Methods for the estimation and application of biological networks

Patrick Danaher

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2013

Reading Committee:

Pei Wang, Chair

Daniela Witten

Li Hsu

Program Authorized to Offer Degree:
UW Biostatistics

University of Washington

Abstract

Methods for the estimation and application of biological networks

Patrick Danaher

Chair of the Supervisory Committee:
Affiliate Associate Professor Pei Wang
Department of Biostatistics, University of Washington

The advent of high-dimensional biological data from technologies like microarrays and mass spectrometers has transformed both biology and statistical theory; however, the tremendous potential of these datasets to explore the interactive behavior of genes or proteins has been largely unexplored. This dissertation describes two advances in the study of biological networks in these datasets, introducing improved methods for estimating network structure and for describing changes in pathway behavior in disease. The first method, the "Joint Graphical Lasso," is an extension of existing network estimation methods to datasets with multiple classes of observations, for example cancer and healthy cells. We describe a convex penalized likelihood equation whose solution has desirable properties for joint network estimation, and we detail an algorithm for its solution. The second method is a test for biologically meaningful changes in the pattern of co-regulation in biological pathways. Analysis of biological pathways has been almost entirely restricted to investigation of marginal effects; our method instead focuses on the joint behavior of features, examining important and previously unexplored aspects of pathway behavior.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOTATION

Except where defined otherwise, variables in this dissertation adhere to the convention of representing matrices with bold, upper case letters, vectors with bold, lower case letters, and scalars with lower case letters. For example, a data matrix is written $\mathbf{Y}_{n\times p}$, a multivariate observation from that matrix is written $\mathbf{y}_i$, and an observation of a single feature from the matrix is written $y_{i,j}$.

Unless otherwise stated, rows and columns of matrices are indexed with the letters i and j, respectively, and classes of data are indexed with the letter k. n denotes the sample size of a data matrix, and p denotes the number of features, or the dimension.

ACKNOWLEDGMENTS

The author wishes to express sincere appreciation to Pei Wang for her mentoring over the past years, and to Daniela Witten for her considerable contributions to the work in Chapter 2. Additionally, gratitude is due to Noah Simon, Holger Hoefling, Jacob Bien, and Ryan Tibshirani for helpful conversations about the Joint Graphical Lasso and for sharing with us unpublished results; to Jian Guo and Ji Zhu for providing software for the proposal in Guo et al. [2011]; and to Karthik Mohan, Mike Chung, Su-In Lee, Maryam Fazel, and Seungyeop Han for helpful comments.

Chapter 1

INTRODUCTION

Research in statistical methodology for high-dimensional biological data has focused heavily on the marginal behavior of features in these datasets, seeking for example to identify genes whose mean expression changes with disease or proteins whose mean expression is correlated with drug dose. However, marginal approaches ignore the vast amount of information available in the joint behavior of these datasets' features. Examination of joint behavior promises to identify sets of genes and proteins that belong to the same pathways, that act upon each other, or that are regulated by the same entities. By applying the correct tools to high-dimension biological datasets, statisticians are poised to make a massive contribution to this central project of molecular biology. Examination of the joint behavior of features will also increase our understanding of disease. In general, we can expect that disease state will not only affect the mean expression levels of genes and proteins, but also their pattern of co-expression. For example, loss of an important regulatory element in cancer may lead to a tightly co-regulated pathway falling into disarray, a phenomenon that may manifest more prominently in the covariance matrix than the mean vector. Alternatively, as quiescent pathways are activated in response to disease, correlation amongst their genes may be induced by variable degrees of pathway activity.

The initial focus on marginal behavior in high-dimensional biological datasets makes sense logistically: estimating the behavior of the means of each of 20,000

genes is a hard enough problem; estimating their joint distribution is a monumental endeavor. We can simplify our task by focusing on the covariance matrix of the features and ignoring higher moments and non-linear behavior of the features' joint distribution. The challenges of estimating the covariance matrix in high-dimensional data are still considerable, of course. The sample covariance matrix becomes highly unstable and ill-defined when dimension exceeds sample size. Simply the number of parameters required to estimate the covariance matrix, $p(p+1)/2$ instead of the already very large p, creates logistical difficulties and makes overfitting even more of a concern than in marginal analyses. Despite the difficulties involved, our analyses of these datasets will remain incomplete until we include serious investigation of their features' joint behavior alongside the use of our already well-developed techniques for examining their means.

There are more facets to joint behavior than to marginal behavior, and consequently more questions to ask. Below is a brief survey of the types of statistical problems posed by joint behavior. The work described in this dissertation will advance many of these goals.

Estimation

Quantities we can estimate include the features' correlation and covariance matrices, and more usefully, their partial correlation and inverse covariance matrices. These values help identify pairs of genes or proteins that interact or that are subject to the same regulatory element. A number of approaches to point estimation have been described for these quantities. (The project of interval estimation is much more difficult in the high-dimension, low sample size setting typical of these datasets.)

Classification

There has been great effort applied to the problem of using gene expression or protein levels to predict disease state, with several successful diagnostics already in clinical

use. But current methods largely ignore covariance and only look at feature means. Incorporating covariance into these algorithms would make better use of all available information, if it could be done accurately and without overfitting. This approach could be as simple as moving from Linear Discriminant Analysis (LDA) to Quadratic Discriminant Analysis (QDA), though the challenges of accurate inverse covariance estimation in high dimensions will require modifications to standard methods.

Clustering

Clustering is already a very difficult problem in high-dimensional data, and bringing covariance into the process is an ambitious goal. However, if covariance information can be used successfully, clustering algorithms will have access to worlds of previously ignored information. A tractable goal is to use Mahalanobis distance instead of Euclidean distance in distance-based clustering methods. This modest use of covariance information is achievable with existing methodology, though to our knowledge no one has applied the tools of high-dimensional covariance estimation to the creation of a Mahalanobis metric. A more ambitious use of covariance information would be the use of model-based clustering instead of simpler distance-based clustering. The dimensionality and complexity of this goal are daunting, but some authors have made advances on this front [Zhou et al., 2009].

Hypothesis testing

The joint behavior of genes or proteins can be described with numerous hypotheses. We can test for interaction between genes, or for changes in genes' interaction with disease. We can test for interactive effects of genes on disease. We can also test for differences in genes' covariance between disease states, testing both the strong null hypothesis that two covariance matrices are identical and more focused hypotheses targeting aspects of covariance structure that reflect on meaningful biological processes like pathway dysregulation.

This dissertation describes two advances to the investigation of the covariance of features in high-dimensional biological datasets. Chapter 2 discusses the Joint Graphical Lasso (JGL), an extension of existing network estimation methods to multiple classes of data, for example cancer and healthy cells. Since most datasets include at least two classes of data, JGL is a crucial extension, allowing for the first time comparisons of high-dimensional networks between disease states.

Chapter 3 describes a test for between-class differences in biologically meaningful aspects of the covariance structure of the genes in biological pathways. This important scientific goal has been almost entirely ignored by statistics so far, and this test promises to allow more complete pathway-level analyses of high-dimensional biological data.

Chapter 2

THE JOINT GRAPHICAL LASSO FOR INVERSE COVARIANCE ESTIMATION ACROSS MULTIPLE CLASSES

2.1 Background

In recent years, much interest has focused upon estimating an undirected graphical model on the basis of a $n \times p$ data matrix $\mathbf{X}$, where n is the number of observations and p is the number of features. Suppose that the observations $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^p$ are independent and identically distributed $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\mu} \in \mathbb{R}^p$ and $\boldsymbol{\Sigma}$ is a positive definite $p \times p$ matrix. Then zeros in the inverse covariance matrix $\boldsymbol{\Sigma}^{-1}$ correspond to pairs of features that are conditionally independent – that is, pairs of variables that are independent of each other, given all of the other variables in the data set. In a Gaussian graphical model [Lauritzen, 1996], these conditional dependence relationships are represented by a graph in which nodes represent features and edges connect conditionally dependent pairs of features.

A natural way to estimate the *precision* (or *concentration*) matrix $\boldsymbol{\Sigma}^{-1}$ is via maximum likelihood. Letting $\mathbf{S}$ denote the empirical covariance matrix of $\mathbf{X}$, the Gaussian log likelihood takes the form (up to a constant)

$$\frac{n}{2}\left(\log\det\boldsymbol{\Sigma}^{-1} - \operatorname{trace}(\mathbf{S}\boldsymbol{\Sigma}^{-1})\right). \tag{2.1.1}$$

Maximizing (2.1.1) with respect to $\boldsymbol{\Sigma}^{-1}$ yields the maximum likelihood estimate $\mathbf{S}^{-1}$.

However, two problems can arise in using this maximum likelihood approach to

estimate $\mathbf{\Sigma}^{-1}$. First, in the high-dimensional setting where the number of features p is larger than the number of observations n, the empirical covariance matrix $\mathbf{S}$ is singular and so cannot be inverted to yield an estimate of $\mathbf{\Sigma}^{-1}$. If $p \approx n$, then even if $\mathbf{S}$ is not singular, the maximum likelihood estimate for $\mathbf{\Sigma}^{-1}$ will suffer from very high variance. Second, one often is interested in identifying pairs of variables that are unconnected in the graphical model, i.e. that are conditionally independent; these correspond to zeros in $\mathbf{\Sigma}^{-1}$. But maximizing the log likelihood (2.1.1) will in general yield an estimate of $\mathbf{\Sigma}^{-1}$ with no elements that are exactly equal to zero.

In recent years, a number of proposals have been made for estimating $\mathbf{\Sigma}^{-1}$ in the high-dimensional setting in such a way that the resulting estimate is *sparse*. Edwards [2000] suggests a backward stepwise selection method for edges in the graphical model. It begins with a fully connected model, and in successive iterations resets partial correlations with above-threshold p-values to zero and then recalculates the partial correlation matrix and its p-values. This stepwise selection procedure distorts the p-values, making a meaningful threshold impossible to define. Drton and Perlman [2004] apply Fisher's variance-stabilizing z-transformation to the sample correlations, allowing calculation of simultaneous confidence intervals for all partial correlations. They can thereby provide meaningful p-values for all edges. Drton and Perlman [2007] extend this approach to estimation of other types of graphical models.

The testing-based methods described above fail when $p > n$ and the MLE of $\mathbf{\Sigma}^{-1}$ becomes undefined. A number of recent papers extend sparse $\mathbf{\Sigma}^{-1}$ estimation to the high dimension, low sample size setting. Meinshausen and Bühlmann [2006] propose doing this via a penalized regression approach, which is improved upon by Peng et al. [2009]. A number of authors instead take a penalized log likelihood approach [Yuan and Lin, 2007a, Friedman et al., 2007b, Rothman et al., 2008]: rather than maximizing

(2.1.1), one can instead solve the problem

$$\text{maximize}_{\mathbf{\Theta}} \left\{ \log\det \mathbf{\Theta} - \text{trace}(\mathbf{S\Theta}) - \lambda ||\mathbf{\Theta}||_1 \right\}, \tag{2.1.2}$$

where λ is a nonnegative tuning parameter. This is a convex optimization problem, and its solution [Yuan, 2008, Friedman et al., 2007b, Rothman et al., 2008] provides an estimate for $\mathbf{\Sigma}^{-1}$. The use of an ℓ_1 or *lasso* [Tibshirani, 1996] penalty on the elements of $\mathbf{\Theta}$ has the effect that when the tuning parameter λ is large, some elements of the resulting precision matrix estimate will be exactly equal to zero. Moreover, (2.1.2) can be solved even if $p \gg n$. The solution to the problem (2.1.2) is referred to as the *graphical lasso*. Some authors have proposed applying the ℓ_1 penalty in (2.1.2) only to the off-diagonal elements of $\mathbf{\Theta}$.

Graphical models are especially of interest in the analysis of gene expression data, since it is believed that genes operate in pathways, or networks. Graphical models based on gene expression data can provide a useful tool for visualizing the relationships among genes and for generating biological hypotheses. The standard formulation for estimating a Gaussian graphical model assumes that each observation is drawn from the same distribution. However, in many datasets the observations may correspond to several distinct classes, so the assumption that all observations are drawn from the same distribution is inappropriate. For instance, suppose that a cancer researcher collects gene expression measurements for a set of cancer tissue samples and a set of normal tissue samples. In this case, one might want to estimate a graphical model for the cancer samples and a graphical model for the normal samples. One would expect the two graphical models to be similar to each other, since both are based upon the same type of tissue, but also to have important differences stemming from the fact that gene networks are often dysregulated in cancer. Estimating separate graphical models for the cancer and normal samples does not exploit the similarity between the

true graphical models. And estimating a single graphical model for the cancer and normal samples ignores the fact that we do not expect the true graphical models to be identical, and that the differences between the graphical models may be of interest.

In this chapter, we propose the *joint graphical lasso*, a technique for jointly estimating multiple graphical models corresponding to distinct but related conditions, such as cancer and normal tissue. Our approach is an extension of the graphical lasso (2.1.2) to the case of multiple data sets. It is based upon a penalized log likelihood approach, where the choice of penalty depends on the characteristics of the graphical models that we expect to be shared across conditions.

We illustrate our method with a small toy example that consists of observations from two classes. Within each class, the observations are independent and identically distributed according to a normal distribution. The two classes have distinct covariance matrices. When we apply the graphical lasso separately to the observations in each class, the resulting graphical model estimates are less accurate than when we use our joint graphical lasso approach. Results are shown in Figure 2.1.

The rest of this chapter is organized as follows. In Section 2.2, we present the joint graphical lasso optimization problem. Section 2.3 contains an alternating directions method of multipliers algorithm for its solution. In Section 2.4, we present theoretical results that lead to massive gains in the algorithm's computational efficiency. Section 2.5 contains a discussion of related approaches from the literature, and in Section 2.6 we discuss tuning parameter selection. In Section 2.7, we illustrate the performance of our proposal in a simulation study. Section 2.8 contains an application to a lung cancer gene expression dataset. The discussion is in Section 2.9.

2.2 The joint graphical lasso

We briefly introduce some notation that will be used throughout this chapter.

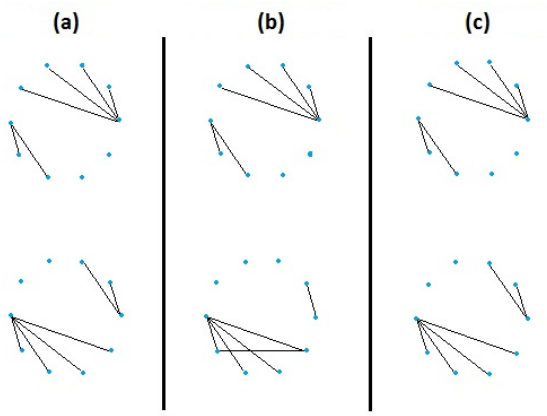

Figure 2.1: *Comparison of the graphical lasso with our joint graphical lasso in a toy example with two conditions, $p = 10$ variables, and n=200 observations per condition.* **(a)***: True networks.* **(b)***: Networks estimated by applying the graphical lasso separately to each class.* **(c)***: Networks estimated by applying our joint graphical lasso proposal.*

We let K denote the number of classes in our data, and let $\mathbf{\Sigma}_k^{-1}$ denote the true precision matrix for the kth class. We will seek to estimate $\mathbf{\Sigma}_1^{-1}, \ldots, \mathbf{\Sigma}_K^{-1}$ by formulating convex optimization problems with arguments $\{\mathbf{\Theta}\} = \mathbf{\Theta}^{(1)}, \ldots, \mathbf{\Theta}^{(K)}$. The solutions $\hat{\mathbf{\Theta}}^{(1)}, \ldots, \hat{\mathbf{\Theta}}^{(K)}$ to these optimization problems will constitute estimates of $\mathbf{\Sigma}_1^{-1}, \ldots, \mathbf{\Sigma}_K^{-1}$.

We will index matrix elements using $i = 1, \ldots, p$, $j = 1, \ldots, p$, and will index classes using $k = 1, \ldots, K$. $\|\mathbf{A}\|_F$ will denote the Frobenius norm of matrix $\mathbf{A}$, *i.e.* $\|\mathbf{A}\|_F = \sqrt{\sum_i \sum_j A_{ij}^2}$.

2.2.1 The general formulation for the joint graphical lasso

Suppose that we are given K data sets, $\mathbf{Y}^{(1)}, \ldots, \mathbf{Y}^{(K)}$, with $K \geq 2$. $\mathbf{Y}^{(k)}$ is a $n_k \times p$ matrix consisting of n_k observations with measurements on a set of p features, which are common to all K data sets. Furthermore, we assume that the $\sum_{k=1}^K n_k$

observations are independent, and that the observations within each data set are identically distributed: $\mathbf{y}_1^{(k)}, \ldots, \mathbf{y}_{n_k}^{(k)} \sim N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$. Without loss of generality, we assume that the features within each data set are centered such that $\boldsymbol{\mu}_k = \mathbf{0}$. We let $\mathbf{S}^{(k)} = \frac{1}{n_k}(\mathbf{Y}^{(k)})^T\mathbf{Y}^{(k)}$, the empirical covariance matrix for $\mathbf{Y}^{(k)}$. The log likelihood for the data takes the form (up to a constant)

$$\ell(\{\boldsymbol{\Theta}\}) = \frac{1}{2}\sum_{k=1}^{K} n_k \left(\log\det \boldsymbol{\Theta}^{(k)} - \text{trace}(\mathbf{S}^{(k)}\boldsymbol{\Theta}^{(k)})\right). \tag{2.2.3}$$

Maximizing (2.2.3) with respect to $\boldsymbol{\Theta}^{(1)}, \ldots, \boldsymbol{\Theta}^{(K)}$ yields the maximum likelihood estimate $(\mathbf{S}^{(1)})^{-1}, \ldots, (\mathbf{S}^{(K)})^{-1}$.

However, depending on the application, the maximum likelihood estimates that result from (2.2.3) may not be satisfactory. When p is smaller than but close to n_k, the maximum likelihood estimate can have very high variance, and no elements of $(\mathbf{S}^{(1)})^{-1}, \ldots, (\mathbf{S}^{(K)})^{-1}$ will be zero, leading to difficulties in interpretation. In addition, when $p > n_k$, the maximum likelihood estimate becomes ill-defined. Moreover, if the K data sets correspond to observations collected from K distinct but related classes, then one might wish to borrow strength across the K classes to estimate the K precision matrices, rather than estimating each precision matrix separately.

Therefore, instead of estimating $\boldsymbol{\Sigma}_1^{-1}, \ldots, \boldsymbol{\Sigma}_K^{-1}$ by maximizing (2.2.3), we consider the penalized log likelihood and seek $\{\hat{\boldsymbol{\Theta}}\}$ solving

$$\text{maximize}_{\{\boldsymbol{\Theta}\}} \left\{ \sum_{k=1}^{K} n_k \left(\log\det \boldsymbol{\Theta}^{(k)} - \text{trace}\left(\mathbf{S}^{(k)}\boldsymbol{\Theta}^{(k)}\right)\right) - P(\{\boldsymbol{\Theta}\}) \right\} \tag{2.2.4}$$

subject to the constraint that $\boldsymbol{\Theta}^{(1)}, \ldots, \boldsymbol{\Theta}^{(K)}$ are positive definite. Here $P(\{\boldsymbol{\Theta}\})$ denotes a convex penalty function, so that the objective in (2.2.4) is strictly concave in $\{\boldsymbol{\Theta}\}$. We propose to choose a penalty function P that will encourage $\hat{\boldsymbol{\Theta}}^{(1)}, \ldots, \hat{\boldsymbol{\Theta}}^{(K)}$ to

share certain characteristics, such as the locations or values of the nonzero elements; moreover, we would like the estimated precision matrices to be sparse. In particular, we will consider penalty functions that take the form $P(\{\mathbf{\Theta}\}) = \lambda_1 \sum_k \sum_{i \neq j} |\theta_{ij}^{(k)}| + \tilde{P}(\{\mathbf{\Theta}\})$, where $\tilde{P}$ is a convex function and λ_1 is a nonnegative tuning parameter. When $\tilde{P}(\{\mathbf{\Theta}\}) = 0$, (2.2.4) amounts to performing K uncoupled graphical lasso optimization problems (2.1.2). The $\tilde{P}$ penalty is chosen to encourage similarity across the K estimated precision matrices; therefore, we refer to the solution to (2.2.4) as the *joint graphical lasso* (JGL). We discuss specific forms of the penalty function in (2.2.4) in the next section.

2.2.2 Two useful penalty functions

In this section, we introduce two particular choices of the convex penalty function P in (2.2.4) that lead to useful graphical model estimates. In Appendix 1, we further extend these proposals to work on the scale of partial correlations.

The fused graphical lasso

The *fused graphical lasso* (FGL) is the solution to the problem (2.2.4) with the penalty

$$P(\{\mathbf{\Theta}\}) = \lambda_1 \sum_{k=1}^{K} \sum_{i \neq j} |\theta_{ij}^{(k)}| + \lambda_2 \sum_{k<k'} \sum_{i,j} |\theta_{ij}^{(k)} - \theta_{ij}^{(k')}|, \tag{2.2.5}$$

where λ_1 and λ_2 are nonnegative tuning parameters. This is a *generalized fused lasso* penalty [Hoefling, 2010b], and results from applying ℓ_1 penalties to (1) each off-diagonal element of the K precision matrices, and (2) differences between corresponding elements of each pair of precision matrices. Like the graphical lasso, FGL results in sparse estimates $\hat{\mathbf{\Theta}}^{(1)}, \ldots, \hat{\mathbf{\Theta}}^{(K)}$ when the tuning parameter λ_1 is large. In addition, many elements of $\hat{\mathbf{\Theta}}^{(1)}, \ldots, \hat{\mathbf{\Theta}}^{(K)}$ will be identical across classes when the

tuning parameter λ_2 is large [Tibshirani et al., 2005]. Thus FGL borrows information aggressively across classes, encouraging not only similar network structure but also similar edge values.

The group graphical lasso

We define the *group graphical lasso* (GGL) to be the solution to (2.2.4) with

$$P(\{\mathbf{\Theta}\}) = \lambda_1 \sum_{k=1}^{K} \sum_{i \neq j} |\theta_{ij}^{(k)}| + \lambda_2 \sum_{i \neq j} \sqrt{\sum_{k=1}^{K} {\theta_{ij}^{(k)}}^2}. \tag{2.2.6}$$

Again, λ_1 and λ_2 are nonnegative tuning parameters. A lasso penalty is applied to the elements of the precision matrices and a *group lasso* penalty is applied to the (i, j) element across all K precision matrices [Yuan and Lin, 2007b]. This group lasso penalty encourages a similar pattern of sparsity across all of the precision matrices – that is, there will be a tendency for the zeros in the K estimated precision matrices to occur in the same places. Specifically, when $\lambda_1 = 0$ and $\lambda_2 > 0$, each $\hat{\mathbf{\Theta}}^{(k)}$ will have an identical pattern of non-zero elements. On the other hand, the lasso penalty encourages further sparsity within each $\hat{\mathbf{\Theta}}^{(k)}$.

GGL encourages a weaker form of similarity across the K precision matrices than does FGL: the latter encourages shared edge values across the K matrices, whereas the former encourages only a shared pattern of sparsity.

2.3 Algorithm for the joint graphical lasso problem

2.3.1 An ADMM algorithm

We solve the problem (2.2.4) using an *alternating directions method of multipliers* (ADMM) algorithm. We refer the reader to Boyd et al. [2010] for a thorough exposition of ADMM algorithms as well as their convergence properties, and to Simon and

Tibshirani [2011] and Mohan et al. [2012] for recent applications of ADMM to related problems.

