiple classes of nodes and edges in the network.

Finally, it is important to note that the network view of systems may not fully represent the complex nature of interactions among components. However, more complex mechanistic methods (e.g. dynamical systems) require additional information and experiments in order to determine the nature of interactions, and may be computationally prohibitive in high dimensional settings. The increase in available information about mechanisms of interactions among components of the system, and improvements in computing power could justify the use of more sophisticated mathematical models for the analysis of complex biological systems.

Bibliography

A. Alexa, J. Rahnenfuhrer, and T. Lengauer. Improved scoring of functional groups from gene expression data by decorrelating GO graph structure. *Bioinformatics*, 22(13):1600–1607, 2006.

A. Arnold, Y. Liu, and N. Abe. Temporal causal modeling with graphical granger methods. In *Proceedings of the 13th ACM SIGKDD*, pages 66–75, 2007.

M. Ashburner, C. Ball, J. Blake, D. Botstein, H. Butler, J. Cherry, A. Davis, K. Dolinski, S. Dwight, J. Eppig, et al. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet*, 25(1):25–9, 2000.

J. S. Bader, A. Chaudhuri, J. M. Rothberg, and J. Chant. Gaining confidence in high-throughput protein interaction networks. *Nature Biotechnology*, 2004.

P. Baldi, S. Brunak, Y. Chauvin, C. Andersen, and H. Nielsen. Assessing the accuracy of prediction algorithms for classification: an overview. *Bioinformatics*, 16(5):412, 2000.

O. Banerjee, L. El Ghaoui, and d'Aspremont Alexandre. Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data. *Journal of Machine Learning Research*, 9:485–516, 2008.

W. T. Barry, A. B. Nobel, and F. A. Wright. Significance analysis of functional categories in gene expression studies: a structured permutation approach. *Bioinformatics*, 21(9):1943–1949, 2005. ISSN 1367-4803.

M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.

Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. Roy. Statist. Soc. Ser. B*, 57(1):289–300, 1995. ISSN 0035-9246.

J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36(2):192–236, 1974.

J. Besag. Statistical analysis of non-lattice data. *The statistician*, pages 179–195, 1975.

S. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge University Press, 2004.

N. Breslow and X. Lin. Bias correction in generalised linear mixed models with a single component of dispersion. *Biometrika*, 82(1):81–91, 1995.

A. Buja, T. Hastie, and R. Tibshirani. Linear smoothers and additive models. *Ann. Statist*, 17(2):453–555, 1989.

H. Causton, B. Ren, S. Koh, C. Harbison, E. Kanin, E. Jennings, T. Lee, H. True, E. Lander, and R. Young. Remodeling of Yeast Genome Expression in Response to Environmental Changes. *Molecular Biology Of The Cell*, 12(2):323–337, 2001.

S. Chaudhuri, M. Drton, and T. Richardson. Estimation of a covariance matrix with zeros. *Biometrika*, 94(1):199–216, 2007.

F. Chung. *Spectral graph theory.* American Mathematical Society, 1997.

F. Chung, R. Graham, and S. Yau. On sampling with Markov chains. *Random Structures and Algorithms*, 9(1-2):55–77, 1996.

M. Culp, G. Michailidis, and K. Johnson. On multi-view learning with additive models. *Annals Of Applied Statistics*, 3(1):292–318, 2009.

J. Dattorro. *Convex optimization & Euclidean distance geometry.* Meboo Publishing USA, 2005.

J. de Leeuw. Block-relaxation algorithms in statistics. In *Information System and Data Analysis*, pages 308–325. Springer-Verlag, 1994.

E. Demidenko. *Mixed models: theory and applications.* Wiley-IEEE, 2004.

A. Dempster, N. Laird, D. Rubin, et al. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1):1–38, 1977.

R. Diestel. *Graph Theory.* Springer-Verlag, 2006.

B. Efron and R. Tibshirani. On testing the significance of sets of genes. *Annals of Applied Statistics*, 1(1):107–129, 2007.

J. Fiorot and P. Huard. Composition and Union of General Algorithms of Optimization. *Mathematical Programming Study*, 10(1):69–85, 1979.

F. Fouss, A. Pirotte, J. Renders, and M. Saerens. A novel way of computing dissimilarities between nodes of a graph, with application to collaborative filtering and subspace projection of the graph nodes. In *European Conference on Machine Learning Proceedings, ECML.* Citeseer, 2004.

