ESTIMATION OF GRAPHICAL MODELS WITH BIOLOGICAL APPLICATIONS

Yuying Xie

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Statistics and Operations Research.

Chapel Hill
2015

Approved by:

Yufeng Liu

William Valdar

Shankar Bhamidi

J. S. Marron

Wei Sun

ABSTRACT

Yuying Xie: ESTIMATION OF GRAPHICAL MODELS WITH BIOLOGICAL APPLICATIONS
(Under the direction of Yufeng Liu and William Valdar)

Graphical models are widely used to represent the dependency relationship among random variables. In this dissertation, we have developed three statistical methodologies for estimating graphical models using high dimensional genomic data. In the first two, we estimate undirected Gaussian graphical models (GGMs) which capture the conditional dependence among variables, and in the third, we describe a novel method to estimate a Gaussian Directed Acyclic Graph (DAG).

In the first project, we focus on estimating GGMs from a group of dependent data. A motivating example is that of modeling gene expression collected on multiple tissues from the same individual. Existing methods that assume independence among graphs are not applicable in this setting. To estimate multiple dependent graphs, we decompose the problem into two graphical layers: the systemic layer, which is the network affecting all outcomes and therefore describing cross-graph dependency, and the category-specific layer, which represents the graph-specific variation. We propose a new graphical EM technique that estimates the two layers jointly; and also establish the estimation consistency and selection sparsistency of the proposed estimator. We confirm by simulation and real data analysis that our EM method is superior to a naive one-step method

Next, we consider estimating GGMs from noisy data. A notable drawback of existing methods for estimating GGMs is that they ignore the existence of measurement error which is common in biological data. We propose a new experimental design using technical replicates, and develop a new methodology using an EM algorithm to efficiently estimate the sparse GGM by taking account the measurement error. We systematically study the asymptotic properties of the proposed method in high dimensional settings. Simulation

study suggests that our method have substantially higher sensitivity and specificity to estimate the underlying graph than existing methods.

Lastly, we consider the estimation of the skeleton of a Directed Acyclic Graph (DAG) using observational data. We propose a novel method named AdaPC to efficiently estimate the skeleton of a DAG by a two-step approach. The performance of our method is systematically evaluated by numerical examples.

ACKNOWLEDGEMENTS

A great many people have helped me throughout my graduate research. I finally finished the journey to my second and also potentially last Ph.D degree. I owe my gratitude to all those who gave me direction, instruction, and courage. Thank you very much!

First, I would like to sincerely thank my co-mentor Dr. Yufeng Liu. I am extremely fortunate to have an advisor who gave me the freedom to explore, and at the same time, the guidance to recover when I lost confidence in my research. I still remember how excited and nervous I was when I first talked to Yufeng about the possibility of working with him for my second Ph.D in Statistics. He taught me, encouraged me, and led me into the area of statistical machine learning. Without his constant encouragement and support, I would not have accomplished as much. I value Yufeng's creativity in developing novel machine learning methodologies to analyze biological data with complicated structure, and admire his passion for both work and life. He has been a role model for me as a researcher and a teacher.

I also would like to thank my other mentor, Dr. William Valdar, for his support, advice, criticism, and encouragement for the past five years. It was Will who dragged me into a dry lab. He taught me not only about how to program efficiently, but also about how express ideas clearly. Most importantly, he taught me how to question thoughts and progress a topic beyond its current understanding. It has been a pleasure and great honor to be your student.

I am very grateful to my excellent committee members: Dr. Shankar Bhamidi is an excellent teacher. The knowledge I gained from his measure theory class provided a strong foundation for my future research in statistics. Dr. J. S. Marron is an expert in OODA, which is particularly instrumental for cancer research. I learned a great deal from his course on consulting: How to translate a client's problem into a statistical framework, and how to address biological problems using different statistical methods. Dr. Wei Sun is the only

professor who to sit on the committees of both my Ph.D.s. He greatly cares for his students, even letting me borrow his office for my phone interviews. He gave me great suggestions for my project and was always willing to shared his code.

Amazing lab members and classmates were another treasure I got during my study. We went through all the up and down together. I greatly value their friendship and I deeply appreciate their belief in me. I have to give a special mention for the support given by Dr. Allen Larnacic, Dr. Zhaojun Zhang, Dr. Jeremy Sabourin, Dr. Wonyul Lee, Dr. Qiang Sun, Dr. Chang Zhang, Dr. Guanhua Chen, Dr. Sunyoung Shin, Dr. Patrick Kimes, Guan Yu, Daniel Oreper, Greg Keele, Dr Jeff Roach, Paul Maurizio, and Robert Corty.

Most importantly, none of this would have been possible without the love and patience of my departed grandma Chunying Huang, my parents Shengbao Xie and Chanyang Feng, my wife Yangfan Phoebe Liu, and my son Dongchen Terence Xie. They have been my constant source of strength through this journey. Thank you, my loves.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction

With the advance of technology in recent years, we have witnessed a data explosion in many fields including biological science, social network and engineering. As a result, in order to better explore and understand the information behind large and maybe noisy data sets, there is a great need for the development of new statistical methodologies. A graphical model is a probabilistic model which uses a graph to denote the conditional dependence structure among random variables. Since graphs are conveniently used to provide insight and capture complex dependencies among random variables, graphical models have become an important focus of research in recent years. For example, in biological research, genes can be represented by the nodes of a graph, and the correlations between genes can be represented by the edges. In the area of finance, nodes of a graph can represent different stocks in Nasdaq and edges can represent the partial correlation between stocks.

There are two commonly used branches of graphical models. The first is Markov Random Fields, i.e. Markov networks. A Markov random field (MRF) is a model over an undirected graph representing conditional independence between variables (Pearl, 1995). The simplest continuous MRF is Gaussian graphical model (GGM), since the first two moments can fully specify the corresponding distribution. In Sections 1.2 and 1.3, some basic background and extension of GGM in the statistical machine learning literature are introduced. The second is Bayesian networks, also known as Directed Acyclic Graph (DAG) which are models over directed graphs and commonly used for causal inference (Pearl, 2009). In Sections 1.4 and 1.5, the background and recent advances in Baysian networks are briefly discussed. For more details on graphical model please refer to the comprehensive book by Pearl (2009).

1.1 Gaussian Graphical Models

Gaussian Graphical models (GGMs) are widely used to represent conditional dependencies among sets of normally distributed outcome variables that are observed together. For example, observed, and potentially dense, correlations between measurements of expression for multiple genes, stock market prices of different asset classes, or blood flow for multiple voxels in functional Magnetic Resonance Imaging (fMRI)-measured brain activity can often be more parsimoniously explained by an underlying graph corresponding to the partial correlation. The partial correlation matrix is normally assumed to be sparse, since most of the time two variables interact with each other through other variables, hence the partial correlation between these two variables given all other variables is zero. As methods for estimating these underlying graphs have matured, a number of elaborations to the basic GGM have been proposed; these include elaborations that seek either to model more closely the sampling distribution of the data, or to model prior expectations of the analyst about structural similarities among graphs representing data sets that are related.

Introduced more formally, the conditional dependence relationships among a set of p outcome variables, $y = (y_1, ..., y_p)$, can be represented by a graph $\mathcal{G} = (X, E)$ where each variable corresponds to a node in the set V and conditional dependencies are represented by the edges in the set E. If we further assume that the joint distribution of the outcome variables is multivariate Gaussian, $y \sim \mathcal{N}(0, \Sigma)$, then conditional dependencies are reflected in the non-zero entries of the precision matrix $\Omega = \Sigma^{-1}$. Specifically, variables i and j are conditionally independent given the other variables if and only if the (i, j)-th element of Ω is zero. Inferring the dependence structure of such a Gaussian graphical model is thus equivalent to estimating which elements of its precision matrix are non-zero.

When the underlying graph is sparse, as is often assumed, the ordinary maximum likelihood estimate (MLE) is dominated by shrinkage methods: The MLE of Ω typically results a graph that is fully connected, and so gives a result that is unhelpful for estimating graph topology. To impose sparsity, and thereby provide a more informative inference about network structure, a number of methods have been introduced that estimate the precision matrix under ℓ_1 regularization. For example, Meinshausen and Bühlmann (2006) proposed

to determine iteratively the edges of each node in $\mathcal{G}$ by regressing the corresponding variable y_j on the remaining variables y_{-j} under an ℓ_1 penalty, an approach which can be viewed as optimizing a pseudo-likelihood (Rocha et al., 2008; Ambroise et al., 2009; Peng et al., 2009). Several groups have proposed sparse penalized maximum likelihood to estimate GGMs(see for example Yuan and Lin, 2007; Banerjee et al., 2008; d'Aspremont et al., 2008; Rothman et al., 2008). Several efficient implementations solving this problem have also been published including the graphical-LASSO (GLASSO) algorithm (Friedman et al., 2008) and the QUadratic Inverse Covariance (QUIC) algorithm (Hsieh et al., 2011). The asymptotic properties of such penalized estimation schemes have also been described in theoretical studies (for example, Rothman et al., 2008; Lam and Fan, 2009).

1.2 Extensions of Gaussian Graphical Models

Although a single graph provides a useful representation for an underlying dependence structure, several extensions of GGMs have been proposed. In the context where the precision matrix, and hence the graph, is dynamic over time, Zhou et al. (2010) proposed a weighted method to estimate the graph's temporal evolution. Another practical extension is the simultaneous estimation of multiple graphs that may share some common structure. For example, when inferring how brain regions interact using fMRI data, each subject's brain corresponds to a different graph, but we would nonetheless, expect some interaction patterns to be common across subjects, as well as patterns specific to an individual. In such cases, joint estimation of multiple related graphs can be more efficient than estimating graphs separately. For joint estimation of Gaussian graphs, Varoquaux et al. (2010) and Honorio and Samaras (2010) proposed methods using group-LASSO (Yuan and Lin, 2006), and multitask-LASSO respectively. Both methods assume that all graphs share the same pattern, namely that the precision matrices have the same pattern of zeros. To provide greater flexibility, Guo et al. (2011) proposed a joint penalized method using a hierarchical penalty, and derived the convergence rate and sparsistency properties for the resulting estimators. Under the same setting, Danaher et al. (2014) extended the graphical

lasso (Friedman et al., 2008) to estimate multiple graphs from independent data sets using penalties based on the generalized fused lasso or, alternatively, the sparse group lasso.

However, in some applications, data from different categories are naturally dependent, hence the methods mentioned above are not valid. In Chapter 2, we develop a new graphical EM method to estimate the GGMs from dependent data sets.

1.3 Bayesian Networks and Directed Acyclic Graphs

Causality is an important topic in scientific research. Bayesian networks have become popular in recent years for their application in causal inference (Glymour, 1987; Koller and Friedman, 2009; Pearl, 1995, 2000, 2009). Though estimating causal effect requires experimental data, when the causal structure, the DAG of the Bayesian network, is given, the post-intervention distributions and causal effects can be estimated from observational data using various existing methods (Pearl, 2000).

A Bayesian network is a probabilistic graphical model that uses a directed acyclic graph (DAG) to represent the conditional dependencies of a set of random variables. More formally, a DAG is a mathematical object consisting of a pair (V, E), where V is the set of vertices indicating random variables and E contains all the directed edges representing direct causal relationship among the variables. A directed edge is an ordered pair of nodes: For example, the edge from node $X \rightarrow$ node Y can be represented as (X, Y). Implicit in the notation (X, Y) are several additional constraints on the relationship between X and Y: X is said to be a parent of Y, Y is a child of X, and the two node are adjacent with one another. A directed path in $\mathcal{G}$ is a sequence of distinct vertices with directed edge pointing from each vetex to its successor. X is called the an ancestor of Y, and Y a descendant of X, if there is a directed path from X to Y. There are no directed cycles in a DAG, namely, there are no two distinct vertices that are ancestors of each other. This requirement is a necessary condition for causal inference (Spirtes et al., 2000).

Given a Bayesian network including the DAG $\mathcal{G}(V, E)$ and corresponding probability distribution $\mathcal{P}$, it is well known that there are other DAGs which can describe exactly the same conditional independence information of $\mathcal{P}$ (Chickering and Boutilier, 2002). Hence

we could only identify an class of DAGs called the Markov equivalence class given the data. The DAGs in a Markov equivalence class share the same skeleton structure and v-structures (Pearl, 2009). Here, skeleton of a DAG is the undirected version of DAG, and a v-structure (X_i, X_j, X_k) is a triple structure in a DAG with the edges oriented as $(X_i \rightarrow X_j \leftarrow X_k)$. Using the shared skeleton structure and v-structures in a Markov equivalence class, we define a completed partially directed acyclic graph (CPDAG) which uniquely represents the corresponding Markov equivalence class (Andersson et al., 1997). A CPDAG has the following properties: 1) The skeleton of a CPDAG is the same for each DAG in the Markov equivalence class; 2) If an edge in such a CPDAG is directed, all the DAGs in the equivalence class have the same directed edge; 3) For every undirected edge $(X_i - X_j)$ in such a CPDAG, there exists at least a DAG with $X_i \rightarrow X_j$ and a DAG with $X_i \leftarrow X_j$.

1.4 Estimation of Directed Acyclic Graphs and Skeleton

Estimating DAG can be challenging, since the size of the space of DAGs is super-exponential in the number of nodes (Kalisch and Bühlmann, 2007). However, when the dimension is small or moderate, there are several quite successful methods using greedy or structurally restricted approaches (See for example, Chickering and Boutilier, 2002; Chow et al., 1968; Heckerman and Chickering, 1995; Spiegelhalter et al., 1993).

In order to handle high dimensional data, Spirtes et al. (2000) proposed a sequential method called Peter-Clark algorithm also known as PC-algorithm. Starting from a fully connected graph, the PC algorithm recursively removes edges based on conditional independence test to obtain an undirected graph, namely the skeleton of the DAG. The resulting skeleton can then be partially directed via additional tests to obtain further information of the corresponding DAG. After proposed, the PC-algorithm has gained a lot of attentions especially in high-dimensional settings among different areas(See for example, Kalisch et al., 2010; Stekhoven et al., 2012; Zhang et al., 2012),since it is computationally feasible for thousands of variables when the underlying graph is sparse and efficient. There is also an R package available (Kalisch et al., 2012). Moreover, the PC-algorithm estimation consistency has also been studied (Kalisch and Bühlmann, 2007). As a drawback of the PC-algorithm,

the results of PC-algorithm are order-dependent, in the sense that different initial nodes would lead to different outputs. To overcome this drawback, Colombo and Maathuis (2013) proposed a modified PC-algorithm called PC-stable algorithm, which is order-invariant.

Another drawback of the PC-algorithm is the large number of tests for high dimensional data, since it starts from a fully connected graph. To address this problem, one can start with the so called moral graph (also known as the independence graph) instead of the fully connected graph. A moral graph is a undirected graph generated from a DAG by connecting two parents of the same node corresponding to the v-strucure, and then removing the direction from all edges. Therefore, the moral graph of a DAG contains or equals to its skeleton. Namely, the skeleton could be obtained by removing extra edges in the corresponding moral graph. Based on this fact, Spirtes et al. (2000) proposed the the Independence Graph (IG) algorithm, which first estimates the independence graph, i.e. the moral graph, and then removes extra edges using conditional independence tests. Under the multivariate Gaussian assumption, the moral graph becomes the partial correlation graph which can be uniquely determined by Ω as described in Section 1.1. Under Gaussian assumption, Ha et al. (2014) proposed the PenPC algorithm which is similar to the IG algorithm. The concept behind PenPC is that they first estimate the precision matrix Ω via penalized regression, and then use a modified PC-stable algorithm to delete the extra edges due to v-structures. The advantage of the PenPC algorithm relies on the fact that it screens out most of the extra edges in the first step leaving much few conditional independence tests to be performed in the following step. Thus the PenPC algorithm enjoys better accuracy and faster computational speed.

A network is denoted as scale-free when its degree distribution (asymptotically) follows a power law. The biological networks and social networks are conjectured to be scale-free, and hence have many v-structures which leads to a great amount of extra edges in the first step of PenPC algorithm. To address this shortcoming, we proposed a new method, denoted as AdaPC, to estimate the intercept of precision and covariance matrices and then remove extra edges to recover the skeleton in Chapter 4. Since in scale-free networks, the covariance matrices are relative sparse, thus the resulting interception between precision and covariance matrices are much sparser than precision matrices alone.

CHAPTER 2

Joint Estimation of Multiple Dependent Gaussian Graphical Models with Applications to Mouse Genomics

2.1 Introduction

Gaussian Graphical models (GGMs) are widely used to represent conditional dependencies among sets of normally distributed outcome variables that are observed together. For example, observed, and potentially dense, correlations between measurements of expression for multiple genes, stock market prices of different asset classes, or blood flow for multiple voxels in functional Magnetic Resonance Imaging (fMRI)-measured brain activity can often be more parsimoniously explained by an underlying graph whose structure may be relatively sparse. As methods for estimating these underlying graphs have matured, a number of elaborations to the basic GGM have been proposed; these include elaborations that seek either to model more closely the sampling distribution of the data, or to model prior expectations of the analyst about structural similarities among graphs representing data sets that are related. In this paper, we propose an elaboration that seeks to model an additional feature of the sampling distribution — a feature increasingly encountered in biomedical data — whereby correlations between the observed outcome variables are more realistically considered to be the byproduct of multiple underlying conditional dependencies acting at different levels: Specifically, the case where, for example, observed correlations between expressed genes in different tissues (e.g., liver, kidney, brain) measured on the same individual result from distinct dependence structures existing not only within the specific tissue but also across tissues (system-wide) at the level of the whole body. We describe these independent graphical strata as the "category-specific" and the "systemic" layers, and use latent outcome variables to approach their estimation.

Introduced more formally, the conditional dependence relationships among a set of p outcome variables, $Y = (Y_1, \ldots, Y_p)$, can be represented by a graph $\mathcal{G} = (\Gamma, E)$ where each variable corresponds to a node in the set Γ and conditional dependencies are represented by the edges in the set E. If we further assume that the joint distribution of the outcome variables is multivariate Gaussian, $Y \sim \mathcal{N}(0, \Sigma)$, then conditional dependencies are reflected in the non-zero entries of the precision matrix $\Omega = \Sigma^{-1}$. Specifically, variables i and j are conditionally independent given the other variables if and only if the (i, j)-th element of Ω is zero. Inferring the dependence structure of such a Gaussian graphical model is thus equivalent to estimating which elements of its precision matrix are non-zero.

When the underlying graph is sparse, as is often assumed, the ordinary maximum likelihood estimate (MLE) is dominated by shrinkage methods: The MLE of Ω typically implies graph that is fully connected, and so gives a result that is unhelpful for estimating graph topology. To impose sparsity, and thereby provide a more informative inference about network structure, a number of methods have been introduced that estimate the precision matrix under ℓ_1 regularization. For example, Meinshausen and Bühlmann (2006) proposed to determine iteratively the edges of each node in $\mathcal{G}$ by regressing the corresponding variable Y_j on the remaining variables Y_{-j} under an ℓ_1 penalty, an approach which can be viewed as optimizing a pseudo-likelihood (Rocha et al., 2008; Ambroise et al., 2009; Peng et al., 2009). More recently, a large number of papers have proposed for estimation of GGMs using sparse penalized maximum likelihood (see for example Yuan and Lin, 2007; Banerjee et al., 2008; d'Aspremont et al., 2008; Rothman et al., 2008; Ravikumar et al., 2011). Efficient implementations to address this problem include the graphical-LASSO (GLASSO) algorithm (Friedman et al., 2008) and the QUadratic Inverse Covariance (QUIC) algorithm (Hsieh et al., 2011). The convergence rate and selection consistency of such penalized estimation schemes have also been described in theoretical studies (for example, Rothman et al., 2008; Lam and Fan, 2009).

Although a single graph provides a useful representation for an underlying dependence structure, several extensions of GGMs have been proposed. In the context where the precision matrix, and hence the graph, is dynamic over time, Zhou et al. (2010) proposed a weighted method to estimate the graph's temporal evolution. Another practical extension

is the simultaneous estimation of multiple graphs that may share some common structure. For example, when inferring how brain regions interact using fMRI data, each subject's brain corresponds to a different graph, but we would nonetheless, expect some interaction patterns to be common across subjects, as well as patterns specific to an individual. In such cases, joint estimation of multiple related graphs can be more efficient than estimating graphs separately. For joint estimation of Gaussian graphs, Varoquaux et al. (2010) and Honorio and Samaras (2010) proposed methods using group-LASSO (Yuan and Lin, 2006), and multitask-LASSO respectively. Both methods assume that all graphs share the same pattern, namely that the precision matrices have the same pattern of zeros. To provide greater flexibility, Guo et al. (2011) proposed a joint penalized method using a hierarchical penalty, and derived the convergence rate and sparsistency properties for the resulting estimators. Under the same setting, Danaher et al. (2014) extended the graphical lasso (Friedman et al., 2008) to estimate multiple graphs from independent data sets using penalties based on the generalized fused lasso or, alternatively, the sparse group lasso.

The methods discussed above for estimating multiple Gaussian graphs focus on settings in which data collected from different categories are stochastically independent. In some applications, however, data from different categories are more naturally considered stochastically dependent. In a study considered here, gene expression data have been collected on multiple tissues in multiple mice. Specifically, for each mouse, we have expression measurements for p genes in each of K different tissues (or categories, in our terminology), represented by the p-vector Y_k $(k = 1, \ldots, K)$. Gene expression profiles between mice may be from a similar network structure, but are otherwise stochastically independent. Gene expression profiles for different tissues within the same mouse, however, are stochastically dependent. For this type of data, increasingly common in biomedical research, the above methods are not applicable.

To explore the gene network structure across different tissues, and to characterize the dependence among tissues, we consider a decomposition of the observed gene expression Y_k into two latent vectors

$$Y_k = Z + X_k \tag{2.1}$$

where $Z, X_1, \ldots, X_K$ are mutually independent. Because $\text{cov}(Y_k, Y_l) = \text{var}(Z)$ for any $k \neq l$, Z represents the sample dependence across different tissues. Letting Ω_j denote the precision matrix of X_j for tissue j, and defining $\text{var}(Z) = \Omega_0^{-1}$, we aim to estimate Ω_k for all $k = 0, 1, \ldots, K$ from the observed outcome data $\{Y_1 = y_1, \ldots, Y_K = y_K\}$. To accomplish joint estimation of multiple dependent networks, two new methods are proposed: a one-step method and an expectation-maximization (EM) method. To our knowledge, this is the first work proposing joint estimation of such systemic and category-specific networks.

In the above decomposition, z can be viewed as representing "systemic" variation in gene expression, that is, variation manifesting simultaneously in all measured tissues of the same mouse, whereas x_k represents "category-specific" variation, that is, variation unique to tissue k. An important property of this two-layer model is that sparsity in the systemic and category-specific networks can produce graphs for the outcome variable y that are highly connected (i.e., not sparse). Conversely, highly connected graphs for the outcome y can easily arise from relatively sparse underlying dependencies acting at two levels. This phenomenon is illustrated in Figure 2.1. Category-specific networks Ω_1 and Ω_2 are depicted for the two categories C1 and C2 (Figure 2.1a,b); these might correspond to, for example, liver and brain tissue-types. The systemic network Ω_0 is depicted in Figure 2.1c; this reflects relationships affecting all tissues at once, for example, gene interactions that are responsive to hormone levels or other globally-acting processes. Despite the fact that all three underlying networks, Ω_0, Ω_1 and Ω_2, are sparse, the precision matrix of observed variables within each tissue — that is, the "aggregate" network $\Omega_{Y_k} = (\Omega_0^{-1} + \Omega_k^{-1})^{-1}$ (following Eq (2.1)) — is nonetheless highly connected. Existing methods aiming to estimate a single sparse network layer are therefore ill-suited to this problem because they impose sparsity in the wrong place — on the aggregate network rather than on the two simpler layers that generate it. Even methods that seek a common structure shared by categories will be inappropriate because the dependence between the categories is not only structural but also stochastic: Even if a mouse has biological replicates (e.g., genetically identical but distinct mice) that exhibit the same structure of systemic network, that mouse's own tissues are affected by individual-specific system-wide stochastic variation.

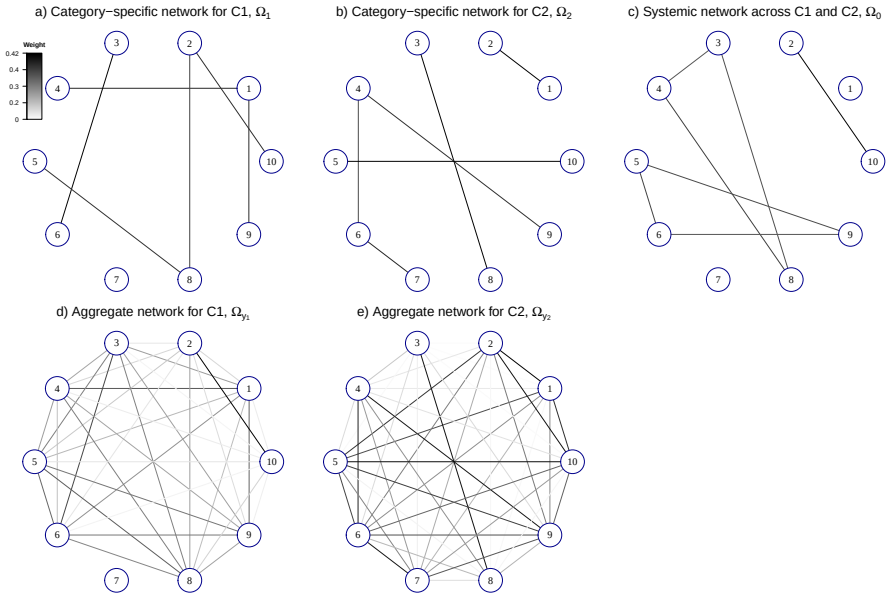

Figure 2.1: Illustration of systemic and category-specific networks using a toy example with two categories ($C1$ and $C2$) and $p = 10$ variables. a) Category-specific network for $C1$. b) Category-specific network for $C2$. c) Systemic network affecting variables in both C_1 and C_2. d)Aggregate network, $\Omega_{Y_1} = (\Omega_1^{-1} + \Omega_0^{-1})^{-1}$, for category C_1. e) Aggregate network, $\Omega_{Y_2} = (\Omega_2^{-1} + \Omega_0^{-1})^{-1}$, for C_2.

The remainder of the article is organized as follows. In Section 2.2, we introduce our dependent Gaussian graphical model, its implementation, and the one-step and EM methods. In Section 2.3, we study the asymptotic properties of the proposed methods. In Section 2.4, we illustrate the performance of our methods through simulations and real mouse study.

2.2 Methodology

For convenience the following notations are used throughout the paper. We denote the true precision and covariance matrices respectively as Ω^* and Σ^*. For any matrix $W = (\omega_{ij})$, we denote the determinant as $\det(W)$, the trace as $\mathrm{tr}(W)$ and W^- as the off-diagonal entries of W. We further denote the jth eigenvalue of W as $\phi_j(W)$, and the minimum and maximum eigenvalues of W as $\phi_{\min}(W)$ and $\phi_{\max}(W)$. The Frobenius norm $\|W\|_F$ is defined as $\sum_{i,j} \omega_{ij}^2$; the operator/spectral norm $\|W\|^2$ is defined as $\phi_{\max}(WW^{\mathrm{T}})$; the infinity norm $\|W\|_\infty$ is defined as $\max|w_{ij}|$, and we also define $|W|_1 = \sum_{i,j} |\omega_{ij}|$.

2.2.1 Problem formulation

In the problem we address, measurements are available on the same p outcome variables in each of K distinct categories on each of n subjects. Some dependency is anticipated among outcomes both at the level of the category and at the level of the subject. We describe dependency at the level of the category as "category-specific". Drawing an analogy with physiology, we describe dependency at the level of the subject (i.e., the individual, the mouse, etc) as "systemic"; that is, modelled as if affecting outcomes in all categories of the same subject simultaneously. Our primary example is the measurement of gene expression giving rise to transcript abundance readings on p genes on K tissues (e.g., liver, kidney, brain) in n laboratory mice. Letting $Y_{k,i}$ be the i-th data vector for the k-th category, we model

$$Y_{k,i} = X_{k,i} + Z_i \quad (i = 1, \ldots, n; \quad k = 1, \ldots, K), \tag{2.2}$$

where Z_i is the random vector corresponding to the shared systemic random effect, and $X_{k,i}$ is the random effect corresponding to the k-th category. We assume that $X_{k,i}$ and Z_i are independent and identically distributed $p-$dimensional random vectors with mean 0, and covariance matrices Σ_k and Σ_0 respectively, for $i = 1, \ldots, n$ and $k = 1, \ldots, K$. For simplicity, we further assume that $X_{k,i}$, and Z_i are independent of each other and each follows a multivariate Gaussian distribution.

For the i-th sample in the k-th category, we observe the p-dimensional realization of $Y_{k,i}$, vector $y_{k,i} = (y_{k,i1}, \ldots, y_{k,ip})^{\mathrm{T}}$. Without loss of generality, we assume these observations are centered, i.e., $\sum_{i=1}^{n} y_{k,ij} = 0$ $(j = 1, \ldots, p; \ k = 1, \ldots, K)$. Let $y_{\cdot,i}$ be the combined data vector with $y_{\cdot,i} = (y_{1,i}^{\mathrm{T}}, \ldots, y_{K,i}^{\mathrm{T}})^{\mathrm{T}}$, such that $y_{\cdot,i}$ follows a Gaussian distribution with covariance $\Sigma_Y = \{_d\Sigma_k\} + J \otimes \Sigma_0 = \{\Sigma_{Y(l,m)}\}_{1 \le l,m \le K}$, where $\{_{d\cdot}\}$ is a block diagonal matrix, J is a square matrix with all $1'$s as the entries, $\otimes$ is the Kronecker product and $\Sigma_{Y(l,m)}$ is the covariance matrix between Y_l and Y_m. We denote the n by Kp dimensional data matrix by $y = (y_{\cdot,1}, \ldots, y_{\cdot,n})^{\mathrm{T}}$, and let $\Omega_k = (\Sigma_k)^{-1} = (\omega_{k(i,j)})_{p \times p}$, and $\Omega_Y = (\Sigma_Y)^{-1}_{Kp \times Kp}$. Our goal is to estimate Ω_k. Although X_k and Z are latent variables, we can show that Ω_k is identifiable under the model setup in (2.2) with $K \ge 2$. More details can be found in

Section 2.6. For simplicity, we write Ω and Σ for $\{\Omega_k\}_{k=0}^K$ and $\{\Sigma_k\}_{k=0}^K$ respectively in the following derivation.

The log-likelihood of the data can be written as

$$\mathcal{L}(\Omega \mid y) = -\frac{npK}{2}\log(2\pi) + \frac{n}{2}\left[\log\{\det(\Omega_Y)\} - \mathrm{tr}(\hat{\Sigma}_Y\Omega_Y)\right],$$

where

$$\hat{\Sigma}_Y = n^{-1}\sum_{i=1}^{n} y_i y_i^{\mathrm{T}} = \{\hat{\Sigma}_{Y(l,m)}\}_{1\le l,m\le K} \tag{2.3}$$

is the $Kp \times Kp$ sample covariance matrix. Under our setting, the log-likelihood can also be expressed as

$$\begin{aligned}\mathcal{L}(\Omega \mid y) \propto &\sum_{k=1}^{K}\left[\log\{\det(\Omega_k)\} - \mathrm{tr}(\hat{\Sigma}_{Y(k,k)}\Omega_k)\right] + \log\{\det(\Omega_0)\} \\ &- \log\{\det(A)\} + \sum_{l,m=1}^{K}\mathrm{tr}\left(\Omega_l\hat{\Sigma}_{Y(l,m)}\Omega_m A^{-1}\right),\end{aligned} \tag{2.4}$$

where $A = \sum_{k=0}^K \Omega_k$. The detailed derivation can be found in the Supplementary material.

A natural way to achieve a sparse estimate of Ω is to maximize the penalized log-likelihood

$$\hat{\Omega} = \underset{\Omega\succ 0}{\mathrm{argmax}}\,\mathcal{P}(\Omega) = \underset{\Omega\succ 0}{\mathrm{argmax}}\,\mathcal{L}(\Omega \mid y) - \lambda_1\sum_{k=1}^{K}|\Omega_k^-|_1 - \lambda_2|\Omega_0^-|_1. \tag{2.5}$$

Because the likelihood is complicated in its full form, direct estimation of the precision matrices in (2.5) is difficult. Estimation can proceed directly, however, given the values z of the latent outcome vector Z. Using this observation and recalling that $Z \sim \mathcal{N}(0, \Sigma_0)$, we can first estimate Σ_0 and then the other parameters subsequently. In Sections 2.2.2 and 2.2.3, we consider estimation of these multiple dependent graphs using a one-step procedure and a method based on the EM algorithm.

2.2.2 One-step method

The idea behind our one-step method is to generate a good initial estimate for Σ and then obtain estimates for Ω through a subsequent one-step optimization. Because $\mathrm{var}(Z) =$

$\text{cov}(Y_l, Y_m)$, for any $m \neq l$, it is natural to use the covariance matrix $\Sigma_{Y(l,m)}$ between all pairs of Y_l and Y_m to estimate Σ_0 as

$$\hat{\Sigma}_0 = \frac{1}{K(K-1)} \sum_{m \neq l} \hat{\Sigma}_{Y(m,l)} = \frac{1}{K(K-1)n} \sum_{m \neq l} \sum_{i=1}^{n} \left(y_{m,i} y_{l,i}^{\mathrm{T}} \right). \tag{2.6}$$

Using the fact that $\text{var}(X_k) = \text{var}(X_k) - \text{var}(Z)$, we can then obtain an estimate for Σ_k as

$$\hat{\Sigma}_k = \hat{\Sigma}_{Y(k,k)} - \hat{\Sigma}_0 = \frac{1}{n} \sum_{i=1}^{n} \left(y_{k,i} y_{k,i}^{\mathrm{T}} \right) - \hat{\Sigma}_0. \tag{2.7}$$

Note that although $\hat{\Sigma}_k$ is symmetric, it is not necessarily positive semidefinite. Positive definiteness can be ensured, however, using the projection approach of Xu and Shao (2012): for any possible non-positive definite matrix $\hat{\Sigma}_k$, we obtain the projection $\hat{\Sigma}'_k$ by solving

$$\hat{\Sigma}'_k = \underset{\Sigma \succeq 0}{\text{argmin}} \, \|\Sigma - \hat{\Sigma}_k\|_\infty. \tag{2.8}$$

Lastly, we estimate Ω by minimizing $K+1$ separate functions, $\mathcal{W}_k$, defined as follows:

$$\mathcal{W}_k(\Omega_k) = \text{tr}(\hat{\Sigma}'_k \Omega_k) - \log\{\det(\Omega_k)\} + \lambda \sum_{i \neq j} \left| \omega_{k(i,j)} \right|, \tag{2.9}$$

where $k = 0, 1, \ldots, K$, and $\lambda = \lambda_2$ when $k = 0$ and $\lambda = \lambda_1$ otherwise. The minimization problem of (2.9) can be solved efficiently by various algorithms such as GLASSO as proposed by Friedman et al. (2008) or by QUIC as proposed by Hsieh et al. (2011). We refer to this approach as the "one-step" method and compare its performance with the EM method defined next.

2.2.3 Graphical EM method

The one-step method provides an estimate of Ω. In the spirit of the classic Expectation-Maximization (EM) algorithm (Dempster et al., 1977), this estimate of Ω can be used to obtain a better estimate of Σ, which in turn can be used to obtain a better estimate of Ω. This procedure can then be iterated until the estimates of Ω converge, leading to a graphical EM, described in detail below.

First, we rewrite the sampling model as

$$\begin{pmatrix} Z \\ Y_1 - Z \\ \vdots \\ Y_K - Z \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & 0 & \dots & 0 \\ 0 & \Sigma_1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & \Sigma_K \end{pmatrix} \right),$$

and the log-likelihood on the observed data $Y = y$ as

$$\begin{aligned} \mathcal{L}(\Omega \mid y, z) \propto & \log\big\{\det(\Omega_0)\big\} - \operatorname{tr}\big(\Omega_0 z z^{\mathrm{T}}/n\big) \\ & + \sum_{k=1}^{K} \Big[\log\big\{\det(\Omega_k)\big\} - \operatorname{tr}\big(\Omega_k \sum_{i=1}^{n} (y_{k,i} - z_i)(y_{k,i} - z_i)^{\mathrm{T}}/n\big) \Big]. \end{aligned}$$

The above log likelihood cannot be calculated directly because the values of z_i and $z_i z_i^{\mathrm{T}}$ are unobserved. But we can calculate a function $\mathcal{Q}(\Omega \mid \Omega^{(t)}, y)$ in which z_i and $z_i z_i^{\mathrm{T}}$ are substituted by their expected values conditional on Ω and y. This implies the following EM steps.

E step:

$$\begin{aligned} \mathcal{Q}(\Omega \mid y, \Omega^{(t)}) \propto & \sum_{k=1}^{K} \left(\log\{\det(\Omega_k)\} - \operatorname{tr}\left[\Omega_k E\left\{ \sum_{i=1}^{n} (y_{k,i} - z_i)(y_{k,i} - z_i)^{\mathrm{T}}/n \mid y, \Omega^{(t)} \right\} \right] \right) \\ & + \log\{\det(\Omega_0)\} - \operatorname{tr}\left\{ \Omega_0 E\left(\sum_{i=1}^{n} z_i z_i^{\mathrm{T}}/n \mid y, \Omega^{(t)} \right) \right\} \\ = & \sum_{k=0}^{K} \left\{ \log\{\det(\Omega_k)\} - \operatorname{tr}\left(\Omega_k \dot{\Sigma}_k^{(t)} \right) \right\}. \end{aligned}$$

M step:

$$\Omega^{(t+1)} = \underset{\Omega}{\operatorname{argmin}} - \mathcal{Q}(\Omega \mid y, \Omega^{(t)}) + \lambda_1 \sum_{i \neq j} \sum_{k=1}^{K} \left| \omega_{k(i,j)} \right| + \lambda_2 \sum_{i \neq j} \left| \omega_{0(i,j)} \right|, \tag{2.10}$$

where $\Omega^{(t)}$ and $\omega_{k(i,j)}^{(t)}$ denote the estimates from the t-th iteration, and $\dot{\Sigma}_k^{(t)}$ is defined as

$$\begin{aligned} \dot{\Sigma}_k^{(t)} = & E\left\{ \sum_{i=1}^{n} (y_{k,i} - z_i)(y_{k,i} - z_i)^{\mathrm{T}}/n \mid y, \Omega^{(t)} \right\} \\ = & \ddot{\Sigma}_{Y(k,k)} - \sum_{l=1}^{K} \left(\ddot{\Sigma}_{Y(k,l)} \Omega_l^{(t)} \right) (A^{(t)})^{-1} - (A^{(t)})^{-1} \sum_{l=1}^{K} \left(\Omega_l^{(t)} \ddot{\Sigma}_{Y(l,k)} \right) \end{aligned}$$

$$+ (A^{(t)})^{-1} \sum_{l,k=1}^{K} \Big(\Omega_l^{(t)} \ddot{\Sigma}_{Y(l,k)} \Omega_k^{(t)}\Big)(A^{(t)})^{-1} + (A^{(t)})^{-1}, \qquad (k = 1, \ldots, K). \quad (2.11a)$$

$$\dot{\Sigma}_0^{(t)} = \sum_{i=1}^{n} E\big(z_i z_i^{\mathrm{T}}/n \mid y, \Omega^{(t)}\big) = (A^{(t)})^{-1} + (A^{(t)})^{-1} \sum_{l,k=1}^{K} \big(\Omega_l^{(t)} \ddot{\Sigma}_{Y(l,k)} \Omega_k^{(t)}\big)(A^{(t)})^{-1}, \quad (2.11b)$$

where $\ddot{\Sigma}_y$ is an estimator for Σ_y^*, here $\ddot{\Sigma}_y = \hat{\Sigma}_y$. Thus, at the $t+1$ iteration, problem (2.74) is decomposed into $K+1$ separate optimization problems:

$$\Omega_k^{(t+1)} = \underset{\Omega_k}{\operatorname{argmin}} \left\{ \operatorname{tr}\left(\Omega_k \dot{\Sigma}_k^{(t)}\right) - \log\{\det(\Omega_k)\} + \lambda \sum_{i \neq j} \big|\omega_{k(i,j)}\big| \right\}, \qquad (2.12)$$

where $\lambda = \lambda_2$ when $k = 0$, otherwise $\lambda = \lambda_1$ for $k = 0, \ldots, K$. We then can use GLASSO (Friedman et al., 2008) to solve (2.12).