To solve the problem (2.2.4) subject to the constraint that $\mathbf{\Theta}^{(k)}$ is positive definite for $k = 1, \ldots, K$ using ADMM, we note that the problem can be rewritten as

$$\underset{\{\mathbf{\Theta}\},\{\mathbf{Z}\}}{\text{minimize}} \left\{ -\sum_{k=1}^{K} n_k \left(\log\det \mathbf{\Theta}^{(k)} - \text{trace}(\mathbf{S}^{(k)}\mathbf{\Theta}^{(k)}) \right) + P(\{\mathbf{Z}\}) \right\}, \tag{2.3.7}$$

subject to the positive-definiteness constraint as well as the constraint that $\mathbf{Z}^{(k)} = \mathbf{\Theta}^{(k)}$ for $k = 1, \ldots, K$, where $\{\mathbf{Z}\} = \{\mathbf{Z}^{(1)}, \ldots, \mathbf{Z}^{(K)}\}$. The scaled augmented Lagrangian [Boyd et al., 2010] for this problem is given by

$$\begin{aligned} L_\rho(\{\mathbf{\Theta}\}, \{\mathbf{Z}\}, \{\mathbf{U}\}) = \; & - \sum_{k=1}^{K} n_k \left(\log\det \mathbf{\Theta}^{(k)} - \text{trace}(\mathbf{S}^{(k)}\mathbf{\Theta}^{(k)}) \right) + P(\{\mathbf{Z}\}) \\ & + \frac{\rho}{2} \sum_{k=1}^{K} ||\mathbf{\Theta}^{(k)} - \mathbf{Z}^{(k)} + \mathbf{U}^{(k)}||_F^2, \end{aligned} \tag{2.3.8}$$

where $\{\mathbf{U}\} = \{\mathbf{U}^{(1)}, \ldots, \mathbf{U}^{(K)}\}$ are dual variables. Roughly speaking, an ADMM algorithm corresponding to (2.3.8) results from iterating three simple steps. At the ith iteration, they are as follows:

1. $\{\mathbf{\Theta}_{(i)}\} \leftarrow \arg\min_{\{\mathbf{\Theta}\}} \left\{ L_\rho \left(\{\mathbf{\Theta}\}, \{\mathbf{Z}_{(i-1)}\}, \{\mathbf{U}_{(i-1)}\} \right) \right\}$.

2. $\{\mathbf{Z}_{(i)}\} \leftarrow \arg\min_{\{\mathbf{Z}\}} \left\{ L_\rho \left(\{\mathbf{\Theta}_{(i)}\}, \{\mathbf{Z}\}, \{\mathbf{U}_{(i-1)}\} \right) \right\}$.

3. $\{\mathbf{U}_{(i)}\} \leftarrow \{\mathbf{U}_{(i-1)}\} + (\{\mathbf{\Theta}_{(i)}\} - \{\mathbf{Z}_{(i)}\})$.

We now present the ADMM algorithm in greater detail.

ADMM algorithm for solving the joint graphical lasso problem

1. Initialize the variables: $\mathbf{\Theta}^{(k)} = \mathbf{I}$, $\mathbf{U}^{(k)} = \mathbf{0}$, $\mathbf{Z}^{(k)} = \mathbf{0}$ for $k = 1, \ldots, K$.

2. Select a scalar $\rho > 0$.

3. For $i = 1, 2, 3, \ldots$ until convergence:

   (a) For $k = 1, \ldots, K$, update $\mathbf{\Theta}_{(i)}^{(k)}$ as the minimizer (with respect to $\mathbf{\Theta}^{(k)}$) of

   $$-n_k \left(\log\det \mathbf{\Theta}^{(k)} - \text{trace}(\mathbf{S}^{(k)}\mathbf{\Theta}^{(k)})\right) + \frac{\rho}{2}||\mathbf{\Theta}^{(k)} - \mathbf{Z}_{(i-1)}^{(k)} + \mathbf{U}_{(i-1)}^{(k)}||_F^2.$$

   Letting $\mathbf{V}\mathbf{D}\mathbf{V}^T$ denote the eigendecomposition of $\mathbf{S}^{(k)} - \rho\mathbf{Z}_{(i-1)}^{(k)}/n_k + \rho\mathbf{U}_{(i-1)}^{(k)}/n_k$, the solution is given [Witten and Tibshirani, 2009] by $\mathbf{V}\tilde{\mathbf{D}}\mathbf{V}^T$, where $\tilde{\mathbf{D}}$ is the diagonal matrix with jth diagonal element

   $$\frac{n_k}{2\rho}\left(-D_{jj} + \sqrt{D_{jj}^2 + 4\rho/n_k}\right).$$

   (b) Update $\{\mathbf{Z}_{(i)}\}$ as the minimizer (with respect to $\{\mathbf{Z}\}$) of

   $$\frac{\rho}{2}\sum_{k=1}^{K}||\mathbf{Z}^{(k)} - (\mathbf{\Theta}_{(i)}^{(k)} + \mathbf{U}_{(i-1)}^{(k)})||_F^2 + P(\{\mathbf{Z}\}). \tag{2.3.9}$$

   (c) For $k = 1, \ldots, K$, update $\mathbf{U}_{(i)}^{(k)}$ as $\mathbf{U}_{(i-1)}^{(k)} + (\mathbf{\Theta}_{(i)}^{(k)} - \mathbf{Z}_{(i)}^{(k)})$.

The final $\hat{\mathbf{\Theta}}^{(1)}, \ldots, \hat{\mathbf{\Theta}}^{(K)}$ that result from this algorithm are the JGL estimates of $\mathbf{\Sigma}_1^{-1}, \ldots, \mathbf{\Sigma}_K^{-1}$. This algorithm is guaranteed to converge to the global optimum [Boyd et al., 2010]. We note that the positive-definiteness constraint on the estimated precision matrices is naturally enforced by the update in Step (c)(i).

Details of the minimization of (2.3.9) will depend on the form of the convex penalty function P. We note that the task of minimizing (2.3.9) can be re-written as

$$\underset{\{\mathbf{Z}\}}{\text{minimize}}\left\{\frac{\rho}{2}\sum_{k=1}^{K}||\mathbf{Z}^{(k)} - \mathbf{A}^{(k)}||_F^2 + P(\{\mathbf{Z}\})\right\}, \tag{2.3.10}$$

where

$$\mathbf{A}^{(k)} = \mathbf{\Theta}_{(i)}^{(k)} + \mathbf{U}_{(i-1)}^{(k)}. \tag{2.3.11}$$

We will see in Section 2.3.2 that for the FGL and GGL penalties, solving (2.3.10) is a simple task, regardless of the value of K.

The algorithm given above involves computing the eigendecomposition of a $p \times p$ matrix, which can be computationally demanding when p is large. However, in Section 2.4, we will present two theorems that reveal that when the values of the tuning parameters λ_1 and λ_2 are large, one can obtain the *exact* solution to the JGL optimization problem without ever computing the eigen decomposition of a $p \times p$ matrix. Therefore, solving the JGL problem is fast even when p is quite large. In Section 2.8, we will see that one can perform FGL with $K = 2$ classes and almost 18,000 features in under 2 minutes.

2.3.2 Solving (2.3.10) for the joint graphical lasso

We now consider the problem of solving (2.3.10) if P is a generalized fused lasso or group lasso penalty.

Solving (2.3.10) for FGL

If P is the penalty given in (2.2.5), then (2.3.10) takes the form

$$\underset{\{\mathbf{Z}\}}{\text{minimize}} \left\{ \frac{\rho}{2} \sum_{k=1}^{K} ||\mathbf{Z}^{(k)} - \mathbf{A}^{(k)}||_F^2 + \lambda_1 \sum_{k=1}^{K} \sum_{i \neq j} |Z_{ij}^{(k)}| + \lambda_2 \sum_{k<k'} \sum_{i,j} |Z_{ij}^{(k)} - Z_{ij}^{(k')}| \right\}. \tag{2.3.12}$$

Now (2.3.12) is completely separable with respect to each pair of matrix elements (i, j): that is, one can simply solve, for each (i, j),

$$\underset{Z_{ij}^{(1)},\ldots,Z_{ij}^{(K)}}{\text{minimize}} \left\{ \frac{\rho}{2} \sum_{k=1}^{K} (Z_{ij}^{(k)} - A_{ij}^{(k)})^2 + \lambda_1 1_{i \neq j} \sum_{k=1}^{K} |Z_{ij}^{(k)}| + \lambda_2 \sum_{k<k'} |Z_{ij}^{(k)} - Z_{ij}^{(k')}| \right\}. \quad (2.3.13)$$

This is a special case of the *fused lasso signal approximator* [Hoefling, 2010b] in which there is a fusion between each pair of variables. A very efficient algorithm for this special case, which can be performed in $O(K \log K)$ operations, is available [Hocking et al., 2011, Hoefling, 2010a, Tibshirani, 2012].

In fact, when $K = 2$, (2.3.13) has a very simple closed form solution. When $\lambda_1 = 0$, it is easy to verify that the solution to (2.3.13) takes the form

$$(\hat{Z}_{ij}^{(1)}, \hat{Z}_{ij}^{(2)}) = \begin{cases} (A_{ij}^{(1)} - \lambda_2/\rho, A_{ij}^{(2)} + \lambda_2/\rho) & \text{if } A_{ij}^{(1)} > A_{ij}^{(2)} + 2\lambda_2/\rho \\ (A_{ij}^{(1)} + \lambda_2/\rho, A_{ij}^{(2)} - \lambda_2/\rho) & \text{if } A_{ij}^{(2)} > A_{ij}^{(1)} + 2\lambda_2/\rho \\ (\frac{A_{ij}^{(1)} + A_{ij}^{(2)}}{2}, \frac{A_{ij}^{(1)} + A_{ij}^{(2)}}{2}) & \text{if } |A_{ij}^{(1)} - A_{ij}^{(2)}| \leq 2\lambda_2/\rho \end{cases}. \quad (2.3.14)$$

And when $\lambda_1 > 0$, the solution to (2.3.13) can be obtained through soft-thresholding (2.3.14) by λ_1/ρ [see Friedman et al., 2007a]. Here the soft-thresholding operator is defined as $S(x, c) = \text{sgn}(x)(|x| - c)_+$, where $a_+ = \max(a, 0)$.

Solving (2.3.10) for GGL

If P is the group lasso penalty (2.2.6), then (2.3.10) takes the form

$$\underset{\{\mathbf{Z}\}}{\text{minimize}} \left\{ \frac{\rho}{2} \sum_{k=1}^{K} ||\mathbf{Z}^{(k)} - \mathbf{A}^{(k)}||_F^2 + \lambda_1 \sum_{k=1}^{K} \sum_{i \neq j} |Z_{ij}^{(k)}| + \lambda_2 \sum_{i \neq j} \sqrt{\sum_k {Z_{ij}^{(k)}}^2} \right\}. \quad (2.3.15)$$

First, for all $i = 1, \ldots, p$ and $k = 1, \ldots, K$, it is easy to see that the solution to (2.3.15) has $\hat{Z}_{ii}^{(k)} = A_{ii}^{(k)}$. And one can show that the off-diagonal elements take the

form [Friedman et al., 2010]

$$\hat{Z}_{ij}^{(k)} = S(A_{ij}^{(k)}, \lambda_1/\rho)\left(1 - \frac{\lambda_2}{\rho\sqrt{\sum_{k=1}^{K} S(A_{ij}^{(k)}, \lambda_1/\rho)^2}}\right)_+, \tag{2.3.16}$$

where S denotes the soft-thresholding operator.

2.4 Faster computations for FGL and GGL

We now present two theorems that lead to substantial computational improvements to the JGL algorithm presented in Section 2.3. Using these theorems, one can inspect the empirical covariance matrices $\mathbf{S}^{(1)}, \ldots, \mathbf{S}^{(K)}$ in order to determine whether the solution to the JGL optimization problem is block diagonal after some permutation of the features. Then one can simply perform the JGL algorithm on the features within each block separately, in order to obtain *exactly* the same solution that would have been obtained by applying the algorithm to all p features. This leads to huge speed improvements since it obviates the need to ever compute the eigen decomposition of a $p \times p$ matrix. Our results mirror recent improvements in algorithms for solving the graphical lasso problem [Witten et al., 2011, Mazumder and Hastie, 2012].

For instance, suppose that for a given choice of λ_1 and λ_2, we determine that the estimated inverse covariance matrices $\hat{\mathbf{\Theta}}^{(1)}, \ldots, \hat{\mathbf{\Theta}}^{(K)}$ are block diagonal, each with the same R blocks, the rth of which contains p_r features, $\sum_{r=1}^{R} p_r = p$. Then in each iteration of the JGL algorithm, rather than having to compute the eigen decomposition of K $p \times p$ matrices, we need only compute eigen decompositions of matrices of dimension $p_1 \times p_1, \ldots, p_R \times p_R$. This leads to a potentially massive reduction in computational complexity from $O(p^3)$ to $\sum_{r=1}^{R} O(p_r^3)$.

We begin with a very simple lemma for which the proof follows by inspection of (2.2.4). The lemma can be extended by induction to any number of blocks.

Lemma 2.4.1. *Suppose that the solution to the FGL or GGL optimization problem is block diagonal with known blocks. That is, the features can be reordered in such a way that each estimated inverse covariance matrix takes the form*

$$\hat{\mathbf{\Theta}}^{(k)} = \begin{pmatrix} \hat{\mathbf{\Theta}}_1^{(k)} & 0 \\ 0 & \hat{\mathbf{\Theta}}_2^{(k)} \end{pmatrix} \tag{2.4.17}$$

where each of $\hat{\mathbf{\Theta}}_1^{(1)}, \ldots, \hat{\mathbf{\Theta}}_1^{(K)}$ *has the same dimension. Then,* $\hat{\mathbf{\Theta}}_1^{(1)}, \ldots, \hat{\mathbf{\Theta}}_1^{(K)}$ *and* $\hat{\mathbf{\Theta}}_2^{(1)}, \ldots, \hat{\mathbf{\Theta}}_2^{(K)}$ *can be obtained by solving the FGL or GGL optimization problem on just the corresponding set of features.*

We now present the key results. Theorems 1 and 2 outline necessary and sufficient conditions for the presence of block diagonal structure in the FGL and GGL optimization problems, and are proven in Appendix 2.

Theorem 1. *Consider the FGL optimization problem with* $K = 2$ *classes. Let* C_1 *and* C_2 *be a non-overlapping partition of the* p *variables such that* $C_1 \cap C_2 = \emptyset$, $C_1 \cup C_2 = \{1, \ldots, p\}$. *The following conditions are necessary and sufficient for the variables in* C_1 *to be completely disconnected from those in* C_2 *in each of the resulting network estimates:*

1. $|n_1 S_{ij}^{(1)}| \leq \lambda_1 + \lambda_2$ *for all* $i \in C_1$ *and* $j \in C_2$,
2. $|n_2 S_{ij}^{(2)}| \leq \lambda_1 + \lambda_2$ *for all* $i \in C_1$ *and* $j \in C_2$, *and*
3. $|n_1 S_{ij}^{(1)} + n_2 S_{ij}^{(2)}| \leq 2\lambda_1$ *for all* $i \in C_1$ *and* $j \in C_2$.

Furthermore, if $K > 2$, *then*

$$|n_k S_{ij}^{(k)}| \leq \lambda_1 \text{ for all } i \in C_1,\ j \in C_2,\ k = 1, \ldots, K \tag{2.4.18}$$

is a sufficient condition for the variables in C_1 to be completely disconnected from those in C_2.

Theorem 2. *Consider the GGL optimization problem with $K \geq 2$ classes. Let C_1 and C_2 be a non-overlapping partition of the p variables, such that $C_1 \cap C_2 = \emptyset$, $C_1 \cup C_2 = \{1, \ldots, p\}$. The following condition is necessary and sufficient for the variables in C_1 to be completely disconnected from those in C_2 in each of the resulting network estimates:*

$$\sum_{k=1}^{K} (|n_k S_{ij}^{(k)}| - \lambda_1)_+^2 \leq \lambda_2^2 \text{ for all } i \in C_1,\ j \in C_2\ . \tag{2.4.19}$$

Theorems 1 and 2 allow us to quickly check if, given a partition of the features C_1 and C_2, the solution to the JGL optimization problem is block diagonal with one block corresponding to features in C_1 and one block corresponding to features in C_2. In practice, for any given (λ_1, λ_2), we can quickly perform the following two-step procedure to identify any block structure in the FGL or GGL solution:

1. Create $\mathbf{M}$, a $p \times p$ matrix with $M_{ii} = 1$ for $i = 1, \ldots, p$. For $i \neq j$, let $M_{ij} = 0$ if the conditions specified in Theorem 1 are met for that pair of variables and the FGL penalty is used, or if the condition of Theorem 2 is met for that pair of variables and the GGL penalty is used. Otherwise, set $M_{ij} = 1$.

2. Identify the connected components of the undirected graph whose adjacency matrix is given by $\mathbf{M}$. Note that this can be performed in $O(|M|)$ operations, where $|M|$ is the number of nonzero elements in $\mathbf{M}$ [Tarjan, 1972].

Theorems 1 and 2 guarantee that the connected components identified in the second step correspond to distinct blocks in the FGL or GGL solutions. Therefore, one can quickly obtain these solutions by solving the FGL or GGL optimization problems on

the submatrices of these K $p \times p$ empirical covariance matrices that correspond to that block diagonal structure. Consequently, we can obtain the *exact* solution to the JGL optimization problem on extremely high-dimensional data sets that would otherwise be computationally intractable. For instance, in Section 2.8 we performed FGL on a gene expression data set with almost $18,000$ features in under two minutes.

Theorems 1 and 2 lead to speed improvements only if the tuning parameters λ_1 and λ_2 are sufficiently large. We argue that this will in fact be the case in most practical applications of JGL. When network estimation is performed for the sake of data exploration and when p is large, only a very sparse network estimate will be useful; otherwise, interpretation of the estimate will be impossible. Even when data exploration is not the end goal of the analysis, large values of λ_1 and λ_2 will generally be used, since most data sets cannot reasonably support estimation of $Kp(p+1)/2$ nonzero parameters when $n \ll p$.

2.5 Relationship to previous proposals

Several past proposals have been made to jointly estimate graphical models on the basis of observations drawn from distinct conditions. Some proposals have used time-series data to define time-varying networks in the context of continuous or binary data [Zhou et al., 2008, Song et al., 2009a, Ahmed and Xing, 2009, Kolar and Xing, 2009, Song et al., 2009b, Kolar et al., 2010]. Guo et al. [2011] instead describe a likelihood-based method for estimating precision matrices across multiple related classes simultaneously. They employ a hierarchical penalty that forces similar patterns of sparsity across classes, an approach that is similar in spirit to GGL.

Our FGL and GGL proposals have a number of advantages over these existing approaches. Methods for estimating time-varying networks cannot be easily extended to the setting where the classes lack a natural ordering. Guo et al. [2011]'s proposal

is a closer precursor to our method, and can in fact be stated as an instance of the problem (2.2.4) with a *hierarchical group lasso penalty*

$$P(\{\mathbf{\Theta}\}) = \lambda \sum_{i \neq j} \sqrt{\sum_k |\theta_{ij}^{(k)}|} \tag{2.5.20}$$

that encourages a shared pattern of sparsity across the K classes. But the approach of Guo et al. [2011] has a number of disadvantages relative to FGL and GGL. (1) The penalty (2.5.20) is not convex, so convergence to the global optimum is not guaranteed. (2) Because (2.5.20) is not convex, it is not possible to achieve the speed improvements described in Section 2.4. Consequently, the Guo et al. [2011] proposal is quite slow relative to our approach, as seen in Figures 2.2(e), C.1(e), and C.2(e), and essentially cannot be applied to very high-dimensional data sets. (3) Unlike FGL and GGL, it uses just one tuning parameter, and is unable to control separately the sparsity level and the extent of network similarity. (4) In cases where we expect edge values as well as network structure to be similar between classes, FGL is much better suited than GGL and Guo et al. [2011]'s proposal, both of which encourage shared patterns of sparsity but do not encourage similarity in the signs and values of the nonzero edges.

Guo et al. [2011]'s proposal is included in the simulation study in Section 2.7.

2.6 Tuning parameter selection

One can select tuning parameters for JGL using an approximation of the Akaike Information Criterion (AIC),

$$AIC(\lambda_1, \lambda_2) = \sum_{k=1}^{K} \left[n_k \text{trace}(\mathbf{S}^{(k)} \hat{\mathbf{\Theta}}^{(k)}_{\lambda_1,\lambda_2}) - n_k \log\det \hat{\mathbf{\Theta}}^{(k)}_{\lambda_1,\lambda_2} + 2E_k \right], \tag{2.6.21}$$

where $\{\hat{\mathbf{\Theta}}^{(k)}_{\lambda_1,\lambda_2}\}$ is the set of estimated inverse covariance matrices based on tuning parameters λ_1 and λ_2, and E_k is the number of non-zero elements in $\hat{\mathbf{\Theta}}^{(k)}_{\lambda_1,\lambda_2}$. A grid search can then be performed to select λ_1 and λ_2 minimizing the $AIC(\lambda_1, \lambda_2)$ score. The simulation study in Section 7 suggests that this criterion tends to select models whose Kullback-Leibler (dKL) divergence from the true model is low. When the number of variables p is very large, computing $AIC(\lambda_1, \lambda_2)$ over a range of values for λ_1 and λ_2 may prove computationally onerous. If this is the case, we suggest a dense search over λ_1 while holding λ_2 at a fixed, low value, followed by a quick search over λ_2, holding λ_1 at its optimal value.

It is worth noting that in most cases, network estimation is performed as a part of exploratory data analysis and hypothesis generation. For these purposes, approaches such as AIC, BIC, and cross-validation may tend to choose models too large to be useful. In this setting, model selection should be guided by practical considerations, such as network interpretability, stability, and the desire for an edge set with a low false discovery rate [Meinshausen and Buhlmann, 2010, Li et al., 2011].

2.7 Simulation study

We compare the performances of FGL and GGL to two existing methods, graphical lasso and Guo et al. [2011]'s proposal, in Section 2.7.1. When applying the graphical lasso, networks are fitted for each class separately. We investigate the effects of n and p on FGL and GGL's performances in Section 2.7.2. Additional simulation results are presented in Appendix C.

The effects of the FGL and GGL penalties vary with the sample size. For ease of presentation of the simulation study results, we multiply the reported tuning parameters λ_1 and λ_2 by the sample size of each class before performing JGL.

To ease interpretation, we reparametrize the GGL penalties in our simulation

study. The motivation is to summarize the regularization for "sparsity" and for "similarity" separately. In FGL, this is nicely achieved by just using λ_1 and λ_2, as the former drives network sparsity and the latter drives network similarity. In contrast, in GGL, both tuning parameters contribute to sparsity: λ_1 drives individual network edges to zero whereas λ_2 simultaneously drives network edges to zero across all K network estimates. We reparameterize our simulation results for GGL in terms of $\omega_1 = \lambda_1 + \frac{1}{\sqrt{2}}\lambda_2$ and $\omega_2 = \frac{1}{\sqrt{2}}\lambda_2/(\lambda_1 + \frac{1}{\sqrt{2}}\lambda_2)$, which we found to reasonably reflect the levels of "sparsity" and "similarity" regularization, respectively.

