Universidad Carlos III de Madrid

Department of Statistics

PHD in Business Administration and Quantitative Methods

Doctoral Thesis

New Estimation Methods for High-Dimensional Inverse Covariance Matrices

Author:
Vahe Avagyan

Advisors:
Francisco J. Nogales
Andrés M. Alonso

December 2015

Contents

List of Figures

List of Tables

List of Acronyms

AP	Average Proportion
ADT	Adaptive D-trace
BIC	Bayesian Information Criterion
CLIME	Constrained ℓ_1 Minimization for Inverse Matrix Estimation
CV	Cross-Validation
DT	D-trace
GGM	Gaussian Graphical Model
GLASSO	Graphical LASSO
KLL	Kullback-Leibler Loss
LASSO	Least Absolute Shrinkage and Selection Operator
LDA	Linear Discriminant Analysis
MCC	Matthews Correlation Coefficient
MVP	Minimum Variance Portfolio
MSE	Mean Squared Error
R-GLASSO	Rooted GLASSO
SCAD	Smoothly Clipped Absolute Deviation
SPACE	Sparse Partial Correlation Estimation
WADT	Weighted Adaptive D-trace

List of Notations (Symbols)

p	Number of variables
n	Number of observations
$\mathbb{N}$	Set of natural numbers
$\mathbb{Q}$	Set of rational numbers
$\mathbb{R}$	Set of real numbers
$\mathbb{R}^p$	Space of p-dimensional vectors with real valued elements
$\mathbb{R}^{p\times p}$	Space of $p \times p$-dimensional symmetric matrices with real valued elements
a_i	i-th element of vector $a \in \mathbb{R}^p$
a_{ij}	(i,j)-th element of matrix $A \in \mathbb{R}^{p\times p}$
A^T	Transpose of matrix A
I	Identity matrix
$\mathbf{1}_p$	p-dimensional vector of ones
$\mathbf{X}$	Sample data matrix $\mathbf{X} = (X_1^T, ..., X_n^T)^T$, where $X_i \in \mathbb{R}^p$
Ω	Precision matrix $\Omega = [\omega_{ij}]_{1\leq i,j\leq p}$, where $\omega_{ij} \in \mathbb{R}$ for $1 \leq i, j \leq p$
Σ	Covariance matrix $\Sigma = [\sigma_{ij}]_{1\leq i,j\leq p}$
S	Sample covariance matrix $S = [s_{ij}]_{1\leq i,j\leq p}$
P	Partial correlation matrix $P = [\rho_{ij}]_{1\leq i,j\leq p}$

List of Notations (Definitions)

Vector ℓ_2 (Euclidean) norm	$\|\|a\|\|_2 = \sqrt{\sum_{i=1}^{p} a_i^2}$
Matrix entrywise ℓ_2 (Frobenius) norm	$\|\|A\|\|_2 = \sqrt{\sum_{i=1}^{p}\sum_{j=1}^{p} a_{ij}^2}$
Matrix entrywise ℓ_1 norm	$\|\|A\|\|_1 = \sum_{i=1}^{p}\sum_{j=1}^{p} \|a_{ij}\|$
Matrix ℓ_1 norm	$\|\|A\|\|_{\ell_1} = \max_{1\leq j\leq p} \sum_{i=1}^{p} \|a_{ij}\|$
Matrix entrywise ℓ_∞	$\|\|A\|\|_\infty = \max_{1\leq i,j\leq p} \|a_{ij}\|$
Matrix spectral (operator) norm	$\|\|A\|\|_{\text{spec}} = \sup_{\|\|x\|\|_2\leq 1} \|\|Ax\|\|_2$
Positive definite matrix A	$A \succ 0$
Positive semidefinite matrix A	$A \succeq 0$
Trace of a matrix A	$\text{trace}(A)$
Determinant of a matrix A	$\det(A)$
Expected value of a random matrix A	$E(A)$
Cardinality of a set Z	$\text{card}(Z)$
Sign function	$\text{sign(a)} = \begin{cases} -1 & \text{if } a < 0 \\ 0 & \text{if } a = 0 \\ 1 & \text{if } a > 0 \end{cases}$
Indicator function	$\mathbb{I}_\alpha = \begin{cases} 1 & \text{if } \alpha \text{ is true} \\ 0 & \text{if } \alpha \text{ is false} \end{cases}$
Maximum function	$\max(a,b) = \begin{cases} a & \text{if } a > b \\ b & \text{if } a \leq b \end{cases}$

To my family

Abstract

The estimation of inverse covariance matrix (also known as precision matrix) is an important problem in various research fields and methodologies, especially in the current age of high-dimensional data abundance. In addition, the classical estimation methods are no longer stable and applicable in high dimensional settings, i.e., when the dimensionality has the same order as the sample size or is much larger.

This thesis focuses on the estimation of the precision matrices as well as their applications. In particular, the goal of this thesis is to develop and analyse accurate precision matrix estimators for problems in high-dimensional settings. Moreover, the proposed precision matrix estimators should emulate the existing prominent estimators in terms of different statistical measures without being computationally more extensive.

This thesis is comprised of two articles on estimation of precision matrices in high dimensional settings. In what follows, we summarize the main contributions of this thesis.