We summarize the proposed EM method in the following steps:

Step 1 (Initial value). Initialize $\hat{\Sigma}_0$ and $\hat{\Sigma}_k$ for all $k = 1, \ldots, K$ using (2.3), (2.6), (2.7) and (2.8).

Step 2 (Updating rule: the M step). Update Ω_k using (2.12) for all $k = 0, \ldots, K$ using GLASSO.

Step 3 (Updating rule: the E step). Update $\dot{\Sigma}_k$ using (2.76a) and (2.76a).

Step 4 (Iteration). Iterate Steps 2 and 3 until convergence is achieved.

The next proposition demonstrates the convergence property of our graphical EM algorithm.

Proposition 2.1. *With $\lambda_1 > 0$ and $\lambda_2 > 0$, the graphical EM algorithm solving* (2.5) *has the following properties:*

1. *The penalized log-likelihood in* (2.5) *is bounded above;*
2. *For each iteration, the penalized log-likelihood is non-decreasing;*
3. *For a prespecified threshold δ, after finite steps, the objective function in* (2.5) *converges in the sense that*

$$|\mathcal{P}(\Omega^{(t+1)}) - \mathcal{P}(\Omega^{(t)})| < \delta.$$

The detail of proof could be found in the Proof section.

2.2.4 Model selection

We consider two options for selecting the tuning parameter $\lambda = (\lambda_1, \lambda_2)$, minimization of the extended BIC (Chen and Chen, 2008), and cross-validation. Extended BIC is quick to compute and explicitly takes into account both goodness of fit and model complexity. Cross-validation, by contrast, is more computationally demanding and focuses on the predictive power of the model.

In our model we define extended BIC as

$$\mathrm{BIC}_\gamma(\lambda) = -2\mathcal{L}(\{\hat{\Omega}_k\}_{k=0}^K) + \nu(\lambda)\log n + 2\gamma \log \binom{Kp(p-1)/2}{\nu(\lambda)}, \quad 0 \le \gamma \le 1,$$

where $\{\hat{\Omega}_k\}_{k=0}^K$ are the estimates with the tuning parameter set at λ, $\mathcal{L}(\cdot)$ is the likelihood function described in (2.4), the degrees of freedom, $\nu(\lambda)$, is the sum of the number of non-zero off-diagonal elements on $\{\hat{\Omega}_k\}_{k=0}^K$. The criterion is indexed by a parameter $\gamma \in [0, 1]$. The tuning parameter λ is selected by

$$\hat{\lambda} = \mathrm{argmin}\{\mathrm{BIC}_\gamma(\lambda) : \lambda_1, \lambda_2 \in (0, \infty)\}.$$

In describing the cross-validation procedure, we define the predictive negative log-likelihood function as follows:

$$\mathcal{F}(\Sigma, \Omega) = \mathrm{tr}(\Sigma\Omega) - \log\{\det(\Omega)\}.$$

To select λ using cross-validation, we first randomly split the dataset equally into J groups, and denote the sample covariance matrix from the j-th group as $\Sigma_{(j,\lambda)}$ and the precision matrix estimated from the remaining groups as $\hat{\Omega}_{(-j,\lambda)}$. Then we choose λ as

$$\underset{\lambda}{\mathrm{argmin}} \left\{ \sum_{j=1}^{J} \mathcal{F}(\Sigma_{(j,\lambda)}, \hat{\Omega}_{(-j,\lambda)}) : \quad \lambda_1, \lambda_2 \in (0, \infty) \right\}.$$

The performance of these two selection methods is provided in Section 2.4.

2.3 Asymptotic properties

In this section we study the asymptotic properties of our proposed methods. First, we introduce the notation and the regularity conditions on the true precision matrices $\{\Omega_k^*\}_{k=0}^K$, where $\Omega_k^* = (\omega_{k(j,j')}^*)_{p\times p}$. Let $T_k = \left\{(j,j') : j \neq j', \omega_{k(j,j')}^* \neq 0\right\}$ be the set of indices of all nonzero off-diagonal elements in Ω_k^*, $q_k = |T_k|$ be the cardinality of T_k, and $q = \sum_{k=0}^K q_k$. Let $\{\Sigma_k^*\}_{k=0}^K$ be the true covariance ma

trices for z and $\{x_k\}_{k=1}^K$, and $\Sigma_Y^* = \{\Sigma_{Y(l,m)}^*\}_{1\leq l,m\leq K}$ be the true covariance matrices for y. We assume that the following regularity conditions hold.

Condition 1. There exist constants τ_1, τ_2 such that for all $k = 0, 1, \ldots, K$, $0 < \tau_1 < \phi_{\min}(\Omega_k^*) \leq \phi_{\max}(\Omega_k^*) < \tau_2 < \infty$.

Condition 2. There exists a constant $\tau_3 > 0$, such that $\min_{k=0,\ldots,K} \min_{(j,j')\in T_k} \left|\omega_{k(j,j')}^*\right| \geq \tau_3$.

Condition 1 bounds the eigenvalues of Ω_k^*, thereby guaranteeing the existence of its inverse and facilitating the proof of consistency. Condition 2 is needed to bound the nonzero elements away from zero.

The following theorems discuss estimation consistency and selection sparsistency of our methods.

Theorem 2.1 (Consistency of the one-step method)**.** *Under Conditions 1-2,* $(p+q)\log p/n = o(1)$*, and* $a_1(\log p/n)^{1/2} \leq \lambda_1, \lambda_2 \leq b_1\{(1+p/q)\log p/n\}^{1/2}$ *for some constants* a_1 *and* b_1*. Let* $\{\hat{\Omega}_k\}_{k=0}^K$ *be the minimizer defined by* (2.9) *using the one-step method, then*

$$\sum_{k=0}^K \left\|\hat{\Omega}_k - \Omega_k^*\right\|_F = O_p\left[\left\{\frac{(p+q)\log p}{n}\right\}^{1/2}\right].$$

Before we introduce the main theorem of the EM algorithm, we first present a corollary of Theorem 2.1 which gives a good estimator of Σ_Y^*.

Corollary 2.1. *Suppose that Conditions 1-2 hold, and* $\hat{\Omega}_k$ *is the one-step solution on Theorem 2.1, then* $\check{\Sigma}_k = (\hat{\Omega}_k)^{-1}$ *satisfies*

$$\left\|\check{\Sigma}_k - \Sigma_k^*\right\|_F = O_p\left[\left\{\frac{(p+q)\log p}{n}\right\}^{1/2}\right].$$

To study our EM estimator, we need a good estimator for Σ_Y^* which specifies in the following condition.

Condition 3. We assume there exists an estimator $\tilde{\Sigma}_Y$ such that

$$\|\tilde{\Sigma}_Y - \Sigma_Y^*\|_F = O_p\left[\left\{\frac{(p+q)\log p}{n}\right\}^{1/2}\right].$$

The rate in Condition 3 is required to control the convergence rate of the E-step estimate $\dot{\Sigma}_k$ and thus the consistency of the estimate from the EM method. Under the conditions in Theorem 2.1, we can use the one-step estimator $\hat{\Omega}_0, \ldots, \hat{\Omega}_K$ to obtain $\tilde{\Sigma}_Y = J \otimes (\hat{\Omega}_0)^{-1} + \{_d(\hat{\Omega}_k)^{-1}\}$, which satisfies Condition 3 by Corollary 2.1.

Theorem 2.2 (Consistency of the EM method)**.** *Suppose Conditions 1-3 hold, and* $(p+q)\log p/n = o(1)$*, and* $a_2(\log p/n)^{1/2} \le \lambda_1, \lambda_2 \le b_2\{(1+p/q)\log p/n\}^{1/2}$ *for some constants* a_2 *and* b_2*. Then the solution,* $\{\hat{\Omega}_k\}_{k=0}^K$*, of the EM method satisfies*

$$\sum_{k=0}^K \left\|\hat{\Omega}_k - \Omega_k^*\right\|_F = O_p\left[\left\{\frac{(p+q)\log p}{n}\right\}^{1/2}\right].$$

Theorem 2.3 (Sparsistency of the one-step method)**.** *Under the assumptions of Theorem 2.1. If we further assume that* $\sum_{k=0}^K \left\|\hat{\Omega}_k - \Omega_k^*\right\| = O_p(\eta_n)$ *for a sequence of* $\eta_n \to 0$*, and* $(\log p/n + \eta_n^2)^{1/2} = O(\lambda_1) = O(\lambda_2)$*, then with probability tending to 1, the minimizer* $\{\hat{\Omega}_k\}_{k=0}^K$ *satisfies* $\hat{\omega}_{k(j,j')} = 0$ *for all* $(j,j') \in T_k^c$*,* $k = 0, \ldots, K$*.*

To obtain sparsistency we require a lower bound on the rate of the regularization parameters λ_1 and λ_2. For consistency, we need an upper bounds for λ_1 and λ_2 to control the biases. In order to have both consistency and sparsistency to hold simultaneously, we need the bounds to be compatible, that is, we need $(\log p/n + \eta_n^2)^{1/2} = O(\lambda_1, \lambda_2) = \{(1+p/q)\log p/n\}^{1/2}$. Using the inequalities $\|W\|_F^2/p \le \|W\|^2 \le \|W\|_F^2$, there are two scenarios describing the rate of η_n, as in Lam and Fan (2009). In the worst case, where $\sum_{k=0}^K \|\hat{\Omega}_k - \Omega_k^*\|$ has the same rate as $\sum_{k=0}^K \|\hat{\Omega}_k - \Omega_k^*\|_F$, the two bounds are compatible only when $q = O(1)$. In the most optimistic case, where $\sum_{k=0}^K \|\hat{\Omega}_k - \Omega_k^*\|^2 = \sum_{k=0}^K \|\hat{\Omega}_k - \Omega_k^*\|_F^2/p$, we have $\eta_n^2 = (1+q/p)\log p/n$, and compatibility of the bounds requires $q = O(p)$.

Theorem 2.4 (Sparsistency of EM method)**.** *Under the assumptions of Theorem 2.2, and if we further assume that* $\sum_{k=0}^{K} \left\| \hat{\Omega}_k - \Omega_k^* \right\| = O_p(\eta_n)$ *and* $\sum_{k=0}^{K} \left\| \tilde{\Sigma}_k - \Sigma_k^* \right\| = O_p(\zeta_n)$ *for sequences of* $\eta_n \to 0$ *and* $\zeta_n \to 0$*. Moreover let* $\zeta_n + \eta_n = O(\lambda_1) = O(\lambda_2)$*, then with probability tending to 1, the minimizer* $\{\hat{\Omega}_k\}_{k=0}^{K}$ *in Theorem 2.2 satisfies* $\hat{\omega}_{k(j,j')} = 0$ *for all* $(j, j') \in T_k^c$*,* $k = 0, \ldots, K$*.*

Similar to the discussion above, for the EM algorithm, we have both consistency and sparsistency when $q = O(p)$ for the best scenario, and $q = O(1)$ for the worst case. Details of the proof can be found in the Supplementary material.

2.4 Simulation

We assessed the performance of the one-step and EM methods by applying them to simulated data generated by two types of synthetic network: a chain network and a nearest-neighbor (NN) network (Figure 2.2). Twenty-four simulation settings were considered. These varied the base architecture of the category-specific network, the degree to which the actual structure could deviate from this basic architecture, and the number of outcome variables (nodes).

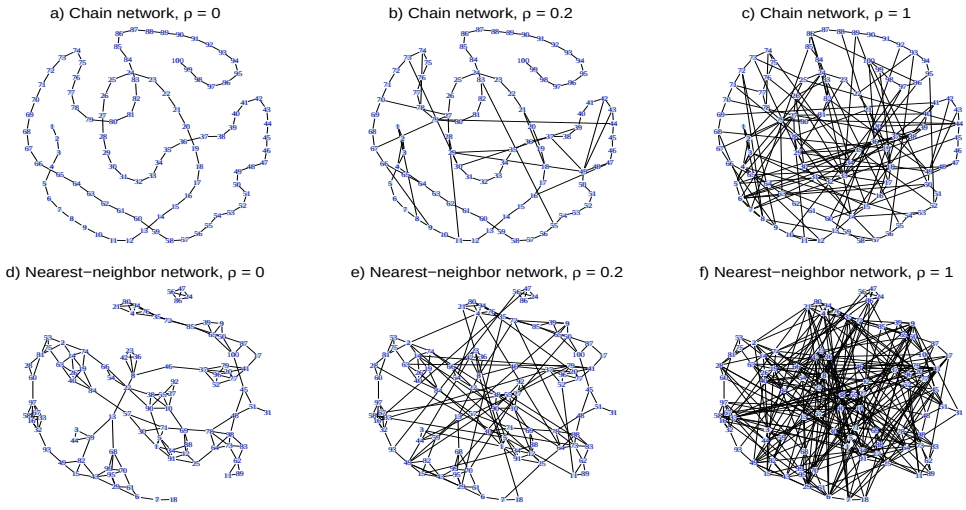

Figure 2.2: Network topologies generated in the simulations. Top row (a-c) shows chain networks with noise ratios $\rho = 0$, 0.2, and 1. Bottom row (d-e) shows nearest-neighbor (NN) networks with $\rho = 0$, 0.2, and 1.

2.4.1 Simulating category-specific and systemic networks

Under each of the 24 simulation conditions, i.i.d. samples were generated, with systemic outcomes generated as $Z_i \sim \mathcal{N}(0, \Omega_0^{-1})$, category-specific outcomes as $x_{ki} \sim \mathcal{N}(0, \Omega_k^{-1})$, and observed outcomes as $y_{ki} = x_{ki} + z_i$, for $k = 1, \ldots, K = 4$, and $i = 1, \ldots, n$. The following base architectures were considered for the five networks $(\Omega_k)_{k=0}^K$:

- Chain/Chain network with $p = 30$ and $p = 100$ nodes: the K category-specific networks and the systemic network are all chain-networks.
- NN/Chain network with $p = 30$ and $p = 100$ nodes: the K category-specific networks are NN-networks and the systemic network is a chain-network.
- Chain/NN network with $p = 30$ and $p = 100$ nodes: the K category-specific networks are chain-networks and the systemic network is an NN-networks.
- NN/NN network with $p = 30$ and $p = 100$ nodes: the K category-specific networks and the systemic network are all NN-networks.

Chain networks were generated using the algorithm in Fan et al. (2009), constructing $\Sigma = \{\sigma_{i,j}\}$ by generating $\sigma_{i,j} = \exp(-|s_i - s_j|/2)$ for $s_1 < s_2 < \ldots < s_p$ as $s_i - s_{i-1} \sim$ Uniform$(0.5, 1)$ for $i = 2, \ldots, p$. NN networks were generated using the method of Li and Guo (2006), sampling p points uniformly on $[0, 1]^2$ and then calculating all pairwise distances to find the m nearest neighbors of each point. Pairs of nodes were linked if they are mutual m-nearest neighbors, with $m = 5$ in our model. Under this construction, elements in the precision matrix for each edge are first generated from uniform $[0.5, 1]$ or $[-1, -0.5]$. The diagonal entry of each row is taken as the sum of the absolute values of that row's elements. Then, the numbers in each row are divided by their corresponding diagonal entry so that the final precision matrix has diagonal elements of 1 and is positive definite. The structures of the chain and NN network are shown in Figures 2.2a and 2.2d respectively.

Simulated networks were allowed to deviate from their base architectures by a specified degree ρ, through random addition of edges following the method of Guo et al. (2011). Specifically, for each Ω_k for $k = 0, 1, \ldots, K$ generated above, we randomly picked a symmetric pair of zero elements and replaced them with a random value generated uniformly

from $[-1, -0.5] \cup [0.5, 1]$. This procedure was repeated ρT times, with T being the number of links in the initial structure, and $\rho \in \{0, 0.2, 1\}$. For each simulation trial we generate two independent realizations of the data tensor y, each corresponding to sample size $n = 300$. The first realization is used for tuning and training, and the second realization is for testing.

2.4.2 Criteria for evaluating performance

The performance of the one-step and EM methods are compared by examining of their receiver operating characteristic (ROC) curves, and numerically using a number of metrics including the entropy loss

$$\text{EL} = \text{tr}\{(\Omega)^{-1}\hat{\Omega}\} - \log[\det\{(\Omega)^{-1}\hat{\Omega}\}] - p\,,$$

and the Frobenius loss

$$\text{FL} = \|\Omega - \hat{\Omega}\|_F^2 / \|\Omega\|_F^2\,,$$

where Ω is the true precision matrix and $\hat{\Omega}$ is the corresponding estimate. We also report the false positive rate (FP), the false negative rate (FN) and the Hamming distance (HD).

2.4.3 Estimation of category-specific Ω_k and systemic networks Ω_0

As shown in our toy example in Figure 2.1, existing methods are designed to estimate aggregate network Ω_{Y_k} instead of category-specific (Ω_k) and systemic (Ω_0) networks. In this section, we focus only on our proposed one-step and EM methods.

Results of the simulations are reported in Table 2.1. Summary statistics are based on 50 replicate trials under each of the 24 conditions, and given for model fitting under both extended BIC and under cross-validation criteria (as described in Section 2.2.4). In general, the one-step method under either model selection criteria resulted in higher values of entropy loss, Frobenius loss, false negative rates and hamming distance. For both methods, cross-validation tended to select models with more false positive links but fewer false negative links leading to a denser graph.

ROC curves for the one-step and EM methods are plotted in Figure 2.3; each curve is based on 100 replications with the constraint $\lambda_1 = \lambda_2$. In the plots, the ROC curves of

the EM method are seen to dominate those of the one-step method. Under all the settings, the EM method outperforms the one-step method as the structures become increasingly complicated (i.e., as ρ is increased). In general, the EM method delivers more accurate results than the one-step method.

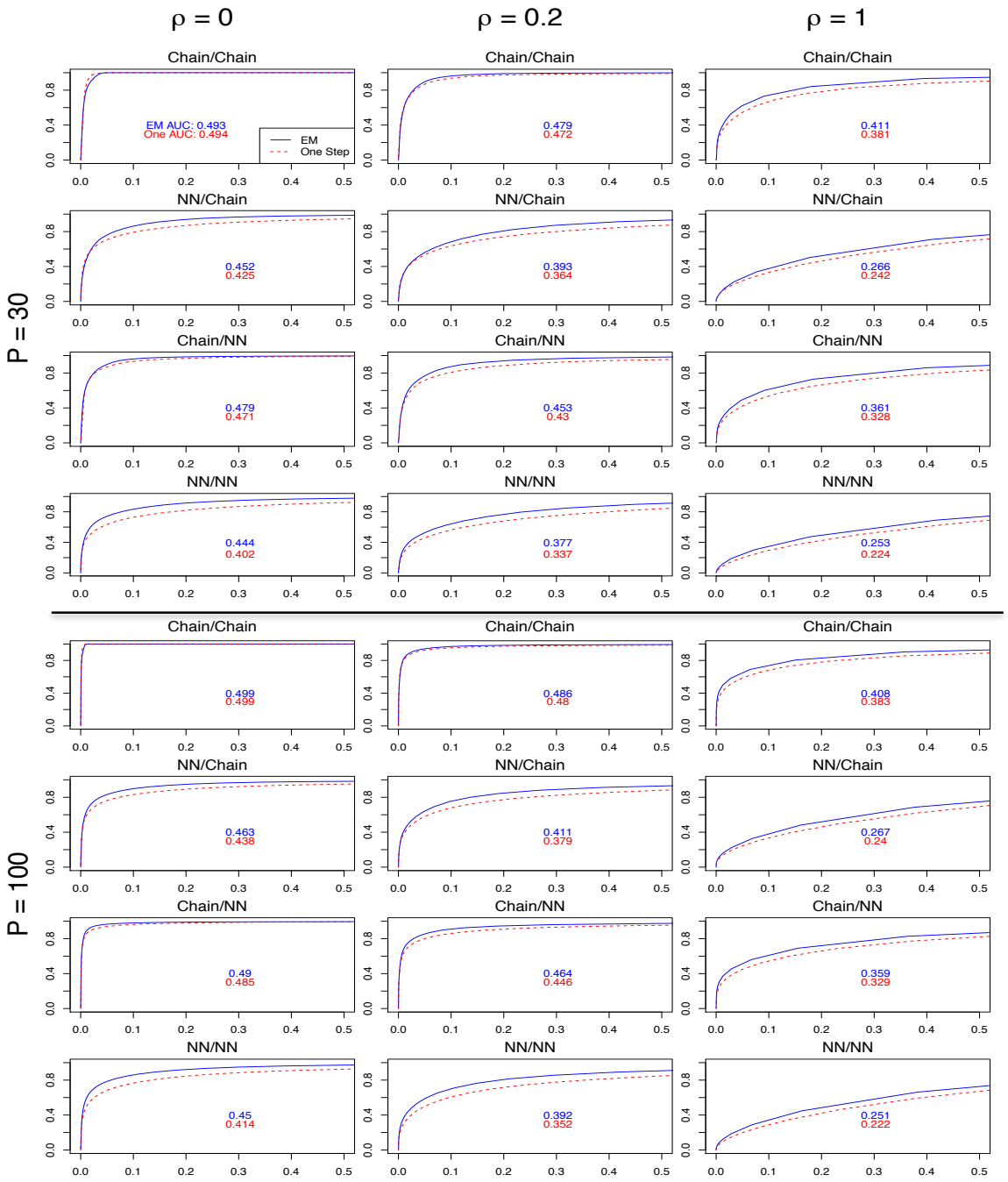

Figure 2.3: Receiver operating characteristic (ROC) curves assessing power and discrimination of graphical inference under different simulation settings. Each panel reports performance of the EM method (blue line) and the one-step method (dashed line), plotting true positive rate (y-axis) against false positive rate (x-axis) for a given noise ratio ρ.

2.4.4 Estimation of aggregate networks Ω_{Y_k}

Although our goal is to estimate the two layers of network that generate the data, we can also use our estimators of Ω_0 and $\Omega_k, k = 1, \ldots, K$ to estimate the aggregate network $\Omega_{Y_k} = (\Omega_k^{-1} + \Omega_0^{-1})^{-1}$ as a derived statistic. Doing so allows us to compare our method with existing methods that aim explicitly to estimate the aggregate network Ω_{Y_k}; these methods would otherwise be incomparable.

We compare the performance for our EM method with two existing methods for estimating multiple graphs: the hierarchical penalized likelihood method (HL) proposed by Guo et al. (2011), and the joint group lasso (JGL) proposed by Danaher et al. (2014). The results of this comparison are reported in Table 2.3, which gives Frobenius and Entropy loss for the four methods under each setting. False positive rates and false negative rates are not reported: These are inapplicable here because the aggregate networks are not sparse. Under most simulation settings examined, including Chain/NN, NN/Chain, and NN/NN, the graphical EM method performs the best in terms of both losses. For the Chain/Chain setting, the EM performs second best, with the best results achieved by HL. The stronger performance of HL under this setting is explained by the fact that when both Ω_k and Ω_0 are chain networks, the corresponding Ω_{Y_k} can have strong banding structure with large absolute value within the band and small absolute value outside the band; the HL method performs well because it is designed to work on such structures.

2.5 Application to gene expression data in mouse

To demonstrate the potential utility of our approach, we apply the EM method to mouse genetic data from two experimental studies, those of Dobrin et al. (2009) and Crowley et al. (2014). In each case we aim to infer systemic and category-specific gene co-expression networks from transcript abundance measurements collected by microarray. In describing our inference on these datasets we find it helpful to distinguish two interpretations of a network.

Potential Network: The network of biologically possible interactions in the type of system under study.

Induced Network: The subgraph of the potential network that could be inferred in the population under study. This is a statistical (not physical) phenomenon, and describes the dependency structure induced by the interventions (or perturbations) applied to the system.

A trivial example is the relationship between caloric intake, sex, and body weight. Body weight is influenced by both the state of being male or female and the degree of calorie consumption; these relations constitute edges in the potential network. Yet in a population where caloric intake varies but where individuals are exclusively male, the effect of sex is undefined, and corresponding edges relating sex to body weight would be undetectable (and therefore absent) in the induced network. More generally, the induced network for a system is defined both by the potential network and the intervention applied to it: Two populations of mice could have the same potential network but when subject to distinct interventions could have different induced networks. Conversely, when estimating the dependency structure of variables arising from population data, the degree to which the induced network reflects the potential network is a function of the intervention bias.

The Dobrin et al. (2009) dataset comprises expression measurements for 23,698 transcripts on 301 male mice in four tissues: adipose, liver, brain and muscle. These mice arose from an F_2 cross between two contrasting inbred founder strains, one with normal body weight physiology and the other with a heritable tendency for rapid weight-gain. As is the nature of an F_2 cross of inbred strains, the analyzed offsprings constitute an i.i.d. sample of individuals who are genetically distinct, and are subject to a randomized allocation of normal and weight-inducing DNA variants (alleles) at multiple locations along their genomes. Any gene expression network inferred on such a population would in turn be expected to emphasize more strongly those subgraphs of the underlying potential network related to body weight. Moreover, since the intervention alters a property affecting the entire individual, we might expect it to exert at least some of its effect systemically — that is, globally across all tissues in each individual.

We illustrate the effect of this intervention bias on network inference using a subset of the F_2 data, inferring the dependency structure of gene co-expression among three groups

of well-annotated genes in brain and liver. The first group, an obesity-related gene set, includes the gene *Aif1*, which has been implicated in obesity in a genomewide association study (Thorleifsson et al., 2009), and seven other genes that are in the co-expressed network of *Aif1* (Dobrin et al., 2009): *Pycard*, *Ctss*, *Psmb9*, *Ifi44*, *Rgl2* ,*H2-Eb1* and *Hck*. The second group, an imprinting-related gene set, includes the gene *Igf2*, which has been implicated in genetic imprinting and parent-of-origin effects, and seven other genes that are involved in co-expression network of *Igf2* (Obayashi et al., 2008): *Peg3*, *Prkra*, *Tab1*, *Nrk*, *Pde4b*, *Mdk* and *Rab8a*. The third group comprises seven genes implicated in the functioning of the extracellular matrix (the ECM-related gene set): *Col1A2*, *Ltbp1*, *Ptrf*, *Cdh11*, *Dcn*, *Fbln5*, and *bgn*. These groups were chosen based on criteria independent of our analysis, and represent three groups whose respective effects would be exaggerated under very different interventions. Specifically, we would expect pronounced edges within the following gene sets in the following types of populations: the obesity-related set in F_2 populations derived from founders contrasting in obesity-related outcomes; the ECM-related set in F_2 populations from founders contrasting in ECM characteristics; and the imprinting-related set in offspring of a reciprocal cross, that is, a population in which the factor being deliberately varied is the parent-of-origin (i.e., the mother or father) from which a DNA variant is inherited.

Tissue-specific and systemic networks inferred on the Dobrin et al. (2009) dataset by our EM method are shown Figure 2.4. Each node represents a gene and the thickness of an edge is proportional to the magnitude of the associated partial correlation. The systemic network in Figure 2.4c includes edges on the *Aif1* obesity-related pathway only, which is consistent with the F_2 exhibiting a dependency structure largely induced by an obesity-related genetic intervention that acts systemically. The category-specific networks are shown in Figures 2.4a and b. After taking account of the systemic network, part of the *Aif1* pathway can still be found on all category-specific networks. This suggests that the genetic effect induces both systemic and tissue-specific variation related to the *Aif1* pathway. Figure 2.4 also illustrates the heterogeneity between different categories: It shows that, for instance, *Aif1* and *Rgl2* are linked on brain but not on liver. In their analysis, Dobrin et al. (2009) used a correlation network approach whereby (unconditional) correlations with statistical significance above a

predefined threshold were declared as edges, and found evidence supporting a role for *Aif1* in tissue-to-tissue co-expression.

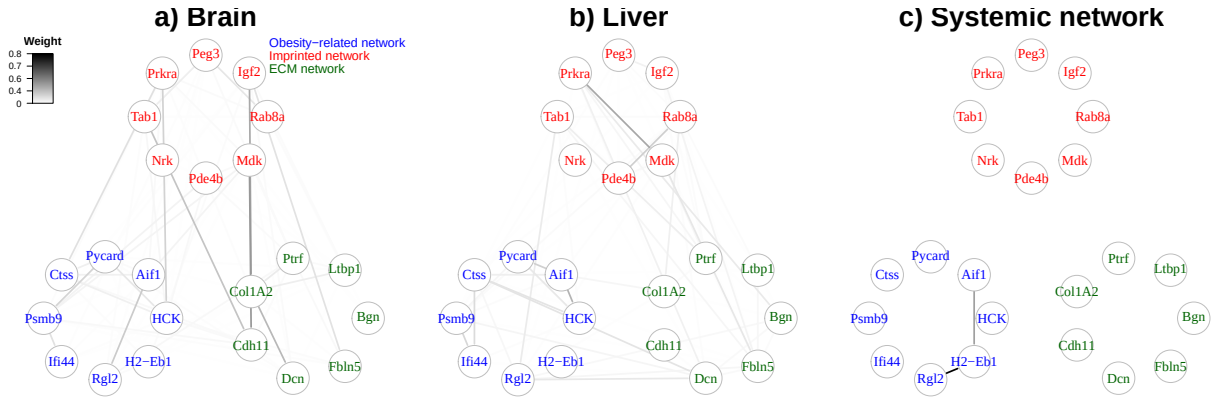

Figure 2.4: Topology of gene co-expression networks inferred by the EM-method for the data from a population of F_2 mice with randomly allocated high-fat vs normal gene variants. Panels a) and b) display the estimated brain-specific and liver-specific dependency structures respectively. Panel c) shows the estimated systemic structure describing whole body interaction that simultaneously affect variables in both tissues.

The Crowley et al. (2014) data set measures mRNA expression levels of $23,000$ genes in four tissues (brain, liver, lung and kidney) in 45 mice arising from three independent reciprocal F_1 crosses. A reciprocal F1 cross between two inbred strains — say, strains P1 and P2 — involves generating two sub-populations: one group comprises offspring from mating P1 females with P2 males; the other comprises offspring from mating P2 females with P1 males. Across the two groups, all mice have the same DNA (i.e., the same genetic background), with each mouse having one copy of each gene from P1 and another copy from P2; but between groups the parent from whom DNA is inherited differs, with, for example, one group inheriting P1 through the mother and the other group inheriting P1 through the father. The underlying intervention in a reciprocal cross is therefore not the varying of genetics as such but the varying of parent-of-origin (i.e., epigenetics). In the Crowley et al. (2014) experiment, three independent reciprocal crosses were performed using all pairs from the three inbred strains CAST, PWK and WSB, i.e., CAST with PWK, CAST with WSB, and PWK with WSB. Note that although the three strains used were not selected specifically for their contrast on a particular outcome measure (they were in fact selected for mutual genetic dissimilarity), across the different crosses we would nonetheless expect

differences in expression entirely due to genetic background. To remove this genetic effect, and therefore focus primarily on the varying of parent-of-origin, for each reciprocal pairing (e.g., CAST with PWK, which comprises CAST x PWK and the reciprocal PWK x CAST) we centered the data for each gene: For jth gene expression for the ith mouse in CAST x PWK, we subtracted the mean of jth gene among all CAST x PWD and PWD x CAST mice. Moreover, to be comparable with our analysis of Dobrin et al. (2009) above, we include only brain and liver data, and restrict attention to the same set of genes as in the analysis of the F_2 mice. Applying our EM method to this reciprocal cross data identifies three edges on the systemic network (Figure 2.5 c) that include the genes *Igf2*, *Tab1*, *Nrk* and *Pde4b*, all from the imprinting-related set implicated in mediating parent-of-origin effects. Thus, the inferred patterns of systemic-level gene relationships in the two studies coincide with the intervention biases we would expect based on the structure of those studies, with genes affecting body weight in the Dobrin et al. (2009) data and genes affected by parent-of-origin in the Crowley et al. (2014) data.

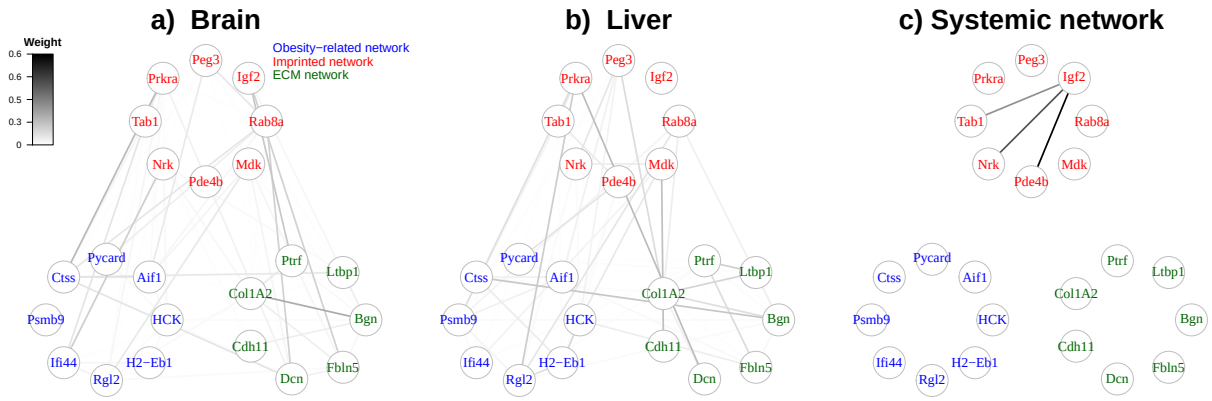

Figure 2.5: Topology of gene co-expression networks inferred by the EM-method for the data from a population of reciprocal F_1 mice. Panels a) and b) display the estimated brain-specific and liver-specific dependency structures respectively. Panel c) shows the estimated systemic structure describing whole body interaction that simultaneously affect variables in both tissues.

To demonstrate the use of our method for higher dimensional data, we examine a larger subset of genes from the Dobrin et al. (2009) study. Selecting the $p = 1000$ genes that had the largest within-group variance among the four tissues in the F_2 population, we applied our graphical EM method, using extended BIC to select the tuning parameter λ. The topologies

of the tissue-specific and systemic networks are shown in Figures 2.6a-d, with a zoomed-in view of the edges of the systemic network shown in Figure 2.6f. The systemic network is sparse, with 249 edges connecting 62 of the 1000 genes (Figure 2.6e). Its sparsity may reflect there being few interactions simultaneously occurring across all tissues in this F_2 population, with one contributing reason being that some genes are being expressed primarily in one tissue and not others. We note that the systemic network also includes a connection between two genes, *Ifi44* and *H2-Eb1*, that are members of the *Aif1* network of Figure 2.4. To characterize more broadly the genes identified in the systemic network, we conducted an analysis of gene ontology (GO) enrichment (Shamir et al., 2005): The distribution of GO terms associated with connected genes in the systemic network was contrasted against the background distribution of GO terms in the entire 1000-gene set. This showed that the systemic network is significantly enriched for genes associated with immune and metabolic processes, which accords with recent studies linking obesity to strong negative impacts on immune response to infection (Milner and Beck, 2012; Lumeng, 2013). In their study, Dobrin et al. (2009) also showed that the enrichment of inflammatory response processes in co-expression from liver and adipose, again using unconditional correlations. We note that obesity is a complex trait that affects and is affected by interactions between and cooperation among different tissues. By looking at the four tissues collectively, we can build a systemic network revealing dependencies common to all tissues; these dependencies may closely associate with the obesity phenotype but may not be so easily identified (or distinguished) in networks inferred on tissues analyzed singly, or on different tissues model without a shared graphical component.

2.6 Discussion

Herein we consider joint estimation of multiple Gaussian graphical models that are stochastically dependent, and we propose a decomposition of the modeled graphs into two layers: a systemic layer, characterizing the sample dependency, and a category-specific layer, representing graph-specific variation. We then propose novel one-step and EM methods that jointly estimate the two layers using only data observed on their combined outcomes. We

evaluate performance of our methods in theory and simulation, and demonstrate promising application of these models in mouse genetics. We are exploring several extensions to this work. First, for the EM method, we currently use estimates from the one-step method generate a single set of starting values. To better explore the domain of the joint penalized likelihood, we could instead initialize from multiple start-points. Second, we can extend our ℓ_1 penalized maximum likelihood approach to general non-convex penalties such as the SCAD penalty (Fan and Li, 2001), MCP (Zhang, 2010), and the truncated ℓ_1-function (Shen et al., 2012). Furthermore, we believe it would be both practicable and useful to extend these methods beyond the Gaussian assumption (as in for example Cai et al., 2011; Liu et al., 2012; Xue and Zou, 2012).

Table 2.1: Summary statistics reporting performance of the EM and one-step methods inferring graph structure for different networks. In each cell, the number before and after the slash correspond to the results using extended BIC and cross-validation, respectively.

	True networks Category-specific / Systemic	ρ	Method	EL	FL	FP(%)	FN(%)	HD (%)
$p = 30$	Chain / Chain	0	One step	0.9 /0.8	0.06 /0.05	20.9 /24.6	0.0 /0.0	20.9/24.6
		0	EM	**0.6/0.5**	**0.04/0.03**	**17.5/22.8**	**0.0/0.0**	**17.5/22.8**
		0.2	One step	0.9 /0.8	0.05 /0.05	21.4 /24.7	3.5 /2.5	24.9/27.2
		0.2	EM	**0.6/0.6**	**0.04/0.03**	**16.6/22.7**	**2.6/1.3**	**19.2/24.0**
		1	One step	1.1 /0.9	0.07 /0.05	14.0 /**23.0**	32.5 /21.1	46.5/44.1
		1	EM	**0.8/0.6**	**0.05/0.04**	**12.9**/25.2	**28.5/14.2**	**41.4/39.4**
	NN / Chain	0	One step	1.2 /0.9	0.07 /0.06	22.2 /**17.6**	27.0 /15.8	49.2/33.4
		0	EM	**0.8/0.6**	**0.05/0.04**	**15.3**/21.4	**17.4/6.8**	**32.7/28.2**
		0.2	One step	1.2 /0.8	0.06 /0.05	21.8 /**21.4**	42.8 /27.5	64.6/48.9
		0.2	EM	**0.8/0.6**	**0.05/0.04**	**16.0**/27.0	**34.9/17.0**	**50.9/44.0**
		1	One step	1.0 / 0.8	0.06 /0.05	9.1 /**16.5**	86.6 /63.4	95.7 / 79.9
		1	EM	**0.9/0.6**	**0.05/0.04**	**8.6** /29.4	**84.2/43.5**	**92.8/ 72.9**
	Chain / NN	0	One step	0.8 /0.8	0.04 /0.05	23.4 /24.7	**1.4**/3.3	24.8/28.0
		0	EM	**0.5/0.5**	**0.03/0.03**	**17.5/22.0**	1.8 /**3.1**	**19.3 / 25.1**
		0.2	One step	0.8 /0.8	0.05 /0.05	21.1 /**24.3**	9.3 /10.1	30.4 /34.4
		0.2	EM	**0.6/0.6**	**0.03/0.03**	**15.8**/24.8	**8.2/7.0**	**24.0/ 31.8**
		1	One step	1.0 /0.8	0.07 /0.05	10.7 /**21.8**	43.1 /30.4	53.8 / 52.2
		1	EM	**0.8/0.6**	**0.05/0.04**	**10.6**/24.9	**36.3/22.7**	**46.9/ 47.6**
	NN / NN	0	One step	1.0 /0.8	0.06 /0.05	17.9 /**16.0**	28.7 /20.8	46.6 / 36.8
		0	EM	**0.7/0.5**	**0.04/0.03**	**13.9** /20.4	**18.9/8.6**	**32.8 29.0**
		0.2	One step	1.0 /0.8	0.06 /0.05	16.9 /**19.4**	44.1 /31.8	61.0/51.2
		0.2	EM	**0.7/0.6**	**0.04/0.03**	**13.7** /24.7	**36.6/18.9**	**50.3/ 43.6**
		1	One step	1.0 /0.8	0.06 /0.05	**3.6** /**18.9**	87.3 /58.2	90.9 / 77.1
		1	EM	**0.9/0.6**	**0.05/0.04**	4.8 /25.9	**83.2/45.3**	**88.0/71.2**

Table 2.2: Summary statistics reporting performance of the EM and one-step methods inferring graph structure for different networks. In each cell, the number before and after the slash correspond to the results using extended BIC and cross-validation, respectively.