2.7.1 Performance as a function of tuning parameters

Simulation set-up

In this simulation, we consider a three-class problem. We first generate three networks with $p = 500$ features belonging to ten equally sized unconnected subnetworks, each with a power law degree distribution. Power law degree distributions are thought to mimic the structure of biological networks [Chen and Sharp, 2004] and are generally harder to estimate than simpler structures [Peng et al., 2009]. Of the ten subnetworks, eight have the same structure and edge values in all three classes, one is identical between the first two classes and missing in the third (i.e. the corresponding features are singletons in the third network), and one is present in only the first class. The topology of the networks generated is shown in Figure D.1 in Appendix 4.

Given a network structure, we generate a covariance matrix for the first class as follows [Peng et al., 2009]. We create a $p \times p$ matrix with ones on the diagonal, zeroes on elements not corresponding to network edges, and values from a uniform distribution with support on $\{[-.4, -.1] \cup [.1, .4]\}$ on elements corresponding to edges. To ensure positive definiteness, we divide each off-diagonal element by 1.5 times the sum of the absolute values of off-diagonal elements in its row. Finally, we average

the matrix with its transpose, achieving a symmetric, positive-definite matrix $\mathbf{A}$. We then create the (i, j) element of $\mathbf{\Sigma}_1$ as

$$d_{ij}(\mathbf{A}^{-1})_{ij}/\sqrt{(\mathbf{A}^{-1})_{ii}(\mathbf{A}^{-1})_{jj}},$$

where $d_{ij} = 0.6$ if $i \neq j$ and $d_{ij} = 1$ if $i = j$. We create $\mathbf{\Sigma}_2$ equal to $\mathbf{\Sigma}_1$, then reset one of its ten subnetwork blocks to the identity. We create $\mathbf{\Sigma}_3$ equal to $\mathbf{\Sigma}_2$, and reset an additional subnetwork block to the identity. Finally, for each class we generate independent, identically distributed samples from a $N(\mathbf{0}, \mathbf{\Sigma}_k)$ distribution.

We present two additional simulations studies involving two-class datasets in Appendix 3. The first additional simulation uses the same network structure described above, and the second uses a single power law network with no block structure.

Simulation results

Our first set of simulations illustrates the effect of varying tuning parameters on the performances of FGL and GGL. We generated 100 three-class data sets with $p = 500$ features and $n = 150$ observations per class, as described in Section 2.7.1. Class 1's network had 490 edges, class 2's network is missing 49 of those edges, and class 3's network is missing an additional 49 edges. Figure 2.2 shows the results, averaged over the 100 data sets. In each plot, the lines for FGL and for GGL indicate the results obtained with a single value of the similarity tuning parameters λ_2 and ω_2. The graphical lasso and the proposal of Guo et al. [2011] are included in the comparisons.

Figure 2.2(a) displays the number of true edges selected against the number of false edges selected. We say the edge represented by $\theta_{ij}^{(k)}$ is selected if $\hat{\theta}_{ij}^{(k)} \neq 0$, and we say $\hat{\theta}_{ij}^{(k)}$ identifies a true edge if $\hat{\theta}_{ij}^{(k)} \neq 0$ and a false edge if $\hat{\theta}_{ij}^{(k)} = 0$. As the sparsity tuning parameters λ_1 and ω_1 decrease, the number of edges selected increases. At many values of the similarity tuning parameter λ_2, FGL dominates the

other methods. At some choices of the similarity tuning parameter ω_2, GGL performs as well as Guo et al. [2011]. FGL, GGL, and Guo et al. [2011]'s proposal dominate the graphical lasso.

Figure 2.2(b) displays the sum of squared errors (SSE) between estimated edge values and true edge values: $\sum_{k=1}^{K}\sum_{i\neq j}(\hat{\theta}_{ij}^{(k)} - (\mathbf{\Sigma}_k^{-1})_{ij})^2$. Unlike the proposal of Guo et al. [2011], FGL, GGL, and the graphical lasso tend to overshrink edge values towards zero due to the use of convex penalty functions. Thus, while FGL and GGL attain SSE values that are as low as those of Guo et al. [2011], they do so when estimating much larger networks. When simultaneous edge selection and estimation are desired, it may be useful to run FGL or GGL once and then to re-run them with smaller penalties on the selected edges, as in Meinshausen [2007].

Figure 2.2(c) evaluates each method's success in detecting *differential edges*, or edges that differ between classes. We say the pair of edges $(\theta_{ij}^{(k)}, \theta_{ij}^{(k')})$ is estimated to be differential if if $\hat{\theta}_{ij}^{(k)} \neq \hat{\theta}_{ij}^{(k')}$, and we say it is truly differential if $\theta_{ij}^{(k)} \neq \theta_{ij}^{(k')}$ and falsely differential if $\theta_{ij}^{(k)} = \theta_{ij}^{(k')}$. For FGL, the number of differential edges is computed as the number of pairs $k < k', i < j$ such that $\hat{\theta}_{ij}^{(k)} \neq \hat{\theta}_{ij}^{(k')}$. Since GGL, the proposal of Guo et al. [2011], and the graphical lasso cannot yield edges that are exactly identical across classes, for those approaches the number of differential edges is computed as the number of pairs $k < k', i < j$ such that $|\hat{\theta}_{ij}^{(k)} - \hat{\theta}_{ij}^{(k')}| > 10^{-2}$. The number of true positive differential edges is plotted against the number of false positive differential edges. Note that by controlling the total number of non-zero edges, the sparsity tuning parameters λ_1 and ω_1 have a large effect on the number of edges that are estimated to differ between the two networks. FGL yields fewer false positives than the competing methods, since it shrinks between-class differences in edge values to zero. Since neither GGL nor Guo et al. [2011]'s method are designed to shrink edge values towards each other, by this measure neither method outperforms

even the graphical lasso.

Figure 2.2(d) displays the sum of the dKL's of the estimated distributions from the true distributions, as a function of the ℓ_1 norm of the off-diagonal elements of the estimated precision matrices, i.e. $\sum_k \sum_{i \neq j} |\hat{\theta}_{ij}^{(k)}|$. The dKL from the multivariate normal model with inverse covariance estimates $\hat{\mathbf{\Theta}}^{(1)}, \ldots, \hat{\mathbf{\Theta}}^{(K)}$ to the multivariate normal model with the true precision matrices $\mathbf{\Sigma}_1^{-1}, \ldots, \mathbf{\Sigma}_K^{-1}$ is

$$\frac{1}{2} \sum_{k=1}^{K} \Big(- \log \det(\hat{\mathbf{\Theta}}^{(k)} \mathbf{\Sigma}_k) + \mathrm{trace}(\hat{\mathbf{\Theta}}^{(k)} \mathbf{\Sigma}_k) \Big) .$$

At most values of λ_2, FGL attains a lower dKL than the other methods, followed by Guo et al. [2011]'s method, then by GGL. The graphical lasso has the worst performance, since it estimates each network separately.

Figure 2.2(e) compares the methods' running times. Computation time (in seconds) is plotted against the total number of non-zero edges estimated. The graphical lasso is fastest, but FGL and GGL are much faster than the proposal of Guo et al. [2011], due to the results from Section 2.4. Timing comparisons were performed on an Intel Xeon x5680 3.3 GHz processor. It is worth mentioning that the FGL algorithm is much faster in problems with only two classes, since in that case there is a closed-form solution to the generalized fused lasso problem (Section 2.3.2). This can be seen in Figures C.1(e) and C.2(e) in Appendix 3.

We examined the FGL and GGL models with tuning parameters selected as described in Section 2.6. For FGL, AIC selected the tuning parameters $\lambda_1 = 0.175$, $\lambda_2 = 0.025$. Over the 100 replicate datasets, the FGL models with these tuning parameters averaged a dKL of 774, 884 true positive edges, 2406 false positive edges, 77 true positive differential edges, and 4977 false positive differential edges. In GGL, AIC selected the tuning parameters $\omega_1 = 0.225$, $\omega_2 = 1$. Over the 100 replicate

datasets, the GGL models with these tuning parameters averaged a dKL of 776, 898 true positive edges, 736 false positive edges, 53 true positive differential edges, and 1456 false positive differential edges.

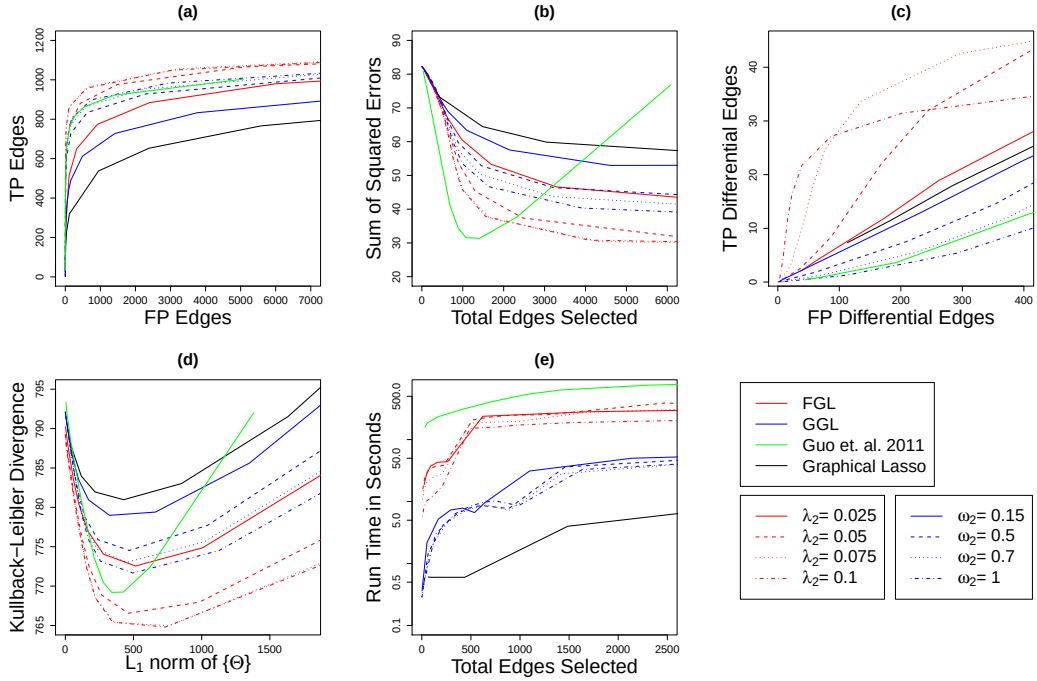

Figure 2.2: *Performance of FGL, GGL, Guo et al. [2011]'s method, and the graphical lasso on simulated data with* 150 *observations in each of 3 classes, and* 500 *features. Black lines display models derived using the graphical lasso, green lines display the proposal of Guo et al. [2011], red lines display FGL, and blue lines display GGL.* **(a)***: The number of edges correctly identified to be nonzero (TP Edges) is plotted against the number of edges incorrectly identified to be nonzero (FP edges).* **(b)***: The sum of squared errors in edge values is plotted against the total number of edges estimated to be nonzero.* **(c)***: The number of edges correctly found to have values differing between classes (TP Differential Edges) is plotted against the number of edges incorrectly found to have values differing between classes (FP Differential Edges).* **(d)***: The dKL of the estimated models from the true models is plotted against the ℓ_1 norm of the off-diagonal entries of the estimated precision matrices.* **(e)***: Running time (in seconds) is plotted against the number of non-zero edges estimated. Note the use of a log scale on the y-axis.*

2.7.2 Performance as a function of n and p

We now evaluate the effect of sample size n and dimension p on the performances of FGL and GGL.

Simulation set-up

We generate a pair of networks with $p = 500$ much as described in Section 2.7.1, but with $K = 2$ instead of $K = 3$. The first network has 10 equal-sized components with power law degree distributions, and the second network is identical to the first in both edge identity and value, but with two components removed.

In addition to the 500-feature network pair, we generate a pair of networks with $p = 1000$ features, each of which is block diagonal with 500×500 blocks corresponding to two copies of the 500-feature networks just described. We generate covariance matrices from the networks exactly as described in Section 2.7.1.

Simulation results

For both the $p = 500$ and the $p = 1000$ networks, we simulate 100 datasets with $n = 50$, $n = 200$, and $n = 500$ samples in each class. We run FGL with $\lambda_1 = 0.2, \lambda_2 = 0.1$ and GGL with $\lambda_1 = 0.05, \lambda_2 = 0.25$. We record in Table 2.1 dKL as well as the sensitivity and false discovery rate associated with detecting non-zero edges and detecting differential edges. In this simulation setting, accuracy of covariance estimation (as measured by dKL) improves significantly from $n = 50$ to $n = 200$, and improves only marginally with a further increase to $n = 500$. Detection of edges improves throughout the range of n's sampled: for both FGL and GGL, sensitivity improves slightly with increased sample size, and FDR decreases dramatically. Detection of edge differences is a more difficult problem, for which FGL performs well.

Table 2.1: *Performance of FGL and GGL as a function of n and p. Means over 100 replicates are shown for dKL, and for sensitivity (Sens.) and false discovery rate (FDR) of detection of edges (DE) and differential edge detection (DED).*

	p	n	dKL	DE Sens.	DE FDR	DED Sens.	DED FDR
FGL	500	50	545.1	0.502	0.966	0.262	0.996
		200	517.5	0.570	0.053	0.228	0.485
		500	516.6	0.590	0.001	0.192	0.036
	1000	50	1119.3	0.600	0.970	0.245	0.998
		200	1035.0	0.666	0.063	0.223	0.557
		500	1033.3	0.681	0.000	0.194	0.025
GGL	500	50	549.8	0.490	0.973	0.337	0.996
		200	520.8	0.505	0.060	0.244	0.903
		500	519.7	0.524	0.010	0.194	0.921
	1000	50	1127.9	0.587	0.976	0.316	0.998
		200	1041.7	0.615	0.061	0.239	0.908
		500	1039.4	0.629	0.007	0.197	0.920

2.8 Analysis of lung cancer microarray data

We applied FGL to a dataset containing $22,283$ microarray-derived gene expression measurements from large airway epithelial cells sampled from 97 smokers with lung cancer and 90 smokers without cancer [Spira et al., 2007]. The data are publicly available from the Gene Expression Omnibus [Barrett et al., 2005] at accession number GDS2771. We omitted genes with standard deviations in the bottom 20% since a greater share of their variance is likely attributable to non-biological noise. The remaining genes were normalized to have mean zero and standard deviation one within each class. To avoid disparate levels of sparsity between the classes and to prevent the larger class from dominating the estimated networks, we weighted each class equally instead of by sample size in (2.2.4). Since our goal was data visualization and hypothesis generation, we chose a high value for the sparsity tuning parameter, $\lambda_1 = 0.95$, to yield very sparse network estimates. We ran FGL with a range of

λ_2 values in order to identify the edges that differed most strongly, and settled on $\lambda_2 = 0.005$ as providing the most interpretable results. Application of Theorem 1 revealed that only 278 genes were connected to any other gene using the chosen tuning parameters. Identification of block diagonal structure using Theorem 1 and application of the FGL algorithm took less than two minutes. (Note that this data set is so large that it would be computationally prohibitive to apply the proposal of Guo et al. [2011]!) FGL estimated 134 edges shared between the two networks, 202 edges present only in the cancer network, and 18 edges present only in the normal tissue network. The results are displayed in Figure 2.3.

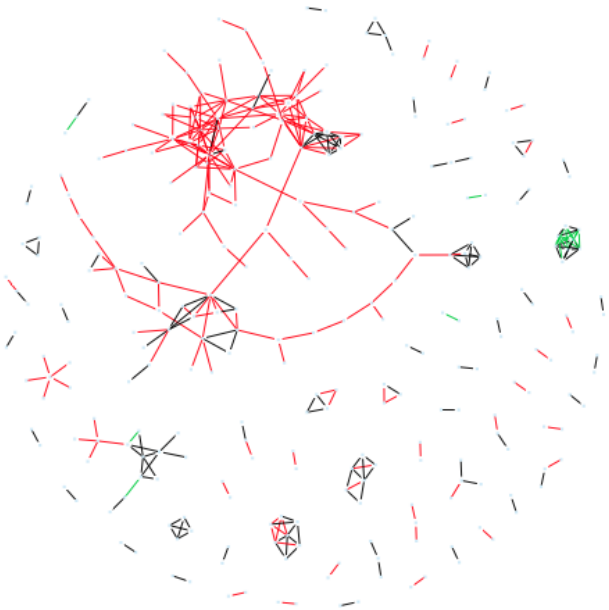

Figure 2.3: *Conditional dependency networks inferred from 17,772 genes in normal and cancerous lung cells. 278 genes have nonzero edges in at least one of the two networks. Black lines denote edges common to both classes. Red and green lines denote tumor-specific and normal-specific edges, respectively.*

The estimated networks contain many two-gene subnetworks common to both

classes, a few small subnetworks, and one large subnetwork specific to tumor cells. Reassuringly, 45% of edges, including almost all of the two-gene subnetworks, connect multiple probes for the same gene. Many other edges connect genes that are obviously related, involved in the same biological process, or even coding for components of the same enzyme. Examples include TUBA1B and TUBA1C, PABPC1 and PABPC3, HLA-B and HLA-G, and SERPINB3 and SERPINB4. Recovery of these pairs suggests that FGL (and other network analysis tools) can generate high-quality hypotheses about gene co-regulation and functional interactions. This increases our confidence that some of the non-obvious two-gene subnetworks detected in this analysis may merit further investigation. Examples include DAZAP2 and TCP1, PRKAR1A and CALM3, and BCLAF1 and SERPB1. A complete list of subnetworks detected is available in the Supplementary Materials.

The small black and green network in Figure 2.3 suggests an interesting phenomenon. It contains multiple probes for two hemoglobin genes, HBA2 and HBB. In the normal tissue network, the probes for these genes are heavily interconnected both within and between the genes. In the tumor cells, while edges between HBA2 probes and between HBB probes are preserved, no edges connect the two genes. The abundance of connections between the two genes in healthy cells and the absence of connections in tumor cells may indicate a possible direction of future investigation.

The most promising results of this analysis arise from the large subnetwork (104 nodes for 84 unique genes) unique to tumor cells. Many of the subnetwork's genes are involved in constructing ribosomes, including RPS8, RPS23, RPS24, RPS7p11, RPL3, RPL5, RPL10A, RPL14P1, RPL15, RPL17, RPL30 and RPL31. Other genes in the subnetwork further involve ribosome functioning: SRP14 and SRP9L1 are involved in recruiting proteins from ribosomes into the ER, and NACA inhibits the SRP pathway. Thus this subnetwork portrays a detailed web of relationships con-

sistent with known biology. More interestingly, this network also contains two genes in the RAS oncogene family: RAB1A and RAB11A. Genes in this family have been linked to many types of cancer, and are considered promising targets for therapeutics [Adjei, 2008]. These genes' connections with ribosome activity in the tumor samples may indicate a relationship common to an important subset of cancers. Many other genes belong to this network, each indicating a potentially novel interaction in cancer biology.

2.9 Discussion

We have introduced the joint graphical lasso, a method for estimating sparse inverse covariance matrices on the basis of observations drawn from distinct but related classes. We employ an ADMM algorithm to solve the joint graphical lasso problem with any convex penalty function, and we provide explicit and efficient solutions for two useful penalty functions. Our algorithm is tractable on very large datasets ($> 20,000$ features), and usually converges in seconds for smaller problems (500 features). Our joint estimation methods outperform competing approaches on a range of simulated datasets.

In the JGL optimization problem (2.2.4), the contribution of each class to the penalized log likelihood is weighted by its size; consequently, the largest class can have outsize influence on the estimated networks. By omitting the n_k term in (2.2.4), it is possible to weight the classes equally to prevent a single class from dominating estimation.

We note that FGL and GGL's reliance on two tuning parameters is a strength rather than a drawback: unlike the proposal of Guo et al. [2011], which involves a single tuning parameter that controls both sparsity and similarity, in performing FGL and GGL one can vary separately the amount of similarity and sparsity to enforce in

the network estimates.

The joint graphical lasso has potential applications beyond those discussed in this chapter. For instance, one could use it to shrink multiple classes' precision matrices towards each other in order to define a classifier intermediate between quadratic discriminant analysis (QDA) and linear discriminant analysis (LDA) [Hastie et al., 2009]. In fact, a similar approach has been taken in recent work [Simon and Tibshirani, 2011]. In the unsupervised setting, it can be used in the maximization step of Gaussian model-based clustering in order to reduce the variance associated with estimating a separate covariance matrix for each cluster.

"JGL", an R package implementing FGL and GGL, is available on `CRAN`, `http://cran.r-project.org/`.

Chapter 3

A TEST FOR CHANGES IN CO-REGULATION OF BIOLOGICAL PATHWAYS

3.1 Overview

Research with high-dimensional biological data has expanded from single-gene analyses to include a focus on the behavior of known biological pathways. Working on the level of pathways makes use of known biological relationships, improving analyses' biological interpretability and compensating for genomic data's high level of noise by combining the signals of variable single genes into more stable statements about pathways. Existing pathway analysis methods examine a very limited set of questions, focusing almost exclusively on the mean (marginal) behavior of pathway genes, for example looking for pathways that are up or down-regulated or that are enriched with differentially expressed genes.

In this chapter, we propose to broaden the examination of biological pathways by testing for differences in the covariance matrices of pathway genes between disease states. By examining the rich information available in the joint, or interactive behavior of pathway genes, we hope to allow more complete examination of biological pathways. We begin by introducing a model for the joint behavior of pathway genes that will suggest biological interpretations for a number of functions of the eigenvalues of the pathway genes' covariance matrix. These aspects of a pathway's eigenstructure will estimate important biological quantities like the variability of pathway activity and the extent of pathway dysregulation. We then design a test for between-class differences in these functions of the eigenvalues of the covariance matrix. We thereby

hope to provide for the identification of meaningful differences in the pattern of co-regulation of pathway genes, capturing important information in the genes' covariance and complimenting existing, mean-focused pathway-level analyses.

The covariance matrix $\mathbf{\Sigma}$ for a set of genes provides a relatively parsimonious description of their joint behavior. In multivariate normal (MVN) data, the covariance matrix defines their joint behavior completely; under other distributions, it provides a partial but useful description. The eigendecomposition of the covariance matrix of a MVN distribution directly defines the shape, scale and orientation of the elliptical contours of the distribution. The sum of the eigenvalues determines the scale, the relative sizes of the eigenvalues determine the shape, and the eigenvectors determine the orientation. As the eigenvectors of empirical covariance matrices are very difficult to interpret and somewhat unstable unless $n \gg p$ [Paul, 2007], we will focus on properties determined by the eigenvalues: scale and shape.

The scale of a pathway's covariance matrix reflects a quantity of obvious interest: its overall variability. We will measure a pathway's variability with the trace of its covariance matrix, or equivalently with the sum of its eigenvalues.