S. H. Friedberg, A. J. Insel, and L. E. Spence. *Linear Algebra.* Prentice Hall, 1996.

J. Friedman, T. Hastie, H. Höfling, and R. Tibshirani. Pathwise coordinate optimization. *Annals of Applied Statistics*, 1(2):302–332, 2007.

J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Technical Report, Department of Statistics Stanford University*, 2008a.

J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–41, 2008b.

A. Fujita, J. Sato, H. Garay-Malpartida, R. Yamaguchi, S. Miyano, M. Sogayar, and C. Ferreira. Modeling gene expression regulatory networks with the sparse vector autoregressive model. *BMC Systems Biology*, 1(1):39, 2007.

A. P. Gasch and M. Werner-Washburne. The genomics of yeast responses to environmental stress and starvation. *Funct. Integr. Genomics*, 2(4-5):181–192, 2002. ISSN 0090-5364.

A. P. Gasch, P. T. Spellman, C. M. Kao, O. Carmel-Harel, M. B. Eisen, G. Storz, D. Botstein, and P. O. Brown. Genomic expression programs in the response of yeast cells to environmental changes. *Mol Bio Cell*, 11, 2000. ISSN 4241-4257.

J. J. Goeman and P. Bühlmann. Analyzing gene expression data in terms of gene sets: methodological issues. *Bioinformatics*, 23(8):980–987, 2007.

C. Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, pages 424–438, 1969.

S. Haberman. Concavity and Estimation. *Annals of Statistics*, 17(4):1631–1661, 1989.

C. Harbison, D. Gordon, T. Lee, N. Rinaldi, K. Macisaac, T. Danford, N. Hannett, J. Tagne, D. Reynolds, J. Yoo, et al. Transcriptional regulatory code of a eukaryotic genome. *Nature*, 431(7004):99–104, 2004.

D. Heckerman, D. Geiger, and D. Chickering. Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20(3):197–243, 1995.

S. Hohmann and W. Mager. *Yeast Stress Responses*. Springer, 2003.

F. Hong and H. Li. Functional Hierarchical Models for Identifying Genes with Different Time-Course Expression Profiles. *Biometrics*, 62(2):534–544, 2006.

J. Huang, N. Liu, M. Pourahmadi, and L. Liu. Covariance matrix selection and estimation via penalised normal likelihood. *Biometrika*, 93(1):85–98, 2006.

J. Huang, S. Ma, and C. Zhang. Adaptive Lasso For Sparse High-Dimensional Regression Models. *Statistica Sinica*, 18:1603–1618, 2008.

T. Ideker, V. Thorsson, J. Ranish, R. Christmas, J. Buhler, J. Eng, R. Bumgarner, D. Goodlett, R. Aebersold, and L. Hood. Integrated Genomic and Proteomic Analyses of a Systematically Perturbed Metabolic Network, 2001.

T. Ideker, O. Ozier, B. Schwikowski, and A. Siegel. Discovering regulatory and signalling circuits in molecular interaction networks. *Bioinformatics*, 18(1):S233–S240, 2002.

Z. Jiang and R. Gentleman. Extensions to gene set enrichment. *Bioinformatics*, 23 (3):306–313, 2007.

M. Kalisch and P. Bühlmann. Estimating high-dimensional directed acyclic graphs with the PC-algorithm. *The Journal of Machine Learning Research*, 8:613–636, 2007.

M. Kanehisa and S. Goto. KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Research*, 28(1):27–30, 2000.

K. Kao, Y. Yang, R. Boscolo, C. Sabatti, V. Roychowdhury, and J. Liao. Transcriptome-based determination of multiple transcription regulator activities in Escherichia coli by using network component analysis. *Proceedings of the National Academy of Sciences*, 101(2):641–646, 2004.

M. Kerr and A. Churchill. Statistical design and the analysis of gene expression microarray data. *Genetics Research*, 77(02):123–128, 2001a.

M. Kerr and G. Churchill. Experimental design for gene expression microarrays. *Biostatistics*, 2(2):183–201, 2001b.

K. Knight and W. Fu. Asymptotics for lasso-type estimators. *Annals Of Statistics*, 28(5):1356–1378, 2000.

C. Lam and J. Fan. Sparsity and rate of convergence in large covariance matrix estimation. *Annals Of Statistics*, 37(6B):4254–4278, 2009.