First, we propose a simple improvement of the popular Graphical LASSO (GLASSO) framework that is able to attain better statistical performance without increasing significantly the computational cost. The proposed improvement is based on computing a root of the sample covariance matrix to reduce the spread of the associated eigenvalues. Through extensive numerical results, using both simulated and real datasets, we show that the proposed modification improves the GLASSO procedure. Our results reveal that the square-root improvement can be a reasonable choice in practice.

Second, we introduce two adaptive extensions of the recently proposed ℓ_1 norm penalized D-trace loss minimization method. It is well known that the ℓ_1 norm penalization often fails to control the bias of the obtained estimator because of its overestimation behavior. Our proposed extensions are based on the adaptive and weighted adaptive thresholding operators and intend to diminish the bias produced by the ℓ_1 penalty term. We present the algorithm

for solving our proposed approaches, which is based on the alternating direction method. Extensive numerical results, using both simulated and real datasets, show the advantage of our proposed estimators.

Acknowledgements

First of all, I would like to extend my gratitude to my advisors Francisco Javier Nogales and Andrés M. Alonso. I have benefited immensely from their valuable suggestions, countless support and guidance throughout the whole time of my PhD. It has been a great pleasure working with Javier and Andres.

I am also indebted to Francisco Javier Prieto for his time and comments. His insightful advices and feedbacks significantly influenced on my research.

I would like to thank the members of the Statistics Department at Universidad Carlos III de Madrid. I want to express my sincere thanks to the friends I have made during the Master and PhD. Special thanks to my PhD fellows and friends Zhu, Tang and Mei for all the wonderful moments we shared together and enjoyable discussions we had over the last years.

Most importantly, I am deeply thankful to my family for their constant encouragement and endless love. This thesis would not have been possible without their support and motivation. I especially thank to my brother Vardan for his priceless advices. I dedicate this thesis to my family.

Last, I gratefully acknowledge the financial support from the Statistics Department at Universidad Carlos III de Madrid and the Government of Spain for the research grant MTM2013-44902-P.

Chapter 1

Introduction

An accurate estimation of high-dimensional inverse covariance matrix (also known as *precision* or *concentration* matrix) has a crucial role in the current age of high-dimensional data explosion. It is an important problem in various research fields and statistical methodologies. In the recent decade, the high-dimensional precision matrix has attracted a growing interest due to the massive flow of voluminous datasets spanning several intensely developing scientific areas (e.g., medicine, genetics, finance, sociology, etc.). Properly estimated high-dimensional precision matrix is fundamental in linear and quadratic discriminant analysis, forecasting, clustering and several other statistical methodologies when dealing with a vast amount of variables (Mardia et al. 1979; McLachlan 2004). One of the a real-world application which requires an accurate and stable precision matrix estimate is the computation of optimal portfolios for large number of assets (Stevens 1998; Frahm and Memmel 2010; Goto and Xu 2013). We describe other important applications in the next section.

Without loss of generality, we assume that $\mathbf{X}$ is a $n \times p$ mean-centered sample data matrix. Each row $X_i = (X_i^1, ..., X_i^p)$ is a realization of a p-variate random vector, *independent and identically distributed* for $i = 1, ..., n$, and has an unknown $p \times p$ covariance matrix Σ with the corresponding precision matrix $\Omega = \Sigma^{-1}$. In what follows, we thoroughly describe the importance

and virtues of precision matrix and the main drawbacks of the classical estimations in high-dimensional settings. After, we provide a detailed literature review of previously studied estimation approaches and methods for the precision matrix and related concepts.

1.1 Background and Literature Review

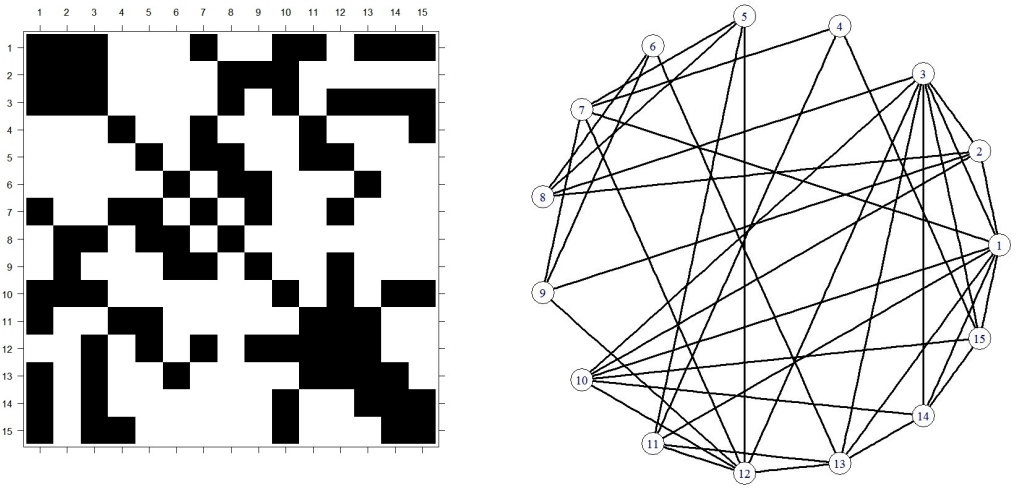
Unlike the covariance matrix, the precision matrix contains proper "multivariate" information. It is well known that each entry of the covariance matrix represents the pair relationship of variables regardless the influence of the other variables. On the other hand, each entry of the precision matrix represents a "correlation indication" between two variables given all the other variables. Moreover, the precision matrix is associated with the *partial correlation matrix*. This statement can be seen through the following statistical property of the precision matrix. The partial correlation between two variables X^i and X^j can be expressed as $\rho_{ij} = -\frac{\omega_{ij}}{\sqrt{\omega_{ii}}\sqrt{\omega_{jj}}}$, for $1 \leq i, j \leq p$. Therefore, the non-zero off-diagonal precision matrix entries indicate the conditional dependence of the corresponding variables.