Describing the shape of the covariance matrix is a more complex problem. The standardized vector of eigenvalues provides a complete description of the shape of $\mathbf{\Sigma}$; however, we argue that a more parsimonious statistic would be more valuable, both for interpretation and for statistical stability. We will rely on our understanding of biology to guide a simpler approach. First, we distinguish between two primary sources of variability in genes' measured expression levels: variability resulting from changing levels of pathway activity, and variability resulting from biological and technical noise. Second, we work from the premise that most biological pathways have a primary function and that co-regulation of pathway genes is largely driven by the level of activity of that function. Thus is it reasonable to suppose that the first principal

component of a pathway would correspond to variability due to changing levels of pathway activity, and that the remaining eigenvectors would mainly map noise. We therefore expect that a pathway's first eigenvalue will be prominent and most of the remaining eigenvalues will be small. In this setting, we can reasonably narrow our description of the shape of $\mathbf{\Sigma}$ to focus only on the share of variability captured by the first eigenvector. We will therefore base our test for differences in the joint behavior of pathway genes on two quantities: the sum of the eigenvalues, and the first eigenvalue. In doing so, we will capture differences in both the shape and scale of pathway genes' covariance structure.

The rest of this chapter is organized as follows. In Section 3.2, we detail existing theoretical results upon which we will rely, and we review earlier tests for differences in joint behavior. In Section 3.3 we define our test, motivating it from a biological model described in Section 3.3.1, detailing how biological phenomena are reflected in the eigenvalues of the pathway genes' covariance matrix under this model in Section 3.3.2, providing a heuristic justification of our method in Section 3.3.3, and fully defining our testing procedure in Section 3.3.4. In Section 3.3.5 we derive an alternative version of our procedure for pathways where no co-regulation is apparent. In Section 3.3.6 we discuss normalization of datasets in preparation for our test. In Section 3.4 we evaluate our test in simulated datasets, and in Section 3.5 we apply our test to a prostate cancer gene expression dataset. In Section 3.6 we suggest a number of extensions to this method.

3.2 Background

3.2.1 Behavior of sample eigenvalues

Estimating the scale of a covariance matrix is an easy task: the sum of the sample eigenvalues is an unbiased estimate of the sum of the population eigenvalues. However,

individual sample eigenvalues provide highly biased estimates of their population counterparts, and this bias depends on n, p, and $\mathbf{\Sigma}$ in complex ways [Marčenko and Pastur, 1967]. The relationship between sample size and the bias of the first sample eigenvalue will complicate the task of testing for differences in the shape of covariance matrices. A brief review of the behavior of the first sample eigenvalue follows.

Consider MVN data with n observations of p features, and assume that $n, p \to \infty$ and $\frac{p}{n} \to \gamma \in (0, \infty)$. Denote the population eigenvalues $\{\lambda_1, \ldots, \lambda_p\}$, and the sample eigenvalues $\{\hat{\lambda}_1, \ldots, \hat{\lambda}_p\}$. In the simple case that $\mathbf{\Sigma} = \mathbf{I}$, the first sample eigenvalue $\hat{\lambda}_1$ converges almost surely to $(1 + \sqrt{\gamma})^2$ [Bai and Yin, 1993]. In the slightly more complex case that most eigenvalues are equal to one and a small number of "spiked" eigenvalues are greater than one, the sample eigenvalues corresponding to the spiked population eigenvalues exhibit two distinct behaviors depending on the value of the spiked population eigenvalues. Specifically, consider the j^{th} population eigenvalue λ_j. If $\lambda_j > 1 + \sqrt{\gamma}$, then $\hat{\lambda}_j \to \lambda_j + \frac{\gamma \lambda_j}{\lambda_j - 1}$; whereas if $\lambda_j \in (1, 1 + \sqrt{\gamma})$, then its corresponding sample eigenvalue behaves as if the population eigenvalue were not spiked and converges to a value within the support of the Marčenko-Pastur density, $[(1 - \sqrt{\gamma})^2, (1 + \sqrt{\gamma})^2]$ [Baik and Silverstein, 2006]. More generally, El Karoui [2007] showed that for a very large class of $\mathbf{\Sigma}$'s, the first sample eigenvalue will converge in distribution to the Tracy-Widom law after appropriate centering and scaling.

In Section 3.3.1, we argue that after appropriate scaling, most biological settings will be well-described by a $\mathbf{\Sigma}$ with mainly unit eigenvalues and with a small number of high, spiked eigenvalues. We will motivate our test from this setting. Paul [2007] details the asymptotic behavior of the sample eigenvalues in this spiked eigenvalue setting, a result we restate in Theorem 3.

Theorem 3. *Assume that $p, n \to \infty$ and $\frac{p}{n} \to \gamma \in (0, \infty)$. Further assume that the eigenvalues of the covariance matrix are all 1 except for a fixed number of spikes with*

value greater than $1+\sqrt{\gamma}$ *(and with multiplicity 1). Then for a sample eigenvalue* $\hat{\lambda}_j$ *corresponding to a population eigenvalue* $\lambda_j > 1+\sqrt{\gamma}$*,*

$$\sqrt{n}(\hat{\lambda}_j - g(\lambda_j)) \rightarrow_d N(0, \sigma^2(\lambda_j)), \tag{3.2.1}$$

where $g(\lambda_j) = \lambda_j + \gamma\frac{\lambda_j}{\lambda_j - 1}$ *and* $\sigma^2(\lambda_j) = 2\lambda_j^2(1 - \frac{\gamma}{(\lambda_j-1)^2})$*.*

We will rely heavily on these results.

A few authors have attempted to correct the bias of the first sample eigenvalue for general covariance matrices. For a large class of covariance matrices, the Marčenko-Pastur equation describes the bias of the sample eigenvalues by detailing a complex relationship between the Stieltjes transform of the empirical distribution of the sample eigenvalues and the distribution of population eigenvalues [Marčenko and Pastur, 1967]. El Karoui [2006] bases a sample eigenvalue bias correction method on this relationship. Hendrikse et al. [2009] rely less on theory and use a bootstrap-like approach to eigenvalue bias correction. They use an iterative algorithm to define a covariance matrix whose sample eigenvalues closely approximate the data's sample eigenvalues, and they take the eigenvalues of this known covariance matrix as estimates of the true population eigenvalues. Neither of these approaches dominates the other, and while both improve on the sample eigenvalues, simulations in Hendrikse et al. [2009] show that neither reliably estimates the population eigenvalues. Furthermore, both are very computationally intensive. For these reasons, we will not incorporate these generalized methods into our test. Instead, we will rely on the assumptions of the spiked eigenvalue case laid out in Paul [2007].

3.2.2 Existing tests for differing joint behavior of pathway genes

Now suppose we have observations from two distinct distributions with covariance matrices $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$. Our goal is to test the null hypothesis that $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2$. (Here, $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$ are $p \times p$ matrices, where p is the number of genes in the pathway.) A number of authors define tests for differences in covariance matrices. We will compare our work to two recent tests designed with modern high-dimensional data in mind.

Schott [2007] introduces a test for equality of covariance matrices that is designed to perform well even when dimension exceeds sample size. Under the assumption that the data are multivariate normal, the product of a class k's empirical covariance matrix $\mathbf{S}_k$ with its sample size n_k will be generated from a Wishart distribution with n_k degrees of freedom and some covariance matrix $\mathbf{\Sigma}_k$. The likelihood ratio test (LRT) for $H_0 : \mathbf{\Sigma}_1 = \mathbf{\Sigma}_2$ includes terms for $\hat{\mathbf{\Sigma}}_1^{-1}$ and $\hat{\mathbf{\Sigma}}_2^{-1}$ and so is undefined when $p > \min(n_1, n_2)$. Since the LRT is unavailable, Schott [2007] bases his test on the sum of squared differences between elements of the classes' empirical covariance matrices. He calculates the bias and variance of this quantity, and shows that the asymptotic null distribution of his normalized statistic is normal.

Srivastava and Yanagihara [2010] extends Schott [2007]'s approach to examine differences in the empirical covariance matrices' traces as well as the squared differences on all their elements. This modification allows the testing of a one-sided alternative hypotheses: based on the traces of $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$, we can say that one is bigger than the other. Srivastava and Yanagihara [2010] also show the asympototic normality of their statistic.

The methods of Schott [2007] and Srivastava and Yanagihara [2010] represent important conceptual steps, but their utility in pathway data is limited by the difficulty in interpreting their results. In the event that either test rejects the null hypothesis, it will be difficult to say what aspect of the covariance structures is driving the out-

come. Srivastava and Yanagihara [2010]'s test at least directly reflects differences in the scales of $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$, but it is entirely uninformative about differences arising from the shape or orientation of these matrices. A test focused on more specific aspects of the covariance matrix with immediate relevance to important biological quantities will be more useful for pathway analysis than these methods.

In this spirit, Eddy et al. [2010] describe a test for changes in the strength of co-regulation of biological pathways. Working from the philosophy that genomics data is generally of very poor quality, they argue for the use of extremely robust methods based on order statistics. Consequently, their method ignores covariance information in its characterization of the joint behavior of pathway genes. They examine the order of the features' expression values in each sample, they measure how consistent this ordering is across samples, and they test the null hypothesis that the consistency of feature ordering is the same across classes. Pathways whose genes or proteins have more inconsistent orderings across samples are labelled as less tightly regulated.

Eddy et al. [2010]'s method is very robust to poor data quality, and is actually unaffected by any normalization scheme that applies a monotone transformation to each observation separately. However, this order statistic-based approach ignores considerable useful information when the dataset is of adequate quality.

The method we propose is intermediate to the very strong null hypotheses of Schott [2007] and Srivastava and Yanagihara [2010] and the very specific null hypothesis of Eddy et al. [2010]. Like the method of Eddy et al. [2010], our null hypothesis reflects specific biological changes, allowing interpretation beyond merely concluding $\mathbf{\Sigma}_1 \neq \mathbf{\Sigma}_2$ as in Schott [2007] and Srivastava and Yanagihara [2010]. However, while the method of Eddy et al. [2010] is designed to detect only changes in the strength of pathway gene co-regulation, our test responds to a number of other interesting differences in co-regulation.

3.3 A test for differences in the eigenstructure of Σ_1 and Σ_2

3.3.1 A model of co-expression in biological pathways

In this section we introduce a number of biological assumptions. These assumptions will motivate our test, and they will allow us to appeal to existing theoretical results by implying a "spiked" eigenvalue model.

We assume that a single biological process is the primary driver of variability of genes within a pathway. We assume that the strength of that process – the pathway's activity level – affects the expression of each gene linearly. And we assume that variability due to causes other than changing pathway activity is uncorrelated between genes and of roughly the same magnitude. The choice of genes to analyze collectively as a pathway will strongly influence the accuracy of these first assumptions. Finally, we further assume that our data are sampled from a multivariate normal distribution.

Below, we make these motivating assumptions more concrete. Consider for the moment a dataset with only one class. Let the data for a pathway's n observations and p features be $\mathbf{Y}_{n\times p}$, and without loss of generality assume $\mathbf{Y}$ is centered so that $E(y_{ij}) = 0$.

We propose to model $\mathbf{Y}$ with a random effect model:

$$y_{ij} = h_j a_i + \epsilon_{ij},$$

where a_i is a random variable with mean 0 and variance σ_a^2, which reflects the pathway activity in the i^{th} sample; h_j is a scaling coefficient for the j^{th} gene in the pathway with $\|h\|_2^2 = 1$; and $\{\epsilon_{ij}\}$ are independent noises with mean 0 and variance c. Then the covariance of an observation $\mathbf{y}_i$ becomes:

$$\mathrm{Cov}(\mathbf{y}) = \mathrm{Cov}(a\mathbf{h} + \boldsymbol{\epsilon}) = \sigma_a^2 \mathbf{h}\mathbf{h}^T + c\mathbf{I}. \tag{3.3.2}$$

So the first eigenvalue of $\mathrm{Cov}(\mathbf{y})$ is $\sigma_a^2 + c$, and the rest are equal to c. Note that this biological model matches the spiked eigenvalue model investigated by Paul [2007] and Bai and Yao [2008], allowing us to appeal to their theoretical results for the asymptotic distribution of the first sample eigenvalue.

In datasets with two classes with covariances $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$, we may wish to test for differences in the shape and scale of these covariance matrices. These quantities are described completely by the eigenvalues of $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$. (The biological relevance of these changes is described in Section 3.3.2.) Based on the above model, to discover between-class differences in eigenvalues we need only test

$$H_{00} : \begin{pmatrix} \sigma_{a1}^2 \\ c_1 \end{pmatrix} = \begin{pmatrix} \sigma_{a2}^2 \\ c_2 \end{pmatrix}.$$

In order to relax the assumptions underlying equation 3.3.2 while retaining our test's biological interpretation, we propose instead to test the null hypothesis

$$H_0 : \begin{pmatrix} \alpha_1 \\ \mathrm{trace}\mathbf{\Sigma}_1 \end{pmatrix} = \begin{pmatrix} \alpha_2 \\ \mathrm{trace}\mathbf{\Sigma}_2 \end{pmatrix},$$

where α_k denotes the first population eigenvalue of class k. When our biological model holds, these two null hypotheses are equivalent; in more general settings, H_0 captures the most salient aspects of the shape and scale of the covariance matrices, while the parameters underlying H_{00} may lose their meaning.

The biological model described in this section is an extremely simplified representation of any real biological system. We have made detailed assumptions that will usually fail. However, our empirical results suggest that modest departures from these assumptions have limited effects on the behavior of our test, with the first sample eigenvalue's mean and variance departing minimally from their theoretical values

and its distribution remaining close to normal. Thus we argue that is appropriate to motivate our test from these assumptions. Section 3.3.6 describes normalization procedures to improve fidelity to these assumptions and details the consequences of certain violations of our assumptions.

3.3.2 Interpreting the eigenvalues of $\mathbf{\Sigma}$

In this section we use our biological model to ascribe meaning to various functions of the eigenvalues of a pathway's covariance matrix, thereby arguing for the biological relevance of the test we propose. Continuing for the moment to consider only a single class of data, let $\mathbf{\Sigma}$ be the population covariance matrix for the data $\mathbf{Y}$. Let the eigenvalues of $\mathbf{\Sigma}$ (the "population eigenvalues") be $\mathbf{\Lambda} = \{\lambda_1, \ldots, \lambda_p\}$, and let the eigenvalues of the sample covariance matrix $\hat{\mathbf{\Sigma}}$ (the "sample eigenvalues") be $\hat{\mathbf{\Lambda}} = \{\hat{\lambda}_1, \ldots, \hat{\lambda}_p\}$. Let $\alpha := \lambda_1$. Under our biological model, the joint behavior of the features captured by the shape and scale of $\mathbf{\Sigma}$ is controlled by only two parameters: the common unspiked eigenvalue c, and the spiked first eigenvalue, which we will call α. By estimating these parameters we can describe a number of biologically relevant characteristics of the joint behavior of pathway genes. And in two-class settings, we will be able to detect changes in important aspects of pathway co-regulation by testing for changes in these parameters between classes.

We can estimate α with $L := \hat{\lambda}_1$, the first sample eigenvalue. Under our biological assumptions, α can be said to describe the variability in the pathway genes due to changing levels of pathway activity. If L differs considerably between two classes, then we might conclude that the pathway lies in a stable equilibrium in one class while having more heterogeneous activity in the other class.

Define $T := \sum_i \hat{\lambda}_i$, the sum of the eigenvalues of $\hat{\mathbf{\Sigma}}$, or equivalently trace$\hat{\mathbf{\Sigma}}$. T will estimate the overall variability of the pathway genes. If the variance of a pathway's

genes differs substantially between two classes, we might conclude that the pathway is experiencing a greater range of activity in one class, or perhaps that biological noise has increased for some reason.

By examining L and T in tandem, we reveal other pertinent aspects of the pattern of co-regulation of the pathway genes. By subtracting L from T, we estimate the variance due to anything other than linear responses to pathway activity. (Under the full biological model, $\frac{T-L}{p-1}$ provides a biased [Paul, 2007] estimate of c, the noise each gene experiences.) So while L captures variability attributable to the functioning of the pathway, $(T-L)$ can be said to capture variability due to noise and other factors unrelated to pathway function. We find $(T-L)$ to be a useful measure of pathway noise or lack of co-regulation. If this quantity differs substantially between classes, we might conclude that in one class the pathway is dysregulated or is participating in an additional, unrelated biological process.

Under our biological model, the proportion of total variability explained by the first eigenvector completely describes the shape of $\mathbf{\Sigma}$. In the case where our full set of biological assumptions fails but the expression levels of the pathway genes primarily respond to a single biological process, this proportion captures the most salient aspects of the shape of $\mathbf{\Sigma}$. We can estimate this quantity with L/T. This "shape" statistic has a geometric interpretation: it reflects the degree of elongation of the ellipse defined by $\mathbf{\Sigma}$. And it has a biological interpretation: it provides a measure of the degree of co-regulation a pathway's genes experience. Biological phenomena like weakened co-regulation, increased variability of pathway activity, and pathway dysregulation will all manifest as between-class differences in L/T.

Thus the pair (L, T) describes a great deal of the biologically interesting aspects of the joint behavior of pathway genes. Furthermore, under our motivating biological model, the pair (L, T) estimates the only parameters of interest, α and c. We therefore

propose to test for differences in (L, T), and argue that this approach will identify pathways whose joint behavior changes in meaningful ways between classes.

3.3.3 Derivation of test

In this section we derive a quadratic form test statistic for testing the null hypothesis that the functions of the eigenvalues described in Section 3.3.2 are unchanged between two classes of data. As we consider two-class datasets, we subscript the parameters and statistics described above to denote class, writing L_1, L_2, α_1, α_2, T_1, T_2, $\mathbf{\Sigma}_1$, $\mathbf{\Sigma}_2$, $\gamma_1 = \frac{p}{n_1}$, $\gamma_2 = \frac{p}{n_2}$, etc. We will test $H_0 : \begin{pmatrix} \alpha_1 \\ \text{trace}\mathbf{\Sigma}_1 \end{pmatrix} = \begin{pmatrix} \alpha_2 \\ \text{trace}\mathbf{\Sigma}_2 \end{pmatrix}$.

Since L_k estimates α_k, and T_k estimates $\text{trace}\mathbf{\Sigma}_k$, it makes sense to rely on $(L_1 - L_2)$ and $(T_1 - T_2)$ to test H_0. A quadratic form statistic promises to effectively combine the information in these two statistics. Below, we derive the details of this quadratic form.

Before we introduce our test statistic, we must address the issue of bias in $(L_1 - L_2)$. Due to the sample size-dependent bias in L_1 and L_2 described in Theorem 3, under H_0,

$$E(L_1 - L_2) \to (\gamma_1 - \gamma_2)\frac{\alpha_0}{\alpha_0 - 1},$$

where α_0 is the first eigenvalue shared by both classes under H_0. When $n_1 \neq n_2$, this expectation is not equal to 0. To center our statistic under H_0, we will attempt to subtract this bias by working with $(L_1 - L_2 - b_L)$, where $b_L = (\gamma_1 - \gamma_2)\frac{\hat{\alpha}_0}{\hat{\alpha}_0 - 1}$.

We therefore define our test statistic as $\mathbf{Q}^T \widehat{\mathbf{\Sigma}}_Q^{-1} \mathbf{Q}$, where

$$\mathbf{Q} := \begin{pmatrix} Q_L \\ Q_T \end{pmatrix} := \begin{pmatrix} L_1 - L_2 - b_L \\ T_1 - T_2 \end{pmatrix},$$

where $\mathbf{\Sigma}_Q := \text{Cov}(\mathbf{Q})$, and where b_L is an estimate of the bias in $(L_1 - L_2)$. It remains

to estimate b_L and $\mathbf{\Sigma}_Q$.

To calculate b_L, we need an estimate of α_0. We define $\hat{\alpha}_0$ as a weighted average of estimates of α_1 and α_2 derived using the method of moments. Specifically, if α_k is the true first spiked eigenvalue for class k, then

$$E(L_k) = \alpha_k + \gamma_k \frac{\alpha_k}{\alpha_k - 1}. \tag{3.3.3}$$

We obtain a method of moments estimate of α_k by replacing $E(L_k)$ with L_k in 3.3.3 and solving for α_k, yielding the estimate

$$\hat{\alpha}_k = \frac{1}{2}\left(1 + L_k - \gamma_k + \sqrt{(1 + L_k - \gamma_k)^2 - 4L_k}\right).$$

Finally, we define

$$\hat{\alpha}_0 := \hat{\alpha}_0(L_1, L_2) := (w_1\hat{\alpha}_1 + w_2\hat{\alpha}_2), \tag{3.3.4}$$

where $w_k = n_k/(n_1 + n_2)$.

Note that if $L_k < (1 + \sqrt{\gamma_k})^2$, then our expression for $\hat{\alpha}_0$ is undefined. When this occurs, we choose a small δ and take $\hat{\alpha}_k = 1 + \sqrt{\max(\gamma_1, \gamma_2)} + \delta$, attaining a value of α_k that is large enough to be consistent with the spiked eigenvalue model and small enough to explain the observed L_k. This arbitrary value of $\hat{\alpha}_k$ strikes a reasonable balance: much smaller values of α would not fit the spiked eigenvalue model, and larger values of α would make the observed L_k less likely. In the event that an L_k is quite low, use of the alternative method described in Section 3.3.5 provides a less ad-hoc approach than this fix.

Now consider estimation of $\mathbf{\Sigma}_Q := \begin{bmatrix} \sigma^2_{Q_L} & \sigma^2_{Q_{LT}} \\ \sigma^2_{Q_{LT}} & \sigma^2_{Q_T} \end{bmatrix}$. We estimate this matrix using the results of Mathai and Pillai [1982], which detailed the asymptotic distribution of the T statistic, and the results of Paul [2007], which detailed the asymptotic

distribution of the L statistic under the spiked eigenvalue setting.

An estimate of $\sigma^2_{Q_T} = \text{Var}(T_1 - T_2)$ follows easily from Mathai and Pillai [1982]'s result showing $\text{Var}(T_k) = 2n_k/(n_k-1)^2\text{trace}(\mathbf{\Sigma}_k^2)$. As T_1 and T_2 are independent, we estimate $\sigma^2_{Q_T}$ with

$$\hat{\sigma}^2_{Q_T} = \widehat{\text{Var}}(T_1) + \widehat{\text{Var}}(T_2) = \sum_k \frac{2n_k}{(n_k-1)^2}\text{trace}(\hat{\mathbf{\Sigma}}_k\hat{\mathbf{\Sigma}}_k). \tag{3.3.5}$$

Next, consider estimation of $\sigma^2_{Q_L} = \text{Var}(Q_L(L_1, L_2))$. Write out $Q_L(L_1, L_2)$ as $L_1 - L_2 - b_L = L_1 - L_2 - (\gamma_1 - \gamma_2)\frac{\hat{\alpha}_0(L_1,L_2)}{\hat{\alpha}_0(L_1,L_2)-1}$, where $\hat{\alpha}_0(L_1, L_2)$ is as described in equation 3.3.4. We can use the delta method to estimate $\text{Var}(Q_L)$. Writing $\mu_k := E(L_k) = \alpha_k + \gamma_k\frac{\alpha_k}{\alpha_k-1}$ and $\text{Cov}\begin{pmatrix} L_1 \\ L_2 \end{pmatrix} = \begin{bmatrix} \sigma^2_{L_1} & 0 \\ 0 & \sigma^2_{L_2} \end{bmatrix}$, we have

$$\text{Var}(Q_L) = \nabla Q_L(\mu_1, \mu_2)^T \begin{bmatrix} \sigma^2_{L_1} & 0 \\ 0 & \sigma^2_{L_2} \end{bmatrix} \nabla Q_L(\mu_1, \mu_2). \tag{3.3.6}$$

Then to estimate $\text{Var}(Q_L)$, we must estimate μ_1, μ_2, $\sigma^2_{L_1}$ and $\sigma^2_{L_2}$, and we must derive $\nabla Q_L(\mu_1, \mu_2)$. We take $\hat{\mu}_k = \hat{\alpha}_0 + \gamma_k\frac{\hat{\alpha}_0}{\hat{\alpha}_0-1}$, the expectation of L_k under the null when $\alpha_0 = \hat{\alpha}_0$. We use the results of Paul [2007] to estimate $\sigma^2_{L_k}$ with

$$\hat{\sigma}^2_{L_k} = 2\hat{\alpha}_0^2((\hat{\alpha}_0 - 1)^2 - \gamma_k)/(n_k(\hat{\alpha}_0 - 1)^2).$$

And we calculate

$$\frac{\partial}{\partial\mu_k}Q_L(\mu_1, \mu_2) = 1 - \frac{w_k(\gamma_1 - \gamma_2)}{2(\hat{\alpha}_0(\mu_1, \mu_2) - 1)^2}\left(1 + \frac{\mu_k - \gamma_k - 1}{\sqrt{(1 + \mu_k - \gamma_k)^2 - 4\mu_k}}\right). \tag{3.3.7}$$

Finally, we write

$$\hat{\sigma}^2_{Q_L} = \widehat{\mathrm{Var}}(Q_L) = \sum_k (\frac{\partial}{\partial \mu_k} Q_L(\hat{\mu}_1, \hat{\mu}_2))^2 \hat{\sigma}^2_{L_k}. \tag{3.3.8}$$

Note that in the case of very small L_k requiring the choice of an arbitrarily low $\hat{\alpha}_k$, the partial derivative of Q_L with respect to L_k is simply 1.