K. Lange. *Optimization*. Springer, 2004.

S. Lauritzen. *Graphical models*. Oxford Univ Press, 1996.

R. Leclerc. Survival of the sparsest: robust gene networks are parsimonious. *Molecular Systems Biology*, 4(213):213:1–6, 2008. doi: 10.1038/msb.2008.52.

I. Lee, Z. Li, and E. Marcotte. An Improved, Bias-Reduced Probabilistic Functional Gene Network of Baker's Yeast, Saccharomyces cerevisiae. *PLoS ONE*, 2(10), 2007.

E. Levina, A. Rothman, and J. Zhu. Sparse estimation of large covariance matrices via a nested Lasso penalty. *Annals of Applied Statistics*, 2(1):245–263, 2008.

C. Li and H. Li. Network-constrained regularization and variable selection for analysis of genomic data. *Bioinformatics*, 24(9):1175, 2008.

C. Li and H. Li. Variable Selection and Regression Analysis for Graph-Structured Covariates with an Application to Genomics. *Annals of Applied Statistics,* in press, 2010.

K.-C. Li. Genome-wide coexpression dynamics: Theory and application. *Proc Natl Acad Sci*, 2002.

X. Lin and N. Breslow. Bias correction in generalized linear mixed models with multiple components of dispersion. *Journal of the American Statistical Association*, 91(435):1007–1016, 1996.

M. J. Lindstrom and D. M. Bates. Newton-Raphson and EM algorithms for linear mixed-effects models for repeated-measures data. *Journal of the American Statistical Association*, 83(404):1014–1022, 1988.

A. Lozano, N. Abe, Y. Liu, and S. Rosset. Grouped graphical Granger modeling for gene expression regulatory networks discovery. *Bioinformatics*, 25(12):i110, 2009.

F. Markowetz and R. Spang. Inferring cellular networks–a review. *BMC Bioinformatics*, 8(Suppl 6):S5, 2007. doi: 10.1186/1471-2105-8-S6-S5.

C. McCulloch. *Generalized Linear Mixed Models*. Ims, 2003.

R. A. McLean and W. L. Sanders. Approximating degrees of freedom for standard errors in mixed linear models. *Proceedings of the Statistical Computing Section, American Statistical Association*, 1988.

L. Meier, S. Van de Geer, and P. Buhlmann. The group lasso for logistic regression. *Journal of Royal Statistical Society. Series B Statistical Methodology*, 70(1):53, 2008.

N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the Lasso. *Annals Of Statistics*, 34(3):1436–1462, 2006.

R. Meyer. Sufficient Conditions for the Convergence of Monotonic Mathematical Programming Algorithms. *JCSS*, 12(1):108–121, 1976.

L. Mirsky. A trace inequality of John von Neumann. *Monatshefte für Mathematik*, 79(4):303–306, 1975.

N. Mukhopadhyay and S. Chatterjee. Causality and pathway search in microarray time series experiment. *Bioinformatics*, 23(4):442, 2007.

K. Murphy. *Dynamic Bayesian networks: representation, inference and learning.* PhD thesis, University Of California, 2002.

W. Oberhofer and J. Kmenta. A General Procedure for Obtaining Maximum Likelihood Estimates in Generalized Regression Models. *Econometrica*, 42(3):579–90, 1974.

I. Ong, J. Glasner, D. Page, et al. Modelling regulatory pathways in E. coli from time series expression profiles. *Bioinformatics*, 18(Suppl 1):S241–S248, 2002.

R. Opgen-Rhein and K. Strimmer. Learning causal networks from systems biology time course data: an effective model selection procedure for the vector autoregressive process. *BMC bioinformatics*, 8(Suppl 2):S3, 2007.

M. Osborne, B. Presnell, and B. Turlach. On the lasso and its dual. *Journal of Computational and Graphical Statistics*, pages 319–337, 2000.

A. Ostrowski. *Solution of Equations and Systems of Equations.* Academic Press, 1966.

A. Ostrowski. *Solution of Equations in Euclidean and Banach Spaces.* Academic Press New York, 1973.

J. Pearl. *Causality: Models, Reasoning, and Inference.* Cambridge Univ Press, 2000.

B. Perrin, L. Ralaivola, A. Mazurie, S. Bottani, J. Mallet, and F. d'Alche Buc. Gene networks inference using dynamic Bayesian networks. *Bioinformatics*, 19(90002): 138–148, 2003.

J. Rahnenführer, F. S. Domingues, J. Maydt, and T. Lengauer. Calculating the statistical significance of changes in pathway activity from gene expression data. *Statistical Applications in Genetics and Molecular Biology*, 2004.