The exceeding attractiveness of the precision matrix estimation emerges under the assumption of multivariate normality of data. It is well known that when the data follow a Gaussian distribution, the zero entries ω_{ij} of the precision matrix indicate the conditional independence between the variables X^i and X^j, given all the other variables (Lauritzen 1996). More specifically, under the normality assumption, the precision matrix is often sparse and represents the statistical dependency among the variables. Therefore, a sparse representation of the precision matrix is an important issue in statistics. The precision matrix is closely related to the *Gaussian Graphical Models* (GGM), which is a prominent framework for representing the structure of the dependencies among vast amount of normally distributed variables (Whittaker 1990) with a low cost. The GGM is an undirected graph[1] $G = (V, E)$, where the set of the nodes, $V = \{1, ..., p\}$, represents the variables. The set of the edges, $E \subseteq V \times V$, consists of the pair indexes (i, j), that correspond

[1] All the edges in undirected graph are undirected.

to $\omega_{ij} \neq 0$, for $1 \leq i, j \leq p$. For example, the concept of the GGM is a convenient way to represent the interactions between large number (usually tens of thousands) of genes. Moreover, in genetic studies the sparsity pattern of the precision matrix and the corresponding GGM is fundamental for the interpretation of the gene interactions, since most of the genes are conditionally independent and do not interact. To illustrate the idea of the GGM, Figure 1.1 depicts a black-and-white heat map of a sparse precision matrix for fifteen variables and the corresponding GGM. Note that the black and white cells of the heat map represent the non-zero and zero entries of the precision matrix, respectively.

FIGURE 1.1: A heat map of a sparse precision matrix and the corresponding GGM



There are several notable applications involving the estimation of intrinsically sparse precision matrix and GGM such as genetic interaction networks through high-dimensional gene expression data (Stifanelli et al. 2013; Yin and Li 2013), brain connectivity networks through neuroimaging techniques (Huang et al. 2010; Ryali et al. 2012), climate networks (Zerenner et al. 2014), etc.

The estimation of the precision matrix is still a challenging problem in high-dimensional statistics. In classical statistics the most ordinary and straightforward precision matrix estimator is the Maximum Likelihood Estimator

(MLE). It is known that under the normality assumption the log-likelihood function for sample data $\mathbf{X}$ is defined (up to a constant) as:

$$\ell(\mathbf{X}, \Omega) \propto \log\det\Omega - \text{trace}(\Omega S), \tag{1.1}$$

where S is the sample covariance matrix.[2] It can be seen that when $n > p$ the MLE of the precision matrix Ω is the inverse of matrix S. We provide more details on derivation of the log-likelihood function and the MLE in remarks E.1 and E.2 of Appendix E, respectively. Although the sample covariance matrix is an unbiased estimator of the covariance matrix Σ, its inverse, S^{-1}, contains a considerable bias. When $n > p$, it is known that (see Anderson 2003)

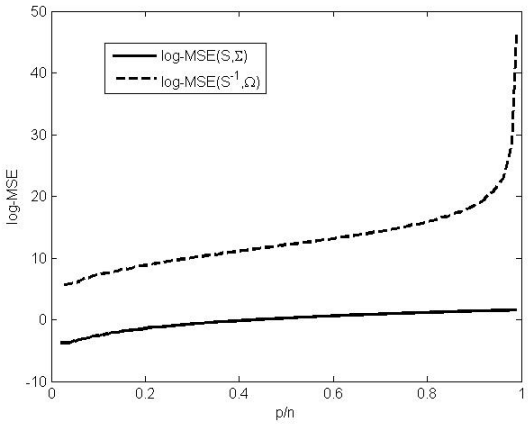
$$E(S^{-1}) - \Omega = \frac{p+2}{n-p-2}\Omega. \tag{1.2}$$

Thus, the traditional precision estimator S^{-1} becomes highly unstable when the ratio $\frac{p}{n}$ increases. For instance, when $p = \frac{n}{2} - 2$, then $E(S^{-1}) - \Omega = \Omega$, therefore, the bias of the classical estimator S^{-1} has the same magnitude as Ω. To illustrate this statement, Figure 1.2 depicts the log-scaled Mean Squared Error[3] (MSE) of the sample covariance matrix (as the estimator of the covariance matrix) and the inverse of the sample covariance matrix (as the estimator of the precision matrix). We observe that the precision estimation error increases exponentially with $\frac{p}{n}$. Moreover, when $\frac{p}{n} > 1$, the matrix S becomes singular and, therefore, the estimator S^{-1} does not exist.