	True networks Category-specific / Systemic	ρ	Method	EL	FL	FP(%)	FN(%)	HD (%)
$p = 100$	Chain / Chain	0	One step	6.2 /4.9	0.13 /0.10	6.4 /12.1	0.0 /0.0	6.4/ 12.1
		0	EM	**3.6/2.9**	**0.08/0.06**	**5.0/9.2**	**0.0/0.0**	**5.0/9.2**
		0.2	One step	6.1 /4.6	0.13 /0.09	5.4 /12.6	7.3 /3.8	12.7 / 16.4
		0.2	EM	**3.6/2.9**	**0.07/0.05**	**4.8/9.9**	**6.9/3.0**	**11.7/12.9**
		1	One step	6.4 /4.6	0.12 /0.08	**2.5/12.5**	50.3 /29.1	52.8/ 41.6
		1	EM	**4.1/3.1**	**0.08/0.05**	3.9 /14.1	**41.3/21.4**	**45.2/35.5**
	NN / Chain	0	One step	5.0 /4.2	0.10 /0.08	**3.1/7.6**	36.7 /21.8	39.8/29.4
		0	EM	**3.7/3.1**	**0.07/0.06**	4.2 /7.8	**25.5/14.0**	**29.7 21.8**
		0.2	One step	5.5 /4.3	0.11 /0.08	**2.5/9.5**	59.1 /35.0	61.6/44.5
		0.2	EM	**4.3/3.2**	**0.09/0.06**	3.9 /10.7	**47.7/24.5**	**51.6 35.2**
		1	One step	4.3 /3.7	0.08 /0.07	**1.1 /5.2**	90.6 /76.6	91.7/81.8
		1	EM	**3.8/3.1**	**0.07/0.05**	1.8 /8.2	**87.1/66.2**	**88.9 / 74.4**
	Chain / NN	0	One step	4.5 /4.1	0.10 /0.08	5.4 /14.9	**5.4**/4.9	10.8/19.8
		0	EM	**3.0/2.8**	**0.06/0.05**	**4.6/8.7**	5.7 /**4.8**	**10.3/ 13.5**
		0.2	One step	4.9 /4.2	0.10 /0.08	4.7 /15.0	19.4 /12.8	24.1/27.8
		0.2	EM	**3.5/3.0**	**0.07/0.05**	**4.2/10.1**	**17.4/10.9**	**21.6/ 21.0**
		1	One step	5.0 /3.8	0.10 /0.07	**1.7**/12.6	66.6 /43.1	68.3/55.7
		1	EM	**3.9/3.0**	**0.07/0.05**	2.0 /**13.3**	**59.6/37.0**	**61.6/50.3**
	NN / NN	0	One step	4.3 /3.7	0.09 /0.07	**2.3**/7.1	43.7 /27.5	46.0/34.6
		0	EM	**3.2/2.7**	**0.06/0.05**	2.6 /**7.1**	**31.5/17.8**	**34.1/24.9**
		0.2	One step	4.9 /3.9	0.10 /0.08	**1.8/8.8**	66.4 /41.1	68.2/49.9
		0.2	EM	**3.9/3.1**	**0.08/0.05**	2.3 /10.4	**55.4/29.2**	**57.7/ 39.6**
		1	One step	4.2 /3.5	0.08 /0.07	**0.2/4.5**	96.3 /81.1	96.5 / 85.6
		1	EM	**3.9/3.1**	**0.07/0.06**	0.3 /6.8	**94.6/71.1**	**94.9/ 77.9**

Table 2.3: Summary statistics reporting performance of HL, JGL, one-step and the EM methods estimating aggregate network ,Ω_Y, under different simulation settings with dimension $p = 30$ and 100.

True networks Category-specific / Systemic	ρ	Method	p	EL	FL	p	EL	FL
Chain / Chain	0	HL	30	**0.277**	**0.013**	100	**1.032**	**0.015**
	0	JGL	30	1.184	0.086	100	3.093	0.066
	0	One step	30	0.795	0.049	100	4.863	0.095
	0	EM	30	0.521	0.031	100	2.940	0.058
	0.2	HL	30	**0.405**	**0.018**	100	**1.557**	**0.021**
	0.2	JGL	30	0.867	0.059	100	3.288	0.063
	0.2	One step	30	0.819	0.050	100	4.649	0.088
	0.2	EM	30	0.555	0.032	100	2.909	0.052
	1	HL	30	0.636	**0.033**	100	**2.370**	**0.035**
	1	JGL	30	0.947	0.058	100	3.658	0.063
	1	One step	30	0.875	0.051	100	4.550	0.082
	1	EM	30	**0.630**	0.035	100	3.132	0.048
NN / Chain	0	HL	30	1.185	0.063	100	5.914	0.096
	0	JGL	30	1.190	0.070	100	5.958	0.101
	0	One step	30	0.888	0.055	100	4.182	0.081
	0	EM	30	**0.629**	**0.037**	100	**3.120**	**0.058**
	0.2	HL	30	1.143	0.065	100	5.640	0.093
	0.2	JGL	30	1.209	0.081	100	5.724	0.101
	0.2	One step	30	0.818	0.050	100	4.292	0.084
	0.2	EM	30	**0.612**	**0.035**	100	**3.245**	**0.058**
	1	HL	30	0.978	0.057	100	4.057	0.070
	1	JGL	30	1.065	0.065	100	4.371	0.074
	1	One step	30	0.790	0.048	100	3.650	0.071
	1	EM	30	**0.640**	**0.036**	100	**3.091**	**0.054**
Chain / NN	0	HL	30	1.060	0.055	100	5.221	0.089
	0	JGL	30	1.085	0.068	100	4.930	0.088
	0	One step	30	0.760	0.045	100	4.141	0.079
	0	EM	30	**0.555**	**0.031**	100	**2.812**	**0.052**
	0.2	HL	30	1.034	0.055	100	4.794	0.075
	0.2	JGL	30	1.209	0.081	100	5.009	0.089
	0.2	One step	30	0.769	0.044	100	4.186	0.078
	0.2	EM	30	**0.572**	**0.031**	100	**3.021**	**0.052**
	1	HL	30	0.850	0.047	100	3.888	0.066
	1	JGL	30	1.065	0.065	100	4.319	0.089
	1	One step	30	0.810	0.050	100	3.794	0.070
	1	EM	30	**0.621**	**0.037**	100	**3.042**	**0.050**
NN / NN	0	HL	30	1.018	0.059	100	4.726	0.082
	0	JGL	30	1.085	0.068	100	4.520	0.074
	0	One step	30	0.780	0.048	100	3.664	0.070
	0	EM	30	**0.563**	**0.033**	100	**2.719**	**0.048**
	0.2	HL	30	0.957	0.053	100	4.690	0.081
	0.2	JGL	30	1.116	0.073	100	4.639	0.082
	0.2	One step	30	0.796	0.048	100	3.943	0.077
	0.2	EM	30	**0.597**	**0.034**	100	**3.085**	**0.053**
	1	HL	30	0.823	0.049	100	3.637	0.065
	1	JGL	30	0.980	0.060	100	3.738	0.064
	1	One step	30	0.760	0.047	100	3.527	0.069
	1	EM	30	**0.643**	**0.038**	100	**3.112**	**0.057**

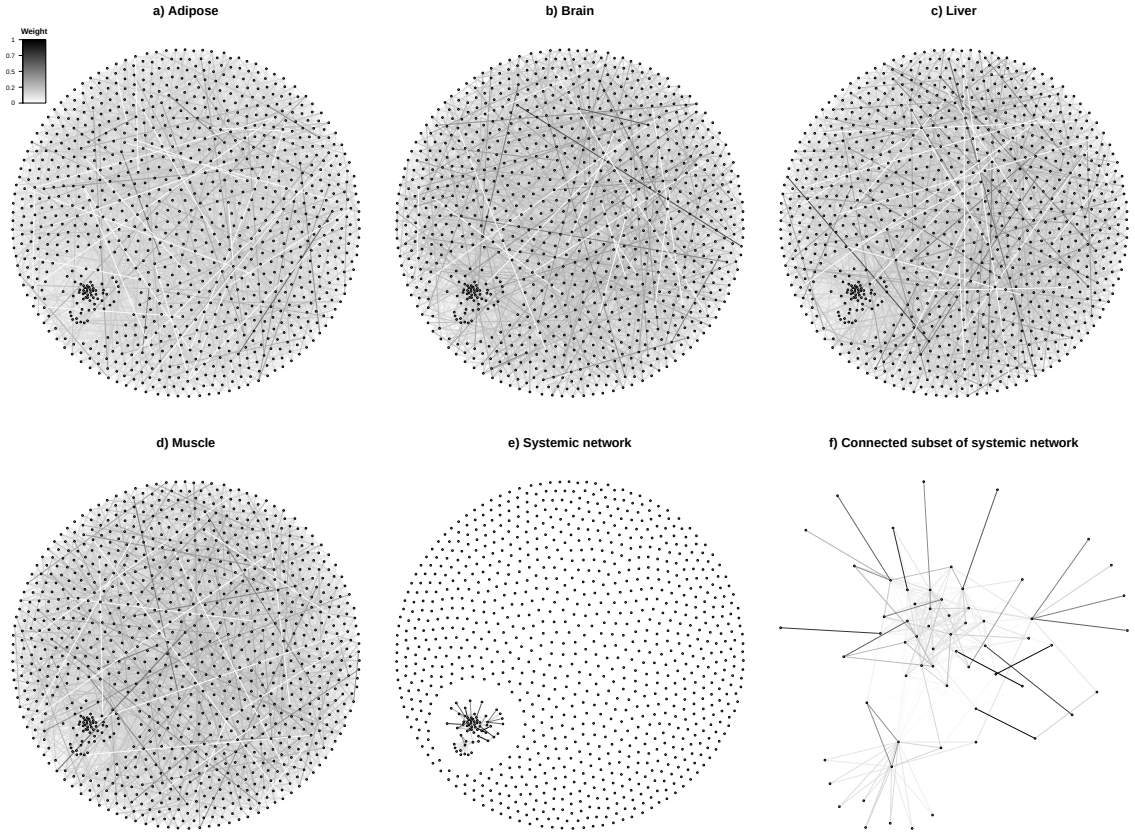

Figure 2.6: Topology of co-expression networks inferred by the EM method applied to measurements of the 1000 genes with highest within-tissue variance in a population of F_2 mice. Panels a), b), c) and d) display the category-specific networks estimated for adipose, hypothalamus, liver and muscle tissues respectively. Panel e) shows the structure of the estimated systemic network, describing across-tissue dependencies, with panel f) showing a zoomed-in view of the connected subset of nodes in this graph.

2.7 Appendix

2.7.1 Derivation of likelihood for y

We first state Sylvester's determinant theorem which is required for this derivation.

Lemma 2.1 (Sylvester theorem)**.** *If A, B are matrices of size $p \times n$ and $n \times p$ respectively, then*

$$det(I_p + AB) = det(I_n + BA),$$

where I_a is the identity matrix of order a.

We would like to derive the expression of Ω_Y using $\{\Omega_k\}_{k=0}^K$. In our setting, Y follows a $Kp-$variate Gaussian distribution with mean 0 and covariance matrix $\Sigma_Y = \{\Sigma_{Y(l,m)}\}_{1\leq l,m\leq K}$; thus $f(Y) \propto \exp(Y^{\mathrm{T}}\Omega_Y Y)$. In addition, we can also derive $f(Y)$ from joint probability $f(Y, Z)$ by integrating out Z. We can write $f(Y)$ as follows,

$$\begin{aligned} f(Y) &= \int f(Y \mid Z) f(Z) \mathrm{d}Z \\ &\propto \int \exp\left[\sum_{k=1}^{K}\left\{(Y_k - Z)^{\mathrm{T}}\Omega_k(Y_k - Z)\right\} + Z^{\mathrm{T}}\Omega_0 Z\right] \mathrm{d}Z. \end{aligned}$$

We then expand the formula and get

$$\begin{aligned} f(Y) &= \exp\left\{\sum_{k=1}^{K}(Y_k^{\mathrm{T}}\Omega_k Y_k)\right\} \int \exp\left[Z^{\mathrm{T}}\left(\sum_{k=0}^{K}\Omega_k\right)Z - 2\left\{\sum_{k=1}^{K}(Y_k^{\mathrm{T}}\Omega_k)\right\}Z\right] \mathrm{d}Z \\ &= \exp\left\{\sum_{k=1}^{K}(Y_k^{\mathrm{T}}\Omega_k Y_k)\right\} \int \exp\left(Z^{\mathrm{T}}AZ - 2c^{\mathrm{T}}Z\right) \mathrm{d}Z \\ &= \exp\left\{\sum_{k=1}^{K}(Y_k^{\mathrm{T}}\Omega_k Y_k)\right\} \exp(-c^{\mathrm{T}}A^{-1}c) \int \exp\left\{(AZ - c)^{\mathrm{T}}A^{-1}(AZ - c)\right\} \mathrm{d}Z \\ &= \exp\left\{\sum_{k=1}^{K}(Y_k^{\mathrm{T}}\Omega_k Y_k)\right\} \exp(-c^{\mathrm{T}}A^{-1}c) \int \exp\left\{(Z - A^{-1}c)^{\mathrm{T}}A(Z - A^{-1}c)\right\} \mathrm{d}Z \\ &\propto \exp\left[\sum_{k=1}^{K}(Y_k^{\mathrm{T}}\Omega_k Y_k) - \{\sum_{k=1}^{K}(Y_k^{\mathrm{T}}\Omega_k)\}A^{-1}\{\sum_{k=1}^{K}(\Omega_k Y_k)\}\right] \\ &= \exp\left\{Y^{\mathrm{T}}\left(\{{}_d\Omega_k\}_{1\leq k\leq K} - \{\Omega_l A^{-1}\Omega_k\}_{1\leq l,k\leq K}\right)Y\right\} \\ &= \exp(Y^{\mathrm{T}}\Omega_Y Y)\,, \end{aligned}$$

where $A = \sum_{k=0}^{K} \Omega_k$ and $c = \sum_{k=1}^{K} \Omega_k Y_k$. We denote $\{\Omega_l A^{-1} \Omega_k\}_{1 \le l,k \le K}$ as a block matrix in which the (l,k)th block is $\Omega_l A^{-1} \Omega_k$.

Thus we have $y \sim \mathcal{N}(0, \left[\{{}_d\Omega_k\}_{k=1}^{K} - \{\Omega_l A^{-1} \Omega_k\}_{1 \le l,k \le K}\right]^{-1})$ and $\Omega_y = \{{}_d\Omega_k\}_{1 \le k \le K} - \{\Omega_l A^{-1} \Omega_k\}_{1 \le l,k \le K}$. Next, we derive the expression for $\det(\Omega_Y)$. We know that

$$\begin{aligned}
\Sigma_Y =& \{{}_d\,\Sigma_k\}_{1 \le k \le K} + \begin{pmatrix} I \\ \vdots \\ I \end{pmatrix} \begin{pmatrix} \Sigma_0 & \cdots & \Sigma_0 \end{pmatrix} \\
=& \{{}_d\,\Sigma_k\}_{1 \le k \le K} \left\{ I_{Kp} + \{{}_d\,\Omega_k\}_{1 \le k \le K} \begin{pmatrix} I \\ \vdots \\ I \end{pmatrix} \begin{pmatrix} \Sigma_0 & \cdots & \Sigma_0 \end{pmatrix} \right\},
\end{aligned}$$

where I and I_{Kp} are $p \times p$ and $Kp \times Kp$ identity matrices respectively. By Lemma 2.1, we have

$$\begin{aligned}
\det(\Sigma_Y) &= \left\{ \prod_{k=1}^{K} \det(\Sigma_k) \right\} \det \left\{ I + \begin{pmatrix} \Sigma_0 & \cdots & \Sigma_0 \end{pmatrix} \{{}_d\Omega_k\}_{1 \le k \le K} \begin{pmatrix} I \\ \vdots \\ I \end{pmatrix} \right\} \\
&= \left\{ \prod_{k=1}^{K} \det(\Sigma_k) \right\} \det \left(I + \Sigma_0 \sum_{k=1}^{K} \Omega_k \right) \\
&= \left\{ \prod_{k=1}^{K} \det(\Sigma_k) \right\} \det \left(\Sigma_0 \Omega_0 + \Sigma_0 \sum_{k=1}^{K} \Omega_k \right) \\
&= \left\{ \prod_{k=0}^{K} \det(\Sigma_k) \right\} \det(A).
\end{aligned}$$

Therefore, we have $\log\{\det(\Omega_Y)\} = \sum_{k=0}^{K} \log\{\det(\Omega_k)\} - \log\{\det(A)\}$. Combining previous results, we can write the log-likelihood as

$$\begin{aligned}
\mathcal{L}(\Omega_Y) =& -\frac{npK}{2} \log(2\pi) + \frac{n}{2} \big[\log\{\det(\Omega_Y)\} - \mathrm{tr}\big(\hat{\Sigma}_Y \Omega_Y\big) \big] \\
=& -\frac{npK}{2} \log(2\pi) + \frac{n}{2} \log\{\det(\Omega_Y)\} - \frac{n}{2} \mathrm{tr}\big(\hat{\Sigma}_Y \Omega_Y\big) \\
=& -\frac{npK}{2} \log(2\pi) + \frac{n}{2} \Big[\sum_{k=0}^{K} \log\det(\Omega_k) - \log\{\det(A)\} \Big] \\
& - \frac{n}{2} \mathrm{tr}\Big(\hat{\Sigma}_Y \Big[\{{}_d\Omega_k\}_{1 \le k \le K} - \{\Omega_l A^{-1} \Omega_k\}_{1 \le l,k \le K} \Big] \Big)
\end{aligned}$$

$$
\begin{aligned}
&= -\frac{npK}{2}\log(2\pi) + \frac{n}{2}\Big[\sum_{k=1}^{K}\log\det(\Omega_k) + \log\{\det(\Omega_0)\} - \log\{\det(A)\}\Big] \\
&\quad - \frac{n}{2}\mathrm{tr}\Big(\hat{\Sigma}_Y\{{}_d\Omega_k\}_{1\le k\le K} - \hat{\Sigma}_Y\{\Omega_l A^{-1}\Omega_k\}_{1\le l,k\le K}\Big) \\
&= -\frac{npK}{2}\log(2\pi) + \frac{n}{2}\sum_{k=1}^{K}\log\{\det(\Omega_k)\} + \frac{n}{2}\Big[\log\{\det(\Omega_0)\} - \log\{\det(A)\}\Big] \\
&\quad - \frac{n}{2}\sum_{k=1}^{K}\mathrm{tr}\Big(\hat{\Sigma}_{Y(k,k)}\Omega_k\Big) + \frac{n}{2}\mathrm{tr}\Big(\hat{\Sigma}_Y\begin{pmatrix}\Omega_1 & \cdots & \Omega_K\end{pmatrix}^{\mathrm{T}} A^{-1}\begin{pmatrix}\Omega_1 & \cdots & \Omega_K\end{pmatrix}\Big) \\
&= -\frac{npK}{2}\log(2\pi) + \frac{n}{2}\mathrm{tr}\Big(\begin{pmatrix}\Omega_1 & \cdots & \Omega_K\end{pmatrix}\hat{\Sigma}_Y\begin{pmatrix}\Omega_1 & \cdots & \Omega_K\end{pmatrix}^{\mathrm{T}} A^{-1}\Big) \\
&\quad + \frac{n}{2}\big[\log\{\det(\Omega_0)\} - \log\{\det(A)\}\big] + \frac{n}{2}\sum_{k=1}^{K}\Big[\log\{\det(\Omega_k)\} - \mathrm{tr}\Big(\hat{\Sigma}_{Y(k,k)}\Omega_k\Big)\Big] \\
&= -\frac{npK}{2}\log(2\pi) + \frac{n}{2}\sum_{l,m=1}^{K}\mathrm{tr}\left(\Omega_l\hat{\Sigma}_{Y(l,m)}\Omega_m A^{-1}\right) + \frac{n}{2}\log\{\det(\Omega_0)\} \\
&\quad - \frac{n}{2}\log\{\det(A)\} + \frac{n}{2}\sum_{k=1}^{K}\Big[\log\{\det(\Omega_k)\} - \mathrm{tr}(\hat{\Sigma}_{Y(k,k)}\Omega_k)\Big].
\end{aligned}
$$

2.7.2 Proof of Identifiability

To demonstrate identifiability of our model, it is sufficient to show the parameters Ω_k are identifiable for all $k = 0, \ldots, K$. To that end, we decompose Y_k in two different ways as follows

$$Y_k = X_k - U + Z + U = X_k^* + Z^* \quad (k = 1, ..., K),$$

where U is a p-dim of random vector. With $U \neq 0$, we have nonunique decompositions of Y_k. Under the model assumption, the resulting X_k^* and Z^* need to satisfy

$$\mathrm{cov}(X_l^*, X_m^*) = 0 \quad (1 \le l, m \le K), \tag{2.13}$$

$$\mathrm{cov}(X_l^*, Z^*) = 0 \quad (l = 1, \ldots, K). \tag{2.14}$$

Expanding (2.13), we have

$$
\begin{aligned}
\mathrm{cov}(X_l^*, X_m^*) &= \mathrm{cov}(X_l, X_m) + \mathrm{var}(U) - \mathrm{cov}(X_l, U) - \mathrm{cov}(X_m, U) \\
&= 0 + \mathrm{var}(U) - \mathrm{cov}(X_i, U) - \mathrm{cov}(X_j, U) = 0,
\end{aligned}
$$

which implies that

$$\text{var}(U) = \text{cov}(X_l, U) + \text{cov}(X_m, U). \tag{2.15}$$

Similarly, from (2.14), we have

$$\begin{aligned}
\text{cov}(X_l^*, Z^*) &= \text{cov}(X_l - U, Z + U) \\
&= \text{cov}(X_k, Z) - \text{var}(U) - \text{cov}(U, Z) + \text{cov}(U, X_l) \\
&= 0 - \text{var}(U) - \text{cov}(U, Z) + \text{cov}(U, X_l) = 0,
\end{aligned}$$

which implies that

$$\text{var}(U) = -\text{cov}(U, Z) + \text{cov}(U, X_l). \tag{2.16}$$

Since (2.16) hod for any l, we have

$$\text{cov}(U, X_l) = \text{cov}(U, X_m) \quad (1 \leq l, m \leq K). \tag{2.17}$$

Combining (2.15), (2.16) and (2.17), we can show

$$\text{cov}(U, X_k) = -\text{cov}(U, Z) \tag{2.18}$$

$$\text{var}(U) = -2\,\text{cov}(U, Z) = 2\,\text{cov}(U, X_l) \quad (1 \leq l \leq K). \tag{2.19}$$

From (2.18) and (2.19) we can further show that

$$\begin{aligned}
\text{var}(X_l^*) &= \text{var}(X_l - U) \\
&= \text{var}(X_l) + \text{var}(U) - 2\,\text{cov}(U, X_l) = \text{var}(X_l) \\
\text{var}(Z^*) &= \text{var}(Z + U) \\
&= \text{var}(Z) + \text{var}(U) - 2\,\text{cov}(U, Z) = \text{var}(Z).
\end{aligned}$$

Therefore, for different decompositions of Y_l the resulting $\text{var}(Z)$ and $\text{var}(X_l)$ remain the same. Consequently, our model is identifiable.

2.7.3 Proof of Proposition 2.1

We divide the proof into two parts. For part I, we first prove that the penalized log-likelihood is bounded, and for part II, we will show that the penalized log-likelihood does not decreases for each step of the graphic EM algorithm.

As shown in (2.4), the log-likelihood can be expressed as

$$\begin{aligned}\mathcal{L}(\{\Omega_k\}_{k=0}^K) \propto & \sum_{k=1}^K \Big[\log\{\det(\Omega_k)\} - \mathrm{tr}(\hat{\Sigma}_{Y(k,k)}\Omega_k) \Big] + \log\{\det(\Omega_0)\} \\ & - \log\{\det(A)\} + \sum_{l,m=1}^K \mathrm{tr}\left(\Omega_l \hat{\Sigma}_{Y(l,m)} \Omega_m A^{-1}\right) \\ = & \sum_{k=0}^K \log\{\det(\Omega_k)\} - \log\{\det(A)\} - \sum_{k=1}^K \mathrm{tr}\big(\hat{\Sigma}_{Y(k,k)}\Omega_k\big) \\ & + \sum_{l,m=1}^K \mathrm{tr}\left(\Omega_l \hat{\Sigma}_{Y(l,m)} \Omega_m A^{-1}\right).\end{aligned}$$

If $\lambda_1 > 0$ and $\lambda_2 > 0$, by Lagrangian duality the problem (2.5) is equivalent to the following constrained optimization problem:

$$\begin{aligned}\max\Bigg\{ & \sum_{k=0}^K \log\{\det(\Omega_k)\} - \log\{\det(A)\} - \sum_{k=1}^K \mathrm{tr}\big(\hat{\Sigma}_{Y(k,k)}\Omega_k\big) \\ & + \sum_{l,m=1}^K \mathrm{tr}\left(\Omega_l \hat{\Sigma}_{Y(l,m)} \Omega_m A^{-1}\right) \Bigg\}, \end{aligned} \tag{2.20}$$

subject to $\Omega_k \succ 0$ and $|\Omega_k^-|_1 \leq C(\lambda_1, \lambda_2)$ for $k = 0, \ldots, K$ and some $C(\lambda_1, \lambda_2) < \infty$. Here Ω^- represents the off-diagonal entries of Ω. Since the $\{\Omega_k^-\}_{k=0}^K$ are bounded, the potential problem comes from the behavior of the diagonal entries which can grow to infinity. Because of the positive-definite requirement, diagonal entries of $\{\Omega_k\}_{k=0}^K$ have to be positive. After some algebra, (2.20) becomes

$$\begin{aligned}& \sum_{k=0}^K \log\{\det(\Omega_k)\} - \log\{\det(A)\} - \sum_{k=1}^K \mathrm{tr}\big(\hat{\Sigma}_{Y(k,k)} A A^{-1} \Omega_k\big) \\ & + \sum_{l,m=1}^K \mathrm{tr}\left(\hat{\Sigma}_{Y(l,m)} \Omega_m A^{-1} \Omega_l\right) \\ & = \sum_{k=0}^K \log\{\det(\Omega_k)\} - \log\{\det(A)\} - \sum_{k=1}^K \mathrm{tr}(\hat{\Sigma}_{Y(k,k)} \Omega_0 A^{-1} \Omega_k)\end{aligned}$$

$$
\begin{aligned}
&- \sum_{l \neq m} \operatorname{tr}\big\{(\hat{\Sigma}_{Y(m,m)} - \hat{\Sigma}_{Y(m,l)})\Omega_l A^{-1}\Omega_m\big\} \\
&= \sum_{k=0}^{K} \log\{\det(\Omega_k)\} - \log\{\det(A)\} - \sum_{k=1}^{K} \operatorname{tr}(\hat{\Sigma}_{Y(k,k)}\Omega_0 A^{-1}\Omega_k) \\
&- \sum_{l > m \geq 1} \operatorname{tr}(M_{(l,m)}\Omega_l A^{-1}\Omega_m), \qquad (2.21)
\end{aligned}
$$

where $M_{(l,m)} = \{\hat{\Sigma}_{Y(l,l)} + \hat{\Sigma}_{Y(m,m)} - 2\hat{\Sigma}_{Y(m,l)}\}$. The equality in (2.21) comes from the fact that

$$
\begin{aligned}
&\operatorname{tr}\big\{(\hat{\Sigma}_{Y(m,m)} - \hat{\Sigma}_{Y(m,l)})\Omega_l A^{-1}\Omega_m\big\} = \operatorname{tr}\big[\big\{(\hat{\Sigma}_{Y(m,m)} - \hat{\Sigma}_{Y(m,l)})\Omega_l A^{-1}\Omega_m\big\}^{\mathrm{T}}\big] \\
&= \operatorname{tr}\big\{\Omega_m A^{-1}\Omega_l(\hat{\Sigma}_{Y(m,m)} - \hat{\Sigma}^{\mathrm{T}}_{Y(m,l)})\big\} = \operatorname{tr}\big\{(\hat{\Sigma}_{Y(m,m)} - \hat{\Sigma}_{Y(l,m)})\Omega_m A^{-1}\Omega_l\big\}.
\end{aligned}
$$

Since all Ω_j are positive definite and $|\Omega_j^-|_1$ are bounded for $j = 0, \ldots, K$, we can decompose them into $\Omega_j = B_j + D_j$, where B_j are matrices with bounded diagonal entries and are invertible, namely $0 \leq \tau_3 \leq \phi_{min}(B_j) \leq \phi_{max}(B_j) \leq \tau_4$, while D_j are diagonal matrices with all entries are greater than some positive number ϵ and possibly grow to infinity, in other words, $0 \leq \|D_j^{-1}\| \leq 1/\epsilon$. We also define $B_A = \sum_{k=0}^{K} B_k$ and $D_A = \sum_{k=0}^{K} D_k$; and by Weyl's inequality

$$
\begin{aligned}
(K+1)\tau_3 \leq \phi_{min}(B_A) \leq \phi_{max}(B_A) \leq (K+1)\tau_4, \\
0 < (K+1)\epsilon \leq \phi_{min}(D_A), \\
\phi_{max}(D_A^{-1}) \leq \frac{1}{(K+1)\epsilon},
\end{aligned}
$$

Then we can consider four different cases:

Case one: $|D_j|_1$ are bounded for all $j = 0, \ldots, K$.

In this case, $\det(\Omega_j)$ and $\|\Omega_j\|_\infty$ are both bounded above for ($j \in \{0, \ldots, K\}$; thus the function in (2.21) is also bounded above.

Case two: All $|D_j|_1$ are bounded except D_l.

In this case, we only need to control the behavior of the following terms

$$
\log\{\det(\Omega_l)\} - \log\{\det(A)\} - \sum_{K \geq k > l} \operatorname{tr}(M_{(k,l)}\Omega_k A^{-1}\Omega_l) - \sum_{l > m \geq 1} \operatorname{tr}(M_{(l,m)}\Omega_l A^{-1}\Omega_m)
$$

$$
- \operatorname{tr}(\hat{\Sigma}_{Y(l,l)}\Omega_0 A^{-1}\Omega_l)
$$

$$= \Big[\log\{\det(\Omega_l)\} - \log\{\det(A)\} \Big] - \sum_{K \geq k > l} \operatorname{tr}\big\{ M_{(k,l)} \Omega_k (B_A + D_A)^{-1} (B_l + D_l) \big\}$$

$$- \sum_{l > m \geq 1} \operatorname{tr}\big\{ M_{(l,m)} (B_l + D_l)(B_A + D_A)^{-1} \Omega_m \big\} - \operatorname{tr}\big\{ \hat{\Sigma}_{Y(l,l)} \Omega_0 (B_A + D_A)^{-1} (B_l + D_l) \big\}$$

$$= \mathrm{I} + \mathrm{II} + \mathrm{III} + \mathrm{IV}.$$

We first want to bound term I which is $\log\{\det(\Omega_l)\} - \log\{\det(A)\}$. Since $A = \sum_{k=0}^{K} \Omega_k$ and all Ω_k are positive definite, by Minkowski determinant theorem it follows that

$$\begin{aligned} \det(A) &\geq \{\det(\sum_{k \neq l} \Omega_k)^{1/p} + \det(\Omega_l)^{1/p}\}^p \\ &\geq \{\det(\Omega_l)^{1/p}\}^p = \det(\Omega_l). \end{aligned}$$

Therefore, we have $\mathrm{I} = \log\{\det(\Omega_l)\} - \log\{\det(A)\} < 0$.

To bound II and III, using Woodbury matrix identity we know

$$\begin{aligned} \|\Omega_l A^{-1}\| &= (B_l + D_l)(B_A + D_A)^{-1} = (B_l + D_l)\{D_A^{-1} - D_A^{-1}(D_A^{-1} + B_A^{-1})^{-1} D_A^{-1}\} \\ &= B_l D_A^{-1} + D_l D_A^{-1} - B_l D_A^{-1}(D_A^{-1} + B_A^{-1})^{-1} D_A^{-1} \\ &\quad - D_l D_A^{-1}(D_A^{-1} + B_A^{-1})^{-1} D_A^{-1}; \end{aligned} \tag{2.22}$$

$$\begin{aligned} \|A^{-1}\Omega_l\| &= (B_A + D_A)^{-1}(B_l + D_l) = \{D_A^{-1} - D_A^{-1}(D_A^{-1} + B_A^{-1})^{-1} D_A^{-1}\}(B_l + D_l) \\ &= D_A^{-1} B_l + D_A^{-1} D_l - D_A^{-1}(D_A^{-1} + B_A^{-1})^{-1} D_A^{-1} B_l \\ &\quad - D_A^{-1}(D_A^{-1} + B_A^{-1})^{-1} D_A^{-1} D_l. \end{aligned} \tag{2.23}$$

We also know $D_A = \sum_{k=0}^{K} (D_k)$ and $D_A^{-1} = \Big\{ d(\sum_{k=0}^{K} D_{k(i,i)})^{-1} \Big\}$. We want to bound the spectral normal of (2.22) and (2.23). In order to do so, we first show

$$\|D_A^{-1}\| = \Big\| \Big\{ \frac{1}{d \sum_{k=0}^{K} D_{k(i,i)}} \Big\} \Big\| \leq \frac{1}{(K+1)\epsilon}; \tag{2.24}$$

$$\|B_l D_A^{-1}\| \leq \|B_l\| \, \|D_A^{-1}\| \leq \frac{\tau_4}{(K+1)\epsilon}; \tag{2.25}$$

$$\|D_A^{-1} B_l\| \leq \|B_l\| \, \|D_A^{-1}\| \leq \frac{\tau_4}{(K+1)\epsilon}; \tag{2.26}$$

$$\|D_l D_A^{-1}\| = \|D_A^{-1} D_l\| = \Big\| \Big\{ \frac{D_{l(i,i)}}{d D_{A(i,i)}} \Big\} \Big\| = \Big\| \Big\{ \frac{D_{l(i,i)}}{d \sum_{k=0}^{K} D_{k(i,i)}} \Big\} \Big\| \leq 1. \tag{2.27}$$

By Weyl's inequality,

$$\begin{aligned}\|(B_A^{-1}+D_A^{-1})^{-1}\| &= \frac{1}{\phi_{\min}(B_A^{-1}+D_A^{-1})} \leq \frac{1}{\phi_{\min}(B_A^{-1})+\phi_{\min}(D_A^{-1})} \\ &\leq \frac{1}{\phi_{\min}(B_A^{-1})} = \phi_{\max}(B_A) \leq (K+1)\tau_4 \end{aligned} \tag{2.28}$$

$$\|B_l D_A^{-1}(B_A^{-1}+D_A^{-1})^{-1}D_A^{-1}\| \leq \|B_l\|\,\|D_A^{-1}\|^2\,\|(B_A^{-1}+D_A^{-1})^{-1}\| \leq \frac{\tau_4^2}{(K+1)\epsilon^2} \tag{2.29}$$

$$\|D_A^{-1}(B_A^{-1}+D_A^{-1})^{-1}D_A^{-1}B_l\| \leq \|B_l\|\,\|D_A^{-1}\|^2\,\|(B_A^{-1}+D_A^{-1})^{-1}\| \leq \frac{\tau_4^2}{(K+1)\epsilon^2} \tag{2.30}$$

$$\|D_l D_A^{-1}(B_A^{-1}+D_A^{-1})^{-1}D_A^{-1}\| \leq \|D_l D_A^{-1}\|\,\|(B_A^{-1}+D_A^{-1})^{-1}\|\,\|D_A^{-1}\| \leq \frac{\tau_4}{\epsilon} \tag{2.31}$$

$$\|D_A^{-1}(B_A^{-1}+D_A^{-1})^{-1}D_A^{-1}D_l\| \leq \|D_A^{-1}D_l\|\,\|(B_A^{-1}+D_A^{-1})^{-1}\|\,\|D_A^{-1}\| \leq \frac{\tau_4}{\epsilon}. \tag{2.32}$$

Combining (2.24)–(2.32), we show that the spectral norm of (2.22) and (2.23) are bounded, namely

$$\begin{aligned}\|\Omega_l A^{-1}\| &= \|(B_l+D_l)(B_A+D_A)^{-1}\| \\ &= \|B_l D_A^{-1} + D_l D_A^{-1} - B_l D_A^{-1}(D_A^{-1}+B_A^{-1})^{-1}D_A^{-1} - D_l D_A^{-1}(D_A^{-1}+B_A^{-1})^{-1}D_A^{-1}\| \\ &\leq \|B_l D_A^{-1}\| + \|D_l D_A^{-1}\| + \|B_l D_A^{-1}(D_A^{-1}+B_A^{-1})^{-1}D_A^{-1}\| \\ &\quad + \|D_l D_A^{-1}(D_A^{-1}+B_A^{-1})^{-1}D_A^{-1}\| \\ &\leq \frac{\tau_4}{(K+1)\epsilon} + 1 + \frac{\tau_4^2}{(K+1)\epsilon^2} + \frac{\tau_4}{\epsilon} \\ &\leq \frac{(K+2)\epsilon\tau_4 + (K+1)\epsilon^2 + \tau_4^2}{(K+1)\epsilon^2} < \infty. \end{aligned} \tag{2.33}$$

$$\begin{aligned}\|A^{-1}\Omega_l\| &= \|(B_A+D_A)^{-1}(B_l+D_l)\| \\ &= \|D_A^{-1}B_l + D_A^{-1}D_l - D_A^{-1}(D_A^{-1}+B_A^{-1})^{-1}D_A^{-1}B_l - D_A^{-1}(D_A^{-1}+B_A^{-1})^{-1}D_A^{-1}D_l\| \\ &\leq \|D_A^{-1}B_l\| + \|D_A^{-1}D_l\| + \|D_A^{-1}(D_A^{-1}+B_A^{-1})^{-1}D_A^{-1}B_l\| \\ &\quad + \|D_A^{-1}(D_A^{-1}+B_A^{-1})^{-1}D_A^{-1}D_l\| \\ &\leq \frac{\tau_4}{(K+1)\epsilon} + 1 + \frac{\tau_4^2}{(K+1)\epsilon^2} + \frac{\tau_4}{\epsilon} \\ &\leq \frac{(K+2)\epsilon\tau_4 + (K+1)\epsilon^2 + \tau_4^2}{(K+1)\epsilon^2} < \infty. \end{aligned} \tag{2.34}$$

Since $M_{(l,k)}$ and $M_{(k,l)}$ only depends on the value of $\hat{\Sigma}_y$ which is calculated from the data, they are bounded above for any $k \neq l$. Based on the assumption that $|\Omega_k|_1$ is bounded

for any $k \neq l$, we can bound $\|\Omega_k\|$ using the fact $\|\Omega_k\| < p\|\Omega_k\|_\infty < p|\Omega_k|_1 < \infty$. We can show that

$$\begin{aligned}
\text{II} &= \sum_{K\geq k\geq l} \Big\{ -\operatorname{tr}\big(M_{(k,l)}\Omega_k A^{-1}\Omega_l\big)\Big\} \\
&= \sum_{K\geq k\geq l}\sum_{j=1}^{p} \Big\{ -\big(M_{(k,l)}\Omega_k A^{-1}\Omega_l\big)_{(j,j)}\Big\} \leq \sum_{K\geq k\geq l}\Big(p\|M_{(k,l)}\Omega_k A^{-1}\Omega_l\|\Big) \qquad (2.35)\\
&\leq \sum_{K\geq k\geq l}\Big(p\|M_{(l,k)}\|\,\|\Omega_k\|\,\|A^{-1}\Omega_l\|\Big) < \infty,
\end{aligned}$$

where the inequality of (2.35) is due to Lemma 2.5; in particular for real matrix B, we have $\max_{i,j}|B_{(i,j)}| \leq \|B\|$. Similarly, we can prove that the term III is also bounded above.

Since $\|\Omega_0\|$ is bounded by assumption, we can bound term IV as follows:

$$\begin{aligned}
\text{IV} &= -\operatorname{tr}\big(\hat{\Sigma}_{Y(l,l)}\Omega_0 A^{-1}\Omega_l\big) \\
&= \sum_{j=1}^{p}\Big\{ -\big(\hat{\Sigma}_{Y(l,l)}\Omega_0 A^{-1}\Omega_l\big)_{(j,j)}\Big\} \leq p\|\hat{\Sigma}_{Y(l,l)}\Omega_0 A^{-1}\Omega_l\| \qquad (2.36)\\
&\leq p\|\hat{\Sigma}_{Y(l,l)}\|\,\|A^{-1}\Omega_l\|\,\|\Omega_0\| < \infty.
\end{aligned}$$

Thus the log-likelihood in (2.21) is bounded above in this Case.

Case three: $|\Omega_0|_1$ is the only matrix not bounded. In this case, following the same argument as (2.22) in Case two, we can get $|\Omega_0 A^{-1}|_1 < \infty$. Combining with the fact that $\log\{\det(\Omega_0)\} - \log\{\det(A)\} < 0$, we prove that the log likelihood in (2.21) is also bounded in this setting.

Case four: At least $|\Omega_r|_1$ and $|\Omega_s|_1$ are not bounded, namely $|D_r|_1$ and $|D_s|_1$ are not bounded with some rate going to infinity. Without lost of generality, we assume $|D_r|_1$ and $|D_s|_1$ have fastest rate going to infinity.