F. Rapaport, A. Zinovyev, M. Dutreix, E. Barillot, and J. Vert. Classification of microarray data using gene networks. *BMC bioinformatics*, 8(1):35, 2007.

S. Raudenbush, M. Yang, and M. Yosef. Maximum Likelihood for Generalized Linear Models With Nested Random Effects via High-Order, Multivariate Laplace Approximation. *Journal Of Computational And Graphical Statistics*, 9(1):141–157, 2000.

R. Robinson. Counting unlabeled acyclic digraphs. In *Combinatorial mathematics V: proceedings of the Fifth Australian Conference, held at the Royal Melbourne Institute of Technology*, pages 28–43. Springer, 28–43, 1977.

A. Rothman, P. Bickel, E. Levina, and J. Zhu. Sparse permutation invariant covariance estimation. *Electronic Journal of Statistics*, 2:494–515, 2008.

H. Rue and L. Held. *Gaussian Markov random fields: theory and applications.* Chapman & Hall, 2005.

K. Sachs, O. Perez, D. Pe'er, D. Lauffenburger, and G. Nolan. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308 (5721):504–506, 2003.

M. Saerens, F. Fouss, L. Yen, and P. Dupont. The principal components analysis of a graph, and its relationships to spectral clustering. *Machine Learning: ECML 2004*, pages 371–383, 2004.

F. Sambo, B. Di Camillo, and G. Toffolo. CNET: an algorithm for reverse engineering of causal gene networks. In *NETTAB2008. Varenna, Italy*, 2008.

G. Sanguinetti, J. Noirel, and P. Wright. MMG: a probabilistic tool to identify submodules of metabolic pathways. *Bioinformatics*, 24(8):1078, 2008.

S. R. Searle. *Linear Models.* New York: John Wiley & Sons, Inc, 1971.

J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905, 2000.

P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search.* MIT Press, 2000.

A. Subramanian, P. Tamayo, V. Mootha, S. Mukherjee, B. Ebert, M. Gillette, A. Paulovich, S. Pomeroy, T. Golub, E. Lander, et al. Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences*, 102(43):15545–15550, 2005.

L. Tian, S. A. Greenberg, S. W. Kong, J. Altschuler, I. S. Kohane, and P. J. Park.

Discovering statistically significant pathways in expression profiling studies. *Proceedings of the National Academy of Sciences*, 102(38):13544–13549, 2005.

I. Tsamardinos, L. Brown, and C. Aliferis. The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning*, 65(1):31–78, 2006.

A. Van der Vaart and J. Wellner. *Weak Convergence and Empirical Processes*. Springer, 1996.

P. Wei and W. Pan. Incorporating gene networks into statistical tests for genomic data via a spatially correlated mixture model. *Bioinformatics*, 24(3):404–411, 2008. ISSN 1367-4803.

Z. Wei and H. Li. A markov random field model for network-based analysis of genomic data. *Bioinformatics*, 2007.

Z. Wei and H. Li. A hidden spatial-temporal Markov random field model for network-based analysis of time course gene expression data. *Annals of Applied Statistics*, 2 (1):408–429, 2008.

M. West. Technical report, Working Paper, 2000.

M. Whitfield, G. Sherlock, A. Saldanha, J. Murray, C. Ball, K. Alexander, J. Matese, C. Perou, M. Hurt, P. Brown, et al. Identification of genes periodically expressed in the human cell cycle and their expression in tumors. *Molecular Biology of the Cell*, 13(6):1977, 2002.

R. Yamaguchi, R. Yoshida, S. Imoto, T. Higuchi, and S. Miyano. Finding module-based gene networks with state-space models-Mining high-dimensional and short time-course gene expression data. *IEEE Signal Processing Magazine*, 24(1):37–46, 2007.

Y. Yang and T. Speed. Design issues for cDNA microarray experiments. *Nature reviews. Genetics(Print)*, 3(8):579–588, 2002.

T. Yoneya and H. Mamitsuka. A hidden Markov model-based approach for identifying timing differences in gene expression under different experimental factors. *Bioinformatics*, 23(7):842, 2007.

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

M. Yuan and Y. Lin. Model selection and estimation in the Gaussian graphical model. *Biometrika*, 94(1):19–36, 2007.

C. Zou and J. Feng. Granger causality vs. dynamic Bayesian network inference: a comparative study. *BMC bioinformatics*, 10(1):122, 2009.

H. Zou. The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.