A straightforward approach could be inverting a well-defined covariance matrix estimate. This approach is known as a two-step or indirect estimation. In this way, several estimators of the covariance and the correlation matrices have been provided with good practical and theoretical properties. Among the most popular ones are the shrinkage estimators (e.g., Ledoit and Wolf 2004; Schafer and Strimmer 2005; Warton 2008; Touloumis 2015), estimators based on thresholding, banding or tapering procedures (e.g., Bickel and Levina 2008; El Karoui 2008; Cai and Yuan 2012; Wang and Daniels 2014) and those based on convex optimization frameworks (e.g., Rothman

[2] We provide the definition of the sample covariance matrix S in the Chapter 2.
[3] See Chapter 2 section 2.5.2 for a formal definition.

FIGURE 1.2: Log-scaled MSE for S and S^{-1}



2012; Xue et al. 2012; Deng and Tsui 2013; Cui et al. 2014). Although the two-step approaches may seem to be convenient and problem-solving, the obtained precision matrix estimators, in general, are not optimal in high-dimensional settings (Ledoit and Wolf 2012). Firstly, by inverting an estimated covariance matrix, we may amplify its estimation error. Secondly, inverting a very large matrix is computationally expensive in terms of the required memory and time. Finally, the two-step approach does not guarantee the sparsity of the precision matrix estimator, even if the estimated covariance matrix is sparse. Thus, to obtain a desirable precision matrix estimate, most of the methodologies in the literature are based on direct estimation techniques. Undoubtedly, direct precision matrix estimation approaches in high-dimensional settings are mathematically more challenging and complex than the two-step approaches because of the absence of a naïve precision matrix estimator.

Substantial research exists related to the problem of precision matrix estimation. A pioneer work has been done by Dempster (1972), who formulated this problem as the *covariance selection* and proposed to study the interdependence of the normally distributed variables through the sparsity notion of the precision matrix. Here we briefly review the main techniques and approaches for estimating precision matrix and associated GGM.

Following the ideas of the shrinkage approaches of the covariance matrix estimation, in essence, the same techniques can also be applied for estimating the precision matrix. In other words, we can estimate the precision matrix by considering different linear combinations between the matrix S^{-1} and a selected target matrix (see, for instance, Haff 1980; Frahm and Memmel 2010; Kourtis et al. 2012). However, as explained above, these approaches rely on $p << n$ assumption, therefore, can not be used in high-dimensional statistics.

To overcome the computational challenges and to deal with the situation of $p = O(n)$, prior research proposed several precision matrix estimators based on a convex optimization framework. To address the sparsity requirement of the matrix and to attain an accurate precision estimator, the LASSO or ℓ_1 regularization can be applied. Originally, Tibshirani (1996) introduced this regularization in the regression framework. However, it has achieved a great interest in the covariance selection study, because it leads to a sparse estimator and computationally convenient due to its convexity. In this way, Banerjee et al. (2006) proposed the ℓ_1 norm penalized log-likelihood function (1.1) maximization approach which is one of the remarkable estimation methods and is defined as the solution of the following problem:

$$\arg\max_{\Omega} \ \log\det\Omega - \text{trace}(S\Omega) - \nu||\Omega||_1, \tag{1.3}$$

where $\nu > 0$ is a penalty parameter and $||\Omega||_1$ is the entrywise ℓ_1 norm[4] of the matrix Ω. Note that the term $||\Omega||_1$ is the convex upper bound of the cardinality of a matrix, therefore, the ℓ_1 norm penalization endorses the sparsity of the estimated precision matrix. Moreover, the log-determinant term guarantees the positive definiteness of the obtained estimator. The precision matrix estimation approach based on the ℓ_1 norm penalized log-likelihood function maximization problem (1.3) is known in the literature as Graphical LASSO[5] or, simply, GLASSO method. Prior work extensively studied this approach. Moreover, some studies considered the original definition of the

[4]We provide the formulation of the entrywise ℓ_1 norm of a matrix in the Notations.

[5]This method is commonly called by the name of the popular and efficient algorithm GLASSO for solving the ℓ_1 norm penalized log-likelihood function maximization problem.

GLASSO method (see, for instance, d'Aspremont et al. 2008; Banerjee et al. 2008), whereas others (see, for instance, Yuan and Lin 2007; Rothman et al. 2008; Yin and Li 2013) defined the GLASSO method by regularizing only the off-diagonal entries of the matrix Ω in the objective function (1.3). In other words, they considered the term $||\Omega||_{1,\text{off}} = \sum_{i=1}^{p} \sum_{j=1, j\neq i}^{p} |\omega_{ij}|$ instead of the term $||\Omega||_1$ in the problem (1.3). Several algorithms have been developed to solve the regularization problem efficiently, such as the Graphical LASSO (Friedman et al. 2008), Project Sub-gradient Method (Duchi et al. 2008), Alternating Linear Minimization (Scheinberg et al. 2010), and Interior Point Method (Li and Toh 2010), among others. Moreover, some scholars proposed approaches to improve the performance of the GLASSO method through adaptive LASSO and non-convex SCAD (Smoothly Clipped Absolute Deviation) penalties (see Fan et al. 2009) or through additional trace norm penalty (see Maurya 2014). Witten et al. (2011) proposed procedure that efficiently speeds-up the algorithm for solving the GLASSO problem. More recently, Banerjee and Ghosal (2015) proposed a Bayesian approach to the GLASSO method.