By the Hadamard's inequality (Horn and Johnson, 1985), we have $\sum_{i=1}^{p}(\log\Omega_{k(i,i)}) \geq \log\{\det(\Omega_k)\}$, and notice that $\sum_{i=1}^{p}(\log\Omega_{k(i,i)})$ has the same rate go to infinity as $\sum_{i=1}^{p}(\log D_{k(i,i)})$, since $\Omega_{k(i,i)} = B_{k(i,i)} + D_{k(i,i)}$ with $B_{k(i,i)}$ being bounded. Thus the order of log likelihood in (2.21) is equivalent to the order of

$$\sum_{k=0}^{K}\sum_{i=1}^{p}\{\log(D_{k(i,i)})\} - \log\{\det(A)\} - \sum_{l>m\geq 1}\big\{\operatorname{tr}(M_{(l,m)}\Omega_l A^{-1}\Omega_m)\big\}$$

$$- \sum_{k=1}^{K} \big\{ \mathrm{tr}(\hat{\Sigma}_{Y(k,k)} \Omega_0 A^{-1} \Omega_k) \big\}. \tag{2.37}$$

Using Minkowski determinant theorem, it follows that

$$\begin{aligned} \det(A) = \det(B_A + D_A) &\geq \{\det(D_A)^{1/p} + \det(B_A)^{1/p}\}^p \\ &\geq \{\det(D_A)^{1/p}\}^p = \det(D_A) = \det(\sum_{k=0}^{K} D_k); \end{aligned}$$

using Woodbury matrix identity we also know that $A^{-1} = (B_A + D_A)^{-1} = D_A^{-1} - D_A^{-1}(D_A^{-1} + B_A^{-1})^{-1} D_A^{-1}$; by definition we have $\Omega_l = B_l + D_l$ and $\Omega_m = B_m + D_m$. Combining these results, we know (2.37) is bounded by:

$$\begin{aligned} &\sum_{k=0}^{K} \sum_{i=1}^{p} \{\log(D_{k(i,i)})\} - \log\big\{\det(\sum_{k=0}^{K} D_k)\big\} - \sum_{l>m\geq 1} \big\{\mathrm{tr}(M_{(l,m)} \Omega_l A^{-1} \Omega_m)\big\} \\ &\quad - \sum_{k=1}^{K} \big\{\mathrm{tr}(\hat{\Sigma}_{Y(k,k)} \Omega_0 A^{-1} \Omega_k)\big\} \\ &= \sum_{k=0}^{K} \sum_{i=1}^{p} \Big\{ \log(D_{k(i,i)}) \Big\} - \log\big\{\det(\sum_{k=0}^{K} D_k)\big\} - \sum_{l>m\geq 1} \big\{\mathrm{tr}(M_{(l,m)} D_l D_A^{-1} D_m)\big\} \\ &\quad + \sum_{l>m\geq 1} \Big[\mathrm{tr}\big\{M_{(l,m)} D_l D_A^{-1} (D_A^{-1} + B_A^{-1})^{-1} D_A^{-1} D_m\big\}\Big] \\ &\quad - \sum_{l>m\geq 1} \big\{\mathrm{tr}(M_{(l,m)} B_l A^{-1} \Omega_m)\big\} - \sum_{l>m\geq 1} \big\{\mathrm{tr}(M_{(l,m)} D_l A^{-1} B_m)\big\} \\ &\quad - \sum_{k=1}^{K} \big\{\mathrm{tr}(\hat{\Sigma}_{Y(k,k)} D_0 D_A^{-1} D_k)\big\} + \sum_{k=1}^{K} \Big[\mathrm{tr}\big\{\hat{\Sigma}_{Y(k,k)} D_0 D_A^{-1} (D_A^{-1} + B_A^{-1})^{-1} D_A^{-1} D_k\big\}\Big] \\ &\quad - \sum_{k=1}^{K} \big\{\mathrm{tr}(\hat{\Sigma}_{Y(k,k)} B_0 A^{-1} \Omega_k)\big\} - \sum_{k=1}^{K} \big\{\mathrm{tr}(\hat{\Sigma}_{Y(k,k)} D_0 A^{-1} B_k)\big\}. \end{aligned} \tag{2.38}$$

For any $k \in \{1, \ldots, K\}$, we have

$$\|D_k D_A^{-1}\| = \|D_A^{-1} D_k\| = \Big\| \Big\{ \frac{D_{k(i,i)}}{D_{A(i,i)}} \Big\}_d \Big\| = \Big\| \Big\{ \frac{D_{k(i,i)}}{\sum_{l=0}^{K} D_{l(i,i)}} \Big\}_d \Big\| \leq 1. \tag{2.39}$$

Combining (2.24)–(2.32) and using Woodbury matrix identity we can show

$$\begin{aligned} \|A^{-1} D_l\| &= \|(B_A + D_A)^{-1} D_l\| \\ &= \|D_A^{-1} D_l - D_A^{-1} (D_A^{-1} + B_A^{-1})^{-1} D_A^{-1} D_l\| \\ &\leq \|D_A^{-1} D_l\| + \|D_A^{-1} (D_A^{-1} + B_A^{-1})^{-1} D_A^{-1} B_l\| \end{aligned}$$

$$\leq 1 + \frac{\tau_4^2}{(K+1)\epsilon^2} < \infty, \quad (l = 0, \ldots, K). \tag{2.40}$$

Using 2.28 and 2.39 we obtain the inequalities:

$$\begin{aligned}
&\sum_{l>m\geq 1} \operatorname{tr}\big\{M_{(l,m)} D_l D_A^{-1} (D_A^{-1} + B_A^{-1})^{-1} D_A^{-1} D_m\big\} \\
&\quad = \sum_{l>m\geq 1} \sum_{j=1}^{p} \big\{M_{(l,m)} D_l D_A^{-1} (D_A^{-1} + B_A^{-1})^{-1} D_A^{-1} D_m\big\}_{(j,j)} \\
&\quad \leq \sum_{l>m\geq 1} p \|M_{(l,m)} D_l D_A^{-1} (D_A^{-1} + B_A^{-1})^{-1} D_A^{-1} D_m\| \\
&\quad \leq \sum_{l>m\geq 1} p \|M_{(l,m)}\| \, \|D_l D_A^{-1}\| \, \|(D_A^{-1} + B_A^{-1})^{-1}\| \, \|D_A^{-1} D_m\| \leq \infty; \qquad (2.41) \\
&\sum_{k=1}^{K} \operatorname{tr}\big\{\hat{\Sigma}_{Y(k,k)} D_0 D_A^{-1} (D_A^{-1} + B_A^{-1})^{-1} D_A^{-1} D_k\big\} \\
&\quad = \sum_{k=1}^{K} \sum_{j=1}^{p} \big\{\hat{\Sigma}_{Y(k,k)} D_0 D_A^{-1} (D_A^{-1} + B_A^{-1})^{-1} D_A^{-1} D_k\big\}_{(j,j)} \\
&\quad \leq \sum_{k=1}^{K} p \|\hat{\Sigma}_{Y(k,k)} D_0 D_A^{-1} (D_A^{-1} + B_A^{-1})^{-1} D_A^{-1} D_k\| \\
&\quad \leq \sum_{k=1}^{K} p \|\hat{\Sigma}_{Y(k,k)}\| \, \|D_0 D_A^{-1}\| \, \|(D_A^{-1} + B_A^{-1})^{-1}\| \, \|D_A^{-1} D_k\| \leq \infty. \qquad (2.42)
\end{aligned}$$

Additionally, using (2.34) and (2.40), we can prove

$$\begin{aligned}
&\sum_{l>m\geq 1} \operatorname{tr}(M_{(l,m)} B_l A^{-1} \Omega_m) \\
&\quad = \sum_{l>m\geq 1} \sum_{j=1}^{p} \big\{M_{(l,m)} B_l A^{-1} \Omega_m\big\}_{(j,j)} \leq \sum_{l>m\geq 1} p \|M_{(l,m)} B_l A^{-1} \Omega_m\| \\
&\quad \leq \sum_{l>m\geq 1} p \|M_{(l,m)}\| \, \|B_l\| \, \|A^{-1} \Omega_m\| \leq \infty; \qquad (2.43) \\
&\sum_{l>m\geq 1} \operatorname{tr}(M_{(l,m)} D_l A^{-1} B_m) \\
&\quad = \sum_{l>m\geq 1} \sum_{j=1}^{p} \big\{M_{(l,m)} D_l A^{-1} B_m\big\}_{(j,j)} \leq \sum_{l>m\geq 1} p \|M_{(l,m)} D_l A^{-1} B_m\| \\
&\quad \leq \sum_{l>m\geq 1} p \|M_{(l,m)}\| \, \|D_l A^{-1}\| \, \|B_m\| \leq \infty; \qquad (2.44) \\
&\sum_{k=1}^{K} \operatorname{tr}\big\{\hat{\Sigma}_{Y(k,k)} B_0 A^{-1} \Omega_k\big\}
\end{aligned}$$

$$= \sum_{k=1}^{K}\sum_{j=1}^{p}\left\{\hat{\Sigma}_{Y(k,k)}B_0A^{-1}\Omega_k\right\}_{(j,j)} \leq \sum_{k=1}^{K} p\|\hat{\Sigma}_{Y(k,k)}B_0A^{-1}\Omega_k\|$$

$$\leq \sum_{k=1}^{K} p\|\hat{\Sigma}_{Y(k,k)}\|\,\|B_0\|\,\|A^{-1}\Omega_k\| \leq \infty \tag{2.45}$$

$$\sum_{k=1}^{K}\mathrm{tr}\left\{\hat{\Sigma}_{Y(k,k)}D_0A^{-1}B_k\right\}$$

$$= \sum_{k=1}^{K}\sum_{j=1}^{p}\left\{\hat{\Sigma}_{Y(k,k)}D_0A^{-1}B_k\right\}_{(j,j)} \leq \sum_{k=1}^{K} p\|\hat{\Sigma}_{Y(k,k)}D_0A^{-1}B_k\|$$

$$\leq \sum_{k=1}^{K} p\|\hat{\Sigma}_{Y(k,k)}\|\,\|D_0A^{-1}\|\,\|B_k\| \leq \infty \tag{2.46}$$

Using (2.41) - (2.46), the order of log likelihood in (2.21) is equivalent to:

$$\begin{aligned}
&\sum_{k=0}^{K}\sum_{i=1}^{p}\Big\{\log(D_{k(i,i)})\Big\} - \log\Big\{\det(\sum_{k=0}^{K}D_k)\Big\} - \sum_{l>m\geq 1}\left\{\mathrm{tr}(M_{(l,m)}D_lD_A^{-1}D_m)\right\}\\
&-\sum_{k=1}^{K}\left\{\mathrm{tr}(\hat{\Sigma}_{Y(k,k)}D_0D_A^{-1}D_k)\right\}\\
=\ &\sum_{i=1}^{p}\sum_{k=0}^{K}\Big\{\log(D_{k(i,i)})\Big\} - \sum_{i=1}^{p}\Big\{\log\big(\sum_{k=0}^{K}D_{k(i,i)}\big)\Big\} - \sum_{i=1}^{p}\sum_{l>m\geq 1}\left(M_{(l,m)(i,i)}\frac{D_{l(i,i)}D_{m(i,i)}}{\sum_{k=0}^{K}D_{k(i,i)}}\right)\\
&-\sum_{i=1}^{p}\sum_{l=1}^{K}\left(\frac{\hat{\Sigma}_{Y(l,l)(i,i)}D_{0(i,i)}D_{l(i,i)}}{\sum_{k=0}^{K}D_{k(i,i)}}\right),
\end{aligned} \tag{2.47}$$

where $\hat{\Sigma}_{Y(l,l)(i,j)}$ and $M_{(l,m)(i,i)}$ represents the entry in ith row and jth column of the matrix $\hat{\Sigma}_{Y(l,l)}$ and $M_{(l,m)}$ respectively.

Next, we will show the diagonal entries of $M_{(l,m)}$ are positive. We know that

$$\begin{pmatrix}\hat{\Sigma}_{Y(l,l)} & \hat{\Sigma}_{Y(l,m)}\\ \hat{\Sigma}_{Y(m,l)} & \hat{\Sigma}_{Y(m,m)}\end{pmatrix} = \begin{pmatrix}Y_l^{\mathrm{T}}\\ Y_m^{\mathrm{T}}\end{pmatrix}\begin{pmatrix}Y_l & Y_m\end{pmatrix}/n$$

is non-negative definite, and $\hat{\Sigma}_{Y(m,l)} = \hat{\Sigma}_{Y(l,m)}^{\mathrm{T}}$. Thus for any vector a, we have

$$a^{\mathrm{T}}\begin{pmatrix}I & -I\end{pmatrix}\begin{pmatrix}\hat{\Sigma}_{Y(l,l)} & \hat{\Sigma}_{Y(l,m)}\\ \hat{\Sigma}_{Y(m,l)} & \hat{\Sigma}_{Y(m,m)}\end{pmatrix}\begin{pmatrix}I\\ -I\end{pmatrix}a = a^{\mathrm{T}}\{\hat{\Sigma}_{Y(l,l)} + \hat{\Sigma}_{Y(m,m)} - \hat{\Sigma}_{Y(m,l)} - \hat{\Sigma}_{Y(m,l)}^{\mathrm{T}}\}a \geq 0.$$

Thus $\hat{\Sigma}_{Y(l,l)} + \hat{\Sigma}_{Y(m,m)} - \hat{\Sigma}_{Y(m,l)} - \hat{\Sigma}^{\mathrm{T}}_{Y(m,l)}$ is non-negative definite, and its diagonal entries, which equals to the diagonal entries of $M_{(l,m)}$, are non-negative.

If we focus on a specific $i \in (1, ..., p)$, then the only positive term is $\sum_{k=0}^{K}\{\log D_{k(i,i)}\}$. Thus, if we could bound it using the remaining of the terms in (2.47), then we will complete the proof.

Without loss of generality, we assume $D_{s(i,i)}$ and $D_{r(i,i)}$ have the highest and second highest rates among those positive term. We have $\log D_{s(i,i)} - \log(\sum_{k=0}^{K} D_{k(i,i)}) < 0$, and also the rate of $M_{(s,r)(i,i)} D_{s(i,i)} D_{r(i,i)} / \{\sum_{k=0}^{K} D_{k(i,i)}\}$ equals to $M_{(s,r)(i,i)} D_{r(i,i)}$. Since $M_{(s,r)(i,i)} > 0$ for any i, the term $\log D_{r(i,i)} - M_{(s,r)(i,i)} D_{s(i,i)} D_{r(i,i)} / \{\sum_{k=0}^{K} D_{k(i,i)}\}$ would go to negative infinity as $D_{r(i,i)} \to \infty$. If the second highest rate for the positive term is $D_{0(i,i)}$, we can simply replace $M_{(s,r)(i,i)}$ with $\hat{\Sigma}_{Y(s,s)(i,i)}$, and the proof can also be carried out. This completes the proof of part I.

In summary, the only condition to bound the likelihood is that $\hat{\Sigma}_{Y(l,l)(i,i)}$ and $M_{(l,m)(i,i)}$ are greater than zero for any l, m, and i. In other words, there is no constant variable in Y_k; and there are no variables that are perfectly correlated between categories l and m.

For part II, we will show that the penalized log-likelihood does not decreases for each step of the graphic EM algorithm. For simplicity, we write Ω for $\{\Omega_k\}_{k=0}^{K}$ in the following derivation. We can write the full log-likelihood as

$$\begin{aligned}\mathcal{L}(\Omega \mid y, z) \propto \log\{\det(\Omega_0)\} - \mathrm{tr}(\Omega_0 z z^{\mathrm{T}}/n) \\ + \sum_{k=1}^{K} \Big[\log\{\det(\Omega_k)\} - \mathrm{tr}(\Omega_k (y_k - z)(y_k - z)^{\mathrm{T}}/n) \Big].\end{aligned}$$

The above log likelihood cannot be calculated directly because the values of z and zz^{T} are unobserved. But we can calculate a function $\mathcal{Q}(\Omega \mid \Omega^{(t)}, y)$ in which z and zz^{T} are substituted by their expected values conditional on Ω and y. We define

$$\begin{aligned}\mathcal{Q}(\Omega \mid y, \Omega^{(t)}) &= E\big\{\mathcal{L}(\Omega \mid y, z) \mid \Omega^{(t)}\big\} \\ &= E\Big[\log\{f(y, z \mid \Omega)\} \mid \Omega^{(t)} \Big] \\ &= E\Big[\log\{f(y \mid \Omega)\} + \log\{f(z \mid y, \Omega)\} \mid \Omega^{(t)} \Big] \\ &= \log\{f(y \mid \Omega)\} + E\Big[\log\{f(z \mid y, \Omega)\} \mid \Omega^{(t)} \Big], \end{aligned} \tag{2.48}$$

where $f(y \mid \Omega)$ and $f(y, z \mid \Omega)$ are the probability of y and joint probability of (y, z) respectively. The equality in (2.48) is due to the fact that expectation is over values of z, and $\log\{f(y \mid \Omega)\}$ is a constant with respect to the expectation since y is observed.

Based on (2.48), we have

$$\begin{aligned}
&\log\{f(y \mid \Omega)\} - Pen(\Omega) \\
&= \mathcal{Q}\big(\Omega \mid y, \Omega^{(t)}\big) - E\Big[\log\{f(z \mid y, \Omega)\} \mid \Omega^{(t)}\Big] - Pen(\Omega),
\end{aligned}$$

where $Pen(\Omega)$ is the penalty function which is $\lambda_1 \sum_{k=1}^{K} |\Omega_k^-|_1 + \lambda_2 |\Omega_0^-|_1$ in our case.

The M step in graphic EM is to update $\Omega^{(t)} \to \Omega^{(t+1)}$ through

$$\Omega^{(t+1)} = \underset{\Omega}{\operatorname{argmax}}\, \mathcal{Q}(\Omega \mid y, \Omega^{(t)}) - Pen(\Omega). \tag{2.49}$$

Comparing the penalized log-likelihoods for steps t and $t+1$, we can get

$$\begin{aligned}
&\log\{f(y \mid \Omega^{(t+1)})\} - Pen\big(\Omega^{(t+1)}\big) - \log\{f(y \mid \Omega^{(t)})\} + Pen(\Omega_k^{(t)}) \\
&= \mathcal{Q}\big(\Omega^{(t+1)} \mid y, \Omega^{(t)}\big) - Pen(\Omega^{(t+1)}) - E\Big[\log\{f(z \mid y, \Omega^{(t+1)})\} \mid \Omega^{(t)}\Big] \\
&\quad - \mathcal{Q}\big(\Omega^{(t)} \mid y, \Omega^{(t)}\big) + Pen(\Omega^{(t)}) + E\Big[\log\{f\big(z \mid y, \Omega^{(t)}\big)\} \mid \Omega^{(t)}\Big].
\end{aligned}$$

By (2.49), it follows that

$$\mathcal{Q}\big(\Omega^{(t+1)} \mid y, \Omega_k^{(t)}\big) - Pen(\Omega^{(t+1)}) - \mathcal{Q}\big(\Omega^{(t)} \mid y, \{\Omega^{(t)}\big) + -Pen(\Omega^{(t)}) > 0, \tag{2.50}$$

since $\Omega^{(t+1)}$ is the maximizer over the term $\mathcal{Q}(\Omega \mid \Omega^{(t)}) - Pen(\Omega)$.

Moreover, by Gibbs' inequality, we have

$$-E\Big[\log\{f(z \mid y, \Omega^{(t+1)})\} \mid \Omega^{(t)}\Big] + E\Big[\log\{f(z \mid y, \Omega^{(t)})\} \mid \Omega^{(t)}\Big] > 0. \tag{2.51}$$

Combining (2.50) and (2.51), we have

$$\log\{f(y \mid \Omega^{(t+1)})\} - Pen(\Omega^{(t+1)}) - \log\{f(y \mid \Omega^{(t)})\} + Pen(\Omega^{(t)}) > 0,$$

which completes the proof for part II.

Thus with an upper bound, d ,for the penalized log-likelihood $\mathcal{P}(\Omega)$ together with an prespecified threshold δ, for at most $\big\{T - \mathcal{P}(\Omega^{(0)})\big\}/\delta$ steps, there are two consecutive steps

t and $t+1$ satisfying

$$|\mathcal{P}(\{\Omega_k^{(t+1)}\}_{k=0}^K) - \mathcal{P}(\{\Omega_k^{(t)}\}_{k=0}^K)| < \delta,$$

which completes the proof.

2.7.4 Proof of Theorem 2.1

In this proof, we need to use Lemma 3 of Bickel and Levina (2008). We state the result here for completeness.

Lemma 2.2. *Let Z_i be i.i.d. $\mathcal{N}(0, \Sigma_p)$ and $\phi_{max}(\Sigma_p) \leq \bar{k} < \infty$. Then , if $\Sigma_p = \{\sigma_{ab}\}$,*

$$\text{pr}\left(\left|\sum_{i=1}^n (Z_{ij}Z_{ik} - \sigma_{jk})\right| \geq n\nu\right) \leq C_1 \exp(-C_2 n\nu^2), \quad \textit{for} \quad |\nu| \leq \delta\,,$$

where C_1, C_2 and δ depend on $\bar{k}$ only.

We first show that $\phi_{max}(\Sigma_Y^*)$ is bounded above. Let $v = (v_0^{\mathrm{T}}, \ldots, v_K^{\mathrm{T}})^{\mathrm{T}} \in \mathbb{R}^{(K+1)p}$ and $v^{\mathrm{T}}v = 1$. With Condition 1, we have

$$\begin{aligned}
v^{\mathrm{T}}\Sigma_Y^* v &= \sum_{k=1}^K v_k^{\mathrm{T}}\Sigma_k^* v_k + (\sum_{k=1}^K v_k^{\mathrm{T}})\Sigma_0^*(\sum_{k=1}^K v_k) \\
&\leq K\tau_2 + (\sum_{k=1}^K v_k^{\mathrm{T}}\sum_{k=1}^K v_k)(\sum_{k=1}^K v_k^{\mathrm{T}})\Sigma_0^*(\sum_{k=1}^K v_k)/(\sum_{k=1}^K v_k^{\mathrm{T}}\sum_{k=1}^K v_k) \\
&\leq K\tau_2 + \tau_2(\sum_{k=1}^K v_k^{\mathrm{T}}\sum_{k=1}^K v_k) \\
&\leq K\tau_2 + 2\tau_2\|\sum_{k=1}^K v_k\|^2 \leq K\tau_2 + 2K^2\tau < \infty.
\end{aligned}$$

In order to estimate Ω_k, we need to minimize (2.9), where $\hat{\Sigma}_k'$ is the only input. First, we would like to bound the maximum absolute entry of matrix $\hat{\Sigma}_k - \Sigma_k^*$. We assume Condition 2 holds, and let $\hat{\Sigma}_0 = \sum_{i=1}^n y_{l,i}y_{m,i}^{\mathrm{T}}/n$ and $\hat{\Sigma}_k = \sum_{i=1}^n y_{k,i}y_{k,i}^{\mathrm{T}}/n - \hat{\Sigma}_0$ ($l \neq m;\ k \in 1...K$). Using the union sum inequality and Lemma 2.2:

$$\begin{aligned}
&\text{pr}\left(\max_{1\leq i,j\leq p} |\hat{\sigma}_{0(i,j)} - \sigma_{0(i,j)}^*| \geq C_3\{(\log p)/n\}^{1/2}\right) \\
=&\text{pr}\left(\bigcup_{1\leq i,j\leq p}\left[|\hat{\sigma}_{0(i,j)} - \sigma_{0(i,j)}^*| \geq C_3\{(\log p)/n\}^{1/2}\right]\right)
\end{aligned}$$

$$\leq \sum_{1\leq i,j\leq p} \mathrm{pr}\left(|\hat{\sigma}_{0(i,j)} - \sigma^*_{0(i,j)}| \geq C_3\{(\log p)/n\}^{1/2}\right) \leq p^2 C_1 \exp\{-C_2 n C_3^2 (\log p)/n\}$$

$$= C_1 p^{2-C_3^2 C_2} \to 0\,,$$

for any sufficiently large C_3. Therefore with probability tending to 1,

$$\|\hat{\Sigma}_0 - \Sigma^*_0\|_\infty \leq C_3\{(\log p)/n\}^{1/2},$$

where $\|W\|_\infty$ denotes $\max_{i,j}|\omega_{i,j}|$ for a matrix W. Similarly we have,

$$\|\hat{\Sigma}_0 + \hat{\Sigma}_k - \Sigma^*_0 - \Sigma^*_k\|_\infty \leq C_4\{(\log p)/n\}^{1/2} \quad (k = 1, \ldots, K).$$

Then by the triangle inequality, we have

$$\|\hat{\Sigma}_k - \Sigma^*_k\|_\infty \leq (C_3 + C_4)\{(\log p)/n\}^{1/2}.$$

Thus $\|\hat{\Sigma}_k - \Sigma^*_k\|_\infty = O_P\big[\{(\log p)/n\}^{1/2}\big]$ $(k = 0, \ldots, K)$. The same rate can also be derived for $\hat{\Sigma}_0 = \sum_{m\neq l}\sum_{i=1}^n y_{m,i}(y_{l,i})^{\mathrm{T}}/\{K(K-1)n\}$ and $\hat{\Sigma}_k = \sum_{i=1}^n y_{k,i}y_{k,i}^{\mathrm{T}}/n - \hat{\Sigma}_0$ following the similar proof strategy. Next, we want to bound $\|\hat{\Sigma}'_k - \Sigma^*_k\|_\infty$ $(k = 0, \ldots, K)$. By triangle inequality, we have

$$\begin{aligned}\|\hat{\Sigma}'_k - \Sigma^*_k\|_\infty &= \|\hat{\Sigma}'_k - \hat{\Sigma}_k + \hat{\Sigma}_k - \Sigma^*_k\|_\infty \\ &\leq \|\hat{\Sigma}'_k - \hat{\Sigma}_k\|_\infty + \|\hat{\Sigma}_k - \Sigma^*_k\|_\infty \\ &\leq 2\|\hat{\Sigma}_k - \Sigma^*_k\|_\infty,\end{aligned}$$

by the definition of projection in (2.8). Thus we also have $\|\hat{\Sigma}'_k - \Sigma^*_k\|_\infty = O_P\big[\{(\log p)/n\}^{1/2}\big]$ $(k = 0, \ldots, K)$.

For simplicity, we will write $\Omega = \Omega_k$, $\Omega^* = \Omega^*_k$, $\hat{\Sigma}' = \hat{\Sigma}'_k$ and $\Delta = \Delta_k$, where $\Delta_k = \Omega_k - \Omega^*_k$ $(k = 0, \ldots, K)$ and $\lambda = \lambda_1$ or λ_2. Let $\hat{\Omega}$ be our estimate minimizing (2.9) and define function $\mathcal{V}(\Omega)$ as a normalized target function of (2.9),

$$\begin{aligned}\mathcal{V}(\Omega) =& \mathrm{tr}(\Omega\hat{\Sigma}') - \log\{\det(\Omega)\} + \lambda|\Omega^-|_1 - \mathrm{tr}(\Omega^*\hat{\Sigma}') + \log\{\det(\Omega^*)\} - \lambda|\Omega^{*-}|_1 \\ =& \mathrm{tr}\{(\Omega - \Omega^*)(\hat{\Sigma}' - \Sigma^*)\} - \big[\log\{\det(\Omega)\} - \log\{\det(\Omega^*)\}\big] \\ &+ \mathrm{tr}\{(\Omega - \Omega^*)\Sigma^*\} + \lambda(|\Omega^-|_1 - |\Omega^{*-}|_1). \qquad (2.52)\end{aligned}$$

For the one-step algorithm, our estimate $\hat{\Omega}$ minimizes $\mathcal{V}(\Omega)$. Notice that $\mathcal{V}(\Omega)$ is also a function of Δ, thus we define $\mathcal{G}(\Delta) \equiv \mathcal{V}(\Omega^* + \Delta)$. It is easy to check $\mathcal{G}(0) = 0$, and $\hat{\Delta} = \hat{\Omega} - \Omega^*$ minimizes the function $\mathcal{G}(\Delta)$. The main idea of the proof is as follows: we first define a closed bounded convex set $\mathcal{A}$ which contains 0, and show that $\mathcal{G}$ is strictly positive on the boundary of $\mathcal{A}$. Since $\mathcal{G}$ is continuous and $\mathcal{G}(0) = 0$, it implies that $\mathcal{G}$ has a local minimum inside $\mathcal{A}$. We define

$$\begin{aligned} \mathcal{A} &= \{\Delta : \Delta = \Delta^{\mathrm{T}}, \|\Delta\|_F \leq M r_n\}, \\ \partial\mathcal{A} &= \{\Delta : \Delta = \Delta^{\mathrm{T}}, \|\Delta\|_F = M r_n\}, \end{aligned} \tag{2.53}$$

where M is a positive constant and $r_n = \{(p+q)(\log p)/n\}^{1/2} \to 0$.

For the logarithm term in (2.52), we use the Taylor expansion $f(t) = \log\det(\Omega + t\Delta)$, the integral form of the remainder, and the symmetry of Δ, Σ^*, and Ω^* to derive

$$\begin{aligned} &\log\{\det(\Omega^* + \Delta)\} - \log\{\det(\Omega^*)\} \\ &= \mathrm{tr}(\Sigma^*\Delta) - \mathrm{vec}(\Delta^{\mathrm{T}}) \left\{ \int_0^1 (1-v)(\Omega^* + v\Delta)^{-1} \otimes (\Omega^* + v\Delta)^{-1} \mathrm{d}v \right\} \mathrm{vec}(\Delta), \end{aligned}$$

where . Thus, we can rewrite (2.52) as,

$$\begin{aligned} \mathcal{G}(\Delta) =& \mathrm{tr}\{\Delta(\hat{\Sigma}' - \Sigma^*)\} + \mathrm{vec}(\Delta^{\mathrm{T}}) \left\{ \int_0^1 (1-v)(\Omega^* + v\Delta)^{-1} \otimes (\Omega^* + v\Delta)^{-1} dv \right\} \mathrm{vec}(\Delta) \\ &+ \lambda(|\Omega^{*-} + \Delta^-|_1 - |\Omega^{*-}|_1) = \mathrm{I} + \mathrm{II} + \mathrm{III}. \end{aligned} \tag{2.54}$$

where where $\mathrm{vec}(\cdot)$ returns the vectorization of a matrix. To show that $\mathcal{G}(\Delta)$ is strictly positive on $\partial\mathcal{A}$, we need to bound I, II and III. First, we bound I using symmetry arguments once again, and the triangular inequality,

$$\begin{aligned} |\mathrm{tr}\{\Delta(\hat{\Sigma}' - \Sigma^*)\}| = |(\mathrm{vec}(\Delta))^{\mathrm{T}} \mathrm{vec}(\hat{\Sigma}' - \Sigma^*)| &= \left| \sum_{i,j} \delta_{ij}(\hat{\sigma}'_{ij} - \sigma^*_{ij}) \right| \\ &\leq \left| \sum_{i \neq j} \delta_{ij}(\hat{\sigma}'_{ij} - \sigma^*_{ij}) \right| + \left| \sum_i \delta_{ii}(\hat{\sigma}'_{ii} - \sigma^*_{ii}) \right| = \mathrm{I}' + \mathrm{II}'. \end{aligned}$$

As discussed above, with probability tending to 1,

$$\max_{i \neq j} |\hat{\sigma}'_{ij} - \sigma^*_{ij}| = \|\mathrm{vec}(\hat{\Sigma}' - \Sigma^*)\|_\infty \leq 2(C_3 + C_4)\{(\log p)/n\}^{1/2},$$

and hence term I′ is bounded by

$$\mathrm{I}' \leq |\Delta^-|_1 \max_{i \neq j} |\hat{\sigma}'_{ij} - \sigma^*_{ij}| \leq 2(C_3 + C_4)\{(\log p)/n\}^{1/2} |\Delta^-|_1. \tag{2.55}$$

We could also bound term II′ with probability tending to 1 using the Cauchy-Schwartz inequality and Lemma 2.2,

$$\begin{aligned}\mathrm{II}' \leq & \left\{ \sum_{i=1}^{p} (\hat{\sigma}'_{ii} - \sigma^*_{ii})^2 \right\}^{1/2} \|\Delta^+\|_F \leq p^{1/2} \max_{1 \leq i \leq p} |\hat{\sigma}'_{ii} - \sigma^*_{ii}| \, \|\Delta^+\|_F \\ \leq & 2(C_3 + C_4)\{p(\log p)/n\}^{1/2} \|\Delta^+\|_F \leq 2(C_3 + C_4)\{(p+q)(\log p)/n\}^{1/2} \|\Delta^+\|_F, \end{aligned} \tag{2.56}$$

where Δ^+ is the digonal entries of Δ.

In order to bound II, we use results established in Rothman et al. (2008, Theorem 1):

$$\mathrm{vec}(\Delta^{\mathrm{T}}) \left\{ \int_0^1 (1-v)(\Omega^* + v\Delta)^{-1} \otimes (\Omega^* + v\Delta)^{-1} \mathrm{d}v \right\} \mathrm{vec}(\Delta) \geq \|\Delta\|_F^2 / (4\tau_2^2) \,. \tag{2.57}$$

Lastly, we would like to bound III. For an index set B and a matrix $M = \{m_{ij}\}$, define $M_B \equiv \{m_{ij} : (i,j) \in B\}$. Recall that $\mathrm{T} = \{(j,j') : j \neq j', \omega^*_{j,j'} \neq 0\}$ and let T^c be its complement. Note that $|\Omega^{*-}|_1 = |\Omega^{*-}_{\mathrm{T}}|_1$, and $|\Omega^{*-} + \Delta^-|_1 = |\Omega^{*-}_{\mathrm{T}} + \Delta^-_{\mathrm{T}}|_1 + |\Delta^-_{\mathrm{T}^c}|_1$. Then, using the triangular inequality, this implies

$$\lambda(|\Omega^{*-} + \Delta^-|_1 - |\Omega^{*-}|_1) \geq \lambda \left(|\Delta^-_{\mathrm{T}^c}|_1 - |\Delta^-_{\mathrm{T}}|_1 \right). \tag{2.58}$$

Combining (2.55), (2.56), (2.57) and (2.58), we can show,

$$\begin{aligned}\mathcal{G}(\Delta) \geq & \, \|\Delta\|_F^2/(4\tau_2^2) - 2(C_3 + C_4)\{(\log p)/n\}^{1/2} |\Delta^-|_1 \\ & - 2(C_3 + C_4)\big\{(p+q)(\log p)/n\big\}^{1/2} \|\Delta^+\|_F + \lambda \left(|\Delta^-_{\mathrm{T}^c}|_1 - |\Delta^-_{\mathrm{T}}|_1 \right) \\ = & \, \|\Delta\|_F^2/(4\tau_2^2) + \Big\{ \lambda - 2(C_3 + C_4)\{(\log p)/n\}^{1/2} \Big\} |\Delta^-_{\mathrm{T}^c}|_1 \\ & - \Big[2(C_3 + C_4)\{(\log p)/n\}^{1/2} + \lambda \Big] \, |\Delta^-_{\mathrm{T}}|_1 - 2(C_3 + C_4)\{(p+q)(\log p)/n\}^{1/2} \|\Delta^+\|_F \\ \geq & \, \|\Delta\|_F^2/(4\tau_2^2) + (a_1 - 2C_3 - 2C_4)\{(\log p)/n\}^{1/2} |\Delta^-_{\mathrm{T}^c}|_1 \\ & - \Big[2(C_3 + C_4)\{(\log p)/n\}^{1/2} + b_1 \big\{ (1 + p/q)(\log p)/n \big\}^{1/2} \Big] \, |\Delta^-_{\mathrm{T}}|_1 \\ & - 2(C_3 + C_4)\big\{(p+q)(\log p)/n\big\}^{1/2} \|\Delta^+\|_F, \end{aligned}$$

where the last inequality uses the condition $a_1\{(\log p)/n\}^{1/2} \leq \lambda_1, \lambda_2 \leq b_1\{(1 + p/q)(\log p)/n\}^{1/2}$. Thus when a_1 is large enough, the term $(a_1 - 2C_3 - 2C_4)\{(\log p)/n\}^{1/2}|\Delta^-_{\mathrm{T}^c}|_1$ is always positive. By using the Cauchy-Schwartz inequality we have

$$|\Delta^-_{\mathrm{T}}|_1 \leq \sqrt{q}\|\Delta^-_{\mathrm{T}}\|_F \leq \sqrt{q}\|\Delta^-\|_F \leq \sqrt{q}\|\Delta\|_F\,, \qquad \|\Delta^+\|_F \leq \|\Delta\|_F\,. \tag{2.59}$$

Thus we have,

$$\begin{aligned}\mathcal{G}(\Delta) \geq & \|\Delta\|_F^2/(4\tau_2^2) - (2C_3 + 2C_4 + b_1)\{(p+q)(\log p)/n\}^{1/2}\|\Delta\|_F \\ & - (2C_3 + 2C_4)\{(p+q)(\log p)/n\}^{1/2}\|\Delta\|_F \\ = & \|\Delta\|_F^2 \left[1/(4\tau_2^2) - (4C_3 + 4C_4 + b_1)\{(p+q)(\log p)/n\}^{1/2}\|\Delta\|_F^{-1}\right]. \end{aligned} \tag{2.60}$$

We know $\Delta \in \partial\mathcal{A}$, where $\partial\mathcal{A} = \{\Delta : \Delta = \Delta^{\mathrm{T}}, \|\Delta\|_F = Mr_n\}$ and $r_n = \{(p+q)(\log p)/n\}^{1/2}$. Thus we have $\|\Delta\|_F^{-1}\{(p+q)(\log p)/n\}^{1/2} = 1/M$ and plug it into (2.60) and get

$$\mathcal{G}(\Delta) \geq \|\Delta\|_F^2 \left[1/(4\tau_2^2) - (4C_3 + 4C_4 + b_1)/M\right] > 0,$$

for any sufficiently large M. Since $\mathcal{G}$ is continuous and $\mathcal{G}(0) = 0$, with the fact that $\mathcal{G}$ is strictly positive on the boundary of $\mathcal{A}$, namely $\partial\mathcal{A}$, it implies that $\mathcal{G}$ has a local minimum inside $\mathcal{A}$. Therefore we have $\|\hat{\Omega} - \Omega^*\|_F \leq Mr_n$, and thus $\|\hat{\Omega} - \Omega^*\|_F = O_p(r_n) = O_p(\{(p+q)(\log p)/n\}^{1/2})$ which completes the proof.

2.7.5 Proof of Corollary 2.1

Let $\hat{\Omega}_k = \Omega_k^* + \Delta_k$ be the one-step solution. By Theorem 2.1, we have

$$\|\hat{\Omega}_k - \Omega_k^*\|_F = \|\Delta_k\|_F = O_p\left[\left\{\frac{(p+q)\log p}{n}\right\}^{1/2}\right].$$

Using the Woodbury matrix identity twice, we know that

$$\begin{aligned}\check{\Sigma}_k &= (\Omega_k^* + \Delta_k)^{-1} = \Sigma_k^* - \Sigma_k^*(\Delta_k^{-1} + \Sigma_k^*)^{-1}\Sigma_k^* \\ &= \Sigma_k^* - \Sigma_k^*(\Delta_k - \Delta_k(\Omega_k^* + \Delta_k)^{-1}\Delta_k)\Sigma_k^*\end{aligned}$$

$$= \Sigma_k^* - \Sigma_k^* \Delta_k \Sigma_k^* + \Sigma_k^* \Delta_k (\Delta_k + \Omega_k^*)^{-1} \Delta_A \Sigma_k^*.$$

By Condition 1, we have ${\tau_2}^{-1} < \phi_{min}(\Sigma_k^*) < \phi_{max}(\Sigma_k^*) < {\tau_1}^{-1}$. Follow the proof of Theorem 2.2, we can get

$$\begin{aligned}
\|\check{\Sigma}_k - \Sigma_k^*\|_F &\leq \|\Sigma_k^* \Delta_k \Sigma_k^*\|_F + \|\Sigma_k^* \Delta_k (\Delta_k + \Omega_k^*)^{-1} \Delta_k \Sigma_k^*\|_F \\
&\leq \|\Sigma_k^*\|^2 \|\Delta_k\|_F + \|\Sigma_k^*\|^2 \|\Delta_k\|_F^2 \|(\Delta_k + \Omega_k^*)^{-1}\| \\
&\leq \|\Delta_k\|_F / (\tau_1^2) + \|\Delta_k\|_F^2 \|(I + \Sigma_k^* \Delta_k)^{-1}\| \, \|\Sigma_k^*\| / (\tau_1^2) \\
&\leq \|\Delta_k\|_F / (\tau_1^2) + \frac{1}{\tau_1^3} \|\Delta_k\|_F^2 (1 - \|\Sigma_k^* \Delta_k\|)^{-1} \qquad (2.61) \\
&\lesssim \|\Delta_k\|_F / (\tau_1^2) + \frac{2}{\tau_1^3} \|\Delta_k\|_F^2 \qquad (2.62) \\
&= O_p \left[\left\{ \frac{(p+q) \log p}{n} \right\}^{1/2} \right],
\end{aligned}$$

the inequality of (2.61) is due to Lemma 2.4, since $\|\Sigma_k^* \Delta_k\| < 1$ when n and p is large enough. The inequality of (2.62) holds when n and p is large enough since

$$\|\Sigma_k^* \Delta_k\| \leq \|\Sigma_k^* \Delta_k\|_F \leq \|\Sigma_k^*\| \, \|\Delta_k\|_F \leq \frac{1}{\tau_1^2} \|\Delta_k\|_F \to 0 \,.$$

Thus we finish the proof.