To complete our estimate of $\mathbf{\Sigma}_Q$, we must estimate $\sigma^2_{Q_{LT}} = \mathrm{Cov}(L_1 - L_2 - b_L, T_1 - T_2)$. Unfortunately, no theory exists detailing the covariance between L_k and T_k. We employ a subsampling approach to estimate this quantity. We choose subsample sizes $n_{0_1} < n_1$ and $n_{0_2} < n_2$ and draw a large number of datasets of sizes n_{0_1} and n_{0_2} without replacement from $\mathbf{Y}_1$ and $\mathbf{Y}_2$. Our empirical experience suggests a choice of $n_{0_k} > \frac{1}{2} n_k$ will lead to accurate estimation. We have found that the best approach is to use these subsampled datasets to estimate $\mathrm{Cor}(L_1 - L_2 - b_L, T_1 - T_2)$, and then to use the theoretical variance estimates to rescale this correlation to an estimate of covariance. Using subsampling to directly estimate $\mathrm{Cov}(L_1 - L_2 - b_L, T_1 - T_2)$ performs poorly, as subsampling estimates of the variance of L tend to be unreliable, possibly due to the relationship between sample size and the expectation of the first sample eigenvalue L. We also found bootstrap estimates of the correlation and covariance of our statistics to be inaccurate, perhaps because repeated observations increase the first eigenvalue of the empirical covariance matrix.

After we obtain $\widehat{\mathbf{\Sigma}}_Q$, we propose to reject H_0 for large values of $\mathbf{Q}^T \widehat{\mathbf{\Sigma}}_Q^{-1} \mathbf{Q}$. A permutation test can then be used to decide the significance cutoff under the null. We also speculate that the null distribution of the proposed test statistic can be well approximated with a χ^2_2 distribution. This property is illustrated in the simulations section. More investigations of the theoretical distribution of our statistic are underway.

3.3.4 Procedure

Our full procedure is as follows:

1. Calculate the classes' first sample eigenvalues L_1 and L_2 and sums of eigenvalues T_1 and T_2. Take $\gamma_1 = \frac{p}{n_1}$ and $\gamma_2 = \frac{p}{n_2}$.

2. Estimate the bias in $(L_1 - L_2)$ under H_0:
   $b_L = (\gamma_1 - \gamma_2)\frac{\hat{\alpha}_0}{\hat{\alpha}_0 - 1}$, where $\hat{\alpha}_0$ is estimated as described in Section 3.3.3 and equation 3.3.4.

3. Define $\mathbf{Q} := \begin{pmatrix} Q_L \\ Q_T \end{pmatrix} := \begin{pmatrix} L_1 - L_2 - b_L \\ T_1 - T_2 \end{pmatrix}$.

4. Estimate $\mathbf{\Sigma}_Q := \begin{bmatrix} \sigma^2_{Q_L} & \sigma^2_{Q_{LT}} \\ \sigma^2_{Q_{LT}} & \sigma^2_{Q_T} \end{bmatrix} := \text{Cov}(\mathbf{Q})$:

   (a) Per the results of Mathai and Pillai [1982], calculate $\hat{\sigma}^2_{Q_T} = \widehat{\text{Var}}(T_1 - T_2) = \sum_k 2n_k/(n_k - 1)^2 \text{trace}(\mathbf{\Sigma}_k^2)$.

   (b) Calculate $\hat{\sigma}^2_{Q_L} = \widehat{\text{Var}}(L_1 - L_2 - b_L)$ using the results of Paul [2007] and the delta method:
   $$\hat{\sigma}^2_{Q_L} = \sum_k \left(\frac{\partial}{\partial \hat{\mu}_k} Q_L(\hat{\mu}_1, \hat{\mu}_2) \right)^2 \hat{\sigma}^2_{L_k},$$
   where $\hat{\mu}_k = \hat{\alpha}_0 + \gamma_k \frac{\hat{\alpha}_0}{\hat{\alpha}_0 - 1}$, where $\frac{\partial}{\partial \hat{\mu}_k} Q_L(\hat{\mu}_1, \hat{\mu}_2)$ is as detailed in equation (3.3.7) and where $\hat{\sigma}^2_{L_k} = 2\hat{\alpha}_0^2((\hat{\alpha}_0 - 1)^2 - \gamma_k)/(n_k(\hat{\alpha}_0 - 1)^2)$.

   (c) Use subsampling to calculate $\sigma^2_{Q_{LT}} = \widehat{\text{Cov}}(L_1 - L_2 - b_L, T_1 - T_2)$:

      i. Pick subsample sizes $n_{0_1} < n_1$ and $n_{0_2} < n_2$ and a number of replicates B, and take B subsamples without replacement of sizes n_{0_1} and n_{0_2} from $\mathbf{Y}_1$ and $\mathbf{Y}_2$.

ii. Calculate $(L_1 - L_2 - b_L)$ and $(T_1 - T_2)$ on each subsampled dataset, and define $\widehat{\text{Cor}}(L_1 - L_2 - b_L, T_1 - T_2)$ as the empirical correlation between these quantities.

iii. Calculate $\hat{\sigma}^2_{Q_{LT}} = \widehat{\text{Cor}}(L_1 - L_2 - b_L, T_1 - T_2)\sqrt{\hat{\sigma}^2_{Q_L}\hat{\sigma}^2_{Q_T}}$.

5. Compute the test statistic $\mathbf{Q}^T\widehat{\mathbf{\Sigma}}_Q^{-1}\mathbf{Q}$. To attain a p-value, compare its value to the quantiles of a χ^2_2 distribution. Alternatively, permute the data's class labels and recompute the test statistic a large number of times, and compare the quantiles of the resulting statistics to the true $\mathbf{Q}^T\widehat{\mathbf{\Sigma}}_Q^{-1}\mathbf{Q}$.

3.3.5 Alternative formulation of the test for the case where there is no spiked eigenvalue

The above theory and procedure allow us to test H_0 in the case where the common first eigenvalue α_0 of $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$ is greater than $1 + \sqrt{\max(\gamma_1, \gamma_2)}$; that is, when α_0 is "spiked." For completeness, in this section we describe an alternative formulation of our test that is appropriate when we wish to test H_0 in the case where $\alpha_1 = \alpha_2 = 1$, that is, when no eigenvalues are spiked. We note that this unspiked, or uniform, eigenvalue setting is rare in biological data: unmeasured biological covariates are likely to introduce some covariance structure in any group of related genes, and so even in the absence of a shared response to changing pathway activity levels we will expect to see covariance matrices with spiked eigenvalues. Furthermore, if we assume an unspiked model, we are assuming that $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2 = \text{I}$, which in fact implies that our null hypothesis $H_0 : \begin{pmatrix} \alpha_1 \\ \text{trace}\mathbf{\Sigma}_1 \end{pmatrix} = \begin{pmatrix} \alpha_2 \\ \text{trace}\mathbf{\Sigma}_2 \end{pmatrix}$ holds. Thus under the unspiked assumption, our procedure effectively tests the very strong null that $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2 = \text{I}$, and not the much weaker null described by H_0. This strengthening of the null hypothesis removes the link between our test and the specific biological quantities, described

in Section 3.1, that we hoped to capture. (However we do expect that even in the unspiked setting our test will have much greater power to detect the eigenstructure changes we are interested in than to detect other departures from $\mathbf{\Sigma} = \mathrm{I}$.) With these limitations in mind, we re-derive our test for use in the unspiked case below.

Under the assumptions that $\mathbf{\Sigma} = \mathrm{I}$, $n, p \to \infty$, and $p/n \to \gamma \in (0, \infty)$, it has been shown [Johansson, 2000, Johnstone and Graybill, 2001] that

$$\frac{L - \mu_{np}}{\sigma_{np}} \to_d TW_1, \tag{3.3.9}$$

where

$$\mu_{np} = \frac{1}{n}\left(\sqrt{n - 1/2} + \sqrt{p - 1/2}\right)^2,$$

$$\sigma_{np} = \sqrt{\frac{\mu_{np}}{n}}\left(\frac{1}{\sqrt{n - 1/2}} + \frac{1}{\sqrt{p - 1/2}}\right)^{1/3},$$

and where TW_1 is the Tracy Widom distribution with parameter $\beta = 1$.

We can use equation (3.3.9) to derive the bias and variance of $(L_1 - L_2)$. Then the asymptotic expectation of L_k is $E(TW_1)\sigma_{n_kp} + \mu_{n_kp}$, which gives us

$$b_L = E(L_1 - L_2) = E(TW_1)(\sigma_{n_1p} - \sigma_{n_2p}) + \mu_{n_1p} - \mu_{n_2p}.$$

Again using (3.3.9), we calculate $\mathrm{Var}(L_k) = \sigma^2_{n_kp} var(TW_1)$, and using the independence of L_1 and L_2 we get

$$\sigma^2_{Q_L} = var(L_1 - L_2 - b_L) = (\sigma^2_{n_1p} + \sigma^2_{n_2p})\mathrm{Var}(TW_1).$$

(Note that in the unspiked setting the expression for b_L depends only on n_1, n_2 and p and has no random components.)

If we assume the eigenvalues are all uniformly 1, then the distribution of T becomes

simpler as well: we can simply write $(n-1)T \sim \chi^2_{(n-1)p}$. Since $(n-1)p$ will tend to be quite large, the distribution of T will be well-approximated by a $N(p, 2p/(n-1))$ distribution. Then we can estimate $\sigma^2_{Q_T} = \mathrm{Var}(T_1 - T_2)$ with $4p/(n-1)$.

Unlike in the spiked eigenvalue setting, we may easily derive a theoretical expression for $\mathrm{Cov}(L, T)$ in the $\mathbf{\Sigma} = \mathrm{I}$ case. We rely on two results in the literature on the behavior of the statistic $U = L/T$ in the case where $\mathbf{\Sigma} = \mathrm{I}$. First, Johnson and Graybill [1972] show that the statistic U is independent of T. Second, Nadler [2011] derives the asymptotic distribution of U, and shows that $E(U) = E(L)/p$. We can calculate $\mathrm{Cov}(L, T) = E(LT) - E(L)E(T) = E(UT^2) - E(L)E(T)$. By the independence of U and T, this is $E(U)E(T^2) - E(L)E(T) = E(L)\left(E(T^2)/p - E(T)\right)$. Using the fact that $E(L) = E(TW_1)\sigma_{np} + \mu_{np}$, and calculating $E(T^2) = \mathrm{Var}(T) + E(T)^2 = 2p/(n-1) + p^2$, we can write

$$\mathrm{Cov}(L, T) = \frac{2}{n-1}(E(TW_1)\sigma_{np} + \mu_{np}).$$

And we can calculate $\sigma^2_{Q_{LT}} = \mathrm{Cov}(L_1 - L_2 - b_L, T_1 - T_2)$ by calculating the above formula using n_1 and n_2 and summing the results.

We have arrived at expressions for b_L and for $\mathbf{\Sigma}_Q$, enabling calculation of the test statistic $\mathbf{Q}^T \hat{\mathbf{\Sigma}}_Q^{-1} \mathbf{Q}$. We note that the asymptotic non-normality of L in this setting will make our test statistic converge to something other than a χ^2_2 distribution. However, the Tracy Widom distribution is roughly normal-shaped (its skew and kurtosis are approximately 0.29 and 0.17, compared to a variance of 1.6), so this departure should not excessively distort the reported p-values.

Though in actual biological data we do not expect this to occur, if a pathway has small L_1 and L_2, a rule is needed to determine whether to run our test under the spiked or the uniform eigenvalue assumption. A number of tests of the null hypothesis that the eigenvalues are uniform ($\lambda_1 = \ldots = \lambda_p$) have been described in the statistical and signal processing literature [Roy, 1953, Johnson and Graybill, 1972], and these

tests would all be appropriate to reject the uniform eigenvalue model in favor of the spiked model.

3.3.6 Pre-analysis normalization

We have motivated our procedure from a specific biological model, and much of the theory we rely upon to establish the asymptotic distribution of our test statistic assumes that the spiked eigenvalue model suggested by this biological model indeed holds. Note that although our biological model implies only a single spiked eigenvalue, the theory we rely on is valid even when there are several spiked eigenvalues. However, we do require that the data are multivariate normal and that the unspiked eigenvalues of the common covariance matrix under H_0 are all equal to 1, which may be unrealistic in real datasets. In this section, we describe normalization procedures to improve the fidelity of a dataset to these assumptions and thereby to better ensure that our test statistic adheres to its conjectured asymptotic χ^2_2 distribution. (We will not discuss the normality assumption beyond recommending log-transforming data when scientifically appropriate.)

Our first concern is that the value of the baseline, unspiked eigenvalue may not be equal to 1. When this assumption fails, the question arises whether L's asymptotic expectation and variance still apply. Let us modify our model to say that apart from a small number of large eigenvalues, the eigenvalues of our covariance matrix are all equal to some $C > 0$. If we knew this C, we could rescale our data and appeal directly to the existing theory. Let L and α retain their meanings as the first sample and first population eigenvalues, call $L_C = L/C$, and call $\alpha_C = \alpha/C$. Then by Paul [2007],

$$\sqrt{n}(L_C - \alpha_C - \gamma\frac{\alpha_C}{\alpha_C - 1}) \to_d N\left(0, 2\alpha_C^2(1 - \frac{\gamma}{(\alpha_C - 1)^2})\right). \tag{3.3.10}$$

We can easily recover the asymptotic distribution of L from (3.3.10): the bias of L

is $C\gamma\frac{\alpha/C}{\alpha/C-1}$, and its variance is $C^2\frac{2}{n}(\alpha/C)^2(1-\frac{\gamma}{(\alpha/C-1)^2})$. Compare these quantities to what we would estimate with a naive approach in which we assumed $C = 1$ and appealed to the asymptotics of Paul [2007] without rescaling. Under the naive approach, we would assume a bias of $\gamma\frac{\alpha}{\alpha-1}$, and a variance of $\frac{2}{n}\alpha^2(1-\frac{\gamma}{(\alpha-1)^2})$. The difference between the true and the "naive" bias is

$$\gamma\alpha^2\frac{C-1}{(\alpha-1)(\alpha-C)},$$

and the difference between the true and the naive variance is

$$\frac{2}{n}\alpha^3\gamma\frac{(C-1)(2C-\alpha(C+1)}{(\alpha-1)^2(\alpha-C)^2}.$$

So we will need $C \approx 1$ for the naive bias and variance to be accurate.

A number of approaches could be used to estimate C and normalize a pathway's data to encourage $C \approx 1$. The median sample eigenvalue could be taken as representative of the common population eigenvalue. Alternatively, the total variance orthogonal to the first eigenvector will estimate $(p-1)C$ under our motivating model. As it allows for several spiked eigenvalues, we favor the first approach. When it is desirable to normalize an entire dataset containing genes on multiple pathways, we advise dividing the entire dataset by a quantity slightly less than the median variance of the features. This approach will yield a median feature variance slightly greater than 1, and in the event that the first principal component of a pathway captures a small portion of the variance of each feature, this normalization will result in the pathway's unspiked eigenvalues equalling approximately 1. Any normalization approach should be applied to both classes equally rather than separately.

Another concern is that the unspiked eigenvalues of the common covariance matrix under H_0 are not uniform. Depending on the normalization strategy employed,

different features in a dataset may have widely divergent variances. A single highly variable feature could then dominate the first eigenvector, overwhelming the effects of co-regulation responding to pathway activity. In this case, the first eigenvalue would lack the biological interpretation we ascribe to it, and the proposed test would reflect less meaningful quantities. Thus the assumptions underlying our theory will be most accurate when all feature variances are on roughly the same scale.

Achieving this simple goal through normalization is actually quite problematic. For example, if we simply standardized each feature's data to have variance equal to 1, the standardized empirical covariance matrix would be a correlation matrix. However, the theory underlying our test was developed for covariance, not correlation matrices, and its results will not apply to data that is so strongly standardized. Furthermore, we are interested in between-class differences in the variance of the features; thus we must be careful to employ standardization that preserves this information.

Thus standardization to achieve relatively uniform variance is inadvisable. To ensure that single features do not dominate the first eigenvalue, we recommend checking each pathway's first eigenvector for dominance of a single feature to ensure that the first eigenvalue retains its biological interpretation. If a single feature appears to be controlling the first principal component, the pathway should be analyzed without it to ensure the pathway's eigenvalues carry the biological meaning we ascribe to them.

We anticipate that the variable unspiked eigenvalues found in real data will not significantly impact our method. We confirm this suspicion in simulations in Section 3.4.1.

3.4 Simulations

3.4.1 Effect of variable unspiked eigenvalues on the distribution of the first sample eigenvalue

In using Theorem 3 to calculate the asymptotic distribution of the first sample eigenvalue L, we make the very implausible assumption that the data have unspiked eigenvalues exactly equal to one. In this section, we perform simulations to evaluate the effect of variable unspiked eigenvalues on the distribution of the first sample eigenvalue. We find that this perturbation of the spiked eigenvalue model has negligible effect on the distribution of the first sample eigenvalue, suggesting the encouraging conclusion that this particular assumption is not important for our method.

For $p = 30$, we defined the covariance matrices $\mathbf{\Sigma}_{constant} = \text{diag}(15, 1, \ldots, 1)$ and $\mathbf{\Sigma}_{variable} = \text{diag}(15, 1 + e_1, \ldots, 1 + e_{p-1})$, in which $e_1, \ldots, e_{p-1}$ were $N(0, 0.5)$ random variables. $\mathbf{\Sigma}_{constant}$ thus has uniform unspiked eigenvalues and conforms perfectly to our assumptions, while $\mathbf{\Sigma}_{variable}$ has variable unspiked eigenvalues and more accurately represents what we expect to see in real data. We generated 100,000 multivariate normal datasets with $n = 50$ from the two covariance matrices, and observed the distribution of the first sample eigenvalue L under both $\mathbf{\Sigma}_{constant}$ and $\mathbf{\Sigma}_{variable}$. Selected empirical quantiles of these distributions are in Table 3.1. It is clear that perturbing the unspiked eigenvalues has a minimal effect on the distribution of L, and therefore on our test statistic.

3.4.2 Performance of the test under the null

The simulations detailed in this section explore the accuracy of the p-values returned by our method. We generate a large number of datasets under the null for varying n and fixed $p = 30$, and we record the actual Type-1 error rate at the nominal $\alpha = 0.05$

Table 3.1: Empirical quantiles of the first sample eigenvalue L from MVN data drawn from a spiked eigenvalue model with (first row) constant unspiked eigenvalues and with (second row) variable unspiked eigenvalues.

Quantile of L	0.01	0.05	0.25	0.50	0.75	0.95	0.99
constant model	9.491	11.057	13.487	15.426	17.548	20.941	23.631
variable model	9.474	11.059	13.521	15.442	17.552	20.974	23.565

level.

We create a covariance matrix for p features under our motivating case as follows. We define a "pathway activity" variable a with variance equal to $c_0 = 20$. We define the effect of pathway activity on each feature as $\mathbf{h}_{p\times 1} = \{-0.5, \frac{1}{p-1} - 0.5, \ldots, \frac{p-2}{p-1} - 0.5, 0.5\}$, that is, at p even increments between -0.5 and 0.5. (See Section 3.3.1 for a detailed explanation of the role of $\mathbf{h}$.) Then the covariance amongst our features induced by the variability in a is $c_0\mathbf{h}\mathbf{h}^T$. To this covariance we add noise on each feature with variance 1. Then by the same argument employed in Section 3.3.1, the covariance matrix under our motivating model is $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2 = c_0\mathbf{h}\mathbf{h}^T + \mathbf{I}$.

For each $n \in \{20, 50, 100, 150, 200, 300, 400\}$ we generate 10,000 pairs of multivariate normal datasets from $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2$ with $n_1 = n_2 = n$, and we run our test on each pair, using p-values calculated both from the theoretical χ^2_2 distribution and from a permutation test. We also run the tests of Schott [2007] and Srivastava and Yanagihara [2010]. The Type-1 errors at the nominal 0.05 levels for each test at each n are in Table 3.2.

With the use of a permutation test to calculate p-values, the proposed method successfully controls Type-1 error. With the use of the quantiles of a χ^2_2 distribution to calculate p-values, the proposed method is conservative. The method of Schott [2007] is anti-conservative in this setting, with Type-1 error rates almost double their

Table 3.2: Type-1 error rates of competing tests at level 0.05 when dimension $p = 30$.

$n_1, n_2 =$	25	50	100	150	200	300
$\mathbf{Q}^T\hat{\mathbf{\Sigma}}_Q^{-1}\mathbf{Q}$, χ^2_2 quantiles	0.0100	0.0125	0.0152	0.0166	0.0148	0.0168
$\mathbf{Q}^T\hat{\mathbf{\Sigma}}_Q^{-1}\mathbf{Q}$, permutation test	0.0508	0.0504	0.0484	0.0484	0.0475	0.0470
method of Schott (2007)	0.0855	0.0877	0.0843	0.0838	0.0827	0.0819
method of Srivastava (2010)	0.0124	0.0239	0.0321	0.0354	0.0356	0.0412

nominal levels. The method of Srivastava and Yanagihara [2010] is conservative at small n and approaches accurate Type-1 error as n increases.

3.4.3 Performance of the test under the alternative

In each simulation setting, we define $\mathbf{\Sigma}_1$, the covariance matrix for the first class, as in section 3.4.2 using $c_0 = 8$. We then perturb $\mathbf{\Sigma}_1$ to create a slightly different covariance matrix $\mathbf{\Sigma}_2$ for the second class. In our first perturbation, we consider the simple case of a pathway becoming subject to greater noise, defining $\mathbf{\Sigma}_2 = \mathbf{\Sigma}_1 + 0.15\mathbf{I}$. This perturbation results in trace$\mathbf{\Sigma}_1 <$ trace$\mathbf{\Sigma}_2$ and has minimal impact on the first eigenvalue. It simulates biological processes in which gene expression is subject to broad disorder. Results under this setting can be seen in the first row of Figure 3.1.