An alternative to the log-likelihood function (1.1) is the so-called *D-trace (DT) function*, which is introduced recently by Zhang and Zou (2014). It has the following definition:

$$f_{DT}(\Omega, \Sigma) = \frac{1}{2}\text{trace}(\Omega^2\Sigma) - \text{trace}(\Omega). \tag{1.4}$$

The function $f_{DT}(\Omega, \Sigma)$ has much simpler structure than the log-likelihood function (1.1). In this way, Zhang and Zou (2014) proposed sparse precision matrix estimation approach through minimization of off-diagonal ℓ_1 norm penalized D-trace function (1.4), defined as the solution of the following optimization problem:

$$\arg\min_{\Omega \succeq \epsilon I} \frac{1}{2}\text{trace}(\Omega^2 S) - \text{trace}(\Omega) + \tau||\Omega||_{1,\text{off}}, \tag{1.5}$$

where $\tau > 0$ is a penalty parameter. The constraint $\Omega \succeq \epsilon I$ guarantees the positive definiteness[6] of the obtained solution. We provide a broad and

[6]We write $\Omega \succeq \epsilon I$ if $\Omega - \epsilon I \succeq 0$.

detailed description of this method in the Chapter 3.

In addition, several other authors studied non-likelihood approaches to estimate the precision matrix or the GGM. For instance, based on the LASSO regression, Meinshausen and Bühlmann (2006) proposed a Neighborhood Selection approach to select the GGM structure and Peng et al. (2009) proposed a Sparse Partial Correlation Estimation (SPACE) method to estimate the partial correlation matrix P. Regarding the precision matrix estimation, Yuan (2010) proposed the use of the Dantzig selector and Cai et al. (2011) proposed Constrained ℓ_1 minimization for Inverse Matrix Estimation (CLIME) method.

Notwithstanding the sizable literature, the estimation of the precision matrix is still attractive and sophisticated problem, especially in high-dimensional settings. The goal of this thesis is to develop and analyse new estimation methods for high-dimensional precision matrices. In particular, our objective is to propose well-defined precision matrix estimators for problems, where dimensionality (e.g., the number of the variables), p, has the same order or can exceed the sample size, n. Moreover, the proposed methods should provide certain desirable properties (such as sparsity) and should ensure competitive performance comparing with the existing state-of-the-art approaches, being suitable for both the precision matrix estimation and the associated GGM prediction.

1.2 Outline of the Thesis

In this work, we focus on the estimation of high-dimensional precision matrix. Our main contribution is that we propose and analyse two new estimation approaches, which provide proper precision matrix estimate and associated GGM prediction for high-dimensional problems. In particular, in order to present the virtues of our proposed methodologies we conduct extensive numerical simulations and real-world applications in high-dimensional framework. The results show that the proposed precision estimators are

well-defined and dominate state-of-the-art estimators in most of the numerical results in terms of different statistical measures. The developed estimators are also appropriate for estimating GGM structures. Moreover, we demonstrate the merits of the proposed estimators when $\frac{p}{n} > 1$. We study the applications of our proposed methods in different research fields, such as genetics (e.g., indicating the state or the type of the tumour) and finance (e.g., selection of large portfolios).

The outline of the thesis is as follows. In the Chapter 2, we propose a simple improvement of the popular GLASSO framework that is able to attain better statistical performance without having to increase significantly the computational cost. Moreover, we can solve the proposed method using any algorithm which solves the original GLASSO method. The proposed improvement is based on computing a root of the sample covariance matrix to reduce the spread of the associated eigenvalues. Through numerical results, using both simulated and real datasets, we show that the proposed technique outperforms the GLASSO estimator. Finally, for the proposed estimator, we establish the convergence rate in the Frobenius norm.

In the Chapter 3, we focus on the recently proposed ℓ_1 norm penalized DT loss minimization method. We introduce two adaptive extensions of this method. The proposed extensions are based on the adaptive and weighted adaptive thresholding operators and intend to diminish the bias produced by the ℓ_1 penalty term. We present the algorithm for solving our proposed approaches, which is based on the alternating direction method. Through comprehensive numerical simulations we show that the methods based on the proposed extensions outperform the original ℓ_1 norm penalized DT loss minimization method. Finally, we study the performance of the proposed estimators using real datasets.

In the Chapter 4, we provide concluding remarks of the thesis and possible future research directions.

Chapter 2

Improving GLASSO Method Using Roots of the Sample Covariance Matrix

2.1 Introduction

Before proceeding with our proposed methodology, we assume that $\mathbf{X}$ is a centered sample data matrix with dimension $n \times p$, where each row $X_i = (X_i^1, ..., X_i^p)$ is a realization of a p-variate *normal* random vector that is *independent and identically distributed* for $i = 1, ..., n$, with covariance matrix Σ and precision matrix $\Omega = \Sigma^{-1}$.