2.7.6 Proof of Theorem 2.2

To prove Theorem 2.2, we use the following Lemma.

Lemma 2.3. *Suppose that Condition 1-2 hold,* $(p+q)(\log p)/n = o(1)$*, and* $\lambda_1, \lambda_2 \leq b_3 \{(1+p/q)(\log p)/n\}^{1/2}$ *for some constant* b_3 *and* $\|\tilde{\Sigma}_k - \Sigma_k^*\|_F = O_p \left[\{(p+q)(\log p)/n\}^{1/2} \right]$*. Let* $\{\tilde{\Omega}_k\}_{k=0}^K$ *be the minimizer define by* (2.9) *replacing* $\hat{\Sigma}_k$ *with* $\tilde{\Sigma}_k$*, then*

$$\sum_{k=0}^{K} \left\| \tilde{\Omega}_k - \Omega_k^* \right\|_F = O_p \left[\left\{ \frac{(p+q) \log p}{n} \right\}^{1/2} \right].$$

Proof. This proof is analogous to the proof of Theorem 2.1. Define $\mathcal{G}(\Delta)$ the same as in (2.52), and define a closed bounded convex set $\mathcal{A}$ as (2.53). We only need to show $\mathcal{G}(\Delta)$ is strictly positive on $\partial \mathcal{A}$, and can write $\mathcal{G}(\Delta) = \mathrm{I} + \mathrm{II} + \mathrm{III}$ as in (2.54). To bound I, we use

matrix symmetry and the Cauchy-Schwarz inequality,

$$|\mathrm{tr}\{\Delta(\tilde{\Sigma}-\Sigma^*)\}| \le \|\tilde{\Sigma}-\Sigma^*\|_F \|\Delta\|_F = D_1 \{(p+q)(\log p)/n\}^{1/2} \|\Delta\|_F,$$

where D_1 is some constant. For II and III, they have the same bound as (2.57) and (2.58) respectively. We can show

$$\begin{aligned}
\mathcal{G}(\Delta) \ge & \|\Delta\|_F^2/(4\tau_2^2) - D_1 \{(p+q)(\log p)/n\}^{1/2} \|\Delta\|_F + \lambda(|\Delta^-_{\mathrm{T}^c}|_1 - |\Delta^-_{\mathrm{T}}|_1) \\
\ge & \|\Delta\|_F^2/(4\tau_2^2) - D_1 \{(p+q)(\log p)/n\}^{1/2} \|\Delta\|_F - \lambda|\Delta^-_{\mathrm{T}}|_1 \\
\ge & \|\Delta\|_F^2/(4\tau_2^2) - D_1 \{(p+q)(\log p)/n\}^{1/2} \|\Delta\|_F \\
& - b_3 \{(p+q)(\log p)/n\}^{1/2} \|\Delta\|_F \qquad (2.63) \\
= & \|\Delta\|_F^2 \left[\frac{1}{4\tau_2^2} - (D_1+b_3) \{(p+q)(\log p)/n\}^{1/2} \|\Delta\|_F^{-1}\right] \\
= & \|\Delta\|_F^2 \left[\frac{1}{4\tau_2^2} - (D_1+b_3)/M\right] > 0,
\end{aligned}$$

for any sufficiently large M defined in (2.53). The inequality of (2.63) uses the result of (2.59) and the condition $\lambda \le b_3\{(1+p/q)(\log p)/n\}^{1/2}$. □

Here we state a known matrix result and provide a short proof for completeness.

Lemma 2.4. *Let F be any $p \times p$ matrix with $\|F\| < 1$. Then $(I-F)^{-1} = \sum_{k=0}^{\infty} F^k$, and*

$$\|(I-F)^{-1}\| \le \frac{1}{1-\|F\|}.$$

Proof. Note that $\left(\sum_{k=0}^{N} F^k\right)(I-F) = I - F^{N+1}$. Since $\|F^k\| \le \|F\|^k$ and $\|F\| < 1$, we have $F^k \to 0$ as $k \to \infty$. As a result of this,

$$\lim_{N\to\infty} \left(\sum_{k=0}^{N} F^k\right)(I-F) = I,$$

thus $(I-F)^{-1} = \sum_{k=0}^{\infty} F^k$, and we can get

$$\|(I-F)^{-1}\| = \|\sum_{k=0}^{\infty} F^k\| \le \sum_{k=0}^{\infty} \|F^k\| \le \sum_{k=0}^{\infty} \|F\|^k \le \frac{1}{1-\|F\|}.$$

□

In the proof of Theorem 2.2, we also need the Lemma 1 from Lam and Fan (2009), and we state the result here for completeness.

Lemma 2.5. *Let A and B be real matrices such that the product AB is defined. Then we have*

$$\|AB\|_F \leq \|A\|\|B\|_F.$$

In particular, if $A = \{a_{ij}\}$, then $|a_{ij}| \leq \|A\|$ for each i, j. When both A and B are symmetric matrices, we also have

$$\|AB\|_F = \|B^T A^T\|_F = \|BA\|_F \leq \|B\| \, \|A\|_F.$$

To prove Theorem 2.2, we assume Conditions 1-3 hold. In the proof of Theorem 2.1, we have shown that for the first M step, we obtain estimators $\hat{\Omega}_k^{(1)}$ such that $\sum_{k=0}^K \left\|\hat{\Omega}_k^{(1)} - \Omega_k^*\right\|_F = O_p\left[\{(p+q)(\log p)/n\}^{1/2}\right]$. In the E-step, if we can show $\|\dot{\Sigma}_k - \Sigma_k^*\|_F = O_p\left[\{(p+q)(\log p)/n\}^{1/2}\right]$, then by Lemma 2.3, the next M-step estimator $\hat{\Omega}_k^{(2)}$ would also be bound by $\sum_{k=0}^K \left\|\hat{\Omega}_k^{(2)} - \Omega_k^*\right\|_F = O_p\left[\{(p+q)(\log p)/n\}^{1/2}\right]$. Since our EM algorithm guarantees to end in finite steps, the estimator from EM algorithm would have the same bound as the one-step algorithm.

From Condition 3, we assume there exists $\tilde{\Sigma}_y$ such that

$$\|\tilde{\Sigma}_y - \Sigma_y^*\|_F = O_p\left[\{(p+q)(\log p)/n\}^{1/2}\right].$$

From the E-step expression (3.11), we know that

$$\dot{\Sigma}_0 = (\hat{A}^{(1)})^{-1} + (\hat{A}^{(1)})^{-1} \sum_{l,k=1}^K \left\{\hat{\Omega}_l^{(1)} \tilde{\Sigma}_{y(l,k)} \hat{\Omega}_k^{(1)}\right\} (\hat{A}^{(1)})^{-1}$$

$$\Sigma_0^* = A^{-1} + A^{-1} \sum_{l,k=1}^K \left\{\Omega_l^* \Sigma_{y(l,k)}^* \Omega_k^*\right\} A^{-1}.$$

Define $\Delta_A = \hat{A}^{(1)} - A$, where $A = \sum_{k=0}^K \Omega_k^*$. We know from Theorem 2.1 that

$$\|\Delta_A\|_F = O_p\left[\{(p+q)(\log p)/n\}^{1/2}\right].$$

Using the Woodbury matrix identity twice, we know that

$$\begin{aligned}(\hat{A}^{(1)})^{-1} &= (A+\Delta_A)^{-1} \\ &= A^{-1} - A^{-1}(\Delta_A^{-1} + A^{-1})^{-1}A^{-1} \\ &= A^{-1} - A^{-1}\{\Delta_A - \Delta_A(A+\Delta_A)^{-1}\Delta_A\}A^{-1} \\ &= A^{-1} - A^{-1}\Delta_A A^{-1} + A^{-1}\Delta_A(\Delta_A + A)^{-1}\Delta_A A^{-1}.\end{aligned}$$

By Condition 1, we have $\tau_1 < \phi_{min}(A) < \phi_{max}(A) < (K+1)\tau_2$. Using Lemma 2.4, we have

$$\begin{aligned}\|(\hat{A}^{(1)})^{-1} - A^{-1}\|_F \leq & \|A^{-1}\Delta_A A^{-1}\|_F + \|A^{-1}\Delta_A(\Delta_A + A)^{-1}\Delta_A A^{-1}\|_F \\ \leq & \|A^{-1}\|^2\|\Delta_A\|_F + \|A^{-1}\|^2\|\Delta_A\|_F^2\|(\Delta_A + A)^{-1}\| \qquad (2.64) \\ \leq & \|\Delta_A\|_F/(\tau_1^2) + \|\Delta_A\|_F^2\|(I + A^{-1}\Delta_A)^{-1}A^{-1}\|/(\tau_1^2) \qquad (2.65) \\ \leq & \|\Delta_A\|_F/(\tau_1^2) + \|\Delta_A\|_F^2\|(I + A^{-1}\Delta_A)^{-1}\|/(\tau_1^3) \\ \leq & \|\Delta_A\|_F/(\tau_1^2) + \frac{1}{\tau_1^3}\|\Delta_A\|_F^2/(1 - \|A^{-1}\Delta_A\|) \qquad (2.66) \\ \lesssim & \frac{1}{\tau_1^2}\|\Delta_A\|_F + \frac{2}{\tau_1^3}\|\Delta_A\|_F^2 \qquad (2.67) \\ = & O_p\left[\left\{\frac{(p+q)\log p}{n}\right\}^{1/2}\right].\end{aligned}$$

The inequality of (2.64) is due to Lemma 2.5; inequality of (2.65) is due to the fact that $\|A^{-1}\|^2 = 1/(\phi_{min}(A))^2 \leq 1/(\tau_1^2)$, inequality of (2.66) is due to Lemma 2.4 and inequality of (2.67) can be achieved when n is large enough since

$$\|A^{-1}\Delta_A\| \leq \|A^{-1}\Delta_A\|_F \leq \|A^{-1}\|\,\|\Delta_A\|_F \leq \frac{1}{\tau_1^2}\|\Delta_A\|_F \to 0\,.$$

Next, define

$$\Delta_1 = (\hat{A}^{(1)})^{-1} - A^{-1},\ \Delta_{k,2} = \hat{\Omega}_k^{(1)} - \Omega_k^* \qquad (2.68)$$

$$\{\Delta_{(l,k),3}\}_{1\leq l,k\leq K} = \{\tilde{\Sigma}_{y(l,k)}\}_{1\leq l,k\leq K} - \{\Sigma^*_{y(l,k)}\}_{1\leq l,k\leq K}, \qquad (2.69)$$

where $\|\Delta_1\|_F, \|\Delta_{k,2}\|_F$ and $\|\Delta_{(l,k),3}\|_F$ all have the same rate of $O_p\left[\{(p+q)(\log p)/n\}^{1/2}\right]$. With (2.68) and (2.69), $\dot{\Sigma}_0^{(1)} - \Sigma_0^*$ can be expressed as follows

$$\begin{aligned}
\dot{\Sigma}_0^{(1)} - \Sigma_0^* =& (\hat{A}^{(1)})^{-1} + (\hat{A}^{(1)})^{-1} \sum_{l,k=1}^{K} \left\{ \hat{\Omega}_l^{(1)} \tilde{\Sigma}_{(l,k)} \hat{\Omega}_k^{(1)} \right\} (\hat{A}^{(1)})^{-1} \\
& - A^{-1} - A^{-1} \sum_{l,k=1}^{K} \left\{ \Omega_l^* \Sigma_{y(l,k)}^* \Omega_k^* \right\} A^{-1} \\
& = \Delta_1 + \Delta_1 \Big(\sum_{l,k=1}^{K} \Omega_l^* \Sigma_{y(l,k)}^* \Omega_k^* \Big) A^{-1} + A^{-1} \Big(\sum_{l,k=1}^{K} \Delta_{l,2} \Sigma_{y(l,k)}^* \Omega_k^* \Big) A^{-1} \\
& + A^{-1} \Big(\sum_{l,k=1}^{K} \Omega_l^* \Delta_{(l,k),3} \Omega_k^* \Big) A^{-1} + A^{-1} \Big(\sum_{l,k=1}^{K} \Omega_l^* \Sigma_{y(l,k)}^* \Delta_{k,2} \Big) A^{-1} \\
& + A^{-1} \Big(\sum_{l,k=1}^{K} \Omega_l^* \Sigma_{y(l,k)}^* \Omega_k^* \Big) \Delta_1 + B,
\end{aligned}$$

where B is the remainder terms with following value

$$\begin{aligned}
B =& \Delta_1 \Big(\sum_{l,k=1}^{K} \Delta_{l,2} \Sigma_{y(l,k)}^* \Omega_k^* \Big) A^{-1} + \Delta_1 \Big(\sum_{l,k=1}^{K} \Omega_l^* \Delta_{(l,k),3} \Omega_k^* \Big) A^{-1} + \Delta_1 \Big(\sum_{l,k=1}^{K} \Omega_l^* \Sigma_{y(l,k)}^* \Delta_{k,2} \Big) A^{-1} \\
& + \Delta_1 \Big(\sum_{l,k=1}^{K} \Omega_l^* \Sigma_{y(l,k)}^* \Omega_k^* \Big) \Delta_1 + A^{-1} \Big(\sum_{l,k=1}^{K} \Delta_{l,2} \Delta_{(l,k),3} \Omega_k^* \Big) A^{-1} + A^{-1} \Big(\sum_{l,k=1}^{K} \Delta_{l,2} \Sigma_{y(l,k)}^* \Delta_{k,2} \Big) A^{-1} \\
& + A^{-1} \Big(\sum_{l,k=1}^{K} \Delta_{l,2} \Sigma_{y(l,k)}^* \Omega_k^* \Big) \Delta_1 + A^{-1} \Big(\sum_{l,k=1}^{K} \Omega_l^* \Delta_{(l,k),3} \Delta_{k,2} \Big) A^{-1} + A^{-1} \Big(\sum_{l,k=1}^{K} \Omega_l^* \Delta_{(l,k),3} \Omega_k^* \Big) \Delta_1 \\
& + A^{-1} \Big(\sum_{l,k=1}^{K} \Omega_l^* \Sigma_{y(l,k)}^* \Delta_{k,2} \Big) \Delta_1 + \Delta_1 \Big(\sum_{l,k=1}^{K} \Delta_{l,2} \Delta_{(l,k),3} \Omega_k^* \Big) A^{-1} + \Delta_1 \Big(\sum_{l,k=1}^{K} \Delta_{l,2} \Sigma_{y(l,k)}^* \Delta_{k,2} \Big) A^{-1} \\
& + \Delta_1 \Big(\sum_{l,k=1}^{K} \Delta_{l,2} \Sigma_{y(l,k)}^* \Omega_k^* \Big) \Delta_1 + \Delta_1 \Big(\sum_{l,k=1}^{K} \Omega_l^* \Delta_{(l,k),3} \Delta_{k,2} \Big) A^{-1} + \Delta_1 \Big(\sum_{l,k=1}^{K} \Omega_l^* \Delta_{(l,k),3} \Omega_k^* \Big) \Delta_1 \\
& + \Delta_1 \Big(\sum_{l,k=1}^{K} \Omega_l^* \Sigma_{y(l,k)}^* \Delta_{k,2} \Big) \Delta_1 + \ldots + \Delta_1 \Big(\sum_{l,k=1}^{K} \Delta_{l,2} \Delta_{(l,k),3} \Delta_{k,2} \Big) \Delta_1.
\end{aligned}$$

Each term of B is product of at least two Δ term, where Δ are Δ_1, $\Delta_{k,2}$ or $\Delta_{(l,k),3}$. Additionally, we know $\|\Delta_1\|_F, \|\Delta_{k,2}\|_F$ and $\|\Delta_{(l,k),3}\|_F$ all have the same rate of $O_p\left[\{(p+q)(\log p)/n\}^{1/2}\right]$, and $\|\Omega_l^*\|_F$, $\|\Sigma_{y(l,k)}^*\|_F$ and $\|A\|_F$ are bounded. Thus $\|B\|_F = O_p(\|\Delta\|_1^2) = o_p(\|\Delta\|_1^2)$ as n and p go to infinity. We then can bound $\|\dot{\Sigma}_0^{(1)} - \Sigma_0^*\|$ as follows

$$\|\dot{\Sigma}_0^{(1)} - \Sigma_0^*\|_F \leq \left\| \Delta_1 \left(\sum_{l,k=1}^{K} \Omega_l^* \Sigma_{y(l,k)}^* \Omega_k^* \right) A^{-1} \right\|_F + \left\| A^{-1} \left(\sum_{l,k=1}^{K} \Delta_{l,2} \Sigma_{y(l,k)}^* \Omega_k^* \right) A^{-1} \right\|_F$$
$$+ \left\| A^{-1} \left(\sum_{l,k=1}^{K} \Omega_l^* \Delta_{(l,k),3} \Omega_k^* \right) A^{-1} \right\|_F + \left\| A^{-1} \left(\sum_{l,k=1}^{K} \Omega_l^* \Sigma_{y(l,k)}^* \Delta_{k,2} \right) A^{-1} \right\|_F$$
$$+ \left\| A^{-1} \left(\sum_{l,k=1}^{K} \Omega_l^* \Sigma_{y(l,k)}^* \Omega_k^* \right) \Delta_1 \right\|_F + \|\Delta_1\|_F + o_p(\|\Delta_1\|_F)$$
$$\leq \|\Delta_1\|_F K^2 \tau_2^2/(\tau_1^2) + \sum_{l=1}^{K} \|\Delta_{l,2}\|_F K \tau_2/(\tau_1^3) + \sum_{l,k=1}^{K} \|\Delta_{(l,k),3}\|_F \tau_2^2/(\tau_1^2) + \sum_{k=1}^{K} \|\Delta_{k,2}\|_F K \tau_2/(\tau_1^3)$$
$$+ \|\Delta_1\|_F K^2 \tau_2^2/(\tau_1^2) + o_p(\|\Delta_1\|_F) = O_p \left[\left\{ \frac{(p+q)\log p}{n} \right\}^{1/2} \right].$$

Similarly for $\dot{\Sigma}_k^{(1)}$, we can prove that $\|\dot{\Sigma}_k^{(1)} - \Sigma_k^*\|_F = O_p\left[\{(p+q)(\log p)/n\}^{1/2}\right]$. Then by Lemma 2.3, the corresponding M-step estimator $\hat{\Omega}_k^{(2)}$ would also be bound by $\sum_{k=0}^{K} \left\|\hat{\Omega}_k^{(2)} - \Omega_k^*\right\|_F = O_p\left[\{(p+q)(\log p)/n\}^{1/2}\right]$, and follow previous step we can show $\|\dot{\Sigma}_k^{(2)} - \Sigma_k^*\|_F = O_p\left[\{(p+q)(\log p)/n\}^{1/2}\right]$ for $k = 0, \ldots, K$ and so on. Since our EM algorithm guarantees to end in finite steps, the estimator from EM algorithm would have the same bound as the One-step algorithm and we finish the proof.

2.7.7 Proof of Theorem 2.3

This proof follows a similar argument to that in Lam and Fan (2009, Theorem 2). The derivative for $\mathcal{W}_k(\Omega_k)$ w.r.t. $\omega_{k(i,j)}$ can be written as

$$\frac{\partial \mathcal{W}_k(\Omega_k)}{\partial \omega_{k(i,j)}} = 2\{\hat{\sigma}'_{k(i,j)} - \sigma_{k(i,j)} + \lambda \operatorname{sign}(\omega_{k(i,j)})\},$$

where $\lambda = \lambda_1$ for $k = 1, \ldots, K$ and $\lambda = \lambda_2$ otherwise. For $(i,j) \in T_k^c$, it is sufficient to show that the sign of $\partial \mathcal{W}_k(\Omega_k)/\partial \omega_{k(i,j)}$ at the local minimum $\hat{\omega}_{k(i,j)}$ only depends on the sign of $\hat{\omega}_{k(i,j)}$ with probability tending to 1. Namely, the rate for λ dominates the rate of $\hat{\sigma}'_{k(i,j)} - \sigma_{k(i,j)}$. To see that, without generality we suppose $\hat{\omega}_{k(i,j)} < 0$ for $(i,j) \in T_k^c$. Then there is a small $\epsilon > 0$ such that $\hat{\omega}_{k(i,j)} + \epsilon < 0$. Since $\hat{\omega}_{k(i,j)}$ is the local minimum, $\partial \mathcal{W}_k(\Omega_k)/\partial \omega_{k(i,j)}$ should be positive at $\hat{\omega}_{k(i,j)} + \epsilon$ for small $\epsilon > 0$. Because $\partial \mathcal{W}_k(\Omega_k)/\partial \omega_{k(i,j)}$ has the same sign at $\hat{\omega}_{k(i,j)}$ and is continuous function, $\partial \mathcal{W}_k(\Omega_k)/\partial \omega_{k(i,j)}$ should be negative

at $\hat{\omega}_{k(i,j)}+\epsilon$ for small $\epsilon > 0$, which contradicts with previous conclusion. Thus, $\hat{\omega}_{k(i,j)}$ would be 0.

Let $\hat{\Omega}_k = \Omega_k^* + \Delta_k$ and $\check{\Sigma}_k = \hat{\Omega}_k^{-1}$. Since $\sum_{k=0}^{K} \left\|\hat{\Omega}_k - \Omega_k^*\right\| = O_p(\eta_n)$, we have $\Delta_k = O_p(\eta_n)$. Using the Woodbury formula twice, we have that

$$\begin{aligned}
\check{\Sigma}_k &= (\Omega_k^* + \Delta_k)^{-1} = \Sigma_k^* - \Sigma_k^*(\Delta_k^{-1} + \Sigma_k^*)^{-1}\Sigma_k^* \\
&= \Sigma_k^* - \Sigma_k^*(\Delta_k - \Delta_k(\Omega_k^* + \Delta_k)^{-1}\Delta_k)\Sigma_k^* \\
&= \Sigma_k^* - \Sigma_k^*\Delta_k\Sigma_k^* + \Sigma_k^*\Delta_k(\Delta_k + \Omega_k^*)^{-1}\Delta_A\Sigma_k^*.
\end{aligned}$$

By Condition 1, we have ${\tau_2}^{-1} < \phi_{min}(\Sigma_k^*) < \phi_{max}(\Sigma_k^*) < {\tau_1}^{-1}$ and we can get

$$\begin{aligned}
\|\check{\Sigma}_k - \Sigma_k^*\| &\leq \|\Sigma_k^*\Delta_k\Sigma_k^*\| + \|\Sigma_k^*\Delta_k(\Delta_k + \Omega_k^*)^{-1}\Delta_k\Sigma_k^*\| \\
&\leq \|\Sigma_k^*\|^2\|\Delta_k\| + \|\Sigma_k^*\|^2\|\Delta_k\|^2\|(\Delta_k + \Omega_k^*)^{-1}\| \\
&\leq \|\Delta_k\|/(\tau_1^2) + \|\Delta_k\|^2\|(I + \Sigma_k^*\Delta_k)^{-1}\|\,\|\Sigma_k^*\|/(\tau_1^2) \\
&\leq \|\Delta_k\|/(\tau_1^2) + \frac{1}{\tau_1^3}\|\Delta_k\|^2(1 - \|\Sigma_k^*\Delta_k\|)^{-1} \qquad (2.70) \\
&\lesssim \|\Delta_k\|/(\tau_1^2) + \frac{2}{\tau_1^3}\|\Delta_k\|^2 \qquad (2.71) \\
&= O_p(\eta_n),
\end{aligned}$$

the inequality of (2.70) is due to Lemma 2.4, since $\|\Sigma_k^*\Delta_k\| < 1$ when n is large enough. The inequality of (2.71) holds when n large enough since

$$\|\Sigma_k^*\Delta_k\| \leq \|\Sigma_k^*\Delta_k\|_F \leq \|\Sigma_k^*\|\,\|\Delta_k\| \leq \frac{1}{\tau_1}\|\Delta_k\| \to 0\,.$$

From the proof of Theorem 2.1, we also know $\|\mathrm{vec}(\hat{\Sigma}_k') - \mathrm{vec}(\Sigma_k^*)\|_\infty = O_p[\{(\log p)/n\}^{1/2}]$ and the $\partial\mathcal{W}_k(\Omega_k)/\partial\omega_{k(i,j)}$ at the local minimum $\hat{\omega}_{k(i,j)}$ is $2\{\hat{\sigma}'_{k(i,j)} - \check{\sigma}_{k(i,j)} + \lambda\,\mathrm{sign}(\hat{\omega}_{k(i,j)})\}$. Combining the results we have

$$\begin{aligned}
\max_{i,j}|\hat{\sigma}'_{k(i,j)} - \check{\sigma}_{k(i,j)}| &= \max_{i,j}|\hat{\sigma}'_{k(i,j)} - \sigma^*_{k(i,j)} + \sigma^*_{k(i,j)} - \check{\sigma}_{k(i,j)}| \\
&\leq \max_{i,j}|\hat{\sigma}'_{k(i,j)} - \sigma^*_{k(i,j)}| + \max_{i,j}|\sigma^*_{k(i,j)} - \check{\sigma}_{k(i,j)}| \\
&\leq |\mathrm{vec}(\hat{\Sigma}_k') - \mathrm{vec}(\Sigma_k^*)|_\infty + \|\check{\Sigma}_k - \Sigma_k^*\| \\
&= O_p[\{(\log p)/n\}^{1/2} + \eta_n].
\end{aligned}$$

Hence if $\lambda \succeq (\log p/n)^{1/2} + \eta_n$, the term $\lambda \operatorname{sign}(\omega_{k(i,j)})$ dominates over $\hat{\sigma}_{k(i,j)} - \check{\sigma}_{k(i,j)}$ with probability tending to 1, making the sign of the $\partial \mathcal{W}_k(\Omega_k)/\partial \omega_{k(i,j)}$ depends only on the sign of $\omega_{k(i,j)}$ at $\hat{\omega}_{k(i,j)}$, thus completes the proof.

2.7.8 Proof of Theorem 2.4

Assume the last iteration for EM minimizes

$$\mathcal{W}'_k(\Omega_k) = \operatorname{tr}(\dot{\Sigma}_k \mathbf{\Omega}_k) - \log(\det(\Omega_k)) + \lambda \sum_{i \neq j} \left|\omega_{k(i,j)}\right|,$$

where $\lambda = \lambda_1$ for $k = 1, \ldots, K$ and $\lambda = \lambda_2$ otherwise. The derivative for $\mathcal{W}'_k$ w.r.t. $\omega_{k(i,j)}$ is

$$\frac{\partial \mathcal{W}'_k(\Omega_k)}{\partial \omega_{k(i,j)}} = 2(\dot{\sigma}_{k(i,j)} - \sigma_{k(i,j)} + \lambda \operatorname{sign}(\omega_{k(i,j)})).$$

Similar to the proof of Theorem 2.3, it is enough to show that for $(i,j) \in T_k^c$, that the sign of $\partial \mathcal{W}_k(\Omega_k)/\partial \omega_{k(i,j)}$ at the local minimum $\hat{\omega}_{k(i,j)}$ only depends on the sign of $\hat{\omega}_{k(i,j)}$ with probability tending to 1. Let $\hat{\Omega}_k$ be the local minimum in Theorem 2.2, and define $\check{\Sigma}_k = \hat{\Omega}_k^{-1}$.

From the proof of Theorem 2.2 and 2.2, we have shown $\|\dot{\Sigma}_k - \Sigma_k^*\|_F = O_p[\{(p+q)(\log p)/n\}^{1/2}]$ and $\|\check{\Sigma}_k - \Sigma_k^*\| = O_p(\eta_n)$. Combining the results yields that

$$\begin{aligned}
\max_{i,j} |\dot{\sigma}_{k(i,j)} - \check{\sigma}_{k(i,j)}| &\leq \max_{i,j} |\dot{\sigma}_{k(i,j)} - \sigma^*_{k(i,j)}| + \max_{i,j} |\sigma^*_{k(i,j)} - \check{\sigma}_{k(i,j)}| \\
&\leq \|\operatorname{vec}(\dot{\Sigma}_k) - \operatorname{vec}(\Sigma^*)_k\|_\infty + \|\check{\Sigma}_k - \Sigma_k^*\| \\
&\leq \|\dot{\Sigma}_k - \Sigma_k^*\|_F + \|\check{\Sigma}_k - \Sigma_k^*\| \\
&= O_p[\{(p+q)(\log p)/n\}^{1/2} + \eta_n].
\end{aligned}$$

Therefore, if $\lambda \succeq \{(p+q)(\log p)/n\}^{1/2} + \eta_n$, the sign of $\frac{\partial \mathcal{W}'_k(\Omega_k)}{\partial \omega_{k(i,j)}}$ only depends on $\operatorname{sgn}(\omega_{k(i,j)})$. This completes the proof.

2.7.9 Extension with Similarity Parameter α_k

As suggested by one of the reviewers, we can extend our model to allow systemic layer varying among tissues. For example, since muscle and adipose both are developed

from mesoderm, they are more close related with each other compared with brain which is developed from Ectoderm. We extend our model to the follows:

$$Y_{k,i} = X_{k,i} + \alpha_k z_i \;\; (k = 1, \ldots, K;\; i = 1, \ldots, n), \tag{2.72}$$

where $X_{k,i} \sim \mathcal{N}(0, \Sigma_k)$, $Z_i \sim \mathcal{N}(0, \Sigma_0)$, α_k is the similarity parameter and $X_{k,i}$ and Z_i are independent with each other. For identifiability issue, we also assume $\max(\mathrm{diag}(\Sigma_0)) = 1$.

Similar to section 2.7.1, the likelihood of Y can be written as follows:

$$\begin{aligned}
f(Y) &= \int f(Y \mid Z) f(Z) \mathrm{d}Z \\
&\propto \int \exp\left\{ \sum_{k=1}^{K} (Y_k - \alpha_k Z)^{\mathrm{T}} \Omega_k (Y_k - \alpha_k Z) + Z^{\mathrm{T}} \Omega_0 Z \right\} \mathrm{d}Z \\
&= \exp\left(\sum_{k=1}^{K} Y_k^{\mathrm{T}} \Omega_k Y_k \right) \int \exp\left\{ Z^{\mathrm{T}} \big(\sum_{k=1}^{K} \alpha_k^2 \Omega_k + \Omega_0 \big) Z - 2 \big(\sum_{k=1}^{K} \alpha_k Y_k^{\mathrm{T}} \Omega_k \big) z \right\} \mathrm{d}z \\
&= \exp\left(\sum_{k=1}^{K} Y_k^{\mathrm{T}} \Omega_k Y_k \right) \int \exp\left(z^{\mathrm{T}} A_{\mathrm{ext}} z - 2 c_{\mathrm{ext}}^{\mathrm{T}} z \right) \mathrm{d}z \\
&= \exp\Big(\sum_{k=1}^{K} Y_k^{\mathrm{T}} \Omega_k Y_k \Big) \exp\Big(- c_{\mathrm{ext}}^{\mathrm{T}} A_{\mathrm{ext}}^{-1} c_{\mathrm{ext}} \Big) \int \exp\left\{ (A_{\mathrm{ext}} Z - c_{\mathrm{ext}})^{\mathrm{T}} A_{\mathrm{ext}}^{-1} (A_{\mathrm{ext}} Z - c_{\mathrm{ext}}) \right\} \mathrm{d}Z \\
&\propto \exp\left\{ \sum_{k=1}^{K} Y_k^{\mathrm{T}} \Omega_k Y_k - (\sum_{k=1}^{K} \alpha_k Y_k^{\mathrm{T}} \Omega_k) A_{\mathrm{ext}}^{-1} (\sum_{k=1}^{K} \alpha_k \Omega_k Y_k) \right\} \\
&= \exp\left[Y^{\mathrm{T}} \left(\{ {}_d\Omega_k \}_{1 \le k \le K} - \{ \alpha_l \alpha_k \Omega_k A_{\mathrm{ext}}^{-1} \Omega_k \}_{1 \le l,k \le K} \right) Y \right] \\
&= \exp(Y^{\mathrm{T}} \Omega_Y Y)\,,
\end{aligned}$$

thus $Y \sim \mathcal{N}(0, \left[\{ {}_d\Omega_k \}_{k=1}^{K} - \{ \alpha_l \alpha_k \Omega_l A_{\mathrm{ext}}^{-1} \Omega_k \}_{1 \le l,k \le K} \right]^{-1})$, where $A_{\mathrm{ext}} = \sum_{k=1}^{K} \alpha_k^2 \Omega_k + \Omega_0$ and $c_{\mathrm{ext}} = \sum_{k=1}^{K} \alpha_k \Omega_k Y_k$. Therefore,

$$\Omega_Y = \{ {}_d\Omega_k \}_{k=1}^{K} - \{ \alpha_l \alpha_k \Omega_k A_{\mathrm{ext}}^{-1} \Omega_k \}_{1 \le l,k \le K}.$$

Similarly, we derive $\det(\Omega_Y)$. We know that

$$\Sigma_Y = \{ {}_d\,\Sigma_k \}_{1 \le k \le K} + \begin{pmatrix} \alpha_1 I \\ \vdots \\ \alpha_K I \end{pmatrix} \begin{pmatrix} \alpha_1 \Sigma_0 & \cdots & \alpha_K \Sigma_0 \end{pmatrix}$$

$$= \{_d\ \Sigma_k\}_{1\le k\le K} \left\{ I + \{_d\ \Omega_k\}_{1\le k\le K} \begin{pmatrix} \alpha_1 I \\ \vdots \\ \alpha_K I \end{pmatrix} \begin{pmatrix} \alpha_1\Sigma_0 & \cdots & \alpha_K\Sigma_0 \end{pmatrix} \right\},$$

and the $\det(\Sigma_Y)$ can be expressed as follows:

$$\begin{aligned}
\det(\Sigma_Y) &= \left\{ \prod_{k=1}^{K} \det(\Sigma_k) \right\} \det \left\{ I + \begin{pmatrix} \alpha_1\Sigma_0 & \cdots & \alpha_K\Sigma_0 \end{pmatrix} \{_d\Omega_k\}_{1\le k\le K} \begin{pmatrix} \alpha_1 I \\ \vdots \\ \alpha_K I \end{pmatrix} \right\} \\
&= \left\{ \prod_{k=1}^{K} \det(\Sigma_k) \right\} \det \left(I + \Sigma_0 \sum_{k=1}^{K} \alpha_k^2 \Omega_k \right) \\
&= \left\{ \prod_{k=1}^{K} \det(\Sigma_k) \right\} \det \left(\Sigma_0\Omega_0 + \Sigma_0 \sum_{k=1}^{K} \alpha_k^2 \Omega_k \right) \\
&= \left\{ \prod_{k=0}^{K} \det(\Sigma_k) \right\} \det(A_{\text{ext}}),
\end{aligned}$$

where $A_{\text{ext}} = \Omega_0 + \sum_{k=1}^{K} \alpha_k^2 \Omega_k$. Therefore, we have $\log\{\det(\Omega_Y)\} = \sum_{k=0}^{K} \log\{\det(\Omega_k)\} - \log\{\det(A_{\text{ext}})\}$. Combining previous result, we can write the log-likelihood as

$$\begin{aligned}
\mathcal{L}(\Omega_Y) =& -\frac{npK}{2}\log(2\pi) + \frac{n}{2}\big[\log\{\det(\Omega_Y)\} - \operatorname{tr}\big(\hat{\Sigma}_Y\Omega_Y\big)\big] \\
=& -\frac{npK}{2}\log(2\pi) + \frac{n}{2}\Big[\sum_{k=0}^{K}\log\{\det(\Omega_k)\} - \log\{\det(A_{\text{ext}})\}\Big] \\
& - \frac{n}{2}\operatorname{tr}\Big(\hat{\Sigma}_Y\Big[\{_d\Omega_k\}_{1\le k\le K} - \{\alpha_l\alpha_k\Omega_l A_{\text{ext}}^{-1}\Omega_k\}_{1\le l,k\le K}\Big]\Big).
\end{aligned}$$

Under this setting, we have

$$\begin{pmatrix} Z \\ Y_1 \\ \vdots \\ Y_K \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & \alpha_1\Sigma_0 & \dots & \alpha_K\Sigma_0 \\ \alpha_1\Sigma_0 & \Sigma_1 + \alpha_1^2\Sigma_0 & \dots & \alpha_1\alpha_k\Sigma_0 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_K\Sigma_0 & \alpha_1\alpha_k\Sigma_0 & \dots & \Sigma_K + \alpha_k^2\Sigma_0 \end{pmatrix} \right).$$

For simplicity, we represent $\{\Omega\}_{k=0}^{K}$ as Ω and $\{\alpha_k\}_{k=1}^{K}$ as α, and thus have

$$Z \mid Y, \Omega, \alpha \sim \mathcal{N}\left((\alpha_1\Sigma_0, \ldots, \alpha_K\Sigma_0)\Sigma_Y^{-1} \begin{pmatrix} Y_1 \\ \vdots \\ Y_K \end{pmatrix}, \Sigma_0 - (\alpha_1\Sigma_0, \ldots, \alpha_K\Sigma_0)\Sigma_Y^{-1}\Omega_Y \begin{pmatrix} \alpha_1\Sigma_0 \\ \vdots \\ \alpha_K\Sigma_0 \end{pmatrix} \right).$$

We can derive $E(Z \mid Y, \Omega, \alpha)$, $\mathrm{Var}(Z \mid Y, \Omega, \alpha)$ and $E(ZZ^{\mathrm{T}} \mid Y, \Omega, \alpha)$ as follows:

$$\begin{aligned}
E(Z \mid Y, \Omega, \alpha) &= (\alpha_1\Sigma_0, \ldots, \alpha_K\Sigma_0)\Omega_Y \begin{pmatrix} Y_1 \\ \vdots \\ Y_K \end{pmatrix} \\
&= (\alpha_1\Sigma_0, \ldots, \alpha_K\Sigma_0)\Big(\{_d\Omega_k\}_{1\le k\le K} - \begin{pmatrix} \alpha_1\Omega_1 \\ \vdots \\ \alpha_K\Omega_k \end{pmatrix} (\alpha_1 A_{\text{ext}}^{-1}\Omega_1, \ldots, \alpha_k A_{\text{ext}}^{-1}\Omega_k) \Big) \begin{pmatrix} Y_1 \\ \vdots \\ Y_K \end{pmatrix} \\
&= \Big((\alpha_1\Sigma_0\Omega_1, \ldots, \alpha_K\Sigma_0\Omega_K) - \Sigma_0\Big(\sum_{k=1}^{K} \alpha_k^2\Omega_k \Big)(\alpha_1 A_{\text{ext}}^{-1}\Omega_1, \ldots, \alpha_k A_{\text{ext}}^{-1}\Omega_k) \Big) \begin{pmatrix} Y_1 \\ \vdots \\ Y_K \end{pmatrix} \\
&= \Big((\alpha_1\Sigma_0\Omega_1, \ldots, \alpha_K\Sigma_0\Omega_K) - \Sigma_0\big(A_{\text{ext}} - \Omega_0 \big)(\alpha_1 A_{\text{ext}}^{-1}\Omega_1, \ldots, \alpha_k A_{\text{ext}}^{-1}\Omega_k) \Big) \begin{pmatrix} Y_1 \\ \vdots \\ Y_K \end{pmatrix} \\
&= (\alpha_1 A_{\text{ext}}^{-1}\Omega_1, \ldots, \alpha_k A_{\text{ext}}^{-1}\Omega_k) \begin{pmatrix} Y_1 \\ \vdots \\ Y_K \end{pmatrix} \\
&= A_{\text{ext}}^{-1} c_{\text{ext}}, \\
\mathrm{Var}(Z \mid Y, \Omega, \alpha) &= \Sigma_0 - (\alpha_1\Sigma_0, \ldots, \alpha_K\Sigma_0)\Omega_Y \begin{pmatrix} \alpha_1\Sigma_0 \\ \vdots \\ \alpha_K\Sigma_0 \end{pmatrix} \\
&= \Sigma_0 - (\alpha_1 A_{\text{ext}}^{-1}\Omega_1, \ldots, \alpha_k A_{\text{ext}}^{-1}\Omega_k) \begin{pmatrix} \alpha_1\Sigma_0 \\ \vdots \\ \alpha_K\Sigma_0 \end{pmatrix}
\end{aligned}$$

$$
\begin{aligned}
&= \Sigma_0 - A_{\text{ext}}^{-1}\Big(\sum_{k=1}^{K}\alpha_k^2\Omega_k\Big)\Sigma_0 \\
&= \Sigma_0 - A_{\text{ext}}^{-1}\Big(A_{\text{ext}} - \Omega_0\Big)\Sigma_0 \\
&= A_{\text{ext}}^{-1} \\
E(ZZ^{\mathrm{T}} \mid Y,\Omega,\alpha) &= \mathrm{Var}(Z \mid Y,\Omega,\alpha) + E(Z \mid Y,\Omega,\alpha)E(Z \mid Y,\Omega,\alpha)^{\mathrm{T}} \\
&= A_{\text{ext}}^{-1} + A_{\text{ext}}^{-1}c_{\text{ext}}c_{\text{ext}}^{\mathrm{T}}A_{\text{ext}}^{-1},
\end{aligned}
$$

where $c_{\text{ext}} = \sum_{k=1}^{K}\alpha_k\Omega_k Y_k$.