In our second perturbation, we define $\mathbf{\Sigma}_2 = 0.7\mathbf{\Sigma}_1$. This setting simulates a dataset in which variability due to all causes has been reduced but the shape of the covariance structure is unchanged. Results under this setting can be seen in the second row of Figure 3.1.

In our third perturbation, we define a $\mathbf{\Sigma}_2$ that simulates pathway dysregulation: well-ordered variability captured by the first eigenvector is reduced, and the unordered noise of each feature is increased. We generate $\mathbf{\Sigma}_2$ with eigenvectors identical to those of $\mathbf{\Sigma}_1$, with a first eigenvalue equal to 0.8 times the first eigenvalue of $\mathbf{\Sigma}_1$, and with

remaining eigenvalues greater than those of $\mathbf{\Sigma}_1$ by 0.25. Results under this setting can be seen in the third row of Figure 3.1.

For each setting, we consider sample sizes $n_1 = n_2 \in \{20, 30, 50, 75, 100, 130\}$, and we define $\mathbf{\Sigma}_1$ and each of the perturbed $\mathbf{\Sigma}_2$ matrices for pathways of size $p = \{20, 50, 100\}$. For each p and each n, we simulate 5,000 multivariate normal datasets from $\mathbf{\Sigma}_1$ and from each of our three perturbed $\mathbf{\Sigma}_2$ matrices. For each replicate we use our test to compare the $\mathbf{\Sigma}_1$ dataset to each of the perturbed $\mathbf{\Sigma}_2$ datasets; for completeness, we also run the tests of Schott [2007] and Srivastava and Yanagihara [2010]. We do not include the test of Eddy et al. [2010]: their method is based on the spread of the features' means rather than on the features' covariance, and therefore any comparison to our method in simulated data will depend entirely on the arbitrary portioning of signal between the means and the covariances.

The simulations under the null (Table 3.2) showed that only our method with permutation-derived p-values had accurate Type-1 error. In order to make the tests comparable, in this simulation we eschew their nominal $p = 0.05$ cutoffs in favor of cutoffs at the true 95^{th} percentile of their statistics under the null. To calculate the test statistic cutoffs that correspond to true 0.05 level tests, we simulate 5,000 datasets under the null setting where the covariance matrix for both classes is $\frac{1}{2}(\mathbf{\Sigma}_1 + \mathbf{\Sigma}_2)$. For each test, we record the 95^{th} percentile of the test statistics returned, and we use that value as the cutoff for a level 0.05 test. In our simulations under the alternative hypothesis, we calculate a test's power as the percentage of its test statistics that exceed the cutoff determined in the null setting. The power of each test in each setting can be seen in Figure 3.1.

The proposed method dominates the methods of Schott [2007] and Srivastava and Yanagihara [2010] in these simulations. In other simulations, we found that the method of Schott [2007] performs well in cases where single elements of the covariance

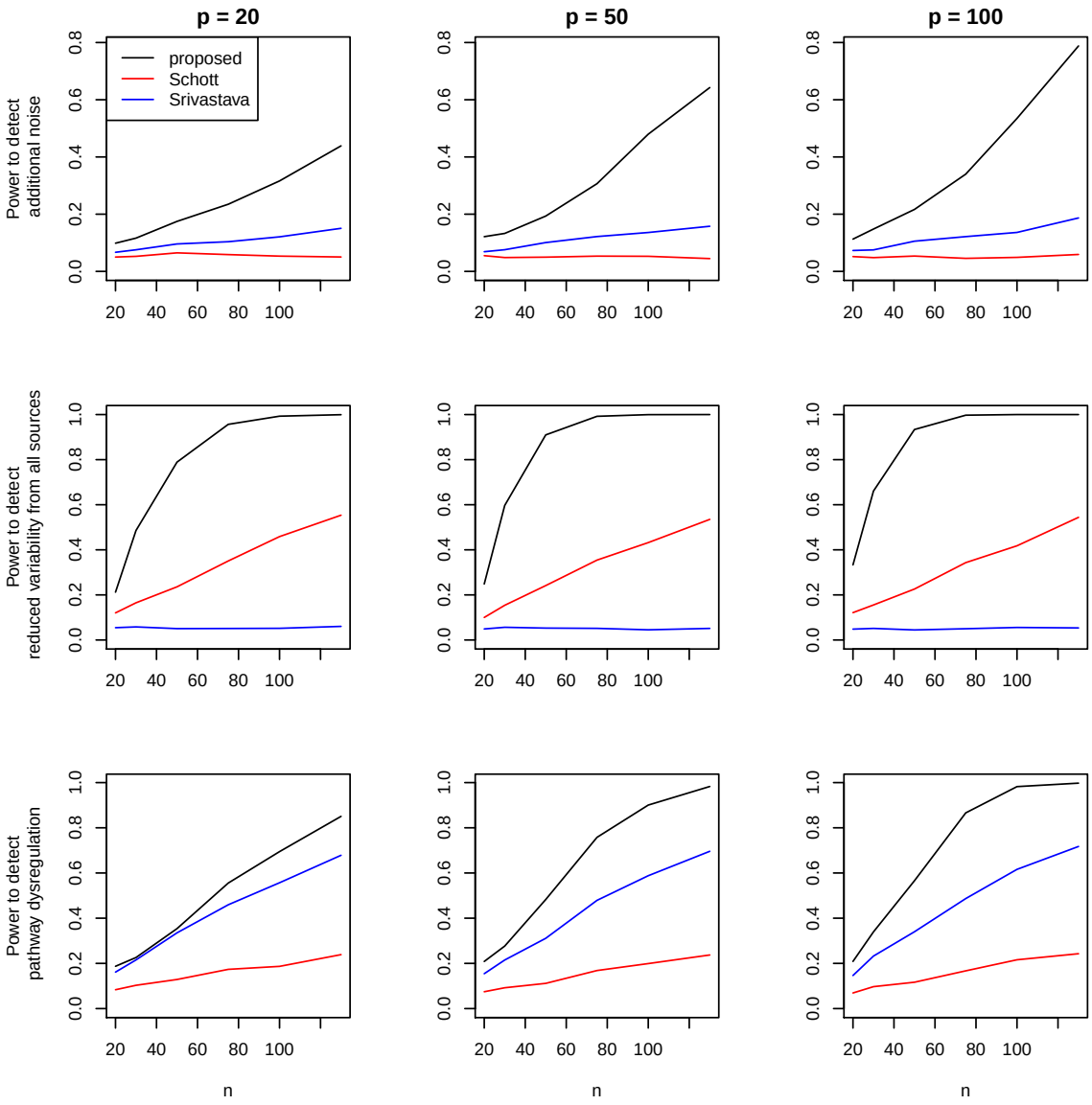

Figure 3.1: *Plots comparing the power of the proposed method (black lines), the method of Schott [2007] (red lines), and the method of Srivastava and Yanagihara [2010] (blue lines).* **First row***: Power under the increased noise setting.* **Second row***: Power under the rescaled covariance setting.* **Third row***: Power under the dysregulation setting.*

matrix differ dramatically between classes. However, changes in the biological quantities we are interested in will most often manifest as widespread, small differences in the covariance matrix, a setting in which these earlier methods perform poorly.

3.5 Application to prostate cancer dataset

We applied our method to a prostate cancer gene expression dataset described in Yu et al. [2004], available on GEO [Barrett et al., 2005] as GDS2545. Their dataset included microarrays from both prostate tumor tissue and healthy prostate tissue adjacent to the tumor from 58 patients. They used Affymetrix HG-U95A arrays to measure 12625 probes for 9461 genes.

We prepared the data for our test as follows. We removed the 67 probes with missing data. In light of the scaling issue described in Section 3.3.6, we sought to ensure that the dataset was on a scale that would yield unspiked pathway eigenvalues approximately equal to 1. To this end, we re-scaled the entire dataset by the inverse of the median standard deviation of genes in the healthy tissue, thereby achieving a median variance of 1 for genes in the healthy tissue and a median variance of 1.144 in the tumor tissue (there was a noticeable trend for genes to be more variable in tumor tissue). After this rescaling, a small number of genes had extreme variances, some as high as 20. To address the concern of single genes dominating the analysis as discussed in Section 3.3.6, we removed the 161 probes with variance greater than 5 in either class. These filters left us with 12397 probes for 9363 genes.

We applied our test to 880 canonical pathways culled by the Broad Institute's Molecular Signatures Database [Subramanian et al., 2005] from KEGG [Kanehisa and Goto, 2000], Reactome [Matthews et al., 2009], Biocarta [Nishimura, 2001], and the Signal Transduction Knowledge Environment (http://stke.sciencemag.org/). Each pathway contained between 5 and 397 genes, with a median of 24 genes, and an

interquartile range of 15 to 48 genes.

Results for top pathways are in Table 3.3. Results for selected, interesting pathways are described in the text and appear in Table 3.4.

Most of the top results appear to be driven by increased trace statistics (T) in tumor tissue, reflecting the trend for this dataset's genes to have greater variability in tumor tissue. This result is somewhat unsurprising, as we might expect gene expression to exhibit less stability in cancer. Simultaneous examination of the L statistic reveals further interesting details. For most of the highly significant pathways, it is also the case that L is noticeably higher in tumor tissue, suggesting that the increased T statistics are at least partly due to increased variability of overall pathway activity. The "ST tumor necrosis factor" pathway typifies this phenomenon: its T statistic is 30% higher in tumor tissue, and its L statistic is 35% higher. As the absolute increase in its L statistic is less than the increase in its T statistic, the pathway's increased variability in tumor tissue cannot be entirely explained by increased variability of pathway activity, perhaps indicating chaotic, dysregulated behavior of pathway genes.

More interestingly, in some pathways, for example "Reactome platelet degranulation," "Reactome post NMDA receptor activation events," "Reactome metabolism of lipids and lipoproteins," and "KEGG type I diabetes mellitus," the increased T statistic in tumor tissue is accompanied by a *smaller* L statistic. For these pathways, we can attribute the greater variability to dysregulation of the pathway, causing greater variability but lower co-regulation. (The shape statistics L/T for these pathways are dramatically higher in healthy tissue.)

A few pathways with low p-values have lower T statistics in tumor tissue. The pathway "Reactome NEF mediated downregulation of MHC class I complex cell surface expression" (p=0.018, 8 probes) has a much higher L statistic in healthy tissue

(7.16 vs. 3.68), and a slightly higher T statistic (16.46 vs. 13.74). This pathway appears to experience much greater co-regulation in healthy cells (shape statistic L/T of 0.44 vs. 0.27) and greater stability of expression in tumors. We might therefore hypothesize that the regulatory mechanisms controlling this pathway have broken down in cancer, leading the pathway to occupy a static activity level and therefore losing the co-expression induced by dynamic pathway activity.

We now turn our attention to biologically pertinent pathways. Three apoptosis-related pathways, "Reactome apoptosis," "Reactome apoptotic cleavage of cell adhesion proteins," and "Reactome apoptotic execution phase," have p-values below 0.05. All three pathways follow the trend of higher T statistics in tumors; however, the "Reactome apoptotic cleavage of cell adhesion proteins" pathway is unique in having a slightly higher L statistic in healthy tissue. Thus we might conclude that this particular component of apoptosis undergoes particuarly dramatic dysregulation. (However the other two apoptosis pathways do have minimal tumor-associated increases in L statistics.) The cancer-related pathways with the lowest p-values are "ST tumor necrosis factor pathway," "Reactome apoptotic cleavage of cell adhesion proteins," and "Reactome cell death signalling via NRAGE NRIF and NADE."

3.6 Discussion

We have described a model for the co-regulation of component genes or proteins of biological pathways. This biological model assumes the eigenvalues of the pathway genes' covariance matrix follow a spiked model, and it suggests biological interpretations for various functions on the sample eigenvalues. We have defined a test to identify changes in these biological properties. Simulations showed our test to have much greater power than existing alternatives to detect these changes; furthermore, our test results have significantly more direct biological interpretations than existing

Table 3.3: Results for the 20 pathways with the lowest p-values

pathway name	dim	p-value	healthy L	tumor L	healthy T	tumor T
Reactome fanconi anemia pathway	12	1.39e-05	2.647	6.417	11.845	19.275
Reactome abortive elongation of HIV1 transcript in the absence of TAT	28	2.32e-05	5.137	6.205	31.215	42.895
Biocarta RAB pathway	19	1.38e-04	2.533	5.699	13.645	19.503
Reactome MTORC1 mediated signalling	13	1.44e-04	3.243	3.871	13.555	18.723
Biocarta TNFR1 pathway	55	2.49e-04	7.415	8.100	57.417	72.310
ST tumor necrosis factor pathway	45	3.92e-04	4.514	6.082	40.387	52.329
Reactome apoptotic cleavage of cell adhesion proteins	13	4.46e-04	3.842	3.725	15.614	20.131
Reactome viral messenger RNA synthesis	19	4.53e-04	3.823	5.550	20.455	28.698
Biocarta HIVNEF pathway	113	4.68e-04	12.488	14.692	118.993	144.613
KEGG RNA polymerase	23	4.75e-04	4.181	5.667	24.284	32.954
KEGG oxidative phosphorylation	99	5.95e-04	10.521	11.114	112.328	132.217
Reactome platelet degranulation	121	9.25e-04	23.045	18.395	172.036	189.709
Reactome influenza viral RNA transcription and replication	101	9.68e-04	15.580	16.707	150.364	175.712
Reactome translation	126	9.86e-04	16.804	16.834	174.976	201.562
Biocarta RHO pathway	40	1.05e-03	7.017	6.569	38.471	47.432
Reactome electron transport chain	54	1.08e-03	6.021	7.680	52.874	66.820
Biocarta SODD pathway	16	1.38e-03	2.641	3.686	12.524	17.653
Reactome metabolism of proteins	212	1.47e-03	25.423	26.424	293.995	337.512
Biocarta NFKB pathway	32	1.58e-03	5.079	6.140	39.390	49.759
KEGG lysine degradation	30	2.12e-03	4.401	8.066	33.601	43.900

Table 3.4: Results for selected pathways discussed in the text

pathway name	dim	p-value	healthy L	tumor L	healthy T	tumor T
Reactome platelet degranulation	121	9.25e-04	23.045	18.395	172.036	189.709
Reactome post NMDA receptor activation events	51	0.013	7.565	6.25	54.692	62.981
Reactome metabolism of lipids and lipoproteins	229	0.013	35.959	30.283	314.603	337.595
KEGG type I diabetes mellitus	61	0.013	10.68	7.354	70.527	76.12
Reactome NEF mediated downregulation of MHC class I complex cell surface expression	8	0.019	7.163	3.676	16.462	13.736
ST tumor necrosis factor pathway	45	3.92e-04	4.514	6.082	40.387	52.329
KEGG colorectal cancer	117	0.040	16.331	19.746	155.104	176.237
KEGG pancreatic cancer	138	0.041	18.217	21.72	183.317	207.765
Reactome apoptosis	182	0.038	20.054	23.56	219.471	245.704
Reactome apoptotic cleavage of cell adhesion proteins	13	4.46e-04	3.842	3.725	15.614	20.131
Reactome apoptotic execution phase	66	0.032	10.052	10.465	77.023	87.683
Reactome cell death signalling via NRAGE NRIF and NADE	68	0.010	11.141	15.861	93.831	112.258
Reactome DNA repair	143	0.036	15.83	19.986	177.771	202.173
Reactome genes involved in apoptotic cleavage of cellular proteins	50	0.027	9.016	8.596	62.902	71.665
Reactome RAS activation upon CA2 infux through NMDA receptor	31	0.024	3.815	4.325	26.408	31.66

methods. Among other quantities of interest, our test can identify changes in the stability of pathway activity, in the strength of co-regulation of pathway genes, and in the level of pathway dysregulation. We demonstrated the utility of our test in a prostate cancer gene expression dataset.

We envision this test to be a powerful complement to traditional, marginal effects-based analyses like Gene Set Enrichment Analysis (GSEA) [Subramanian et al., 2005] or measurements of overall pathway activity levels. Given the high dimension and complex behavior of biological pathways, it seems appropriate to apply a number of analyses focused on different aspects of pathway behavior. A complete analysis of a pathway would include at a minimum a summary of single-gene behavior like GSEA, a comparison of overall pathway activity levels between disease states (see for example Lee et al. [2008]), a test for changes in covariance structure like the method proposed here, and ideally several other tests and descriptive statistics yet to be discovered.

Our test relies on rather strong assumptions arising from our biological model. However, departures from these assumptions may be dealt with relatively easily, as described in Section 3.3.6 and as demonstrated in our analysis of a real gene expression dataset in Section 3.5.

Simulations (not shown) suggest that our test's conservativeness when compared to the quantiles of a χ^2_2 distribution is primarily due to a tendency of formula 3.3.5 to underestimate $\sigma^2_{Q_T}$, the variance of the difference between the Trace statistics. We suspect this phenomenon arises from the poor stability of the empirical covariance matrix when $p \approx n$. This problem may be averted by replacing the maximum likelihood estimates $\hat{\mathbf{\Sigma}}_1$ and $\hat{\mathbf{\Sigma}}_2$ in equation 3.3.5 with regularized estimates such as those described in Friedman et al. [2007b], Peng et al. [2009], or in Chapter 2 of this dissertation.

We are investigating the asymptotic distribution of the statistic $\mathbf{Q} := \begin{pmatrix} L_1 - L_2 - b_L \\ T_1 - T_2 \end{pmatrix}$.

From previous theory [Mathai and Pillai, 1982, Paul, 2007], we know that both the L and T statistics are asymptotically normal. Unfortunately, our bias correction term b_L is a non-linear function of L_1 and L_2, and so we have no reason to suppose it too is asymptotically normal. In the case where $n_1 = n_2$ and therefore $b_L = 0$, we may conclude that $\mathbf{Q}^T\mathbf{\Sigma}_Q^{-1}\mathbf{Q} \rightarrow_d \chi^2_2$. Note that this conclusion depends on the bivariate normality of (L, T), a reasonable proposition which we corroborate with simulated data in Appendix F. In most likely experiments, n_1 and n_2 are reasonably similar, leaving the variance of b_L quite small compared to the variance of $(L_1 - L_2)$ and therefore only minimally distorting the asymptotic normality of $(L_1 - L_2 - b_L)$. Thus we anticipate that $\mathbf{Q}^T\mathbf{\Sigma}_Q^{-1}\mathbf{Q}$ will converge to something very close to a χ^2_2 distribution even when $n_1 \neq n_2$.

A useful extension of this work might be quickly achievable. In cases where a pathway participates in more than one biological process (this would especially be the case for disease-related pathways, for example the KEGG pancreatic cancer pathway), the first eigenvalue might be inadequate to capture variability due to pathway co-regulation. In other cases, the first eigenvalue may do an unsatisfying job of capturing non-linear relationships between pathway activity and expression of pathway genes. A solution to these concerns would be to re-define the L statistic to be the sum of the first $m \ll p$ eigenvalues rather than just the first eigenvalue. So long as the first m population eigenvalues were spikes, the asymptotic distribution of the L_m statistic would have a form very similar to our simple L statistic, it could be simple to derive a test for:

$$H_0 : \begin{pmatrix} \lambda_{1,1} + \ldots + \lambda_{1,m} \\ \text{trace}\mathbf{\Sigma}_1 \end{pmatrix} = \begin{pmatrix} \lambda_{2,1} + \ldots + \lambda_{2,m} \\ \text{trace}\mathbf{\Sigma}_2 \end{pmatrix},$$

where $\lambda_{k,m}$ is the m^{th} eigenvalue for class k. Alternatively, if the first m eigenvalues

are expected to contain information on separate biological quantities, a test for

$$H_0 : \begin{pmatrix} \lambda_{1,1} \\ \vdots \\ \lambda_{1,m} \\ \text{trace}\mathbf{\Sigma}_1 \end{pmatrix} = \begin{pmatrix} \lambda_{2,1} \\ \vdots \\ \lambda_{2,m} \\ \text{trace}\mathbf{\Sigma}_2 \end{pmatrix}$$

could be derived.

Another useful extension would be the development of tests for differences in more targeted quantities than the somewhat broad (L, T). Our current test rejects the null hypothesis in the face of a variety of changes to the eigenstructure of the covariance matrix, and further investigation is required to determine exactly which quantities have changed. Upon rejecting $H_0 : \begin{pmatrix} \alpha_1 \\ \text{trace}\mathbf{\Sigma}_1 \end{pmatrix} = \begin{pmatrix} \alpha_2 \\ \text{trace}\mathbf{\Sigma}_2 \end{pmatrix}$, we might test for changes in L and T separately. These tests would be trivial to derive. More interestingly, we could test for differences in meaningful functions on (L, T). A test for changes in $(T - L)$ could be considered to directly look for increased dysregulation, or non-co-regulated variability, between classes. Similarly, a test for changes in the shape statistic (L/T) could reflect changes in the strength of co-regulation experienced by pathway genes. Given the asymptotic normality of T and L, these tests could be straightforward to derive.

Chapter 4
CONCLUSIONS

4.1 Summary

To summarize the advances described in this dissertation, consider researchers presented with a novel microarray dataset with observations from disease patients and healthy controls. Prior to the introduction of the Joint Graphical Lasso, attempts at network estimation using this dataset would be unsatisfying. Our researchers could apply network estimation techniques like those described in Friedman et al. [2007b] and Peng et al. [2009] to the data from each class, yielding two network estimates with very little overlap, when in truth most network edges are unchanged in disease. Or they could apply a network estimation technique to the entire dataset, yielding a single network estimate that is mute to the important ways network structure does change in disease. The Joint Graphical Lasso allows for the first time network estimates that identify changes between healthy and disease networks while still reflecting our belief that network structure will be largely preserved across disease states. Although previous work [Guo et al., 2011] allowed for joint estimation of multiple graphical models, JGL has a highly flexible and convex penalty, and is the first algorithm with the computational efficiency to allow analysis of an entire microarray dataset.

This microarray dataset would have obvious utility in the search for genes and pathways that change in disease. The researchers would of course examine each gene for significant changes in mean expression in disease, and they would perform gene set enrichment analysis or a similar technique to look for known biological pathways with widespread changes in mean gene expression. But their examination of biological

pathways would be terribly incomplete if they did not look for changes in the interactive behavior of pathway genes. To this end, they might apply Eddy et al. [2010]s test for dysregulation, or Schott [2007] or Srivastava and Yanagihara [2010]'s tests for equality of covariance matrices. However, if the dataset is of reasonable quality, Eddy et al. [2010]s non-parametric approach would throw out a great deal of useful information. And the tests like those of Schott [2007] and Srivastava and Yanagihara [2010] have excessively strong null hypotheses, making interpretation almost impossible and failing to focus on biologically meaningful changes in the covariance matrix. Our new test allows the researchers to test for changes in highly biologically relevant and previously ignored aspects of covariance structure. Compared to existing methods, our new test allows much more direct biological interpretation and has superior power to detect a number of biologically interesting changes.