As mentioned in the Chapter 1, the ℓ_1 norm penalized log-likelihood maximization approach (Banerjee et al. 2006) is one of the state-of-the-art methods for obtaining a sparse and proper precision matrix estimate. This approach is known as the GLASSO method due to the popular solving algorithm of the same name, proposed by Friedman et al. (2008). This algorithm allows a fast, efficient and stable solution of the ℓ_1 norm penalized log-likelihood maximization problem for the high-dimensional problems. As

discussed in the Chapter 1, the GLASSO method has been extensively analysed by several scholars. Moreover, it is one of the frequently applied precision matrix estimation methods in several research fields.[1]

In this chapter, we focus on the GLASSO approach and propose a simple modification that is able to attain a better statistical performance without sacrificing too much the computational cost. According to the dual problem of (1.3), GLASSO method is based on minimization of the log-determinant of the precision matrix subject to its inverse being close to the sample covariance matrix, S. However, it is well known (Johnstone 2001) that in high-dimensional settings the eigenvalues of S are very diffused and hence, its condition number is large. Through simulations, Ledoit and Wolf (2004) show that the condition number and the bias of the largest and smallest sample eigenvalues tend to increase with $\frac{p}{n}$. To improve the stability of the GLASSO estimation, we propose to use a k-root of the sample covariance matrix, with $k \geq 1$, to attain less diffused eigenvalues and therefore, to obtain a more accurate estimation of $\Omega^{1/k}$ and, therefore, of Ω.

Our proposed method is a simple modification of the GLASSO one. Similar to the original GLASSO, it is based on minimization of the log-determinant of the precision matrix, but now subject to its k-root inverse being close to the k-root of the sample covariance matrix. Once the specific k-root and the penalty parameter (associated with the original GLASSO framework) are selected, the proposed procedure requires no additional cost than that of the GLASSO method. Through extensive numerical results, using both simulated and real datasets, we show that the proposed technique outperforms the GLASSO estimator when considering different statistical losses and GGM prediction performance measures. In particular, we use the entropy loss and the Mean Squared Error to measure the statistical performance of the estimators. In addition, we use specificity, sensitivity and Matthews Correlation Coefficient (MCC) to measure the GGM prediction accuracy. Furthermore, we propose a calibration procedure for selecting the k-root of the sample covariance matrix and also the penalty parameter that regularizes

[1]The article by Friedman et al. (2008) has more than 1400 citations as of November 2015 according to `https://scholar.google.com/`.

the log-likelihood function (1.1). Finally, for the proposed k-root GLASSO method, we establish the convergence rate in the Frobenius norm.

The rest of the chapter is organized as follows. Section 2.2 describes the proposed k-root GLASSO (or simply R-GLASSO) methodology to estimate high-dimensional precision matrices. Section 2.3 analyses the convergence rate of the proposed estimator. Section 2.4 proposes a procedure for selecting both the k-root of the sample covariance matrix and the associated penalty parameter that regularizes the log-likelihood function. Section 2.5 exhaustively evaluates the statistical loss and GGM prediction performance of the proposed methodology and compares with that of the GLASSO. Section 2.6 illustrates the solution properties when applying the proposed methodology to three empirical applications: the prediction of breast cancer state, the prediction of the SRBC tumour, and the computation of an optimal financial portfolio. Finally, Appendix A provides the analytical proofs and Appendix B contains the tables of the numerical results.

2.2 Proposed Methodology

Banerjee et al. (2006) have proposed the GLASSO method through maximizing the ℓ_1 norm penalized log-likelihood function (1.1). The GLASSO estimator is the solution of the following optimization problem:

$$\widehat{\Omega}_{\text{GLASSO}} = \arg\max_{\Omega} \ \log\det\Omega - \text{trace}(S\Omega) - \nu||\Omega||_1, \tag{2.1}$$

where $S = \frac{1}{n}\sum_{i=1}^{n} X_i X_i^T$ is the sample covariance matrix and $\nu > 0$ is a penalty parameter which controls the sparsity pattern of $\widehat{\Omega}_{\text{GLASSO}}$. This parameter is unknown in practice and should be selected accurately. We note that in this particular chapter, we follow the original definition of the GLASSO estimator through regularization of all the entries of matrix Ω (see Banerjee et al. 2006, 2008). Note that problem (2.1) is convex, and its dual problem (2.2) is defined as

$$\begin{aligned} &\widehat{\Omega}_{\text{GLASSO}} = \arg\min_{\Omega} \ \log\det\Omega \\ &\text{subject to} \ \ ||\Omega^{-1} - S||_\infty \leq \nu. \end{aligned} \tag{2.2}$$