As in the manuscript, let $y_{k,i}$ be the realization of $Y_{k,i}$ and y be the n by Kp dimensional data matrix. We can modify our EM algorithm to calculate α_k and Ω_k jointly. The modified EM algorithm is described as follows:

The E step calculates:

$$
\begin{aligned}
\mathcal{Q}(\Omega\mid, y, \alpha^{(t)}, \Omega^{(t)}) &= \sum_{k=1}^{K} E\Big[\mathcal{L}\big(x_k \mid y, \alpha^{(t)}, \Omega^{(t)}\big)\Big] + E\Big[\mathcal{L}\big(z \mid y, \alpha^{(t)}, \Omega^{(t)}\big)\Big] \\
&\propto \sum_{k=1}^{K}\left(\log\{\det(\Omega_k)\} - \mathrm{tr}\left[\Omega_k E\left\{\sum_{i=1}^{n}(y_{k,i}-\alpha_k z_i)(y_{k,i}-\alpha_k z_i)^{\mathrm{T}}/n \mid y, \alpha^{(t)}, \Omega^{(t)}\right\}\right]\right) \\
&\quad + \log\{\det(\Omega_0)\} - \mathrm{tr}\left\{\Omega_0 E\left(\sum_{i=1}^{n} z_i z_i^{\mathrm{T}}/n \mid y, \alpha^{(t)}, \Omega^{(t)}\right)\right\} \\
&= \sum_{k=1}^{K}\Bigg(\log\{\det(\Omega_k)\} - \mathrm{tr}\Big[\Omega_k\sum_{i=1}^{n}(y_{k,i}y_{k,i}^{\mathrm{T}})/n - \Omega_k\sum_{i=1}^{n}\Big\{\alpha_k y_{k,i}E\Big(z_i \mid y, \alpha^{(t)}, \Omega^{(t)}\Big)^{\mathrm{T}}\Big\}/n \\
&\quad - \Omega_k\sum_{i=1}^{n}\Big\{\alpha_k E\Big(z_i \mid y, \alpha, \Omega\Big)y_{k,i}^{\mathrm{T}}\Big\}/n + \Omega_k\sum_{i=1}^{n}\Big\{\alpha_k^2 E\big(z_i z_i^{\mathrm{T}} \mid y, \alpha^{(t)}, \Omega^{(t)}\big)\Big\}/n\Big]\Bigg) \\
&\quad + \log\{\det(\Omega_0)\} - \mathrm{tr}\left\{\Omega_0 E\Big(\sum_{i=1}^{n} z_i z_i^{\mathrm{T}}/n \mid y, \alpha^{(t)}, \Omega^{(t)}\Big)\right\} \qquad (2.73) \\
&= \sum_{k=0}^{K}\left[\log\{\det(\Omega_k)\} - \mathrm{tr}\left(\Omega_k\dot{\Sigma}_k^{(t)}\right)\right].
\end{aligned}
$$

The M step: Since the function in (2.73) is biconcave function, we can first fix $\alpha^{(t)}$ and update Ω by solving

$$
\Omega^{(t+1)} = \underset{\Omega}{\mathrm{argmin}} - \mathcal{Q}(\Omega \mid y, \alpha^{(t)}, \Omega^{(t)}) + \lambda_1\sum_{i\neq j}\sum_{k=1}^{K}\big|\omega_{k(i,j)}\big| + \lambda_2\sum_{i\neq j}\big|\omega_{0(i,j)}\big|, \qquad (2.74)
$$

where $\alpha^{(t)}$ and $\Omega^{(t)}$ denote the estimates from the t-th iteration. We then normalize $\Omega_0^{(t+1)}$ by

$$\Omega_0^{(t+1)} = \frac{\Omega_0^{(t+1)}}{\max\{\mathrm{diag}(\Omega_0^{(t+1)})\}}.$$

Then fixing $\Omega^{(t+1)}$, we update α_k by the following equation

$$\hat{\alpha}_k^{(t+1)} = \frac{\mathrm{tr}\left[\hat{\Omega}_k^{(t+1)} \sum_{i=1}^n \left\{ y_{k,i} E(z_i \mid y, \alpha^{(t)}, \Omega^{(t)})^{\mathrm{T}} + E(z_i \mid y, \alpha^{(t)}, \Omega^{(t)}) y_{k,i}^{\mathrm{T}} \right\}\right]}{2\mathrm{tr}\left(\hat{\Omega}_k^{(t+1)} \sum_{i=1}^n E(z_i z_i^{\mathrm{T}} \mid y, \alpha^{(t)}, \Omega^{(t)})\right)}. \quad (2.75)$$

The value of $\dot{\Sigma}_k^{(t)}$ is defined as

$$\begin{aligned}
\dot{\Sigma}_k^{(t)} =& E\left(\sum_{i=1}^n (y_{k,i} - \alpha_k z_i)(y_{k,i} - \alpha_k z_i)^{\mathrm{T}}/n \mid y, \alpha^{(t)}, \Omega^{(t)}\right) \\
=& \sum_{i=1}^n \left\{ y_{k,i} y_{k,i}^{\mathrm{T}} - \alpha_k y_{k,i} E(z_i^{\mathrm{T}} \mid y, \alpha^{(t)}, \Omega^{(t)}) \right\}/n \\
& - \sum_{i=1}^n \left\{ \alpha_k E(z_i \mid y, \alpha^{(t)}, \Omega^{(t)}) y_{k,i}^{\mathrm{T}} + \alpha_k^2 E(z_i z_i^{\mathrm{T}} \mid y, \alpha^{(t)}, \Omega^{(t)}) \right\}/n \\
=& \ddot{\Sigma}_{Y(k,k)} - \sum_{l=1}^K \left(\ddot{\Sigma}_{Y(k,l)} \Omega_l^{(t)} \right) (A_{\mathrm{ext}}^{(t)})^{-1} - (A_{\mathrm{ext}}^{(t)})^{-1} \sum_{l=1}^K \left(\Omega_l^{(t)} \ddot{\Sigma}_{Y(l,k)} \right) \\
& + (A_{\mathrm{ext}}^{(t)})^{-1} \sum_{l,k=1}^K \left(\Omega_l^{(t)} \ddot{\Sigma}_{Y(l,k)} \Omega_k^{(t)} \right) (A_{\mathrm{ext}}^{(t)})^{-1} + (A_{\mathrm{ext}}^{(t)})^{-1}, \qquad (k = 1, \ldots, K). \quad (2.76a)
\end{aligned}$$

$$\begin{aligned}
\dot{\Sigma}_0^{(t)} =& \sum_{i=1}^n E\left(z_i z_i^{\mathrm{T}} \mid y, \alpha^{(t)}, \Omega^{(t)} \right)/n \\
=& (A_{\mathrm{ext}}^{(t)})^{-1} + (A_{\mathrm{ext}}^{(t)})^{-1} \sum_{l,k=1}^K \left(\Omega_l^{(t)} \ddot{\Sigma}_{Y(l,k)} \Omega_k^{(t)} \right) (A_{\mathrm{ext}}^{(t)})^{-1}, \quad (2.76b)
\end{aligned}$$

CHAPTER 3

Estimation of Gaussian Graphical Model from Data with Dependent Noise Structure

3.1 Introduction

Graphical models have been widely used in a broad range of field to investigate the relationships among variables. Gaussian graphical models (GGMs) is the simplest graphical model for contentious random vector which jointly follows a multivariate Gaussian distribution.A central question for Gaussian graphical models (GGMs) is to recover the structure of an undirected Gaussian graph. Let $\mathcal{G} = (V, E)$ be an undirected graph representing the conditional dependence relationship between components of a random vector $y = (y_1, \ldots, y_p)$ as follows. The vertex set $V = \{V_1, \ldots, V_p\}$ represents the components of y. The edge set E consists of pairs (i, j) indicating the conditional dependence between y_i and y_j given all other components. In applications, the fundamental question for GGMs is to recover the edge set E. It has been shown that recovering the structure of GGMs is equivalent to recovering the support of the population precision matrix of the data (Dempster et al., 1977). We assume $y = (y_1, \ldots, y_p)^{\mathrm{T}} \sim \mathcal{N}(\mu, \Sigma)$, where $\Sigma = (\sigma_{ij})$ is the population covariance matrix. The precision matrix, denoted as $\Omega = (\omega_{ij})$, is the inverse of covariance matrix. There is an edge between V_i and V_j, i.e. $(i, j) \in E$, if and only if $\omega_{ij} \neq 0$. Consequently, the support recovery of the precision matrix Ω equals the recovery of the structure of the graph $\mathcal{G}$.

The problems of estimating a large sparse precision matrix and recovering its support have drawn considerable recent attention. There are mainly two approaches in the literature. The first one is a penalized likelihood estimation approach with a lasso-type penalty on entries of the precision matrix (see for example Yuan and Lin, 2007; Banerjee et al., 2008; d'Aspremont et al., 2008; Rothman et al., 2008; Ravikumar et al., 2011). The other one is the neighbourhood based approach, by running a lasso-type regression or Dantzig type

selection of each variable on all the rest of variables to estimate the precision matrix column by column. See for example Meinshausen and Bühlmann (2006), Yuan (2010), Cai et al. (2011) and Sun and Zhang (2013). The optimal convergence rate and selection consistency of such penalized estimation schemes have also been described in theoretical studies (for example, Rothman et al., 2008; Lam and Fan, 2009; Sun and Zhang, 2013).

In spite of an extensive literature on the topic, a notable drawback of existing methods for estimating GGMs is that they ignore the existence of measurement error. Measurement error is both common and varied in biological data. An example of such error is illustrated in Figure 3.1, depicting microarray data. Each blue dot, (x, y), represents two measurements of the expression level of a single gene, in the same individual; i.e., a pair of technical replicates. Each green dot represents two measurements of the expression level of a single gene, but in *different* individuals; i.e., a pair of biological replicates. As shown in Figure 3.1, a large proportion of the total variation among patients is from measurement error. Moreover, Labaj et al. (2011) showed that RNA-seq could reliably quantify expression of only 30% of the genes with a relative error less than 20% of the total variance. Many factors could introduce variation to microarray and RNA-Seq result, including:

1. the degradation rates of RNA among different genes in RNA collection procedure;
2. the amplification efficiency in the PCR step(Hansen et al., 2010);
3. mapping methods (Degner et al., 2009) used in an RNA-seq experiment;
4. molecular constitution and secondary structure of the RNA sample. (Hansen et al., 2010; Li et al., 2010a).

Therefor, new methodology is needed to recover the structure of the graph $\mathcal{G}$ while taking account the underlying measurement error.

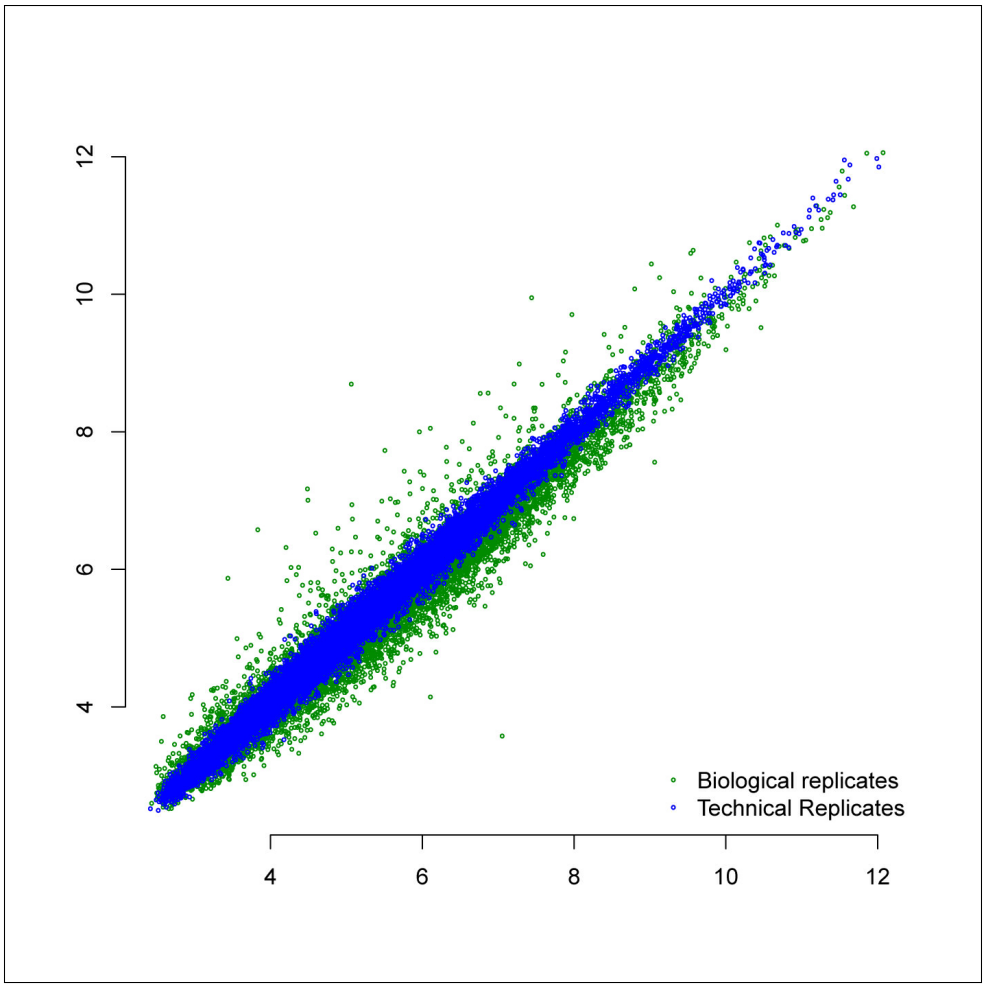

Figure 3.1: Effect of measurement error is illustrated through a scatter plot. Each point represents a gene, X and Y axes are gene expression levels measured by microarray. Blue dots are technical replicates that are the same sample measured twice using two different microarrays. Green dots are biological replicates. A large proportion of the total variance is due to measurement error.

To explore the network structure using noisy data, we consider a decomposition of the observed y into two latent vectors

$$Y = X + \epsilon,$$

where X and ϵ are mutually independent. We further assume $X \sim \mathcal{N}(\mu_X, \Sigma_X)$ and $\epsilon \sim \mathcal{N}(\mu_\epsilon, \Sigma_\epsilon)$. Letting Ω_X and Ω_ϵ denote the precision matrices of X and ϵ respectively, we aim to estimate Ω_X from the observed outcome $Y = y$. With the existence of measurement error, the graph for the outcome variable y is then highly connected (i.e dense), and the

absolute value of off-diagonal entries of Ω_Y are smaller than those of Ω_X. This phenomenon is illustrated in Figure 3.2

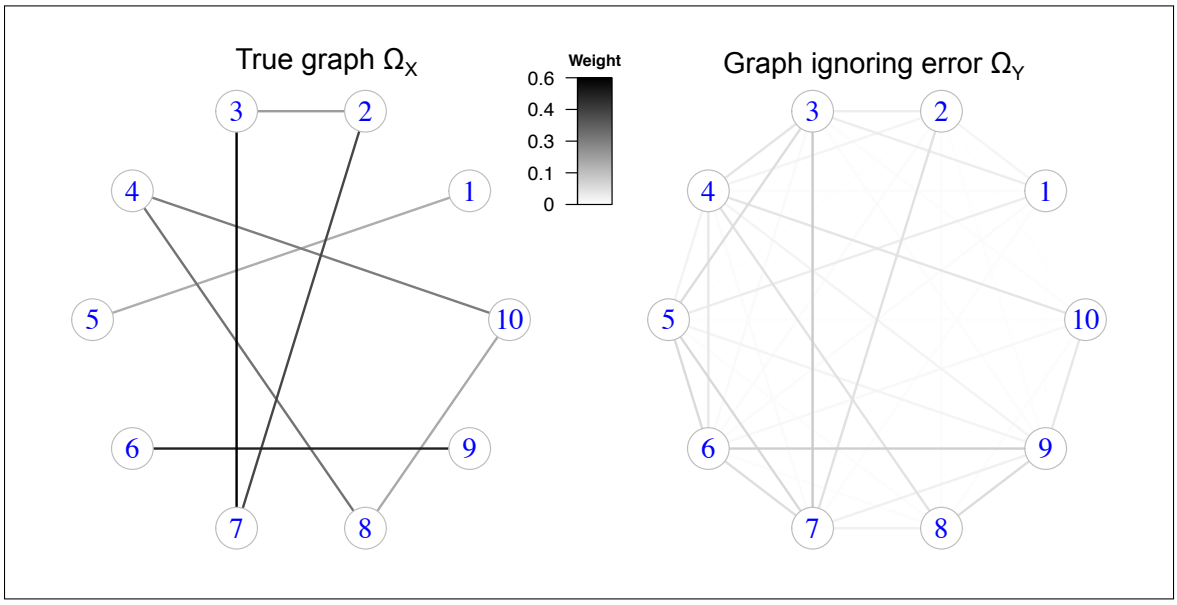

Figure 3.2: Effect of measurement error on Ω_Y with $p = 10$ variables. The left figures is the true Ω_X, and the right figure is $\Omega_Y = (\Omega_X^{-1} + \Omega_\epsilon^{-1})^{-1}$.

Under our setting, We hence have $\Sigma_Y = \Sigma_X + \Sigma_\epsilon$, but Ω_X is not identifiable under this model. To see it, let δ be some small positive number such that $(\Sigma_X - \delta I)$ is positive definite. Then we have

$$\begin{aligned}\Sigma_Y = \Sigma_X + \Sigma_\epsilon &= (\Sigma_X - \delta I) + (\delta I + \Sigma_\epsilon) \\ &= \Sigma'_X + \Sigma'_\epsilon = (\Omega'_X)^{-1} + \Sigma'_\epsilon.\end{aligned}$$

Therefore, Ω_X is not identifiable when only y is available.

To address this issue, we propose a new experimental design using technical replicates (e.g., identical sample but distinct measurements). Specifically, for each sample, we repeat the measurement R times, and denote the results as p-vectors y_r $(r = 1 \dots R)$. For each observed y_r, we decompose it as

$$Y_r = X + \epsilon_r, \tag{3.1}$$

where $\epsilon_r \sim \mathcal{N}(\mu_\epsilon, \Sigma_\epsilon)$. The following proposition guarantees the identifiability of the new experimental design.

Proposition 3.1. *Let $y_{r,i}$ be the ith observed data vector for the rth measurement under the model (3.1) for $R \geq 2$. Then, Ω_x is identifiable.*

The proof is included in Section 3.6.

To estimate dependency networks taking into account measurement error, we propose two new methods: a one-step method and an expectation-maximization (EM) method.

The remainder of the article is organized as follows. In Section 3.2, we introduce our Gaussian graphical model with measurement error, its implementation, and the one-step and EM methods. In Section 3.3, we study the asymptotic properties of the proposed methods. In Section 4, we illustrate the performance of our methods through simulations.

3.2 Methodology

For convenience of the reader, we summarize here notation to be used throughout the chapter. We use $\Omega^* = (\Sigma^*)^{-1}$ and Σ^* to denote the true precision and covariance matrices respectively. Given a matrix $W = (\omega_{ij})$, we use $\det(W)$ to denote the determinant, $\mathrm{tr}(W)$ to denote the trace and W^- to denote the off-diagonal entries of W. We further use $\phi_j(W)$ to denote the jth eigenvalue of W, and $\phi_{\min}(W)$ and $\phi_{\max}(W)$ to denote the minimum and maximum eigenvalues of W. The Frobenius norm of W is $\|W\|_F^2 = \sum_{i,j} \omega_{ij}^2$; the operator/spectral norm $\|W\|^2$ is $\phi_{\max}(WW^{\mathrm{T}})$; and $|W|_1$ is $\sum_{i,j} |\omega_{ij}|$. Finally, we denote $\|W\|_\infty$ as the element-wise maximum $\max_{i,j} |\omega_{i,j}|$.

3.2.1 Problem formulation

In this subsection, there are R measurements on the same p outcome variables on n subjects. One example of this kind of data is the measurement of gene expression on p genes in n human brain samples for $R = 3$ times per sample. Letting $Y_{i,r} = (Y_{i,r1}, \ldots, Y_{i,rp})$ be a p-dimensional random vector denoting the r-th measurement for the i-th sample. We model

$$Y_{i,r} = X_i + \epsilon_{i,r} \quad (i = 1, \ldots, n; \quad r = 1, \ldots, R), \tag{3.2}$$

where $\epsilon_{i,r}$ is the random vector corresponding to the rth random measurement error for the ith individual, and X_i is the random effect of our interest. We assume that X_i and $\epsilon_{r,i}$ are

i.i.d. $p-$dimensional random vectors follow multivariate Gaussian distribution with mean 0, and covariance matrices Σ_X and Σ_ϵ respectively, for $i = 1, \ldots, n$, and $r = 1, \ldots, R$.

Let Y_i be the joint data vector as $Y_i = (Y_{i,1}^{\mathrm{T}}, \ldots, y_{i,R}^{\mathrm{T}})^{\mathrm{T}}$, and thus $Y_i \sim \mathcal{N}(0, \Sigma_Y)$, where $\Sigma_Y = \{_d\Sigma_x + \Sigma_\epsilon\} + J \otimes \Sigma_x = \{\Sigma_{Y(l,m)}\}_{1 \le l,m \le K}$, where $\{_d\cdot\}$ is a block diagonal matrix, J is a square matrix with all $1'$s as the entries and $\otimes$ is the Kronecker product. Our goal is to estimate Ω_X given n i.i.d observations $\{y_i\}_{i=1}^n$. The corresponding likelihood can be written as

$$\mathcal{L}(\Omega_X, \Omega_\epsilon \mid y) = -\frac{nRp}{2}\log(2\pi) + \frac{n}{2}\left\{\log\det(\Omega_Y) - \mathrm{tr}(\hat{\Sigma}_Y\Omega_Y)\right\}, \tag{3.3}$$

where

$$\hat{\Sigma}_Y = n^{-1}\sum_{i=1}^{n} y_i y_i^{\mathrm{T}} = \{\hat{\Sigma}_{Y(l,m)}\}_{1 \le l,m \le K}. \tag{3.4}$$

We can also express the log-likelihood as

$$\begin{aligned}\mathcal{L}(\Omega_X, \Omega_\epsilon \mid y) \propto R\log\det(\Omega_\epsilon) - \sum_{r=1}^{R}\left\{\mathrm{tr}(\hat{\Sigma}_{Y(r,r)}\Omega_\epsilon)\right\} + \log\ \det(\Omega_X) \\ - \log\det(R\Omega_\epsilon + \Omega_X) + \sum_{l,m=1}^{K}\mathrm{tr}\left(\Omega_\epsilon\hat{\Sigma}_{Y(l,m)}\Omega_\epsilon(R\Omega_\epsilon + \Omega_X)^{-1}\right).\end{aligned} \tag{3.5}$$

The detailed derivation can be found in the section 3.6.1. However, the likelihood is complicated and non-concave in its full form. Direct estimation of the precision matrix Ω_X using maximum likelihood is difficult. However, in order to estimate Ω_X following the frame work of Glasso, we only need a good estimate for Σ_X (Friedman et al., 2008). Recalling the model 3.2 and the fact that $\mathrm{cov}(Y_l, Y_m) = \Sigma_X$, we can first estimate Σ_X and then Ω_X subsequently. In Sections 3.2.2 and 3.2.3, we consider estimation of the Gaussian graph with measurement error using a one-step procedure and a method based on the EM algorithm.

3.2.2 One-step method

In this subsection, we first estimate Σ_x and then obtain estimates for Ω_x through a subsequent one-step optimization. By the fact that $\mathrm{var}(x) = \mathrm{cov}(y_m, y_l)$, for any $m \neq l$,we

can estimate Σ_x through

$$\hat{\Sigma}_X = \frac{1}{R(R-1)} \sum_{m \neq l} \hat{\Sigma}_{Y(m,l)} = \frac{1}{K(K-1)n} \sum_{m \neq l} \sum_{i=1}^{n} y_{m,i} (y_{l,i})^{\mathrm{T}}. \tag{3.6}$$

Using the fact that $\Sigma_\epsilon = \mathrm{var}(Y_r) - \mathrm{var}(X)$, we can then obtain an estimate for Σ_ϵ as

$$\hat{\Sigma}_\epsilon = \frac{1}{R} \sum_{r=1}^{R} \left(\hat{\Sigma}_{Y(r,r)} - \hat{\Sigma}_0 \right) = \frac{1}{nR} \sum_{r=1}^{R} \sum_{i=1}^{n} \left(y_{k,i} y_{k,i}^{\mathrm{T}} \right) - \hat{\Sigma}_0. \tag{3.7}$$

While this is not needed for one step estimation, it will be useful in the next section.

Note that $\hat{\Sigma}_X$ preserves symmetry, but does not necessarily guarantee positive definiteness. We use a simple projection approach to make the estimate positive definite. Specifically, let $\hat{\Sigma}_X = \sum_{j=1}^{p} \phi_j(\hat{\Sigma}_X) v_j v_j^{\mathrm{T}}$, where v_j is the j-th eigenvector.

$$\eta_j = \begin{cases} \phi_j(\hat{\Sigma}_k) & \text{if } \phi_j(\hat{\Sigma}_k) > \tau_1 > 0 \ ; \\ \tau_1 & \text{otherwise,} \end{cases}$$

where τ_1 is a pre-specified lower bound on the minimum eigenvalue. Then we have $\hat{\Sigma}'_k = \sum_{j=1}^{p} \eta_j v_j v_j^{\mathrm{T}}$, which is positive definite. We then estimate Ω_x by minimizing the following functions:

$$\mathcal{W}(\Omega_x) = \mathrm{tr}(\hat{\Sigma}'_x \Omega_k) - \log \det(\Omega_k) + \lambda \sum_{j \neq j'} \left| \omega_{x(j,j')} \right|, \tag{3.8}$$

where λ is the tuning parameter. The optimization problem of (3.8) is well studied, and several efficient algorithms are available for example see Friedman et al. (2008) and Hsieh et al. (2011). We denote to this procedure as the "one-step" method and introduce the EM method in next section.

Using (3.7), we can also estimate the Σ_ϵ even though it is not our main interest by minimizing:

$$\mathrm{tr}(\hat{\Sigma}'_\epsilon \Sigma_\epsilon^{-1}) + \log \det(\Sigma_\epsilon) + \lambda_2 \sum_{j \neq j'} \left| \sigma_{x(j,j')} \right|, \hat{\Sigma}_\epsilon \tag{3.9}$$

which can be solved efficiently using the method, spcov, proposed by Bien and Tibshirani (2011).

3.2.3 Graphical EM method

The result of the one-step method is a local solution depending on the initial value of $\hat{\Sigma}_X$. When we treat X as a missing value, our model turns to a classic estimation of parameter under missing data. As introduced by Dempster et al. (1977) in 1970s, the EM algorithm got a lot of attention for dealing with missing value. We introduce this EM method as follows.

The E step calculates:

$$\begin{aligned}\mathcal{Q}(\Omega_X, \Omega_\epsilon \mid y, \Omega_X^{(t)}, \Omega_\epsilon^{(t)}) \propto & -\sum_{r=1}^{R}\left(tr\left[\Omega_k E\left\{\sum_{i=1}^{n}(y_{r,i}-x_i)(y_{r,i}-x_i)^{\mathrm{T}}/n \mid y, \Omega_x^{(t)}, \Omega_\epsilon^{(t)}\right\}\right]\right) \\ & + \log\det(\Omega_x) + R\log\det(\Omega_\epsilon) \\ & - \operatorname{tr}\left\{\Omega_0 E\left(\sum_{i=1}^{n} x_i x_i^{\mathrm{T}}/n \mid y, \Omega_X^{(t)}, \Omega_\epsilon^{(t)}\right)\right\} \\ = & R\left\{\log\det(\Omega_\epsilon) - \operatorname{tr}\left(\Omega_\epsilon \dot{\Sigma}_\epsilon^{(t)}\right)\right\} + \log\det(\Omega_X) - \operatorname{tr}\left(\Omega_X \dot{\Sigma}_X^{(t)}\right).\end{aligned}$$

The M step solves:

$$\begin{aligned}(\Omega_X^{(t+1)}, \Omega_\epsilon^{(t+1)}) = \underset{\Omega_X, \Omega_\epsilon}{\operatorname{argmin}} & -\mathcal{Q}(\Omega_X, \Omega_\epsilon \mid y, \Omega_X^{(t)}, \Omega_\epsilon^{(t)}) \\ & + R\lambda_1 \left|\Sigma_\epsilon^{-}\right| + \lambda_2 \sum_{j \neq j'} \left|\Omega_X^{-}\right|, \end{aligned} \tag{3.10}$$

where $(\Omega_x^{(t)}, \Omega_\epsilon^{(t)}$ denote the estimates from the t-th iteration, and $(\dot{\Sigma}_\epsilon^{(t)}, \dot{\Sigma}_X^{(t)})$ are defined as follows:

$$\begin{aligned}\dot{\Sigma}_\epsilon^{(t)} = & E\left(\frac{1}{nR}\sum_{r=1}^{R}\sum_{i=1}^{n}\left\{(y_{i,r}-x_i)(y_{i,r}-x_i)^{\mathrm{T}}\right\} \mid y, \Omega_X^{(t)}, \Omega_\epsilon^{(t)}\right) \\ = & \frac{1}{nR}\sum_{r=1}^{R}\sum_{i=1}^{n}\left\{y_{r,i}y_{r,i}^{\mathrm{T}} - y_{r,i}E(z_i^{\mathrm{T}} \mid y, \Omega_X^{(t)}, \Omega_\epsilon^{(t)}) - E(z_i \mid y, \Omega_X^{(t)}, \Omega_\epsilon^{(t)})y_{k,i}^{\mathrm{T}}\right\} \\ & + \frac{1}{n}\sum_{i=1}^{n}\left\{E(z_i z_i^{\mathrm{T}} \mid y, \Omega_X^{(t)}, \Omega_\epsilon^{(t)})\right\} \\ = & \frac{1}{R}\sum_{r=1}^{R}\ddot{\Sigma}_{Y(r,r)} - \frac{1}{R}\sum_{r=1}^{R}\sum_{l=1}^{R}\left(\ddot{\Sigma}_{Y(r,l)}\Omega_\epsilon^{(t)}\right)(\Omega_X^{(t)} + R\Omega_\epsilon^{(t)})^{-1}\end{aligned}$$

$$
\begin{aligned}
&- \frac{1}{R}(\Omega_X^{(t)} + R\Omega_\epsilon^{(t)})^{-1} \sum_{r=1}^{R}\sum_{l=1}^{R} \left(\Omega_\epsilon^{(t)} \ddot{\Sigma}_{Y(l,r)}\right) \\
&+ \frac{1}{R}(\Omega_X^{(t)} + R\Omega_\epsilon^{(t)})^{-1}\Omega_\epsilon^{(t)} \sum_{l,r=1}^{R} \left(\ddot{\Sigma}_{Y(l,r)}\right) \Omega_\epsilon^{(t)}(\Omega_X^{(t)} + R\Omega_\epsilon^{(t)})^{-1} \\
&+ (\Omega_X^{(t)} + \Omega_\epsilon^{(t)})^{-1}, \qquad (3.11a)
\end{aligned}
$$

$$
\begin{aligned}
\dot{\Sigma}_X^{(t)} =& \frac{1}{n}\sum_{i=1}^{n} E\left(x_i x_i^{\mathrm{T}} \mid y, \Omega_X^{(t)}, \Omega_\epsilon^{(t)}\right) \\
=& (\Omega_X^{(t)} + R\Omega_\epsilon^{(t)})^{-1}\Omega_\epsilon^{(t)} \sum_{l,r=1}^{K} \left(\ddot{\Sigma}_{Y(l,r)}\right) \Omega_\epsilon^{(t)}(\Omega_X^{(t)} + R\Omega_\epsilon^{(t)})^{-1} \\
&+ (\Omega_X^{(t)} + R\Omega_\epsilon^{(t)})^{-1} \qquad (3.11b)
\end{aligned}
$$

where $\ddot{\Sigma}_Y$ is an estimator for Σ_Y^*, here $\ddot{\Sigma}_Y = \hat{\Sigma}_Y$. Thus, at the $t+1$ iteration, problem (3.10) is decomposed into two separate optimization problems:

$$
\Omega_X^{(t+1)} = \underset{\Omega_X}{\operatorname{argmin}} \left\{ \operatorname{tr}\left(\Omega_X \dot{\Sigma}_X\right) - \log\det(\Omega_X) + \lambda_2 \left|\Omega_X^-\right|_1 \right\}, \qquad (3.12a)
$$

$$
(\Omega_\epsilon^{(t+1)})^{-1} = \Sigma_\epsilon^{(t+1)} = \underset{\Sigma_\epsilon}{\operatorname{argmin}} \left\{ \operatorname{tr}\left(\Sigma_\epsilon^{-1} \dot{\Sigma}_x\right) + \log\det(\Sigma_\epsilon) + \lambda_1 \left|\Sigma_\epsilon^{-1}\right|_1 \right\}. \qquad (3.12b)
$$

The key difference between this Chapter and Chapter 2 is that we put sparsity on Σ_ϵ instead of Ω_ϵ. We then can use GLASSO (Friedman et al., 2008) to solve (3.12a), and use spcov proposed by Bien and Tibshirani (2011) to solve (3.12b).

Th proposed EM method is summarized as follows:

Step 1 (Initial value). Initialize $\hat{\Sigma}_X$ and $\hat{\Sigma}_\epsilon$ (3.4), (3.11a) and (3.11b).

Step 2 (Updating rule: the M step). Update Ω_X using (3.12a) using GLASSO, and Ω_ϵ using (3.12b) using spcov.

Step 3 (Updating rule: the E step). Update $\dot{\Sigma}_X$ and $\dot{\Sigma}_\epsilon$ using (3.11a) and (3.11b).

Step 4 (Iteration). Iterate Steps 2 and 3 until convergence is achieved.

The next proposition demonstrates the convergence property of our penalized EM algorithm. We define $\mathcal{P}(\Omega_X, \Omega_\epsilon)$ as follows

$$\mathcal{P}(\Omega_X \Omega_\epsilon) = \mathcal{L}(\Omega_X, \Omega_\epsilon \mid y) - R\lambda_1 \left|\Sigma_\epsilon^{-1}\right| - \lambda_2 \left|\Omega_X^{-1}\right| \tag{3.13}$$

Proposition 3.2. *With $\lambda_1 > 0$ and $\lambda_2 > 0$, the graphical EM algorithm solving* (2.5) *has the following properties:*

1. *The penalized log-likelihood in* (3.13) *is bounded above;*
2. *For each iteration, the penalized log-likelihood is non-decreasing;*
3. *For a prespecified threshold δ, after finite steps, the objective function in* (3.13) *converges in the sense that*

$$|\mathcal{P}(\Omega_X^{(t+1)}, \Omega_\epsilon^{(t+1)}) - \mathcal{P}(\Omega_X^{(t)}, \Omega_\epsilon^{(t)})| < \delta.$$

This proposition is similar to proposition 2.1 in Chapter 2; Thus the proof is skipped.

3.3 Asymptotic properties

In this section, we study the asymptotic properties of our proposed methods including estimation consistency and sparsistency. Using similar notation as in Chapter 2, we denote Ω_X^* and Ω_ϵ^* to be the true precision matrices, and $T_X = \left\{(j, j') : j \neq j', \omega^*_{X(j,j')} \neq 0\right\}$ $T_\epsilon = \left\{(i, j) : i \neq j, \sigma^*_{\epsilon(i,j)} \neq 0\right\}$ be the set of indices of all nonzero off-diagonal elements in Ω_X^* and Ω_ϵ^* respectively. We define $q_X = |T_X|$ and $q_\epsilon = |T_\epsilon|$ be the cardinality of T_X and T_ϵ , and $q = q_X + q_\epsilon$. Let Σ_X^* and Σ_ϵ^* be the true covariance matrices for X and ϵ, and $\Sigma_Y^* = \{\Sigma^*_{Y(l,m)}\}_{1 \leq l,m \leq K}$ be the true covariance matrices for Y. We assume that the following regularity conditions hold.

Condition 1. There exist constants τ_1, τ_2 such that for all $r \in \{X, \epsilon\}$, $0 < \tau_1 < \phi_{\min}(\Omega_r^*) \leq \phi_{\max}(\Omega_r^*) < \tau_2 < \infty$.

Condition 2. There exists a constant $\tau_3 > 0$, such that $\min_{(i,j) \in T_X} \left|\omega^*_{X(i,j)}\right| \geq \tau_3$ and $\min_{(i,j) \in T_\epsilon} \left|\sigma^*_{\epsilon(i,j)}\right| \geq \tau_3$.

Condition 1 bounds the eigenvalues of Ω_X^* and Ω_ϵ^*, thereby guaranteeing the existence of its inverse and facilitating the proof of consistency. Condition 2 is needed to bound the nonzero elements away from zero.

The following theorems discuss estimation consistency and selection sparsistency of our methods.

Theorem 3.1 (Consistency of the one-step method)**.** *Under Conditions 1-2,* $(p + q_X)\log p/n = o(1)$*, and* $a_3(\log p/n)^{1/2} \leq \lambda_2 \leq b_3\{(1 + p/q_X)\log p/n\}^{1/2}$ *for some constants* a_3 *and* b_3*. Let* $\hat{\Omega}_X$ *be the minimizer defined by* (3.8) *using the one-step method, then*

$$\left\|\hat{\Omega}_X - \Omega_X^*\right\|_F = O_p\left[\left\{\frac{(p+q)\log p}{n}\right\}^{1/2}\right].$$

The proof of Theorem 3.1 is similar to the one for Theorem 2.1, and hence is omitted.

Before we introduce the main theorem of the EM algorithm, we first present a corollary of Theorem 3.1 which gives a good estimator of Σ_Y^*.

Corollary 3.1. *Suppose that Conditions 1-2 hold, let* $\hat{\Sigma}_\epsilon^{one}$ *be the minimizer defined by 3.9, then*

$$\left\|\hat{\Sigma}_\epsilon^{one} - \Sigma_\epsilon^*\right\|_F = O_p\left[\left\{\frac{(p+q_\epsilon)\log p}{n}\right\}^{1/2}\right].$$

To study our EM estimator, we need a good estimator for Σ_Y^* which specifies in the following condition.

Condition 3. We assume there exists an estimator $\tilde{\Sigma}_Y$ such that

$$\|\tilde{\Sigma}_Y - \Sigma_Y^*\|_F = O_p\left[\left\{\frac{(p+q)\log p}{n}\right\}^{1/2}\right].$$

The rate in Condition 3 is required to control the convergence rate of the E-step estimate $\dot{\Sigma}_X$ and $\dot{\Sigma}_\epsilon$, and thus the consistency of the estimate from the EM method. Under the conditions in Theorem 3.1 and Corollary 3.1, we can use the one-step estimator $\hat{\Omega}_X$ and $\hat{\Sigma}_\epsilon^{one}$ to obtain $\tilde{\Sigma}_Y = J \otimes (\hat{\Omega}_X)^{-1} + \{{}_d\hat{\Sigma}_\epsilon^{one}\}$, which satisfies Condition 3 by Corollary 3.1.