4.2 Future directions

There are of course many other statistical goals our investigators may pursue using this dataset. To begin with, they may wish to create a classifier to determine the disease state of future patients. To this end, the Joint Graphical Lasso has potential to improve the utility of the classification method Quadratic Discriminant Analysis (QDA). Unaltered, algorithms like QDA that make use of covariance information suffer as dimension increases, making their use ill-advised. When dimension increases, QDA suffers both from the poor stability of covariance estimation when $n \approx p$ and from the overfitting that results from including $p(p-1)/2$ additional parameters in a classifier. Methods like [Friedman, 1989] and [Simon and Tibshirani, 2011] make some effort to regularize the covariance portion of classifiers. A similar and more flexible strategy would be to use JGL to estimate the covariance of the features in each class. In this approach, varying the sparsity penalty λ_1 would control how aggressively the

covariance matrices were fit to the classes, and varying the similarity penalty λ_2 would control how aggressively differences in covariance were used to separate the classes.

Additionally, our investigators may wish to cluster the datasets' genes or observations. The application of JGL to QDA can be immediately extended to unsupervised learning: in model-based clustering, the M step of the EM algorithm performs a QDA-based classifier. Thus the JGL-dependent QDA method described above could power the model-based clustering of observations using a greater dimension that would have previously been reasonable.

In the same spirit as the above advances, penalized inverse covariance estimation has tremendous promise in any method where the stability of covariance estimates is a concern. For example, in hierarchical clustering of observations in high-dimensional biological datasets we traditionally use a simple Euclidean distance metric, which assumes the features have covariance $\mathbf{I}$. However, by using a regularized estimate of the features' covariance matrix to define a Mahalanobis distance metric, we can more accurately incorporate information about covariance between the features into hierarchical clustering, improving performance without spending excessive degrees of freedom.

In examination of biological pathways, penalized inverse covariance estimation promises an alternative to Gene Set Enrichment Analysis. Fisher's method, a standard technique for combining p-values from multiple independent tests, models the sum of p independent $-2\log$ p-values as a χ^2_p random variable. In a biological pathway, however, p-values or test statistics for genes will often be correlated. By stably estimating the covariance matrix of pathway genes using penalized methods, we may be able to combine genes' test statistics appropriately. We could thereby expand the tractable dimension of methods like Kost and McDermott [2002]. A similar approach has been described in a specific setting in Chen et al. [2011].

Of course, there are diverse penalties that could be applied to inverse covariance estimation. The Lasso is highly appropriate when sparse network estimation is a priority; however, for the above uses, it is likely an entirely different penalty may have better performance. The field would benefit a great deal from a simple investigation of the success of a range of penalties in returning regularized $\mathbf{\Sigma}$ estimates that best capture the true $\mathbf{\Sigma}$.

BIBLIOGRAPHY

A. Adjei. K-ras as a target for lung cancer therapy. *Journal of Thoractic Oncology*, 3(6):S160–S163, 2008.

A. Ahmed and E.P. Xing. Tesla: Recovering time-varying networks of dependencies in social and biological studies. *Proc. Natl. Acad. Sci.*, 29:11878–11883, 2009.

Z. Bai and J. Yao. Central limit theorems for eigenvalues in a spiked population model. *Annales de l'Institut Henri Poincare - Probabilites et Statistiques*, 44(3): 447–474, 2008.

Z. D. Bai and Y. Q. Yin. Limit of the smallest eigenvalue of a large dimensional sample covariance matrix. *Annals of Probability*, 21:1275–1294, 1993.

J. Baik and J. Silverstein. Eigenvalues of large sample covariance matrices of spiked population models. *Journal of Multivariate Analysis*, 97:16431697, 2006.

T. Barrett, T.O. Suzek, D.B. Troup, S.E. Wilhite, W.C. Ngau, P. Ledoux, D. Rudnev, A.E. Lash, W. Fujibuchi, and R. Edgar. NCBI GEO: mining millions of expression profiles–database and tools. *Nucleic Acids Research*, 33:D562–D566, 2005.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

S.P. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2010.

H. Chen and B.M. Sharp. Content-rich biological network constructed by mining pubmed abstracts. *BMC Bioinformatics*, 5:147, 2004.

L.S. Chen, D. Paul, R.L. Prentice, and P. Wang. A regularized hotellings t 2 test for pathway analysis in proteomic studies. *Journal of the American Statistical Association*, 106(496):1345–1360, 2011.

M. Drton and M.D. Perlman. Model selection for Gaussian concentration graphs. *Biometrika*, 91 (3):591–602, 2004.

M. Drton and M.D. Perlman. A SINful approach to Gaussian graphical model selection. *Journal of Statistical Planning and Inference*, 138 (4):1179–1200, 2007.

J.A. Eddy, L. Hood, N.D. Price, and D. Geman. Identifying tightly regulated and variably expressed networks by differential rank conservation. *PLoS Computational Biology*, 6(5), 2010.

D.M. Edwards. *Introduction to Graphical Modelling*. Springer, New York, 2000.

N. El Karoui. Spectrum estimation for large dimensional covariance matrices using random matrix theory. *ArXiv Mathematics e-prints*, 2006.

N. El Karoui. Tracy-Widom limit for the largest eigenvalue of a large class of complex sample covariance matrices. *Annals of Probability*, 35 (2):663–714, 2007.

J. Friedman. Regularized discriminant analysis. *Journal of the American Statistical Association*, 84:165–175, 1989.

J. Friedman, T. Hastie, H. Hoefling, and R. Tibshirani. Pathwise coordinate optimization. *Annals of Applied Statistics*, 1:302–332, 2007a.

J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9:432–441, 2007b.

J. Friedman, T. Hastie, and R. Tibshirani. A note on the group lasso and a sparse group lasso. *Technical report, Department of Statistics, Stanford University*, 2010.

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

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning; Data Mining, Inference and Prediction*. Springer Verlag, New York, 2009.

A. Hendrikse, L. Spreeuwers, and R. Veldhuis. A bootstrap approach to eigenvalue correction. *International Conference on Data Mining*, page 818823, 2009.

L.D. Hocking, A. Joulin, and F. Bach. Clusterpath: An algorithm for clustering using convex fusion penalties. *Proceedings of the 28th International Conference on Machine Learning*, 2011.

H. Hoefling. Personal communication, 2010a.

H. Hoefling. A path algorithm for the fused lasso signal approximator. *Journal of Computational and Graphical Statistics*, 19(4):984–1006, 2010b.

K. Johansson. Shape fluctuations and random matrices. *Communications in Mathematical Physics*, 209(2):437–476, 2000.

D.E. Johnson and F.A. Graybill. An analysis of a two-way model with interaction and no replication. *Journal of the American Statistical Association*, 67:862–868, 1972.

I.M. Johnstone and F.A. Graybill. On the distribution of the largest eigenvalue in principal component analysis. *Annals of Statistics*, 29:295–327, 2001.

M. Kanehisa and S. Goto. KEGG: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Research*, 28(1):27–30, 2000.

M. Kolar and E.P. Xing. Sparsistent estimation of time-varying discrete markov random fields. *Manuscript, arXiv:0907.2337*, 2009.

M. Kolar, L. Song, A. Ahmed, and E.P. Xing. Estimating time-varying networks. *Annals of Applied Statistics*, 4 (1):94–123, 2010.

J.T. Kost and M.P. McDermott. Combining dependent p-values. *Statistics & probability letters*, 60(2):183–190, 2002.

S.L. Lauritzen. *Graphical Models*. Oxford Science Publications, 1996.

E. Lee, H.Y. Chuang, J.W. Kim, T. Ideker, and D. Lee. Inferring pathway activity toward precise disease classification. *PLoS Computational Biology*, 4(11):e1000217, 2008.

S. Li, L. Hsu, J. Peng, and P. Wang. Bootstrap inference for network construction. *Manuscript, arXiv:1111.5028v1*, 2011.

Vladimir A Marčenko and Leonid Andreevich Pastur. Distribution of eigenvalues for some sets of random matrices. *Sbornik: Mathematics*, 1(4):457–483, 1967.

A.M. Mathai and K.C.S. Pillai. Further results on the trace of a noncentral wishart matrix. *Communications in Statistics (Theory and Methods)*, 11 (10):1077–1086, 1982.

L. Matthews, G. Gopinath, M. Gillespie, M. Caudy, D. Croft, B. de Bono, P. Garapati, J. Hemish, H. Hermjakob, and B. Jassal. Reactome knowledgebase of human biological pathways and processes. *Nucleic Acids Research*, 37:D617–D622, 2009.

R. Mazumder and T. Hastie. Exact covariance-thresholding into connected components for large-scale graphical lasso. *Submitted*, 2012.

M. Meinshausen and P. Buhlmann. Stability selection (with discussion). *Journal of the Royal Statistical Society, Series B*, 72:417–473, 2010.

N. Meinshausen. Relaxed lasso. *Computational Statistics and Data Analysis*, 52: 374–393, 2007.

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

K. Mohan, M. Chung, S. Han, D.M. Witten, S.I. Lee, and M. Fazel. Structured learning of Gaussian graphical models. *Advances in Neural Information Processing Systems*, 2012.

B. Nadler. On the distribution of the ratio of the largest eigenvalue to the trace of a Wishart matrix. *Journal of Multivariate Analysis*, 102(2):363–371, 2011.

D. Nishimura. Biocarta. *Biotech Software and Internet Report*, 2:117–120, 2001.

D. Paul. Aymptotics of sample eigenstructure for a large dimension spiked covariance model. *Statistica Sinica*, 17:1617–1642, 2007.

J. Peng, P. Wang, N. Zhou, and J. Zhu. Partial correlation estimation by joint sparse regression model. *Journal of the American Statistical Association*, 104(486):735–746, 2009.

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

S.N. Roy. On a heuristic method of test construction and its use in multivariate analysis. *Annals of Mathematical Statistics*, 24(2):220–238, 1953.

J.R. Schott. A test for the equality of covariance matrices when the dimension is large relative to the sample size. *Computational Statistics and Data Analysis*, 51: 6535–6542, 2007.

N. Simon and R. Tibshirani. Discriminant analysis with adaptively pooled covariance. *Manuscript, arXiv:1111.1687*, 2011.

L. Song, M. Kolar, and E.P. Xing. Keller: Estimating time-evolving interactions between genes. *Bioinformatics*, 25 (12):i128–i136, 2009a.

L. Song, M. Kolar, and E.P. Xing. Time-varying dynamic bayesian networks. *Proceeding of the 23rd Neural Information Processing Systems*, 2009b.

A. Spira, JE. Beane, V. Shah, K. Steiling, G. Liu, F. Schembri, S. Gilman, YM. Dumas, P. Calner, P. Sebastiani, S. Sridhar, J. Beamis, C. Lamb, T. Anderson, N. Gerry, J. Keane, M.E. Lenburg, and J.S. Brody. Airway epithelial gene expression in the diagnostic evaluation of smokers with suspect lung cancer. *Nature Medicine*, 13(3):361–366, 2007.

M.S. Srivastava and H. Yanagihara. Testing the equality of several covariance matrices with fewer observations than the dimension. *Journal of Multivariate Analysis*, 101: 1319–1329, 2010.

A. Subramanian, P. Tamayo, V.K. Mootha, S. Mukherjee, B.L. Ebert, M.A. Gillette, A. Paulovich, S. Pomeroy, T.R. Golub, E.S. Lander, and J.P. Mesirov. Gene set

enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *PNAS*, 104(43):1554515550, 2005.

R.E. Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1(2):146–160, 1972.

R. Tibshirani. Regression shrinkage and selection via the lasso. *J. Royal. Statist. Soc. B.*, 58:267–288, 1996.

R. Tibshirani. Personal communication, 2012.

R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight. Sparsity and smoothness via the fused lasso. *J. Royal. Statist. Soc. B.*, 67:91–108, 2005.

D.M. Witten and R. Tibshirani. Covariance-regularized regression and classification for high-dimensional problems. *J. Royal. Stat. Soc. B.*, 71(3):615–636, PMCID:PMC2806603, 2009.

D.M. Witten, J.H. Friedman, and N. Simon. New insights and faster computations for the graphical lasso. *Journal of Computational and Graphical Statistics*, 20(4): 892–900, 2011.

Y.P. Yu, D. Landsittel, L. Jing, J. Nelson, B. Ren, L. Liu, C. McDonald, R. Thomas, R. Dhir, S. Finkelstein, G. Michalopoulos, M. Becich, and J.H. Luo. Gene expression alterations in prostate cancer predicting tumor aggression and preceding development of malignancy. *Journal of Clinical Oncology*, 22(14):2790–2799, 2004.

M. Yuan. Efficient computation of ℓ_1 regularized estimates in Gaussian graphical models. *Journal of Computational and Graphical Statistics*, 17(4):809–826, 2008.

M. Yuan and Y. Lin. Model selection and estimation in the Gaussian graphical model. *Biometrika*, 94(10:19–35, 2007a.

M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society, Series B*, 68:49–67, 2007b.

H. Zhou, W. Pan, and X. Shen. Penalized model-based clustering with unconstrained covariance matrices. *Electronic journal of statistics*, 3:1473, 2009.

S. Zhou, J. Lafferty, and L. Wasserman. Time varying undirected graphs. *The 21st Annual Conference on Learning Theory (COLT 2008), Helsinki, Finland*, 2008.

Appendix A

MODIFYING JGL TO WORK ON THE SCALE OF PARTIAL CORRELATIONS

Under some circumstances, it may be preferable to encourage the K networks to have shared partial correlations, rather than shared precision matrices. Below, we describe a simple approach for extending our FGL proposal to work on the scale of partial correlations. A similar approach can be taken to extend GGL. The extension relies on two insights.

1. $\rho_{ij} = -\frac{\sigma^{ij}}{\sqrt{\sigma^{ii}\sigma^{jj}}}$, where ρ_{ij} is the true partial correlation between the ith and jth features, and where σ^{ij} is the (i,j)th entry of the true precision matrix.

2. The algorithm for solving the FGL optimization problem can easily be modified to make use of the following penalty function:

$$P(\{\mathbf{\Theta}\}) = \sum_{k=1}^{K}\sum_{i\neq j}\lambda_{1,ij}|\theta_{ij}^{(k)}| + \sum_{k<k'}\sum_{i,j}\lambda_{2,ij}|\theta_{ij}^{(k)} - \theta_{ij}^{(k')}|, \tag{A.0.1}$$

where $\lambda_{t,ij} = \lambda_t/\sqrt{\hat{\sigma}^{ii}\hat{\sigma}^{jj}}$, $t = 1,\ 2$, and where $\hat{\sigma}^{ii}$ is an estimate of the ith diagonal element of the K precision matrices. (Here, we assume the K precision matrices have shared diagonal elements.) The estimate $\{\hat{\sigma}^{ii}\}$ can be obtained in a number of ways, for instance by performing the graphical lasso on the samples from all K data sets together. Then this approach will effectively result in applying a generalized fused lasso penalty to the partial correlations for the K classes.

Appendix B

PROOFS OF THEOREMS SUPPORTING COMPUTATIONAL IMPROVEMENTS TO JGL

Preliminaries to Proofs of Theorems 1 and 2

We begin with a few comments on subgradients. The subgradient of $|\theta_{ij}^{(k)}|$ with respect to $\theta_{ij}^{(k)}$ equals

$$\begin{cases} 1 & \text{if } \theta_{ij}^{(k)} > 0 \\ -1 & \text{if } \theta_{ij}^{(k)} < 0 \\ a & \text{if } \theta_{ij}^{(k)} = 0 \end{cases},$$

for some $a \in [-1, 1]$. The subgradient of $|\theta_{ij}^{(k)} - \theta_{ij}^{(k')}|$ with respect to $(\theta_{ij}^{(k)}, \theta_{ij}^{(k')})$ for $k \neq k'$ equals $(d, -d)$, where

$$d = \begin{cases} 1 & \text{if } \theta_{ij}^{(k)} > \theta_{ij}^{(k')} \\ -1 & \text{if } \theta_{ij}^{(k)} < \theta_{ij}^{(k')} \\ a & \text{if } \theta_{ij}^{(k)} = \theta_{ij}^{(k')} \end{cases},$$

for some $a \in [-1, 1]$. Finally, the subgradient of $\sqrt{\sum_{k=1}^{K} (\theta_{ij}^{(k)})^2}$ with respect to $(\theta_{ij}^{(1)}, \ldots, \theta_{ij}^{(K)})$ is given by

$$\begin{cases} (\theta_{ij}^{(1)}, \ldots, \theta_{ij}^{(K)}) / \sum_{k=1}^{K} (\theta_{ij}^{(k)})^2 & \text{if } \sum_{k=1}^{K} (\theta_{ij}^{(k)})^2 > 0 \\ (\Upsilon_{1,ij}, \ldots, \Upsilon_{K,ij}) & \text{if } \theta_{ij}^{(1)} = \ldots = \theta_{ij}^{(K)} = 0, \end{cases}$$

for some $\Upsilon_{1,ij}, \ldots, \Upsilon_{K,ij}$ such that $\sum_{k=1}^{K} \Upsilon_{k,ij}^2 \leq 1$.

To prove Theorem 1, we will use the following lemma.

Lemma B.0.1. *The following two sets of conditions are equivalent:*

(A): $|n_1 S_1| \leq \lambda_1 + \lambda_2$, $|n_2 S_2| \leq \lambda_1 + \lambda_2$, *and* $|n_1 S_1 + n_2 S_2| \leq 2\lambda_1$.

(B): There exist $\Gamma_1, \Gamma_2, \Upsilon \in [-1, 1]$ *such that* $-n_1 S_1 - \lambda_1 \Gamma_1 - \lambda_2 \Upsilon = 0$, *and* $-n_2 S_2 - \lambda_1 \Gamma_2 + \lambda_2 \Upsilon = 0$.

Proof. We will begin by proving that (B) implies (A), and will then prove that (A) implies (B).
Proof that (B) $\Rightarrow$ (A):

First of all, $-n_1 S_1 - \lambda_1 \Gamma_1 - \lambda_2 \Upsilon = 0$ implies that $|n_1 S_1| \leq \lambda_1 + \lambda_2$, since $\Gamma_1, \Upsilon \in [-1, 1]$. Similarly, $-n_2 S_2 - \lambda_1 \Gamma_2 + \lambda_2 \Upsilon = 0$ implies that $|n_2 S_2| \leq \lambda_1 + \lambda_2$. Finally, summing the two equations in (B) reveals that $n_1 S_1 + n_2 S_2 = -\lambda_1(\Gamma_1 + \Gamma_2)$, which implies that $|n_1 S_1 + n_2 S_2| \leq 2\lambda_1$.

Proof that (A) $\Rightarrow$ (B):

Without loss of generality, assume that $n_1 S_1 \geq n_2 S_2$. We split the proof into two cases.

1. *Case 1:* $n_1 S_1 - n_2 S_2 < 2\lambda_2$.

   Let $\Gamma_1 = \Gamma_2 = \frac{-n_1 S_1 - n_2 S_2}{2\lambda_1}$, and $\Upsilon = \frac{-n_1 S_1 + n_2 S_2}{2\lambda_2}$.

   First, note that by (A), we know that $|n_1 S_1 + n_2 S_2| \leq 2\lambda_1$. Therefore, $\Gamma_1, \Gamma_2 \in [-1, 1]$. Second, note that Case 1's assumption that $n_1 S_1 - n_2 S_2 < 2\lambda_2$ implies that $\Upsilon \in [-1, 1]$. Finally, we see by inspection that $-n_1 S_1 - \lambda_1 \Gamma_1 - \lambda_2 \Upsilon = 0$, and $-n_2 S_2 - \lambda_1 \Gamma_2 + \lambda_2 \Upsilon = 0$.

2. *Case 2:* $n_1S_1 - n_2S_2 \geq 2\lambda_2$.

   Let $\Gamma_1 = \frac{-n_1S_1+\lambda_2}{\lambda_1}$, $\Gamma_2 = \frac{-n_2S_2-\lambda_2}{\lambda_1}$, and $\Upsilon = -1$. Then, by inspection, $-n_1S_1 - \lambda_1\Gamma_1 - \lambda_2\Upsilon = 0$, and $-n_2S_2 - \lambda_1\Gamma_2 + \lambda_2\Upsilon = 0$.

   It remains to show that $\Gamma_1, \Gamma_2, \Upsilon \in [-1, 1]$. Trivially, $\Upsilon = -1 \in [-1, 1]$. From our assumption that $|n_1S_1| \leq \lambda_1 + \lambda_2$, we know that $-1 \leq \Gamma_1$. Moreover, by the assumptions that $n_1S_1 - n_2S_2 \geq 2\lambda_2$ and $|n_1S_1 + n_2S_2| \leq 2\lambda_1$, we have that

$$\Gamma_1 = \frac{-n_1S_1 + \lambda_2}{\lambda_1} \leq \frac{-n_1S_1 + \lambda_2\left(\frac{n_1S_1 - n_2S_2}{2\lambda_2}\right)}{\lambda_1} = \frac{-n_1S_1 - n_2S_2}{2\lambda_1} \leq 1. \quad \text{(B.0.1)}$$

   Therefore $\Gamma_1 \in [-1, 1]$.

   By the assumption that $|n_2S_2| \leq \lambda_1 + \lambda_2$, we know that $\Gamma_2 = \frac{-n_2S_2-\lambda_2}{\lambda_1} \leq 1$. From the assumptions that $n_1S_1 - n_2S_2 \geq 2\lambda_2$ and $|n_1S_1 + n_2S_2| \leq 2\lambda_1$, we have that

$$\Gamma_2 = \frac{-n_2S_2 - \lambda_2}{\lambda_1} \geq \frac{-n_2S_2 - \lambda_2\left(\frac{n_1S_1 - n_2S_2}{2\lambda_2}\right)}{\lambda_1} = \frac{-n_1S_1 - n_2S_2}{2\lambda_1} \geq -1. \quad \text{(B.0.2)}$$

   Therefore $\Gamma_2 \in [-1, 1]$.

Thus we conclude (A) $\Rightarrow$ (B), and our proof of Lemma B.0.1 is complete. □

We will make use of the following lemma in order to prove Theorem 2.

Lemma B.0.2. *The following two conditions are equivalent:*

(A): There exist scalars $a_1, \ldots, a_K$ *such that* $\sum_{k=1}^{K} a_k^2 \leq 1$ *and* $n_k|S_k| \leq \lambda_1 + \lambda_2 a_k$ *for all* $k = 1, \ldots, K$.

(B): There exist scalars $\Gamma_1, \ldots, \Gamma_K \in [-1, 1]$ *and* $\Upsilon_1, \ldots, \Upsilon_K$ *such that* $\sum_{k=1}^{K} \Upsilon_k^2 \leq 1$ *and* $n_kS_k + \lambda_1\Gamma_k + \lambda_2\Upsilon_k = 0$ *for* $k = 1, \ldots, K$.

Proof. We will begin by proving that (B) implies (A), and will then show that (A) implies (B).

Proof that $(B) \Rightarrow (A)$*:*

By (B), $n_k|S_k| = |\lambda_1\Gamma_k + \lambda_2\Upsilon_k| \leq \lambda_1|\Gamma_k| + \lambda_2|\Upsilon_k| \leq \lambda_1 + \lambda_2|\Upsilon_k|$. Letting $a_k = |\Upsilon_k|$, the result holds.