We provide more details on derivation of dual problem (2.2) in remark E.3 of Appendix E. As discussed in Section 2.1, the GLASSO method is sensitive to the eigenvalue structure of the sample covariance matrix, S, especially when p is large. To mitigate this sensitivity, we suggest to shrink the eigenvalue spread by considering a k-root of matrix S defined as $S^{1/k} = BV^{1/k}B'$, where $S = BVB'$ is the eigen-decomposition of S and $k > 1$. In this way, we propose the following R-GLASSO estimator:

$$\begin{aligned} &\widehat{\Omega}_{\text{R-GLASSO}} = \arg\min_{\Omega} \ \log\det\Omega \\ &\text{subject to} \ \ ||\Omega^{-1/k} - S^{1/k}||_\infty \leq \xi_k, \end{aligned} \tag{2.3}$$

where $\xi_k > 0$ is the associated penalty parameter. The problem (2.3) can be rewritten as

$$\begin{aligned} &\widehat{\Gamma} = \arg\min_{\Gamma} \ \log\det\Gamma \\ &\text{subject to} \ \ ||\Gamma^{-1} - S^{1/k}||_\infty \leq \xi_k, \end{aligned} \tag{2.4}$$

and we define our R-GLASSO estimator as $\widehat{\Omega}_{\text{R-GLASSO}} = \widehat{\Gamma}^k$, for a given k and ξ_k. Note that we can write the primal problem of the optimization problem (2.4) as the following:

$$\widehat{\Gamma} = \arg\max_{\Gamma} \ \log\det\Gamma - \text{trace}(S^{1/k}\Gamma) - \xi_k||\Gamma||_1. \tag{2.5}$$

Therefore, we can obtain the proposed estimator $\widehat{\Omega}_{\text{R-GLASSO}} = \widehat{\Gamma}^k$ by solving the problem (2.5) using the same algorithm as for the problem (2.2) without any additional cost. Finally, we note that when $k = 1$, the R-GLASSO estimator coincides with the original one, and, moreover, when $\xi_k = 0$, we obtain the classical naive estimator S^{-1} for any value of k.

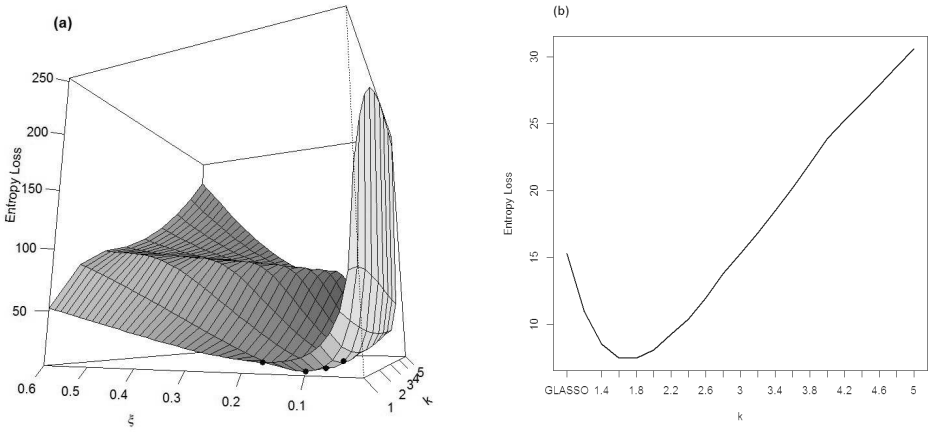
Remark 2.1. It is important to note that the sparsity of the matrix $\widehat{\Gamma}$ does not guarantee the sparsity of the matrix $\widehat{\Omega}_{\text{R-GLASSO}} = \widehat{\Gamma}^k$. However, the main assumption behind the proposed method is that the matrix $\widehat{\Gamma}$ can be considered as an estimator of the matrix $\Omega^{1/k}$, and, therefore, matrix $\widehat{\Gamma}^k$ can be considered as an estimator of the matrix Ω.

To better illustrate the behaviour of the proposed methodology, we show next a particular example. Assume that the true precision matrix Ω has

the following sparse structure: $\omega_{ii} = 1$, $\omega_{i,i-1} = \omega_{i-1,i} = 0.45$, and zero otherwise. We set the values $p = 200$ and $n = 200$.

In Figure 2.1(a), the entropy loss[2] of the proposed estimator is shown as a function of different possible roots (between 1 and 5) and different values of the penalty parameter (between 0.015 and 0.6 with increment of 0.015). Note that, as k moves away from 1 (which corresponds to the GLASSO estimator), it is possible to decrease the loss of the proposed estimator using convenient paths along ξ_k. That is, the minimum possible error of the GLASSO estimator along the ν path is larger than the minimum possible error of the proposed R-GLASSO estimator along the ξ_k path, for some values of k. This improvement can be observed more clearly in Figure 2.1(b), where the entropy loss is plotted against k using the optimal values for ξ_k, i.e, the penalty parameter that minimizes the entropy loss for a given k. Note that we can reduce the statistical loss of the GLASSO estimator by using, for instance, the square-root modification.

FIGURE 2.1: **(a)** Entropy loss of $\widehat{\Omega}_{\text{R-GLASSO}}$ estimator as a function of ξ_k and k. **(b)** Entropy loss of $\widehat{\Omega}_{\text{R-GLASSO}}$ estimator as a function of k (given the optimal ξ_k).



In Section 2.5, through an exhaustive empirical analysis including several sparsity patterns for the precision matrix, we show how the proposed R-GLASSO estimator can outperform the GLASSO under other statistical performance measures covering those for graphical models.