Theorem 3.2 (Consistency of the EM method)**.** *Suppose Conditions 1-3 hold, and* $(p + q)\log p/n = o(1)$*, and* $a_4(\log p/n)^{1/2} \leq \lambda_1, \lambda_2 \leq b_4\{(1 + p/q)\log p/n\}^{1/2}$ *for some constants*

a_2 and b_2. Then the solution, $\hat{\Omega}_X$, of the EM method satisfies

$$\left\|\hat{\Omega}_X - \Omega_X^*\right\|_F = O_p\left[\left\{\frac{(p+q)\log p}{n}\right\}^{1/2}\right].$$

Theorem 3.3 (Sparsistency of the one-step method)**.** *Under the assumptions of Theorem 2.1. If we further assume that $\left\|\hat{\Omega}_X - \Omega_X^*\right\| = O_p(\eta_n)$ for a sequence of $\eta_n \to 0$, and $(\log p/n + \eta_n^2)^{1/2} = O(\lambda_1)$, then with probability tending to 1, the minimizer $\hat{\Omega}_X$ satisfies $\hat{\omega}_{X(i,j)} = 0$ for all $(i,j) \in T_X^c$.*

The sparsistency requires a lower bound on the rate of the regularization parameters λ_2, while the consistency need an upper bounds to control the biases. To achieve both consistency and sparsistency simultaneously, we need the bounds to be compatible, that is, we need $(\log p/n + \eta_n^2)^{1/2} = O(\lambda_2) = \{(1 + p/q_X)\log p/n\}^{1/2}$. Using the same argument as in Chapter 2, there are two scenarios describing the rate of η_n. In the worst case, the two bounds are compatible only when $q_X = O(1)$ and in the most optimistic case, the two bounds are compatible when $q_X = O(p)$.

3.4 Numerical Example

We assessed the performance of the graphical Lasso (glasso), one-step and EM methods by applying them to simulated data generated by two types of noise structures: banded network and block network. We simulate the GGM network for X using nearest neighbour (NN) as described in Chapter 2. The structure of those networks are shown in (Figure 3.3).

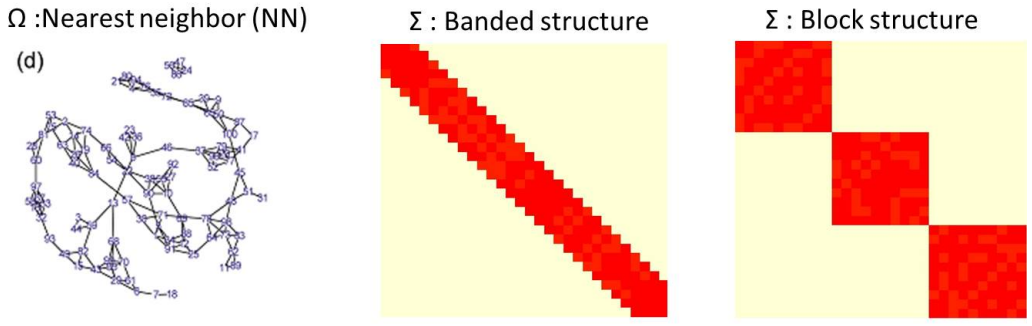

Figure 3.3: Network topologies generated in the simulations.

3.4.1 Simulating X and ϵ networks

Under each of the 12 simulation conditions, i.i.d. samples were generated, with true outcomes generated as $X_i \sim \mathcal{N}(0, \Omega_x^{-1})$, measurement error as $\epsilon_{ri} \sim \mathcal{N}(0, \Sigma_\epsilon)$, and observed outcomes as $y_{i,r} = x_i + \epsilon_{i,r}$, for $r = 1, \ldots, R = 3$, and $i = 1, \ldots, n$. The following base architectures were considered:

- NN/Banded network with $p = 30$ and $p = 100$ nodes: the true outcomes network Ω_X is NN-network and the measurement error network Σ_ϵ is Banded-networks.
- NN/Block network with $p = 30$ and $p = 100$ nodes: the true outcomes networks Ω_X is NN-networks and the measurement error network Σ_ϵ is a block-network.

Banded network for Σ is generated as follows,

$$\sigma_{\epsilon(i,j)} = \begin{cases} 1 & i = j \\ U(0.3, 1) & \text{if } 1 \leq i - j + 1 \leq 3 \text{ and } i \neq j; \\ 0 & \text{otherwise.} \end{cases}$$

We generate block networks for Σ as follows,

$$\sigma_{\epsilon(i,j)} = \begin{cases} 1 & i = j \\ U(0.3, 1) & \text{if } 1 + 4(K-1) \leq i, j \leq 4K \text{ and } i \neq j; \\ 0 & \text{otherwise,} \end{cases}$$

where K is the block size. We set $K = 5$, and 10 for $p = 30$ and 100 respectively. NN networks were generated using the method of Li and Guo (2006), sampling p points uniformly on $[0, 1]^2$ and then calculating all pairwise distances to find the m nearest neighbors of each point. Pairs of nodes were linked if they are mutual m-nearest neighbors, with $m = 5$ in our model. Under this construction, elements in the precision matrix for each edge are first generated from uniform $[0.5, 1]$ or $[-1, -0.5]$. The diagonal entry of each row is taken as the sum of the absolute values of that row's elements. We then calculate the inverse of this matrix, and the numbers in each row of the inverse matrix are divided by their corresponding diagonal entry so that the final covariance matrix has diagonal elements of 1 and is positive definite.

We consider three different noise levels, $\rho \in \{0.3, 0.5, 0.7\}$, for our simulation. The noise level ρ is defined as $\rho = \text{tr}(\Sigma_\epsilon)/\{\text{tr}(\Sigma_x) + \text{tr}(\Sigma_\epsilon)\}$. For each simulation trial we generate two independent realizations of the data tensor y, each corresponding to sample size $n = 300$. The first realization is used for tuning and training, and the second realization is for testing.

We first compare our EM method with one of the most widely used method for GGMs, the Graphical Lasso (Glasso) ignoring the existent of measurement error (Friedman et al., 2008). ROC curves are plotted in Figure 3.4; each curve is based on 100 replications across various tuning parameters. For the comparison purpose, we set $\lambda_1 = \lambda_2$ for our EM method. In the plots, the ROC curves of the EM method are seen to dominate those of the Glasso method especially for high noise level setting. In general, the EM method delivers more accurate results than the Glasso method.

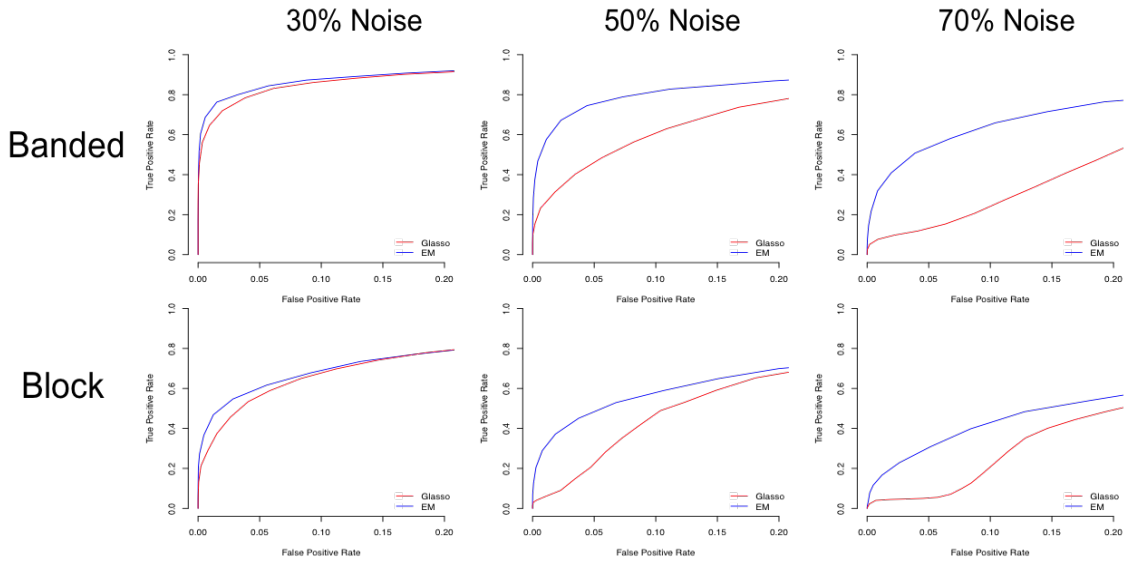

Figure 3.4: Receiver operating characteristic (ROC) curves assessing power and discrimination of graphical inference for $p = 100$ and $n = 500$. Each panel reports performance of the EM method (blue line) and the Glasso method (red line), plotting true positive rate (y-axis)against false positive rate (x-axis) for a given noise ratio.

We then compare the EM method with the one-step method using the same criteria described in Chapter 2. Results of the simulations are reported in Table 3.2. Summary statistics are based on 50 replicate trials under each conditions, and given for model fitting under both extended BIC and under cross-validation criteria. In general, the one-step and EM methods under either model selection criteria resulted in lower values of entropy

loss, Frobenius loss, false negative rates and hamming distance than Glasso ignoring the measurement error. These results are due to the fact that Glasso estimates Ω_Y instead of Ω_X. In most of the setting, the EM method delivers more accurate result than the one-step method.

ROC curves are plotted in Figure 3.5; each curve is based on 100 replications across various tuning parameters with the constraint $\lambda_1 = \lambda_2$ for both EM and one-step methods. In the plots, the EM method has uniformly better sensitivity and specificity than the one-step method in estimating Ω_x.

Table 3.1: Summary statistics reporting performance of the EM and one-step methods inferring graph structure for different networks. In each cell, the number before and after the slash correspond to the results using extended BIC and cross-validation, respectively.

	True networks Category-specific / Systemic	ρ	Method	EL	FL	FP(%)	FN(%)	HD (%)
$p = 30$	NN / Banded	0.3	Glasso	1.2/1.1	0.15/0.15	30.4 /28.6	0.2 /0.0	30.4/28.6
		0.3	One step	0.8/0.8	0.06 /0.06	23.6 /24.7	0.0 /0.0	23.6/24.7
		0.3	EM	**0.6/0.5**	**0.04/0.03**	**15.5/19.8**	**0.0/0.0**	**15.5/19.8**
		0.5	Glasso	2.5 /2.4	0.43/0.42	35.4 /33.6	5.4/3.5	40.8/37.1
		0.5	One step	0.8/0.7	0.05/0.04	25.4 /26.7	3.7 /2.8	29.1/29.5
		0.5	EM	**0.7/0.6**	**0.04/0.03**	**15.3/20.7**	**1.6/1.4**	**16.9/22.1**
		0.7	Glasso	3.8/3.7	0.72 /0.70	34.0 /32.0	27.5 /26.1	61.5/58.1
		0.7	One step	1.2 /1.1	0.08/0.06	15.3/25.1	32.2 /23.1	47.5/48.2
		0.7	EM	**0.8/0.6**	**0.06/0.05**	**13.9/23.2**	**27.4/15.2**	**41.3/38.4**
	NN / Block	0.3	Glasso	1.7/1.5	0.17/0.16	32.2 / 27.6	25.0/22.8	57.2/50.4
		0.3	One step	1.1/1.0	0.08/0.07	23.2/20.6	27.2 /28.8	50.4/49.4
		0.3	EM	**0.8/0.6**	**0.06/0.05**	**17.2/22.3**	**17.3/8.7**	**34.5/31.0**
		0.5	Glasso	2.7 /2.8	0.44 /0.44	43.8 /37.2	30.8 /32.5	74.6/69.7
		0.5	One step	1.2/0.9	0.07 /0.06	21.6 /**20.4**	37.3 / 38.4	58.9/58.8
		0.5	EM	**0.8/0.7**	**0.06/0.05**	**16.4**/26.0	**34.0/18.0**	**50.4/52.0**
		0.7	Glasso	5.8 / 4.8	0.76 /0.75	45.1/40.2	50.6 /53.4	95.7 / 93.6
		0.7	One step	1.0 / 0.9	0.06 /0.05	9.7 /20.0	76.6 /63.5	86.3/ 83.5
		0.7	EM	**0.9/0.8**	**0.05/0.04**	**9.6 /28.4**	**64.2/43.3**	**72.8/ 71.7**

Table 3.2: Summary statistics reporting performance of the EM and one-step methods inferring graph structure for different networks. In each cell, the number before and after the slash correspond to the results using extended BIC and cross-validation, respectively.

	True networks Category-specific / Systemic	ρ	Method	EL	FL	FP(%)	FN(%)	HD (%)
$p = 100$	NN / Banded	0.3	Glasso	14.4/14.1	0.20/0.19	27.4 /24.9	10.4/14.9	37.8/39.8
		0.3	One step	4.1/4.0	0.11 /0.09	25.4/24.9	6.3/6.9	31.7/31.8
		0.3	EM	**3.0/3.0**	**0.06/0.05**	**21.6/25.7**	**5.7 /4.8**	**27.3/ 30.5**
		0.5	Glasso	23.9 /22.6	0.43/0.41	34.7 /35.0	19.4/12.8	54.1/47.8
		0.5	One step	4.8/4.3	0.10 /0.09	10.7 /16.0	20.7/14.8	31.4/30.8
		0.5	EM	**3.5/3.2**	**0.07/0.06**	**5.2/10.7**	**17.6/9.9**	**22.8/ 20.6**
		0.7	Glasso	29.0 /28.7	0.70 /0.67	41.3/42.6	42.6/ 41.1	83.9/83.7
		0.7	One step	5.1 /4.8	0.10 /0.08	7.7/13.3	65.6/45.1	73.3/68.4
		0.7	EM	**3.8/3.5**	**0.07/0.06**	**5.1 /12.0**	**59.1/38.0**	**64.2/50.0**
	NN /Block	0.3	Glasso	17.0 /16.8	0.22 /0.18	23.1/27.4	35.7 /31.8	58.8/59.2
		0.3	One step	4.9 /4.3	0.11 /0.09	16.1/ 17.5	33.7 /28.8	49.8/46.3
		0.3	EM	**3.8/3.2**	**0.06/0.06**	**4.2 /7.8**	**35.3/27.0**	**39.5 / 34.8**
		0.5	Glasso	25.3 /24.3	0.41/0.38	30.5/ 29.1	48.1 /50.0	78.6/79.1
		0.5	One step	5.2 /4.8	0.11 /0.09	3.4/9.1	59.3 /51.0	62.7/60.1
		0.5	EM	**4.3/4.2**	**0.10/0.08**	**3.5 / 7.7**	**55.3/52.2**	**58.8 / 59.5**
		0.7	Glasso	29.3 /28.7	0.74 /0.73	44.1 /50.2	80.1/76.6	124.2/126.8
		0.7	One step	4.2 /3.9	0.09 /0.07	8.1 / 15.2	80.6 /71.6	88.7/86.8
		0.7	EM	**3.8/3.3**	**0.07/0.05**	**1.8 / 8.2**	**81.1/67.3**	**82.9 / 75.5**

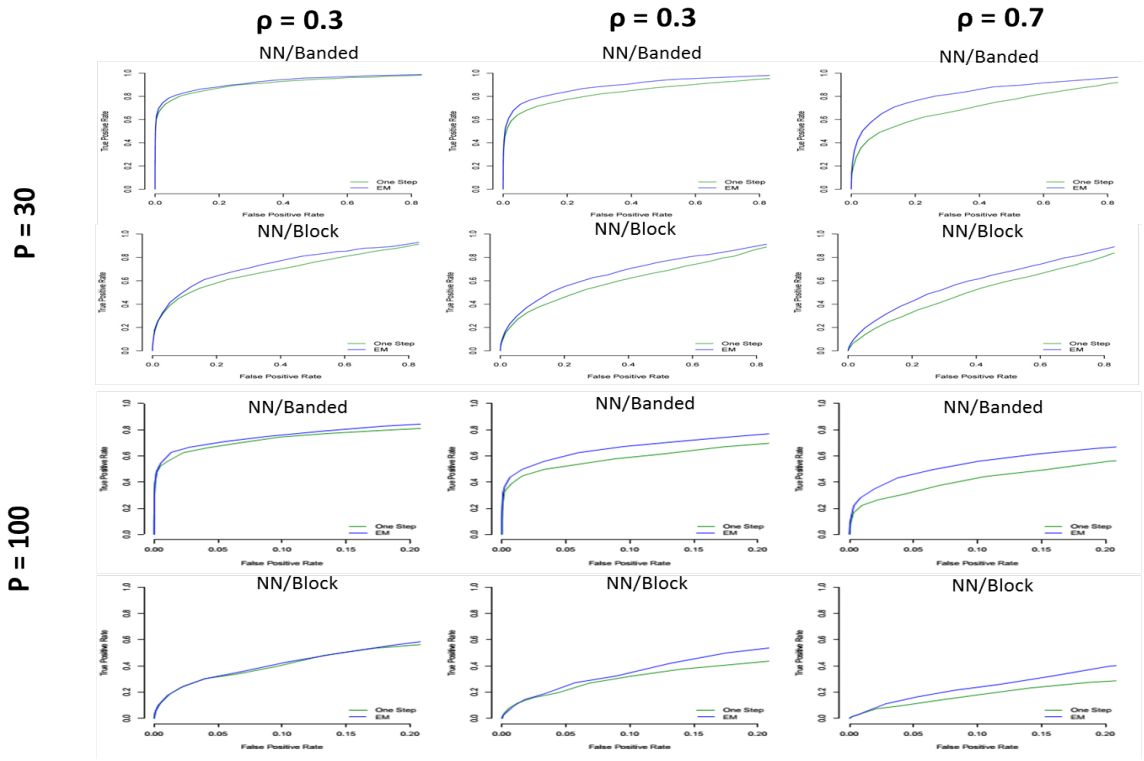

Figure 3.5: Receiver operating characteristic (ROC) curves assessing power and discrimination of graphical inference for $p \in \{30, 100\}$ and $n = 500$. Each panel reports performance of the EM method (blue line) and the one-step method (green line), plotting true positive rate (y-axis)against false positive rate (x-axis) for a given noise ratio.

3.5 Discussion

We propose a new experimental design using technical replicates to estimate GGMs from noisy data which is common in biological science. We then propose novel one-step and EM methods to estimate graph using data with technical replicates. We show that our method is asymptotically consistent for high dimensional GGMs, and also evaluate our method using simulation. There are several interesting directions for further investigation in the future.

First, for the measurement error covariance matrix Σ_ϵ, we currently simply assume it is sparse. However, we know there are several sources for measurement error: RNA preparation procedure, RNA degradation ,PCR amplification steps, and so on. If we can well specify the covariance matrices for those sources, we can propose a more efficient methodology to estimate the underlying graph. Second, how to choose sample size n and number of technical replicate R under limited budget? The choose of n and R would depends on the signal noise ratio, but how to quantify this ratio in high dimension setting is a very interesting topic. Third, with the popularity of RNA-seq, it is practical and useful to extend our methods from Gaussian assumption to high dimensional discrete data. Furthermore, beside obtaining a point estimate of the graph, how to perform statistical inference in graphical models is an important issue. especially in biological science. Recently, Ren et al. (2015) proposed an novel estimator for GGMs, which enjoys asymptotic normality. Extending our frame work with similar estimator would be an important extension.

3.6 Appendix

3.6.1 Derivation of likelihood for y

Since from 2.7.1, we have

$$\mathcal{L}(\Omega \mid y) \propto \sum_{k=1}^{K} \Big[\log\{\det(\Omega_k)\} - \mathrm{tr}\big(\hat{\Sigma}_{Y(k,k)}\Omega_k\big)\Big] + \log\{\det(\Omega_0)\} \\ - \log\{\det(A)\} + \sum_{l,m=1}^{K} \mathrm{tr}\left(\Omega_l \hat{\Sigma}_{Y(l,m)} \Omega_m A^{-1}\right) .$$

Replacing Ω_0 and Ω_k with Ω_X and Ω_ϵ respectively, it follows that

$$\mathcal{L}(\Omega_x, \Omega_\epsilon \mid y) \propto R \log\det(\Omega_\epsilon) - \sum_{r=1}^{R} \left\{ \mathrm{tr}(\hat{\Sigma}_{Y(r,r)}\Omega_\epsilon) \right\} + \log\{\det(\Omega_X)\} \\ - \log\{\det(R\Omega_\epsilon + \Omega_x)\} + \sum_{l,m=1}^{K} \mathrm{tr}\left(\Omega_\epsilon \hat{\Sigma}_{Y(l,m)} \Omega_\epsilon (R\Omega_\epsilon + \Omega_x)^{-1} \right) .$$

3.6.2 Proof of Identifiability

To demonstrate identifiability of our experimental design, we follow the similar proof strategy as shown in Chapter 2. It is enough to prove under different decompositions, the parameters Ω_X and Ω_ϵ have the same value. Given random vector Y_r, we decompose it in two different ways:

$$Y_r = X - U + \epsilon_r + U = X^* + \epsilon_r^* \quad (r = 1, ..., R),$$

where U is a p-dim of random vector. Under our assumption, the two decompositions have the following properties

$$\mathrm{cov}(\epsilon_l^*, \epsilon_m^*) = 0 \quad (1 \le l, m \le R), \tag{3.14}$$

$$\mathrm{cov}(\epsilon_l^*, X^*) = 0 \quad (l = 1, \ldots, R). \tag{3.15}$$

Rearranging (3.14) and (3.15), we have

$$\mathrm{var}(U) = \mathrm{cov}(\epsilon_l, U) + \mathrm{cov}(\epsilon_m, U); \tag{3.16}$$

$$\mathrm{var}(U) = -\mathrm{cov}(U, X) + \mathrm{cov}(U, \epsilon_l). \tag{3.17}$$

We know that (3.17) hold for any l; thus we have

$$\mathrm{cov}(U, \epsilon_l) = \mathrm{cov}(U, \epsilon_m) \quad (1 \le l, m \le R). \tag{3.18}$$

Combining (3.16), (3.17) and (3.18), we can show

$$\mathrm{cov}(U, \epsilon_k) = -\mathrm{cov}(U, X), \tag{3.19}$$

$$\mathrm{var}(U) = -2\,\mathrm{cov}(U, X) = 2\,\mathrm{cov}(U, \epsilon_l) \quad (1 \le l \le R), \tag{3.20}$$

which indicate that

$$\begin{aligned}
\mathrm{var}(\epsilon_l^*) &= \mathrm{var}(\epsilon_l - U) \\
&= \mathrm{var}(\epsilon_l) + \mathrm{var}(U) - 2\,\mathrm{cov}(U, \epsilon_l) = \mathrm{var}(\epsilon_l) \\
\mathrm{var}(X^*) &= \mathrm{var}(X + U) \\
&= \mathrm{var}(X) + \mathrm{var}(U) - 2\,\mathrm{cov}(U, X) = \mathrm{var}(X).
\end{aligned}$$

This completes the proof.

3.6.3 Proof of Corollary 3.1

The proof is similar to the proof of Corollary 1. We only need to prove

$$\|\hat{\Sigma}_\epsilon^{one} - \Sigma_\epsilon^*\|_F = O_p\Big[\Big\{\frac{(p+q)\log p}{n}\Big\}^{1/2}\Big],$$

where is the solution of (3.9). This can be achieved by similar proving strategy as (Lam and Fan, 2009), and we omitted for simplicity.

CHAPTER 4

Estimation of the Skeletons in High Dimensional Directed Acyclic Graphs using Adaptive Group Lasso

4.1 Introduction

Bayesian network is a commonly used probabilistic graphical model that encodes the conditional dependence of a set of random variables. The structure of Bayesian network is represented by a directed acyclic graph (DAG). In a DAG, all the edges are directed without forming a directed loop. The problem of estimating Bayesian networks has received a significant amount of attention, with applications in biological and medical sciences for inferring gene regulatory networks (Friedman, 2004; Glymour, 1987; Koller and Friedman, 2009; Sachs et al., 2005). This popularity is partly attributable to the fact that DAGs can be used to model causal effects (Pearl, 2000).

It has been shown that the skeleton of a DAG, but not the DAG itself, is identifiable when only observational data is available (Chickering and Boutilier, 2002; Pearl, 2009). The skeleton of a DAG is the graph generated by removing all directions from the DAG. The importance of estimating the skeleton of a DAG is fourfold:

1. Skeleton estimation is the first step to construct the DAG.

2. With observational data, the skeleton is still identifiable but the corresponding DAG is not.

3. Skeleton can be used to assess the direction of some edges in the DAG.

4. The skeleton can be used to design follow-up intervention experiments to construct the direction of edges (Maathuis et al., 2009).

In the framework of graphical model, we consider the relationship among a set of p random variables $X = (X_1, \ldots, X_p)$ represented by $\mathcal{G}(V, E)$. Verma and Pearl (1990) proposed

one of the earliest algorithms, called the Inductive Causation (IC) algorithm to estimate the skeleton of DAG. Specifically, for each pair of variables X_i and X_j, they search for the set $S_{ij} \in \{V \setminus (i,j)\}$, such that $X_i \perp\!\!\!\perp X_j | S_{ij}$ (by conditional independence test), and connect X_i and X_j with an undirected edge if and only if no S_{ij} is found. For each edge, IC algorithm has to perform 2^{p-2} tests, and thus, is not feasible for high dimensional data.

To handle high dimensional data, Spirtes et al. (2000) proposed a sequential method called the PC-algorithm. Starting from a fully connected graph, the PC-algorithm recursively removes edges based on conditional independence tests to obtain the skeleton of the DAG. In the PC-algorithm, the search for the separating set S_{ij} is limited to nodes $X_k \in adj(\mathcal{G}, X_i)$, and hence the number of tests is reduced significantly. However, the result of the PC-algorithm is order-dependent in the sense that different initial nodes would lead to different outputs. Recently, (Colombo and Maathuis, 2013) proposed an order-independent PC algorithm, called the PC-stable algorithm, and showed that both the PC and the PC-stable algorithms consistently estimate the skeleton of a sparse DAG with $p = O(n^r)$ for some $r > 0$.

Instead of starting from a fully connected graph, Spirtes et al. (2000) proposed the the IG (Independence Graph) algorithm that starts by estimating the moral graph (independence graph), and then removes the extra edges using conditional independence tests. A moral graph from a DAG is the undirected graph created by connecting two parents of the same node (v-structure). Therefore, the skeleton of a DAG could also be obtained by removing the extra edges due to v-structure. Under the multivariate Gaussian assumption, the moral graph equals the GGM which is uniquely determined by Ω. Under this setting, Ha et al. (2014) proposed the PenPC algorithm, which is similar to the IG algorithm. In the PenPC, they first adapt the neighbourhood selection method to learn the non-zero structure in the precision matrix Ω (Meinshausen and Bühlmann, 2006). Then a modified PC-stable algorithm is applied to delete the extra edges due to v-structure. The advantage of the PenPC algorithm is that by screening out most of the extra edges in the first step, it allows fewer conditional independence tests in the second step. The PenPC algorithm thus provides fewer cumulative mistaken probabilities and faster computing speed. Ha et al. (2014) also

show that the PenPC algorithm is consistent even when the dimension $p = O(\exp(n^a))$ for some $a \geq 0$.

All existing methods tend to have high false negative rate when analyze high dimensional data. These high false negative rates can be undesirable: in certain circumstances, the cost of missing true edges is much higher than including false edges. For example, in biological research, there are many technologies to test the newly discovered edges, but given the large number of candidate edges, a missing edge is much harder recover. This phenomena is exacerbated when analyzing data generated from a DAG with hubs. A hub is defined as a node densely-connected to other nodes. Examples of hubs include the super-hub genes in biology, and google website in media network (Hao et al., 2012; Tan et al., 2014). In these networks, the difference between the GGM and the skeleton of the DAG is large due to the existence of a large number of v-structures induced by the hubs. Consequently, the number of tests in the second step of the PenPC algorithm would be very large. To address this challenging problem, we propose a new method named AdaPC, which uses an adaptive group lasso penalty to efficiently estimate the skeleton of a DAG.

The idea behind our AdaPC is as follows: we first estimate a graph $\mathcal{M}(V, J)$ representing the dependency (including marginal dependence and conditional dependence given all other variables) among a set of random variables. An undirected edge (X_i, X_j) belongs to J, if and only if X_i and X_j are both marginally dependent and conditional dependent given $X_{-\{i,j\}}$. Interestingly, the edge set of the skeleton is a subset of J. Therefore, we remove the extra edges from J to recover the graph of skeleton. A sketch of the AdaPC algorithm is:

1 construct $\mathcal{M}(V, J)$ using penalized regression;

2 remove edges arising from a v-structure plus a common ancestor or directed path using a modified PC-algorithm.

The remainder of the article is organized as follows: in Section 4.2, we discuss some background, terminology, and the concept behind our AdaPC algorithm; in Section 4.3 we introduce the details of the AdaPC algorithm; in Section 4.4, we compares the performance of the AdaPC with other existing methods; in Section 4.5, we illustrate the performance

of our method through human Gilioblastoma multiforme cancer data; and finally close this Chapter with discussion in Section 4.6.

4.2 Preliminaries

4.2.1 Definition and Terminology for DAG

A graph $\mathcal{G}(V, E)$ includes a vertex set $V = 1, ..., p$ and a edge set $E \subseteq V \times V$ that connect some pair of vertices. In our setting, the vertex corresponds to the component of a $p-$dimensional random vector $X = (X_1, \ldots, X_p)$, and the edge $(X_i, X_j) \in E$ denotes the certain relationship between random variable X_i and X_j. If an edge $(X_i, X_j) \in E$ but $(X_j, X_i) \notin E$, we call it a directed edge; If $(X_i, X_j) \in E$ indicates $(X_j, X_i) \in E$, we denote it an undirected edge. A directed graph $\mathcal{G}$ without directed cycle is denoted as directed acyclic graph (DAG). After removing all the directions of a DAG, we obtain its skeleton.

If there is a directed edge (X_i, X_j) in graph $\mathcal{G}$, the node X_i is said to be adjacent with X_j, and is a parent of node X_j. We denote the set of parents of X_j as $pa(X_j)$, and the set of adjacent nodes of X_j as $adj(\mathcal{G}, X_j)$. A directed path in $\mathcal{G}$ is a sequence of distinct vertices in which a directed edge pointing from each vertex in the sequence to its successor. If there is a directed path from X to Y, X is an ancestor of Y, and Y is a descendant of X. A v-structure is the form $X_i \to X_j \leftarrow X_k$, where X_i and X_k are marginally independent but conditionally dependent given their common descendant X_i.

Consider a set of random variables $(X_1, ..., X_p)^{\mathrm{T}}$ following a distribution $\mathcal{P}$. This distribution $\mathcal{P}$ is said to factorize according to a DAG $\mathcal{G}(V, E)$ if $\mathcal{P}$ can be written as the product of a serial of conditional density:

$$\mathcal{P}(X) = \prod_{i=1}^{p} \mathcal{P}_i(X_i | pa(\mathcal{G}, X_i)).$$

The conditional independence relationship in a DAG can be inferred by the concept of d-separation (Pearl, 2000). For example, if $X_i \notin adj(\mathcal{G}, X_j)$, then X_i and X_j are d-separated in $\mathcal{G}$ by a set $S \subseteq (V/(X_i, X_j))$. Namely, $X_i \perp\!\!\!\perp X_j | S$ in any distribution $\mathcal{P}$ which is factorize according to $\mathcal{G}$. A distribution $\mathcal{P}$ is faithful to a DAG $\mathcal{G}(V, E)$, if the conditional independence relationship inferred from $\mathcal{G}(V, E)$ using d-separation is exactly the same as the

conditional independence relationship in $\mathcal{P}$. For simplicity, we assume the random vector $(X_i, ..., X_p)^{\mathrm{T}}$ follows a multivariate Gaussian distribution. It is worth to pointing out that not all distribution $\mathcal{P}$ can be faithfully represented by a DAG (Spirtes et al., 2000). However, Meek (1995) showed that the non-faithful distribution of multivariate Gaussian family form a Lebesgue null set in the space of distributions associated with a DAG $\mathcal{G}(V, E)$.

4.2.2 Gaussian Graphical Model and Correlation Graph

Let X be a p-dimensional random vector following a multivariate normal distribution $\mathcal{N}(\mu, \Sigma)$, and define $\Omega = (\omega_{ij}) = \Sigma^{-1}$. As discussed in Chapter 1.1, X_i is independent with X_j given the rest of variables if and only if $\omega_{ij} = 0$. Thus Ω can be used to construct a Gaussian Graphical Model (GGM), which represents the conditional dependence relationships among a set of variables. The GGM of X can be represented by an undirected graph $\mathcal{C}(V, F)$, where each variable corresponds to a node in the set V and conditional dependencies are represented by the edges in the set F. The undirected edge between vertices i and j, also denoted as $(i, j) \in F$, if variables X_i and X_j are conditionally independent given all the other variables.

A GMM is different from the skeleton of a DAG because of the so called v-structure. Given a v-structure $X_i \rightarrow X_j \leftarrow X_k$, X_i and X_k are conditionally dependent given any separating set S_{ik} containing X_j. Therefore, X_i and X_k are dependent given all other variables i.e. $(i, k) \in F$.

We also define the skeleton of a DAG as graph $\mathcal{G}(V, E^u)$, where V is the set of vertices and E^u is the set of undirected edges. It has been shown that the set of edges F equals the set of edges E^u from the corresponding DAG plus the edges arising from v-structure, and therefore it implies that $E \subset F$ (Ha et al., 2014).

Let $\mathcal{K}(V, H)$ represent the correlation graph (CG) corresponding to marginal correlation relationship between variables. Specifically, an undirected edge $(i, j) \in H$ if and only if variables X_i and X_j are marginally dependent (correlated under Gaussian assumption). While correlation does not equal causality, causal relationship infers correlation. When $\mathcal{P}$ is faithful to $\mathcal{G}(V, E)$, there is an edge between X_i and X_j in a skeleton $\mathcal{G}(V, E^u)$, if and only if X_i and X_j are conditionally dependent given any subset in $\{V \setminus \{i, j\}\}$ including $\emptyset$

(Spirtes et al., 2000). Hence E^u is also a subset of H. The extra edges in H come from two structures:

- a directed path between two nodes ($X \to \ldots \to Z \to \ldots \to Y$),
- two nodes have a common ancestor ($X \leftarrow \ldots \leftarrow Z \to \ldots \to Y$).

We denote $\mathcal{M}(V, J)$ as the graph representing the dependency (including marginal dependence and conditional dependence given all other variables) among a set of random variables. An undirected edge (i, j) belongs to J, if and only if X_i and X_j are both marginally dependent and conditional dependent, namely $J = F \cap H$.

The relationship among a DAG, the skeleton, GGM and CG is illustrated in Figure 4.1 using a toy example. Two different DAGs are shown in Figure 4.1 (a) and (b), with X, Y, Z, and W represent different random variables. The corresponding skeletons, GGMs, CG and GGM $\cap$ CG are shown in the rest of Figure 4.1. In Figure 4.1 (c) and (h), X and Z are connected in GGM due to the v-structure $X \to W \leftarrow Z$. Because of the directed path $Y \to Z \to W$, Y and W are connected in CG as shown in Figure 4.1 (d) and (i). Similarly, because of the co-ancestor structure ($X \leftarrow Y \to Z$) as shown in Figure 4.1 (f), there is an edge between X and Z in CG as shown in Figure 4.1 (i). The common edges shared among GGM and CG, which are the sparest graph and also contain the true DAG structures are shown in Figure 4.1 (e) and (j). In summary, the difference between graph $\mathcal{M}(V, J)$ and skeleton $\mathcal{G}(V, E^u)$ is due to a v-structure plus a common ancestor or a directed path.

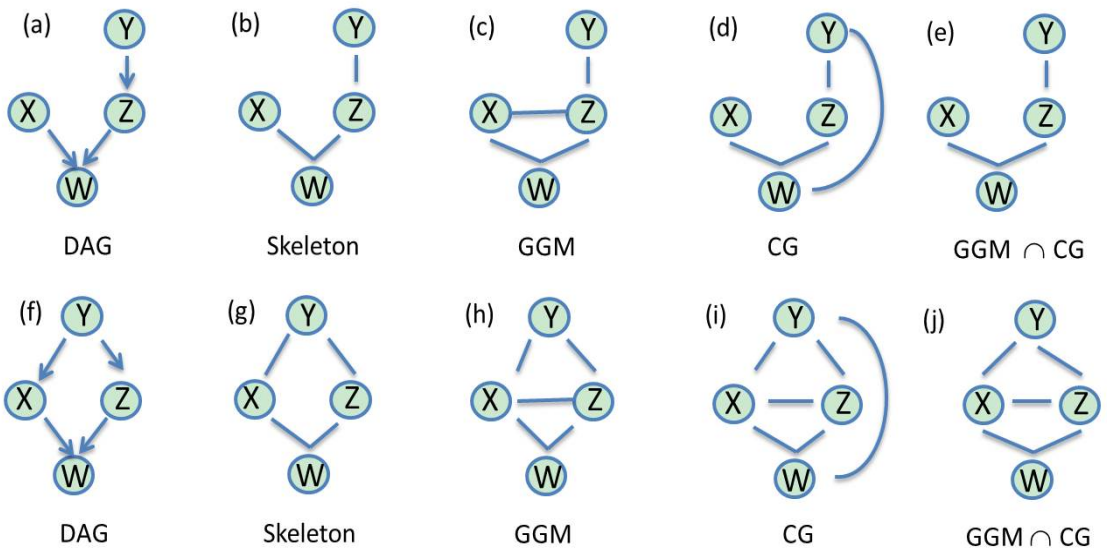

Figure 4.1: Illustration of the relationship among DAG, skeleton, GGN, CG, and GGN ∩ CG.

4.3 Methodology

For convenience, we use the following notation throughout this Chapter. We denote the true precision and covariance matrices respectively as $\Omega^* = (\Sigma^*)^{-1}$. For any matrix $W = (\omega_{ij})$, we denote the determinant as $\det(W)$ and the trace as $\mathrm{tr}(W)$. We further denote the minimum and maximum eigenvalues of W as $\phi_{\min}(W)$ and $\phi_{\max}(W)$. The Frobenius norm of W is defined as $\|W\|_F^2 = \sum_{i,j} w_{ij}^2$, and the operator/spectral norm $\|W\|^2$ is defined as $\phi_{\max}(WW^{\mathrm{T}})$. Given a DAG $\mathcal{G}(V, E)$, we define the corresponding skeleton, GGM, CG and GGM ∩ CG as $\mathcal{G}(V, E^u)$, $\mathcal{C}(V, F)$, $\mathcal{K}(V, H)$ and $\mathcal{M}(V, J)$ respectively.

4.3.1 Problem Formulation

We have n independent observations of the $p-$dimensional random vector X denoting as $\mathbf{X}$ which is a $p \times n$ dimensional matrix. Let $\mathbf{X}_i$ correspond to the vector of n independent observations of X_i. There is some level of dependency among the outcomes variables, which is entailed by an underlying DAG: $\mathcal{G}(V, E)$. Let X_i be the $i-$th random variable, we model

$$X_i = \sum_{j \in pa(\mathcal{G}, X_i)} a_{i,j} X_j + Z_i, \tag{4.1}$$

where the $Z_i \sim \mathcal{N}(0, \sigma_i^2)$ is the latent variables representing the unexplained variation in the node X_i, and $a_{i,j}$ is the direct effect of node X_j on X_i for $j \in pa(\mathcal{G}, x_i)$. Define the adjacency matrix of DAG $\mathcal{G}(V, E)$ as $A = (a_{i,j})$. We assume X follows a multivariate Gaussian distribution with covariance Σ. The adjacency matrix A is not symmetric, and its non zero entries uniquely entail the structure of the corresponding DAG. Additionally, the covariance matrix, Σ, and the precision matrix, Ω, for the random vector X have the following relationship with A:

$$\Sigma = (\{_d\ \sigma_i\} - A)^{-1}(\{_d\ \sigma_i\} - A)^{-T}$$
$$\Omega = (\{_d\ \sigma_i\} - A)^{T}(\{_d\ \sigma_i\} - A)\ ,$$

Our gaol is to estimate the skeleton $\mathcal{G}(V, E^u)$. In the next section, we introduce the AdaPC method using modified penalized regression with adaptive group lasso penalty to estimate the skeleton.

4.3.2 The AdaPC Algorithm

The idea behind our AdaPC algorithm is that we want to first estimate the common edges between $\mathcal{C}(V, F)$ and $\mathcal{K}(V, H)$. Namely, we want our estimate $(\hat{\omega}_{ij}\hat{\sigma}_{ij})_{1 \le i,j \le p} \neq 0$ given $(\omega^*_{ij}\sigma^*_{ij})_{1 \le i,j \le p} \neq 0$; and $(\hat{\omega}_{ij}\hat{\sigma}_{ij})_{1 \le i,j=p} = 0$ given $(\omega^*_{ij}\sigma^*_{ij})_{1 \le i,j \le p} = 0$. One way to solve this problem is using group lasso. However, group lasso is not suitable in this situation, since it attempts to identify common 0 structure from Σ and Ω instead of common nonzero. Therefore, we propose a modified adaptive group lasso to address this issue. Our AdaPC algorithm proceeds in two steps:

1. estimation of $\mathcal{M}(V, J)$ by neighbourhood selection,

2. removal of extra edges by a modified PC-stable algorithm.