Proof that $(A) \Rightarrow (B)$*:*

Let Γ_k and Υ_k take the following forms, for $k = 1, \ldots, K$:

$$\Gamma_k = \begin{cases} -1 & \text{if } \ n_k S_k > \lambda_1 \\ -n_k S_k/\lambda_1 & \text{if } \ -\lambda_1 < n_k S_k < \lambda_1 \\ 1 & \text{if } \ n_k S_k < -\lambda_1 \end{cases} \tag{B.0.3}$$

$$\Upsilon_k = \begin{cases} (-n_k S_k + \lambda_1)/\lambda_2 & \text{if } \ n_k S_k > \lambda_1 \\ 0 & \text{if } \ -\lambda_1 < n_k S_k < \lambda_1 \\ (-n_k S_k - \lambda_1)/\lambda_2 & \text{if } \ n_k S_k < -\lambda_1 \end{cases} \tag{B.0.4}$$

First of all, we note by inspection that $\Gamma_k \in [-1, 1]$ and that $n_k S_k + \lambda_1\Gamma_k + \lambda_2\Upsilon_k = 0$ for $k = 1, \ldots, K$. It remains to show that $\sum_{k=1}^{K} \Upsilon_k^2 \leq 1$. Specifically, we will show that $\Upsilon_k^2 \leq a_k^2$ for $k = 1, \ldots, K$. To see why this is the case, note that if $-\lambda_1 < n_k S_k < \lambda_1$ then $0 = \Upsilon_k^2 \leq a_k^2$. And if $n_k S_k > \lambda_1$ or $n_k S_k < -\lambda_1$, then $\Upsilon_k^2 = \left(\frac{n_k|S_k| - \lambda_1}{\lambda_2}\right)^2 \leq a_k^2$. $\square$

Proof of Theorem 1

We first consider the claim for the case $K = 2$. By the Karush-Kuhn-Tucker [KKT; see e.g. Boyd and Vandenberghe, 2004] conditions, a necessary and sufficient set of conditions for $\{\mathbf{\Theta}\}$ to be the solution to the JGL problem is that

$$\begin{aligned} 0 &= n_1(\mathbf{\Theta}^{(1)})^{-1} - n_1\mathbf{S}^{(1)} - \lambda_1\mathbf{\Gamma}_1 - \lambda_2\mathbf{\Upsilon} \\ 0 &= n_2(\mathbf{\Theta}^{(2)})^{-1} - n_2\mathbf{S}^{(2)} - \lambda_1\mathbf{\Gamma}_2 + \lambda_2\mathbf{\Upsilon}, \end{aligned} \tag{B.0.5}$$

where $\Gamma_{1,ij}$ is the subgradient of $|\theta_{ij}^{(1)}|$ with respect to $\theta_{ij}^{(1)}$, $\Gamma_{2,ij}$ is the subgradient of $|\theta_{ij}^{(2)}|$ with respect to $\theta_{ij}^{(2)}$, and Υ_{ij} is the subgradient of $|\theta_{ij}^{(1)} - \theta_{ij}^{(2)}|$ with respect to $\theta_{ij}^{(1)}$.

Let C_1 and C_2 be a partition of the p variables into two nonoverlapping sets, with $C_1 \cap C_2 = \emptyset$, $C_1 \cup C_2 = \{1, \ldots, p\}$. Consider the matrices

$$\mathbf{\Theta}^{(1)} = \begin{pmatrix} \mathbf{\Theta}_1^{(1)} & 0 \\ 0 & \mathbf{\Theta}_2^{(1)} \end{pmatrix}, \quad \mathbf{\Theta}^{(2)} = \begin{pmatrix} \mathbf{\Theta}_1^{(2)} & 0 \\ 0 & \mathbf{\Theta}_2^{(2)} \end{pmatrix}, \tag{B.0.6}$$

where $\mathbf{\Theta}_1^{(1)}$ and $\mathbf{\Theta}_1^{(2)}$ solve the JGL problem on the features in C_1, and $\mathbf{\Theta}_2^{(1)}$ and $\mathbf{\Theta}_2^{(2)}$ solve the JGL problem on the features in C_2. By inspection of (B.0.5), $\mathbf{\Theta}^{(1)}$ and $\mathbf{\Theta}^{(2)}$ solve the entire JGL optimization problem if and only if for all $i \in C_1$, $j \in C_2$, there exist $\Gamma_{1,ij}, \Gamma_{2,ij}, \Upsilon_{ij} \in [-1, 1]$ such that

$$\begin{aligned} -n_1 S_{ij}^{(1)} - \lambda_1\Gamma_{1,ij} - \lambda_2\Upsilon_{ij} &= 0 \\ -n_2 S_{ij}^{(2)} - \lambda_1\Gamma_{2,ij} + \lambda_2\Upsilon_{ij} &= 0. \end{aligned} \tag{B.0.7}$$

Therefore, by Lemma B.0.1, the proof of the claim for the case $K = 2$ is complete.

The derivation of the sufficient condition for the case $K > 2$ is simple and we omit

it here.

Proof of Theorem 2

We note that Theorem 2's condition (2.4.19) is equivalent to the following:

$$|n_k S_{ij}^{(k)}| \leq \lambda_1 + \lambda_2 a_{ij,k} \text{ for all } i \in C_1,\ j \in C_2,\ k = 1, \ldots, K \tag{B.0.8}$$

where $a_{ij,1}, \ldots, a_{ij,K}$ are scalars that satisfy $\sum_{k=1}^{K} a_{ij,k}^2 \leq 1$. We will prove that (B.0.8) is necessary and sufficient for the variables in C_1 to be completely disconnected from those in C_2 in each of the resulting network estimates.

By the KKT conditions, a necessary and sufficient set of conditions for $\{\mathbf{\Theta}\}$ to be the solution to the JGL problem is that

$$0 = n_k(\mathbf{\Theta}^{(k)})^{-1} - n_k\mathbf{S}^{(k)} - \lambda_1\mathbf{\Gamma}_k - \lambda_2\mathbf{\Upsilon}_k \tag{B.0.9}$$

for $k = 1, \ldots, K$. In (B.0.9), $\Gamma_{k,ij}$ is the subgradient of $|\theta_{ij}^{(k)}|$ with respect to $\theta_{ij}^{(k)}$, and $(\Upsilon_{1,ij}, \ldots, \Upsilon_{K,ij})$ is the subgradient of $\sqrt{\sum_{k=1}^{K}(\theta_{ij}^{(k)})^2}$ with respect to $(\theta_{ij}^{(1)}, \ldots, \theta_{ij}^{(K)})$.

Let C_1 and C_2 be a partition of the p variables into two nonoverlapping sets, with $C_1 \cap C_2 = \emptyset$, $C_1 \cup C_2 = \{1, \ldots, p\}$. Consider the matrices of the form

$$\mathbf{\Theta}^{(k)} = \begin{pmatrix} \mathbf{\Theta}_1^{(k)} & 0 \\ 0 & \mathbf{\Theta}_2^{(k)} \end{pmatrix} \tag{B.0.10}$$

for $k = 1, \ldots, K$, where $\mathbf{\Theta}_1^{(1)}, \ldots, \mathbf{\Theta}_1^{(K)}$ solve the JGL problem on the features in C_1, and $\mathbf{\Theta}_2^{(1)}, \ldots, \mathbf{\Theta}_2^{(K)}$ solve the JGL problem on the features in C_2. By inspection of (B.0.9), $\mathbf{\Theta}^{(1)}, \ldots, \mathbf{\Theta}^{(K)}$ solve the entire JGL optimization problem if and only if for

all $i \in C_1$, $j \in C_2$, there exist $\Gamma_{1,ij}, \ldots, \Gamma_{K,ij} \in [-1, 1]$ and $\Upsilon_{1,ij}, \ldots, \Upsilon_{K,ij}$ satisfying $\sum_{k=1}^{K} \Upsilon_{k,ij}^2 \leq 1$ such that

$$-n_k S_{ij}^{(k)} - \lambda_1 \Gamma_{k,ij} - \lambda_2 \Upsilon_{k,ij} = 0 \tag{B.0.11}$$

Therefore, by Lemma B.0.2, the proof is complete.

Appendix C

ADDITIONAL SIMULATIONS FOR TWO-CLASS DATASETS

We first present results for a simulation study similar to the one in Section 2.7.1, but with only two classes. Taking an approach similar to the one described in Section 2.7.1, we defined two networks with $p = 500$ features belonging to ten equally sized unconnected subnetworks, each with a power law degree distribution. Of the ten subnetworks, eight have the same structure and edge values in both classes, and two are present in only one class. Class 1's network has 490 edges, 94 of which are not present in class 2. We generated covariance matrices as described in Section 2.7.1. Again, we simulated 100 datasets with $n = 150$ observations per class. The results shown in Figure C.1 are similar to the results in Section 2.7.1.

We also simulated data with an entirely different network structure. Instead of the block-diagonal network structure used in the previous simulations, in this simulation we generated data drawn from a single large power law network. We defined class 1's network to be a single power law network with only one component and generated $\mathbf{\Sigma}_1$ as described in Section 2.7.1. We then identified a branch in this network connected to the rest of the network through only one edge. We then let $\mathbf{\Sigma}_2^{-1}$ equal $\mathbf{\Sigma}_1^{-1}$, except for the elements corresponding to the edges in the selected branch, which were set to be zero instead. Finally, we defined $\mathbf{\Sigma}_2$ by inverting $\mathbf{\Sigma}_2^{-1}$, and generated the two classes' data using $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$. This yielded distributions based on two power law networks that were identical except for a missing branch in class 2. Class 1's network has 499 edges, 104 of which are not present in class 2. We simulated 100 datasets

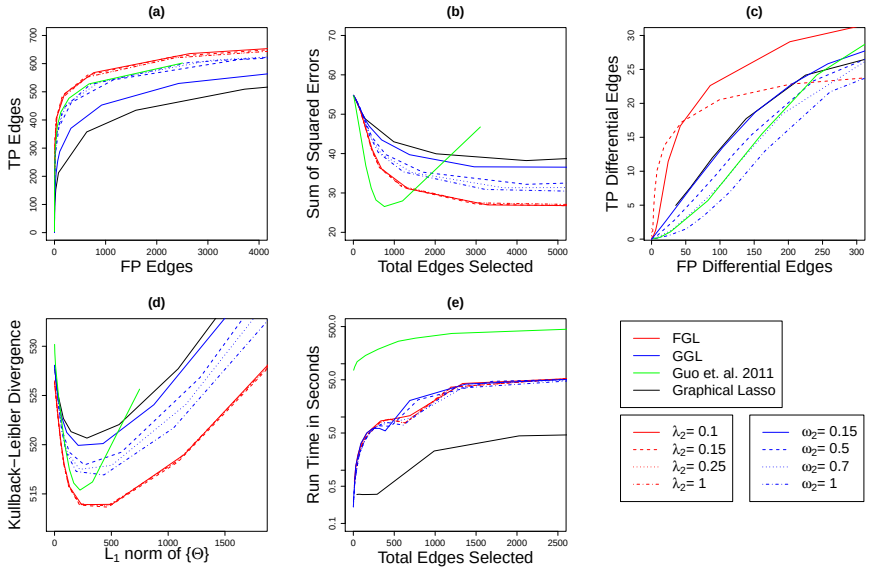

Figure C.1: *Performance of FGL, GGL, Guo et al. [2011]'s method, and the graphical lasso on simulated data with* 150 *observations in each of 2 classes, and* 500 *features corresponding to ten equally sized unconnected subnetworks drawn from a power law distribution. Details are as given in Figure 2.2.*

with $n = 150$ observations per class. Figure C.2 shows the results, averaged over the 100 data sets. Again, FGL and GGL were superior to or competitive with the other methods.

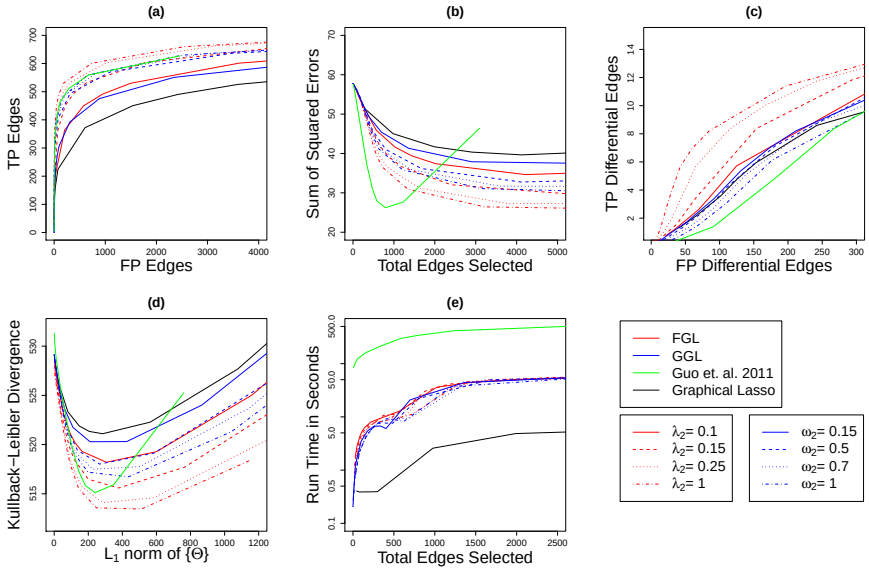

Figure C.2: *Performance of FGL, GGL, Guo et al. [2011]'s method, and the graphical lasso on simulated data with* 150 *observations in each of 2 classes, and* 500 *features corresponding to a single large power law network. Details are as given in Figure 2.2.*

Appendix D

NETWORK STRUCTURE USED IN SIMULATIONS

The network structure for the simulations described in Section 2.7.1 is displayed in Figure D.1. Black edges are shared between all three classes' networks, green edges are present only in classes 1 and 2, and red edges are present only in class 1.

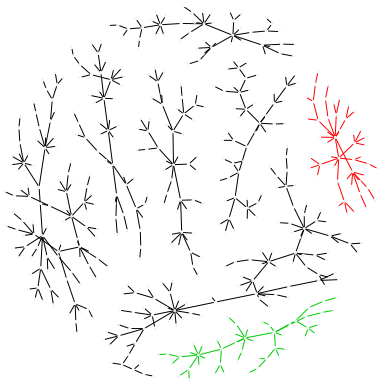

Figure D.1: *Network used to generate simulated datasets for Figure 2.2 in Section 2.7.1. Black edges are common to all three classes, green edges are present only in classes 1 and 2, and red edges are present only in class 1.*

Appendix E

SUBNETWORKS IDENTIFIED IN APPLICATION OF FGL TO A LUNG CANCER GENE EXPRESSION DATASET

The elements of the subnetworks uncovered in Figure 2.3 are as follows. Subnetwork membership is given without regard to subnetwork structure. Each unindented line begins a new subnetwork; each indented line continues a subnetwork. Note that many subnetworks contain multiple probes for the same gene.

Subnetworks in healthy samples

Membership of genes in healthy subnetworks is listed below.

RPL6,RPS7P11,RPL5
RPL10A,RPL15
RHOA,TM9SF2
PRKAR1A,CALM3
EEF1G,EEF1G
NACA,EEF1A1,EEF1A1,BTF3,NACA,RPL3,BTF3
GNAS,GNAS,GNAS,GNAS,GNAS,CD9
LDHB,LDHB
RPLP0,RPLP0
TUBA1B,TUBA1C,TUBA1B,TUBA1B,TUBA1C,TUBA1B,TUBA1B
RPL4,RPL4

PPIA,PPIA,PPIA,PPIA,PPIA

COPB1,PSMA3,MMADHC

ACTG1,ACTG1,ACTG1,ACTG1,ACTG1,ACTG1

ANXA2,ANXA2,ANXA2

PSMA1,PSMA1

EIF1,EIF1,EIF1

CYP1B1,CYP1B1

HBA2,HBB,HBA2,HBB,HBA2,HBA2,HBB,HBA2

AKR1C1,AKR1C2

HBG2,HBG2

UGT1A1,UGT1A1

IDI1,IDI1

HLA-F,HLA-F

ATP5C1,ATP5C1,ATP5C1

SFTPC,SFTPC

ZC3H7B,AK022213,FAM128B

PDE4C,NM_017932,ZNF160,PGF,FBXW12,AK023783,AF222691,AI683552,
HAUS2,SLC35E1

MSMB,MSMB

HINT1,HINT1

CYB5A,CYB5A

PABPC3,PABPC1,LOC652607

HLA-DRB1,LOC100133811,LOC100133811

HLA-B,HLA-G,HLA-G,HLA-B

ATP5L,ATP5L

HLA-C,HLA-C

HLA-DRA,HLA-DRA

IGL@,IGL@,IGLV2-14

TUSC3,TUSC3

SERPINB3,SERPINB4

CD24,CD24

FN1,FN1,FN1,FN1

LOC440926,LOC440926

ACTG1,ACTG1

RPL17,RPL17

GAPDH,GAPDH,GAPDH

CSNK1A1,CSNK1A1

XIST,XIST

MUC5AC,MUC5AC

LOC339047,LOC100132540

SLC38A2,SLC38A2

IGK@,IGK@

SFN,SFN

GGA1,GGA1

Subnetworks in tumor samples

Membership of genes in cancer subnetworks is listed below.

ANXA1,ANXA7,ARL8B,ATP5A1,ATP5C1,ATP5C1,ATP5C1,ATP5F1,
ATP5H,BTF3,BTF3,C11orf58,CALM3,CAST,
CCT2,CD9,CNBP,COPB1,CPNE3,CSNK1A1,CTR9,
CUL3,DPM1,EEF1A1,EEF1A1,FBXL5,GABARAPL2,

GNAS,GNAS,GNAS,GNAS,GNAS,HBXIP,HINT1,
HINT1,HNRNPA2B1,HNRNPK,HSP90AB1,HSP90AB1,
HSPA8,ITM2B,LOC100133775,MARCKS,MATR3,
MDH1,MMADHC,MYL12B,NACA,NACA,NAP1L1,
NAP1L1,NARS,NPTN,PPIA,PPIA,PPIA,PPIA,
PPIA,PPP2CB,PSMA2,PSMA3,PSMD6,PTGES3,
RAB11A,RAB1A,RAN,RHOA,RPL10A,RPL14P1,
RPL17,RPL17,RPL17,RPL3,RPL30,RPL31,RPL5,
RPL6,RPLP0P6,RPS23,RPS24,RPS7P11,RPS8,
SCP2,SEPT2,SF3B1,SF3B1,SPCS1,SRP14,
SRP9L1,SSBP1,TM9SF2,TMCO1,TMED10,TMEM14B,
TOMM20,TPT1,TSG101,UBXN4,UGP2,YWHAZ

RPL15,RPL15,ACTR10,SET

RPL24,RPL35A

PRKAR1A,CALM3

EEF1G,EEF1G

DAZAP2,TCP1

EIF4A2,RTN4

LDHB,LDHB

RPLP0,RPLP0,RPLP0

BCLAF1,SERBP1

TUBA1B,TUBA1C,TUBA1B,TUBA1B,TUBA1C,TUBA1B,TUBA1B

RPL4,RPL4

NQO1,NQO1

ACTG1,ACTG1,ACTG1,ACTG1,ACTG1,ACTG1

ANXA2,ANXA2P2,ANXA2,ANXA2

PSMA1,PSMA1

TWF1,CHMP5

EIF1,EIF1,EIF1

PCM1,TSPAN6,CALM3,HIPK1,AZIN1,C16orf80

DBI,DBI,DBI

CYP1B1,CYP1B1

RPS2,RPS2

HBA2,HBA2,HBA2,HBA2,HBA2

AKR1C1,AKR1C2

HBG2,HBG2,HBG2

UGT1A1,UGT1A1

NPIP,LOC339047,LOC100132540

LOC100133811,HLA-DRB1,LOC100133811,LOC100133811

HLA-F,HLA-F

DBT,PDE4C,NM_017618,ZNF160,PGF,FBXW12,AK023783,AK021514,
AF222691,HAUS2,POLR1B,SLC35E1

TPSB2,TPSB2

SLC27A2,SLC27A2

ZC3H7B,AK022213,FAM128B

OPHN1,RECK

PFDN5,PFDN5

DDR1,DDR1

MSMB,MSMB

ATP5L,ATP5L,ATP5L

NM_017932,AI683552

CYB5A,CYB5A

PABPC3,PABPC1,LOC652607

HLA-B,HLA-G,HLA-G,HLA-B

LOC440926,LOC440926,LOC440926

RPL22,RPL22

HLA-C,HLA-C

HLA-DRA,HLA-DRA

TUBB2C,TUBB3

HBB,HBB,HBB

IGL@,IGL@

TUSC3,TUSC3

SERPINB3,SERPINB4

CD24,CD24

FN1,FN1,FN1,FN1

RPL13A,RPL13A

ACTG1,ACTG1

Appendix F

EVALUATING THE BIVARIATE NORMALITY OF L AND T

The work of Paul [2007] and Mathai and Pillai [1982] establish the asymptotic normality of our statistics L and T. Our quadratic form statistic, however, requires that L and T be jointly bivariate normal. Unfortunately, to our knowledge no theoretical results exist confirming this property. To assess the plausibility of their bivariate normality, we performed a simple simulation. We defined $\mathbf{\Sigma}$ with a spiked eigenvalue of 15 and 29 unspiked eigenvalues equal to 1, and we generated 10,000 MVN datasets from $\mathbf{\Sigma}$ with sample size 100. We saved the L and T statistics from each of these datasets. If L and T are bivariate normal, then L should appear normal after conditioning on T. To test whether bivariate normality holds, we used simple linear regression to predict the simulated L's from the simulated T's. We scaled the residuals from this regression to have standard deviation of 1 and mean 0, and we compared them to the quantiles of a standard normal distribution. As Figure F.1 shows, the residuals conformed nearly perfectly to the quantiles of a normal distribution. Thus is appears very likely that L and T are jointly bivariate normal.

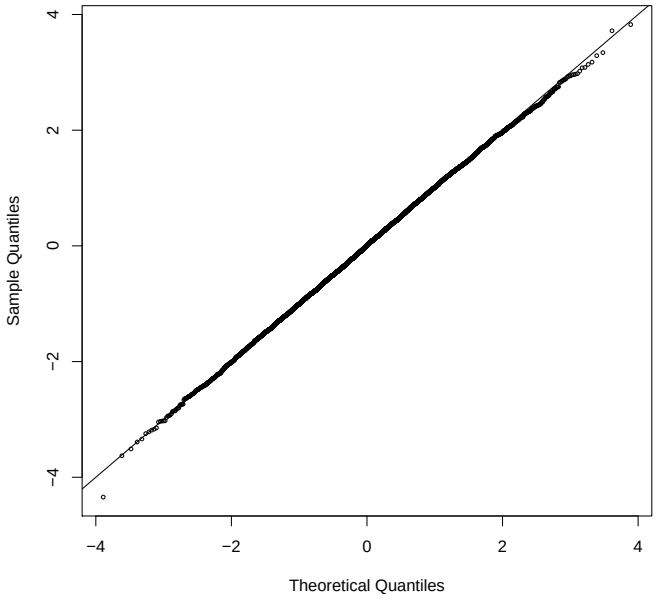

Figure F.1: *QQ plot of standardized residuals from regression predicting L statistics from T statistics calculated on 10,000 simulated datasets.*

VITA

Patrick Danaher grew up in Palo Alto, CA. He graduated from Hamilton College in 2004 with a double major in math and philosophy. Between college and grad school, he hiked the Appalachian Trial, worked two years as a bioinformatics analyst at XDx, a biotech startup, and hiked most of the Pacific Crest Trail. Since arriving in Seattle in 2007 he has spent his time working on his Ph.D. and exploring the Cascades on foot and ski with his wife Sarah.