Step 0 (Initial Estimation). Given a vertex j, we obtain the initial estimators for $\theta_j = (\theta_{1,j}, \ldots, \theta_{j-1,j}, \theta_{j+1,j}, \ldots \theta_{p,j})$ and $\beta_j = (\beta_{j,1}, \ldots, \beta_{j,j-1}, \beta_{j,j+1}, \ldots \beta_{j,p})$ as follows,

$$\theta_j^{\text{int}} = \underset{\theta_j}{\text{argmin}} \left(n^{-1}\|\mathbf{X}_j - \mathbf{X}_{-j}\theta_j)\|_2 + \lambda_1|\theta_j|_1\right),$$
$$\beta_{j,i}^{\text{int}} = \frac{\mathbf{X}_i^{\text{T}}\mathbf{X}_j}{\|\mathbf{X}_i\|\|\mathbf{X}_j\|}, \text{ for } i \neq j,$$

where λ_1 is the tuning parameter. .

Step 1 (Neighborhood Selection). Denote $\gamma_j = (\theta_j^{\text{T}}, \beta_j^{\text{T}})^{\text{T}}$, $\mathbf{Y}_j = (\mathbf{X}_j^{\text{T}}, \mathbf{X}_j^{\text{T}}/\sqrt{p}, \ldots, \mathbf{X}_j^{\text{T}}/\sqrt{p})^{\text{T}}$, and define $\mathbf{Z}_j$ as,

$$\mathbf{Z}_j = \begin{pmatrix} \mathbf{X}_{-j} & 0 & \cdots & 0 \\ 0 & \mathbf{X}_1/\sqrt{p} & \cdots & 0 \\ \vdots & 0 & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{X}_p/\sqrt{p} \end{pmatrix}.$$

We estimate γ_j by solving

$$\{\hat{\theta}_j, \hat{\beta}_j\} = \underset{\gamma_j}{\text{argmin}} \left(n^{-1}\|\mathbf{Y}_j - \mathbf{Z}_j\gamma_j^{\text{T}}\| + \lambda_2 \sum_{i \neq j} w_{ij}\sqrt{\theta_{j,i}^2 + \beta_{j,i}^2}\right), \tag{4.2}$$

where $w_{ij} = (\hat{\theta}_{i,j} \wedge \hat{\beta}_{i,j})^{-1}$ is the weight associated with the ith group. After p penalized regressions, we construct the $\mathcal{M}(V, J)$ as follows: $(i, j) \in J$ if $\hat{\theta}_{i,j}\hat{\beta}_{i,j} \neq 0$ or $\hat{\theta}_{j,i}\hat{\beta}_{j,i} \neq 0$ for $i \neq j$.

Since $\theta_{i,j} = -\omega_{ij}/\omega_{jj}$ and $\beta_{i,j} = \sigma_{ij}$, identifying the non-zero entries in $(\theta_1, \ldots, \theta_p)$ and $(\beta_1, ..., \beta_p)$ equals estimating the non-zero off-diagonal entries in Ω and Σ respectively. If using group lasso, it tends to shrink the group $(\theta_{i,j}, \beta_{i,j})$ to 0, when both elements are close to 0. On the other hand, when one of the elements significantly differs from 0, it leads to make both elements non-zero. However, with the weight w_{ij}, we would place a large weight on the group where at least one of the elements is close to zero, and hence shrink the estimate to zero.

Step 2 (Modified PC-algorithm). We recover the skeleton $\mathcal{G}(V, E^u)$ by applying a modified PC-stable algorithm to remove the extra edges on $\mathcal{M}(V, J)$. Given a undirected graph

$\mathcal{A}(V, L)$, we denote the subgraph of $\mathcal{A}$ on a subset of vertices V_S as $\mathcal{A}(V_S)$ and define some terminologies as follows:

$$\text{adj}(\mathcal{A},\, i) = \{m : (i, m) \in L\},$$

$$\text{adj}(\mathcal{A},\, i,\, j) = \Big\{m : m \in \text{adj}(\mathcal{A},\, i) \cap \text{adj}(\mathcal{A}, j)\Big\},$$

$$\text{Con}(\mathcal{A}, v) = \{\ l\text{: there is a path between } l \text{ and } v \text{ including } v \text{ itself }\},$$

$$\text{Loop}(\mathcal{A}, i, j) = \{l : \text{there is a loop containing } i, j \text{ and } l\ \},$$

$$\mathcal{S}(\mathcal{A})_{ij} = \Big[\bigcup_{v \in \text{adj}(\mathcal{A},i,j)} \text{Con}\big(\mathcal{A}(V \setminus \{i, j\}), v\big)\Big] \bigcap \Big[\text{adj}(\mathcal{A}, i) \bigcup \text{adj}(\mathcal{A}, j)\Big].$$

The modified PC-algorithm is described as follows:

1. Set $k = 0$ and $\mathcal{A}(V, L) = \mathcal{M}(V, J)$.
2. **Repeat:** $k = k + 1$

   2.1 **Repeat:** Select an edge $(i, j) \in L$ with $|\mathcal{S}(\mathcal{A})_{ij}| \geq k$

   2.1.1 **Repeat:** Select an vertex set $V_s \subseteq \mathcal{S}(\mathcal{A})_{ij}$ with $|V_s| = k$

   2.1.1.1 Set $V_k = \Big[\text{adj}(\mathcal{A}, i) \bigcup \text{adj}(\mathcal{A}, j)\Big] \bigcap \text{Loop}(\mathcal{A}, i, j) \setminus [V_s \cup \{i, j\}]$

   2.1.1.2 If X_i and X_j are conditionally independent given $\{X_l : l \in V_s\}$, then $L = L \setminus (i, j)$

   2.1.2 **Until:** $(i, j) \notin L$ or all $|V_s| = k$ have been selected

   2.2 **Until:** all edges $(i, j) \in L$ with $|\mathcal{S}(\mathcal{A})_{ij}| \geq k$ have been selected

3 **Until:** all edges $(i, j) \in L$ have $|\mathcal{S}(\mathcal{A})_{ij}| < k$

The rational behind this step is briefly described as follows: if two nodes i and j connected in $\mathcal{M}$ but not in skeleton, they must come from a structure containing a v-structure (inducing $(i, j) \in F$) and a common ancestor or directed path between i and j (inducing $(i, j) \in H$).

4.4 Simulation Examples

We compare the performances of the AdaPC algorithm with the PC-stable algorithm by applying them to various simulated data sets. The PC-stable algorithm is implemented

using the R-package pcalg. Our results suggest that our algorithms dominate the PC-stable algorithms in both low and high-dimensional settings.

4.4.1 Simulating set-up

We used the following procedure to generate a DAG using the ER model. First, we generated a random adjacency matrix A with all entries equal zero. We then randomly select $pE/2$ entries from the lower triangle and set them to be the independent realizations of a $U([0.1:1])$ random variable, where p is the dimension of graph and E is the expected edges for each node. The nonzero entry $A_{i,j}$ can be interpreted as the weight for the edge (i,j). The structures of the DAG with $E \in \{2,5\}$ and $p = 20$ are shown in Figures 4.2 (a) and (b) respectively.

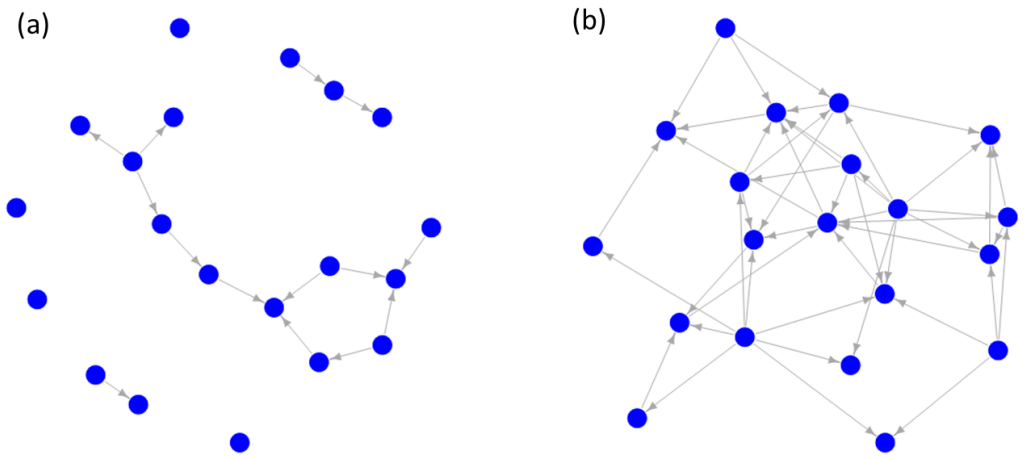

Figure 4.2: DAG topologies used in the simulations. (a) and (b) shows the DAG with $E = 2$ and 5 respectively.

With adjacency matrix A, we generate the corresponding Σ and Ω using the following formulae:

$$\Sigma = (I - A)^{-1}(I - A)^{-T}$$

$$\Omega = (I - A)^{T}(I - A),$$

and generate i.i.d sample by $X \sim \mathcal{N}(0, \Sigma)$.

4.4.2 Relationship between $\mathcal{M}$ and skeleton

In this section, we focus only on the relationship between graph $\mathcal{M}$ and skeleton under different sparsities. We simulate different DAGs ($p = 100$) with various sparsity levels $E \in \{5, 10, 20, 50\}$ following the procedure described above.

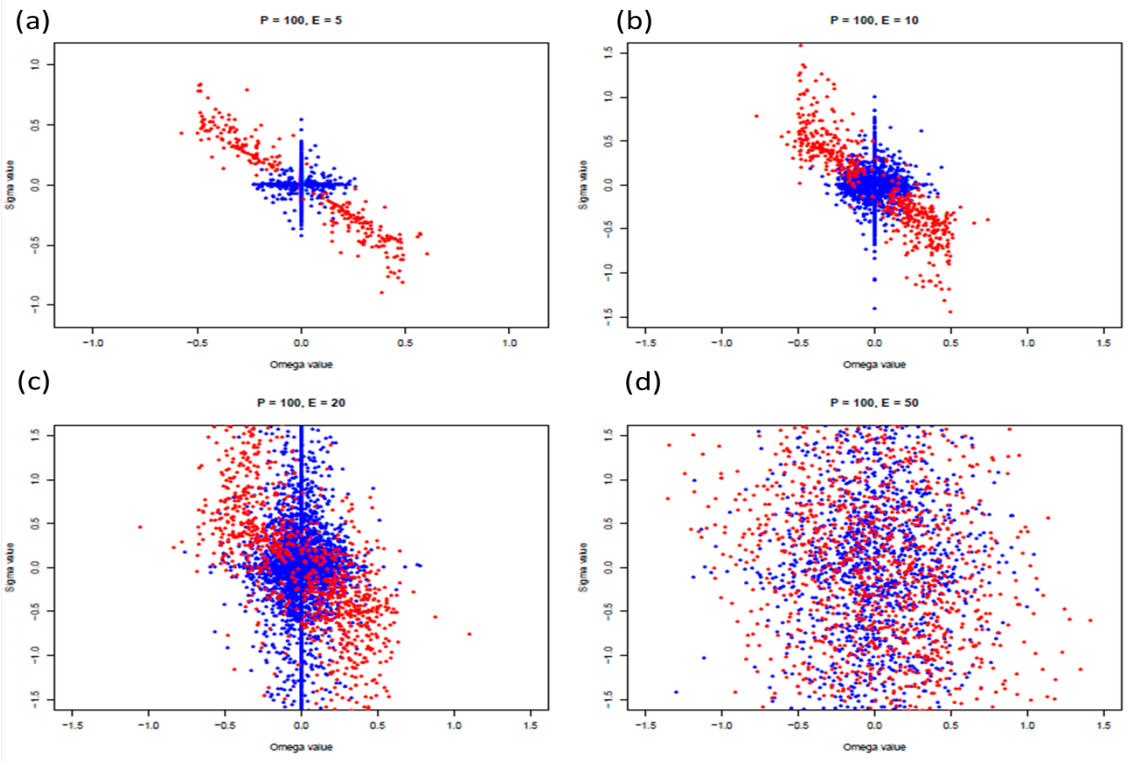

Figure 4.3: Illustration of the relationship between graph $\mathcal{M}$ and skeleton. Each panel plots the Ω value (x-axis) against the Σ value (y-axis) with red dots representing those entries with edges in the skeleton, while blue dots representing entries without edges.

Results reported in Figure 4.3. In general, when the DAG is sparse as in Figure 4.3 (a) and (b), the majority of the red dots (edges in skeleton) are far away from the axes, while the blue dots are mostly on the axes. This corresponds to the fact that graph $\mathcal{M}$ is close to skeleton. With the increase of sparsity, blue and red dots are mixed with each other suggesting that graph $\mathcal{M}$ is significantly different from skeleton as in Figure 4.3 (c) and (d). However in the dense cases, all existing methods would fail to recover skeleton efficiently,

since PC, PenPC and AdaPC algorithms all aim for sparse DAG. This result demonstrates the power of the first step of our AdaPC algorithm.

4.4.3 Estimation of $\mathcal{M}$

We compare the performance of our AdaPC method with a separate method for estimating $\mathcal{M}$. The separate method is described as follows:

1. estimate the sparse precision matrix Ω using neighbourhood selection;
2. estimate the sparse covariance matrix Σ by hard thresholding.

$$\tilde{\sigma}_{ij} = \hat{\sigma}_{ij} I(|\hat{\sigma}_{ij}| > \lambda),$$

   where λ is a prespecified tuning parameter and $\hat{\sigma}_{ij}$ is the sample correlation between X_i and X_j.

3. construct the $\mathcal{M}(V, J)$ by following rule: $(i, j) \in J$, if $\tilde{\sigma}_{ij}\hat{\omega}_{ij} \neq 0$ or $\tilde{\sigma}_{ji}\hat{\Omega}_{ji} \neq 0$.

The results of this comparison are reported in Figure 4.4; each curve is based on 20 replications. In the plots, the ROC curves of the AdaPC method outperform those of the separate method ($\lambda_3 = \{0.01, 0.025, 0.1, \text{eBIC selected}\}$) especially when the structure is sparse.

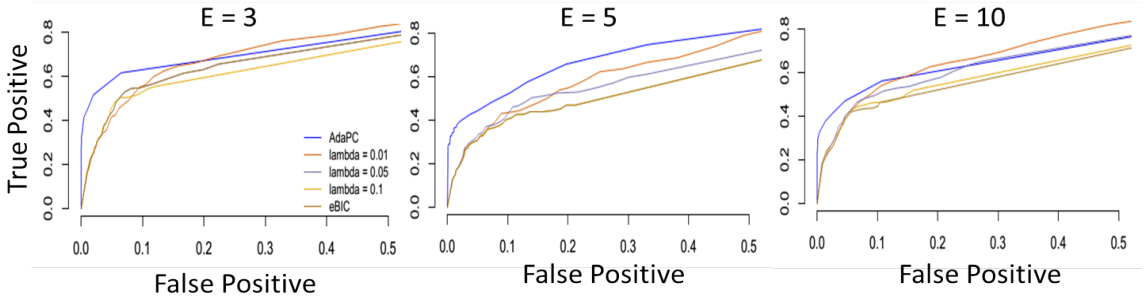

Figure 4.4: Receiver operating characteristic (ROC) curve for estimating $\mathcal{M}$. Each panel reports the performance of the AdaPC method (blue line) and the separate method (orange, grey, yellow and brown lines represent the fixed $\lambda_3$for a given sparsity parameter E.

4.4.4 Estimation of the Skeleton

In this subsection, we compared the performance of the AdaPC and PC-stable algorithm for estimating skeleton. The results is shown in Figure 4.5 based on 100 replications. The ROC result in Figure 4.5 shows the performance of the AdaPC and PC-stable algorithms for different sparsity levels ($E = 3, 5$ and 10). Each curve is the average of 100 replication. In the plots, the AdaPC method consistently performs better than the PC algorithm especially when the graph is relative dense. Additionally, when $E = 10$, PC algorithm can only identify around 45% of the true edges even with $\alpha = 0.90$. This is due to the fact that PC algorithm recursively performs many test for each edge, and hence even with high α value it tends to falsely remove many true edges especially when the underlying structure is relative dense. In the other hand, our AdaPC overcomes this drawback through much least tests in the second step. This is another important advantage for AdaPC.

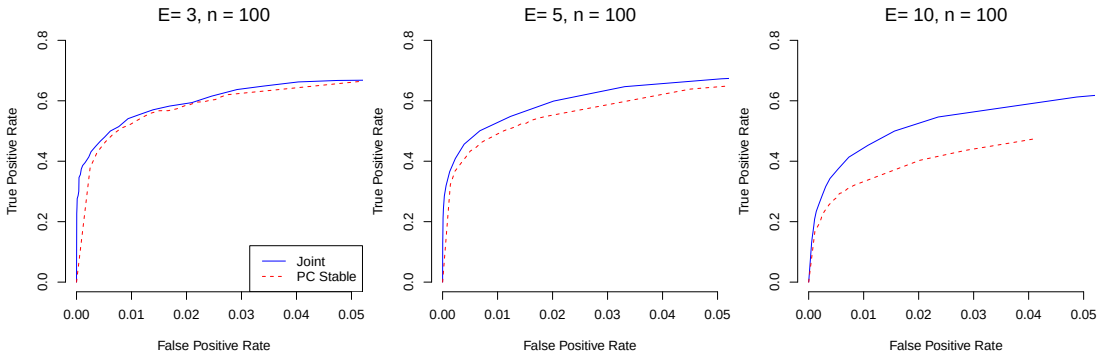

Figure 4.5: ROC curve assessing power and discrimination of estimating the skeleton of a DAG. The blue solid line represents the AdaPC algorithm and dash red line is result from PC-stable algorithm.

4.5 Application

To illustrate the power of our approach in real data, AdaPC was applied to the Glioblastoma multiforme (GBM) cancer data set, consisting of 487 patients with 17814 genes and 534 micro-RNAs, from the Cancer Genome Atlas (TCGA) (The Cancer Genome Atlas Research Network., 2012). Based on microarray data, the patients are classified into four cancer subclasses: 128 Classical,146 Mesenchymal, 86 Neural, and 127 Proneural with sam-

ple size 128, 146, 86 and 127 respectively (Verhaak et al., 2010). Verhaak et al. (2010) selected 840 signature genes(210 genes per class) to best represent each class using ClaNC, on which we base. Our method is designed for single graph, thus we only focus on the largest class, Mesenchymal, with top 394 genes with highest variation in this subclass.

Before estimating the skeleton, we control the effect of micro-RNAs by fitting ℓ_1 penalized linear regression of gene expression against micor-RNA expression. We applied our adaPC method and PC-stable method to the residues. The topologies of the gene networks under different methods are shown in Figures 4.5 a) and b) with the same total amount of edges i.e. 474 edges. The resulting structure from AdaPC is a scale free like structure with only 7 nodes have more than 7 edges (maximum degree is 10). These nodes are often referred as hubs since they are extremely densely connected to other nodes. This scale-free network structure has been shown to be common in biological networks, for example see Hao et al. (2012). The 7 hub genes are *ASCL1, BAI3, GRIA2, GRIK1, KLRC3, NR0B1, SLC4A4*. Here *BAI3* (adhesion G protein-coupled receptor B3), *GRIA2* (glutamate receptor, ionotropic, AMPA 2), *GRIK1* (glutamate receptor, ionotropic, kainate 1) and *NR0B1* (Orphan nuclear receptor), *KLRC3* (killer cell lectin-like receptor subfamily C, member 3), and *NR0B1* (Orphan nuclear receptor) are cell transmembrane receptors which are stimulated by external signals and pass signals by activating downstream proteins. Hence those genes are potential candidates for hubs (Antoni et al., 2010; Choi et al., 2012; Li et al., 2010b; amd Akihisa Imagawa et al., 2013; Tajima et al., 2003). Additionally, *ASCL1*, transcription factors, activates the transcription of target genes through binding to the specific DNA motif, e.g. E box in the promoter region (Augustyn et al., 2014), and thus could be a hub gene.

In comparison, the skeleton inferred from PC-stable algorithm shows a pattern without obvious hub (maximum edge degree is 6). There is no gene connected with more than 6 other genes. Figure 4.7 compares the edge degree distribution between AdaPC and PC-stable methods. The skeleton from the AdaPC method has a heavier right tail compared dwith the one from PC-Stable method. The phenomena is due to the fact for a node with q degrees it need to reject 2^q tests for each edge in order to recover the true structure. However, when q is relative large, e.g in the case of hub, when high probability some of

those test would not reject the null hypothesis and hence lead to removal of the edges. In sum, to estimate skeleton with hubs, AdaPC would be a better choice over PC-stable algorithm.

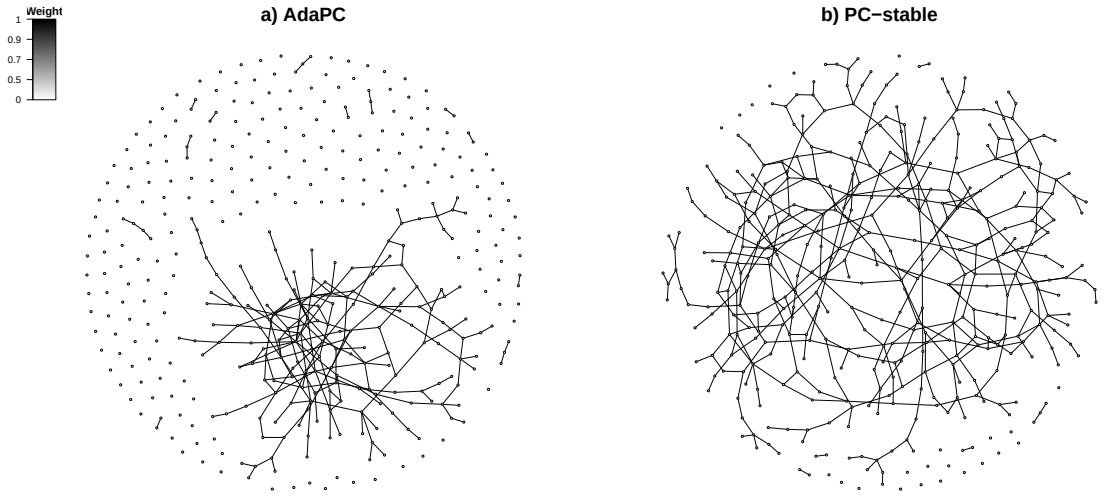

Figure 4.6: Topology of the skeleton networks inferred by the EM method applied to measurements of the 394 genes with highest within-tissue variance in Mesenchymal subclass. Panels a) and b) display the skeleton networks estimated by AdaPC method and PC-stable method respectively.

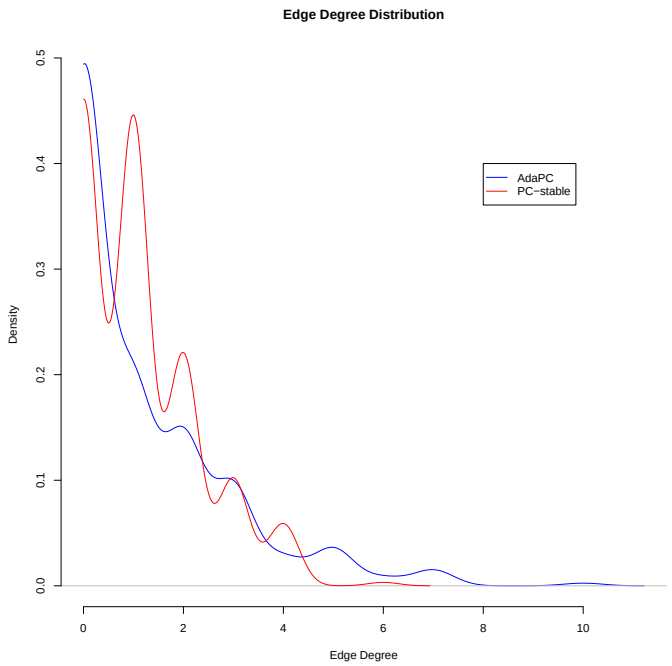

Figure 4.7: Distribution of edge degree from skeleton networks in 4.6.

4.6 Discussion

Due to computational efficiency, causal structure learning algorithms in sparse high dimensional settings are often based on sequential tests such as PC-algorithm and PC-statble algorithm. Due to large amount of tests, those methods only can handle the data with dimension p is polynomial scale of sample size n, and tend to miss a large proportion of true edges when the structure is relatively dense. To address this issue, we propose a two-step approach, the AdaPC, to estimate the skeleton of DAG in high dimensional setting. In the first step, we estimate the $\mathcal{M}$, which is interception of GGM and Covariance graph, using weighted penalized regression. In the following step, we recover the skeleton by removing those extra edges which are only due the special V-structure plus common ancestor, and hence is rare. Numerical studies show that our AdaPC algorithm dominate existing methods. A possible direction of future work is the extend the Gaussian assumption to discrete data such as RNAseq. Another possible direction is to extend the single graph frame work to multiple graphs.

BIBLIOGRAPHY

Ambroise, C., Chiquet, J., and Matias, C. (2009). Inferring sparse gaussian graphical models with latent structure. *Electronic Journal of Statistics*, 3:205–238.

amd Akihisa Imagawa, S. N., Miyata, Y., Yoshikawa, A., Kozawa, J., Okita, K., Funahashi, T., Nakamura, S., Matsubara, K., Iwahashi, H., and Shimomura, I. (2013). Low gene expression levels of activating receptors of natural killer cells (nkg2e and cd94) in patients with fulminant type 1 diabetes. *Immunology L*, 156(1-2):149–155.

Andersson, S. A., Madigan, D., and Perlman, M. D. (1997). A characterization of markov equivalence classes for acyclic digraphs. *The Annals of Statistics*, 25:505–541.

Antoni, G., Morange, P. E., Luo, Y., Saut, N., Burgos, G., Heath, S., Germain, M., Biron-Andreani, C., Schved, J. F., Pernod, G., Galan, P., Zelenika, D., Alessi, M. C., Drouet, L., Visvikis-Siest, S., Wells, P. S., Lathrop, M., Emmerich, J., Tregouet, D. A., and Gagnon, F. (2010). A multi-stage multi-design strategy provides strong evidence that the bai3 locus is associated with early-onset venous thromboembolism. *Journal of Thrombosis and Haemostasis*, 8:2671–2679.

Augustyn, A., Borromeo, M., Wang, T., Fujimoto, J., Shao, C., Dospoy, P., Lee, V., Tan, C., Sullivang, J. P., Larsenh, J. E., Girard, L., Behrens, C., Wistuba, I., Xie, Y., Cobb, M. H., Gazdar, A. F., Johnson, J. E., and Minna, J. D. (2014). Ascl1 is a lineage oncogene providing therapeutic targets for high-grade neuroendocrine lung cancers. *Proceedings of the National Academy of Sciences of the United States of America*, 111(41):14788–14793.

Banerjee, O., Ghaoui, L. E., and d'Aspremont, A. (2008). Model selection through sparse maximum likelihood estimation. *Journal of Machine Learning Research*, 9:485–516.

Bickel, P. and Levina, E. (2008). Regularized estimation of large covariance matrices. *The Annals of Statistics*, 36(1):199–227.

Bien, J. and Tibshirani, R. (2011). Sparse estimation of a covariance matrix. *Biometrika*, 98(4):807–820.

Cai, T., Liu, W., and Luo, X. (2011). A constrained 11 minimization approach to sparse precision matrix estimation. *Journal of American Statistical Association*.

Chen, J. and Chen, Z. (2008). Extended bayesian information criteria for model selection with large model spaces. *Biometrika*, 95:759–771.

Chickering, D. M. and Boutilier, C. (2002). Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3:507–554.

Choi, C., Choi, J.-J., Park, Y.-A., Lee, Y.-Y., Song, S., Sung, C., Song, T., Kim, M.-K., Kim, T.-J., Lee, J.-W., Kim, H.-J., Bae, D.-S., and Kim, B.-G. (2012). Identification of differentially expressed genes according to chemosensitivity in advanced ovarian serous adenocarcinomas: expression of gria2 predicts better survival. *British Journal of Cancer*, 107(1):91–99.

Chow, C. I., Member, S., and Liu, C. N. (1968). Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14:462–467.

Colombo, D. and Maathuis, M. (2013). Order-independent constraint-based causal structure learning. Technical report, Seminar for Statistics, ETH Zurich.

Crowley, J., Zhabotynsky, V., Huang, W. S. S., Pakatci, I., Kim, Y., Wang, J., Morgan, A. P., Calaway, J. D., Aylor, D. L., Yun, Z., Bell, T. A., Buus, R., Calaway, M. E., Didion, J. P., Gooch, T. J., Hansen, S., Robinson, N. N., Shaw, G. D., Spence, J., Quackenbush, C., Barrick, C., Xie, Y., Valdar, W., Lenarcic, A. B., , W. W., Welsh, C. E., Fu, C., Zhang, Z., Holt, J., Guo, Z., Threadgill, D., Tarantino, L. M., Miller, D. R., Zou, F., McMillan, L., Sullivan, P., and de Villena, F. P. M. (2014). Pervasive allelic imbalance revealed by allele-specific gene expression in highly divergent mouse crosses. *Nature Genetics*, revision:revision.

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

d'Aspremont, A., Banerjee, O., and El Ghaoui, L. (2008). First-order methods for sparse covariance selection. *SIAM Journal on Matrix Analysis and Applications*, 30(1):56–66.

Degner, J. F., Marioni, J. C., Pai, A. A., Pickrell, J. K., Nkadori, E., Gilad, Y., and J.K., P. (2009). Effect of read-mapping biases on detecting allele-specific expression from rna-sequencing data. *Bioinformatics*, 25(24):3207–12.

Dempster, A. P., Laird, M. N., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39:1–22.

Dobrin, R., Zhu, J., Molony, C., Argman, C., Parrish, M., Carlson, S., Allan, M., Pomp, D., and Schadt, E. (2009). Multi-tissue coexpression networks reveal unexpected subnetworks associated with disease. *Genome Biology*, 10:55.

Fan, J., Feng, Y., and Wu, Y. (2009). Network exploration via the adaptive lasso and SCAD penalties. *The Annals of Applied Statistics*, 3(2):521–541.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of American Statistical Association*, 96(456):1348–1360.

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

Friedman, N. (2004). Inferring cellular networks using probabilistic graphical models. *Science*, 303(5659):799–805.

Glymour, C. (1987). *Discovering causal structure: artificial intelligence, philosophy of science, and statistical modeling*. Academic Press.

Guo, J., Levina, E., Michailidis, G., and Zhu, J. (2011). Joint estimation of multiple graphical models. *Biometrika*, 98(1):1–15.

Ha, M. J., Sun, W., and Xie, J. (2014). Penpc: A two-step approach to estimate the skeletons of high dimensional directed acyclic graphs. Technical report, University of North Carolina at Chapel Hill.

Hansen, K. D., Brenner, S. E., and Dudoit, S. (2010). Biases in illumina transcriptome sequencing caused by random hexamer priming. *Nucleic Acids Research*, 38:e131.

Hao, D., Ren, C., and Li, C. (2012). Revisiting the variation of clustering coefficient of biological networks suggests new modular structure. *BMC System Biology*, 6:34:1–10.

Heckerman, D. and Chickering, D. M. (1995). Learning bayesian networks: The combination of knowledge and statistical data. In *Machine Learning*, pages 20–197.

Honorio, J. and Samaras, D. (2010). Multi-task learning of gaussian graphical models. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, pages 447–454.

Horn, R. A. and Johnson, C. R. (1985). *Matrix Analysis*. Cambridge University Press.

Hsieh, C.-J., Sustik, M. A., Dhillon, I. S., and Ravikumar, P. (2011). Sparse inverse covariance matrix estimation using quadratic approximation. *Advances in Neural Information Processing Systems*, 24:2330–2338.

Kalisch, M. and Bühlmann, P. (2007). Estimating high-dimensional directed acyclic graphs with the pc-algorithm. *The Journal of Machine Learning Research*, 8:613–636.

Kalisch, M., Fellinghauer, B. A. G., Grill, E., Maathuis, M. H., Mansmann, U., Bühlmann, P., and Stucki, G. (2010). Understanding human functioning using graphical models. *BMC Medical Research Methodology*, 10:1186–1471.

Kalisch, M., Mächler, M., Colombo, D., Maathuis, M. H., and Bühlmann, P. (2012). Causal inference using graphical models with the R package pcalg. *Journal of Statistical Software*, 47:1–26.

Koller, D. and Friedman, N. (2009). *Probabilistic graphical models: principles and techniques*. The MIT Press.

Labaj, P., Leparc, G., Linggi, B., Markillie, L. M., Wiley, S., and Kreil, D. (2011). Characterization and improvement of rna-seq precision in quantitative transcript expression profiling. *Bioinformatics*, 27:383–391.

Lam, C. and Fan, J. (2009). Sparsistency and rates of convergence in large covariance matrices estimation. *The Annals of Statistics*, 37:4254–4278.

Li, H. and Guo, J. (2006). Gradient directed regularization for sparse gaussian concentration graphs, with applications to inference of genetic networks. *Biostatistics*, 7:302–317.

Li, J., Jiang, H., and Wong, W. H. (2010a). Modeling non-uniformity in short-read rates in rna-seq data. *Genome Biology*, 11:R50.

Li, J.-M., Zeng, Y.-J., Peng, F., Li, L., Yang, T.-H., Hong, Z., Lei, D., Chen, Z., and Zhou, D. (2010b). Aberrant glutamate receptor 5 expression in temporal lobe epilepsy lesions. *Brain Rese*.

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

Lumeng, C. (2013). Innate immune activation in obesity. *Molecular Aspects of Medicine*, 34:12–29.

Maathuis, M., Kalisch, M., and Buhlmann, P. (2009). Estimating high-dimensional intervention eeffect from observational data. *The Annals of Statistics*, 37(6A):3133–3164.

Meek, C. (1995). Strong completeness and faithfulness in bayesian networks. In Besnard, P. and Hanks, S., editors, *In Uncertainty in Artificial Intelligence*, pages 411–418. Morgan Kaufmann.

Meinshausen, N. and Bühlmann, P. (2006). High dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, 34:1436–1462.

Milner, J. and Beck, M. (2012). The impact of obesity on the immune response to infection. *Proceedings of the Nutrition Society*, 71:298–306.

Obayashi, T., Hayashi, S., Shibaoka, M., Saeki, M., Ohta, H., and Kinoshita, K. (2008). Coxpresdb: a database of coexpressed gene networks in mammals. *Nucleic Acids Research*, 36:77–82.

Pearl, J. (1995). Causal diagrams for empirical research. *Biometrika*, 82:669–710.

Pearl, J. (2000). *Causality. Models, reasoning, and inference*. Cambridge University Press.

Pearl, J. (2009). *Causality: models, reasoning and inference*. Cambridge University Press.

Peng, J., Wang, P., Zhou, N., and Zhu, J. (2009). Partial correlation estimation by joint sparse regression models. *Journal of the American Statistical Association*, 104:735–746.

Ravikumar, P., Wainwright, M. J., Raskutti, G., and Yu, B. (2011). High-dimensional covariance estimation by minimizing l1-penalized log-determinant divergence. *Electronic Journal of Statistics*, 5:935–980.

Ren, Z., Sun, T., Zhang, C.-H., and Zhou, H. H. (2015). Asymptotic normality and optimalities in estimation of large gaussian graphical model. *Annals of Statistics*.

Rocha, G. V., Zhao, P., and Yu, B. (2008). A path following algorithm for sparse pseudo-likelihood inverse covariance estimation (splice). Technical report, University of California, Berkeley.

Rothman, A. J., Bickel, P. J., Levina, E., and Zhu, J. (2008). Sparse permutation invariant covariance estimation. *Electronic Journal of Statistics*, 2:494–515.

Sachs, K., Perez, O., Pe'er, D., and Douglas A. Lauffenburger, G. P. N. (2005). Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308:523–529.

Shamir, R., Maron-Katz, A., Tanay, A., Linhart, C., Steinfeld, I., Sharan, R., Shiloh, Y., and Elkon, R. (2005). Expander–an integrative program suite for microarray data analysis. *BMC Bioinformatics*, 6:232.

Shen, X., Pan, W., and Zhu, Y. (2012). Likelihood-based selection and sharp parameter estimation. *Journal of American Statistical Association*, 107:223–232.

Spiegelhalter, D. J., Dawid, A. P., Lauritzen, S. L., and Cowell, R. G. (1993). Bayesian analysis in expert systems. *Statistical Science*, 8(3):219–283.

Spirtes, P., Glymour, C., and Scheines, R. (2000). *Causation, Prediction, and Search*. The MIT Press, 2nd edition.

Stekhoven, D. J., Moraes, I., Sveinbjörnsson, G., Hennig, L., Maathuis, M. H., and Bühlmann, P. (2012). Causal stability ranking. *Bioinformatics*, 28(21):2819–2823.

Sun, T. and Zhang, C.-H. (2013). Sparse matrix inversion with scaled lasso. *Jornal of Machine Learning Research*.

Tajima, K., Dantes, A., Yao, Z., Sorokina, K., Kotsuj, F., Seger, R., , and Amsterdam, A. (2003). Down-regulation of steroidogenic response to gonadotropins in human and rat preovulatory granulosa cells involves mitogen-activated protein kinase activation and modulation of dax-1 and steroidogenic factor-1. *The Journal of Clinical Endocrinology & Metabolism*, 88(5):2288–2299.

Tan, K. M., London, P., Mohan, K., Lee, S.-I., Fazel, M., and Witten, D. (2014). Learning graphical models with hubs. Technical report, University of washington.

The Cancer Genome Atlas Research Network. (2012). Comprehensive genomic characterization of squamous cell lung cancers. *Nature*, 489(7417):519–525.

Thorleifsson, G., Walters, G., Gudbjartsson, D., Steinthorsdottir, V., Sulem, P., Helgadotir, A., Styrkarsdottir, U., Gretarsdottir, S., Thorlacius, S., Jonsdottir, I., Jonsdottir, T., and Olafsdottir, E. (2009). Genome-wide association yields new sequence variants at seven loci that associate with measures of obesity. *Nature Genetics*, 41:18–24.

Varoquaux, G., Gramfort, A., Poline, J. B., and Thirion, B. (2010). Brain covariance selection: better individual functional connectivity models using population prior. *Advances in Neural Information Processing Systems 23*, 1050:1–9.

Verhaak, R. G. W., Hoadley, K. A., Purdom, E., Wang, V., Qi, Y., Wilkerson, M. D., Miller, C. R., Ding, L., Golub, T., Mesirov, J. P., Alexe, G., Lawrence, M., O'Kelly, M., Tamayo, P., Weir, B. A., Gabriel, S., Winckler, W., Gupta, S., Jakkula, L., Feiler, H. S., Hodgson, J. G., James, C. D., Sarkaria, J. N., Brennan, C., Kahn, A., Spellman, P. T., Wilson, R. K., Speed, T. P., Gray, J. W., Meyerson, M., Getz, G., Perou, C. M., and Hayes, D. N. (2010). Integrated genomic analysis identifies clinically relevant subtypes of glioblastoma characterized by abnormalities in PDGFRA, IDH1, EGFR, and NF1. *Cancer Cell*, 17(1):98–110.

Verma, T. and Pearl, J. (1990). Equivalence and synthesis of causal models. In Bonissone, P. P., Henrion, M., Kanal, L. N., and Lemmer, J. F., editors, *Sixth Conference on Uncertainty in Artificial Intelligence*, pages 255–270. Elsevier.

Xu, M. and Shao, H. (2012). Solving the matrix nearness problem in the maximum norm by applying a projection and contraction method. *Advances in Operations Research*, 2012:1–15.

Xue, L. and Zou, H. (2012). Regularized rank-based estimation of high-dimensional nonparanormal graphical models. *The Annals of Statistics, to appear*, 40:2541–2571.

Yuan, M. (2010). High dimensional inverse covariance matrix estimation via linear programming. *Journal of Machine Learning Research*, 11:2261–2286.

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

Yuan, M. and Lin, Y. (2007). Model selection and estimation in the gaussian graphical model. *Biometrika*, 94(1):19–35.

Zhang, C.-H. (2010). Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38:894–942.

Zhang, Z., Dehua, L., Dai, G., and Jordan, M. I. (2012). Coherence Functions with Applications in Large-Margin Classification Methods. *Journal of Machine Learning Research*, 13:2705–2734.

Zhou, S., Lafferty, J. D., and Wasserman, L. A. (2010). Time varying undirected graphs. *Machine Learning*, 80(2-3):295–319.