[2]See Section 2.5.2 for a formal definition.

2.3 Convergence Rate

In this section, we analyse the convergence rate of the proposed estimator $\widehat{\Omega}_{\text{R-GLASSO}}$ for rational values of k. First, we state the following main assumptions on the precision matrix Ω:

$$\text{A1} \ : \ \lambda_{\min}(\Omega) \geq \underline{\alpha} > 0,$$

$$\text{A2} \ : \ \lambda_{\max}(\Omega) \leq \bar{\alpha},$$

for some positive values $\bar{\alpha}$ and $\underline{\alpha}$, where $\lambda_{\min}(\Omega)$ and $\lambda_{\max}(\Omega)$ are the minimum and the maximum eigenvalues of matrix Ω, respectively. Note that the assumptions A1 and A2 guarantee the existence of the matrix Ω. Next, we define the set $Z = \{(i,j) : [\Omega^{1/k}]_{ij} \neq 0\}$ and $\text{card}(Z) \leq s$. The following theorem presents the convergence rate of the proposed R-GLASSO estimator.

Theorem 2.2. *Suppose $\widehat{\Omega}_{R\text{-}GLASSO}$ is the solution of problem (2.3) and $k \in \mathbb{Q}$. Under the assumptions A1, A2, if $||\Sigma^{1/k} - S^{1/k}||_\infty = O_P(||\Sigma - S||_\infty)$ and $\xi_k \asymp \sqrt{\dfrac{\log p}{n}}$,*

$$||\widehat{\Omega}_{R\text{-}GLASSO} - \Omega||_2 = O_P\left(\sqrt{\frac{(p+s)\log p}{n}}\right). \tag{2.6}$$

We provide the proof of the Theorem 2.2 in the Appendix A.

2.4 Penalty Parameter Selection

The choice of the penalty parameter has a crucial role in all estimation procedures based on regularization. The penalty parameter controls the properties of the estimator, especially its sparsity level. To account for this sparsity level, we suggest the use of the BIC-type criterion.[3] Yuan and Lin (2007) proposed the following BIC criterion for selecting the penalty parameter of the GLASSO method:

$$\text{BIC}(\nu) = n\left(-\log\det\widehat{\Omega}(\nu) + \text{trace}(S\widehat{\Omega}(\nu))\right) + \log n \times \text{NZ}, \tag{2.7}$$

[3] In one of the empirical applications in Section 2.6, we use a cross-validation procedure to calibrate the penalty parameter, since in this application the sparsity pattern is not relevant.

where $\text{NZ} = \text{card}\{(i,j) : 1 \leq i \leq j \leq p, [\widehat{\Omega}]_{ij} \neq 0\}$. The penalty parameter ν is selected by minimizing $\text{BIC}(\nu)$. Our proposed methodology requires to calibrate two parameters, ξ_k and k. We define the following BIC score to select simultaneously these parameters:

$$\text{BIC}(\xi_k, k) = n\left(-\log\det\widehat{\Omega}(\xi_k, k) + \text{trace}(S\widehat{\Omega}(\xi_k, k))\right) + \log n \times \text{NZ}, \quad (2.8)$$

where $\widehat{\Omega}(\xi_k, k)$ is the estimated precision matrix using the values ξ_k and k. The parameters (ξ_k, k) are selected by minimizing $\text{BIC}(\xi_k, k)$ using a two-dimensional grid search.

2.5 Simulation Study

In this section, we perform a simulation analysis to compare the performance of the proposed estimator $\widehat{\Omega}_{\text{R-GLASSO}}$ with that of the GLASSO estimator $\widehat{\Omega}_{\text{GLASSO}}$. Particularly, in subsection 2.5.1, we detail the considered models for the precision matrix Ω. In subsection 2.5.2, we describe the performance evaluation. Finally, in subsection 2.5.3, we provide the discussion of the results.

2.5.1 Considered Models

We perform an exhaustive simulation study through seven different structures for the precision matrix with varying sizes. We divide the models into random (with random sparsity pattern and elements) and non-random (with fixed sparsity pattern and deterministic elements). The considered models for the precision matrix Ω are the following:

(i) Random models[4]

- *Model* 1. A random p.d. matrix, containing 5% of non-zero entries.
- *Model* 2. A random p.d. matrix, containing 10% of non-zero entries.

[4]All random models are generated using the MATLAB command *sprandsym*.

- *Model* 3. A random p.d. matrix, containing 20% of non-zero entries.
- *Model* 4. A random block-diagonal matrix, with four equally-sized blocks along the diagonal, each containing 50% of non-zero entries.

(ii) Non-random models

- *Model* 5. AR(1) structure: $\omega_{ii} = 1$, $\omega_{i,i-1} = \omega_{i-1,i} = 0.45$, and zero otherwise (Yuan and Lin 2007; Friedman et al. 2008).
- *Model* 6. Decay structure: $\omega_{ij} = 0.6^{|i-j|}$ (Cai et al. 2011; Fan et al. 2009).
- *Model* 7. A block-diagonal matrix, with four equally sized blocks along the diagonal, with a decay model in each block.

For each of the models, we simulate multivariate normal random samples with zero mean, where $n = 200$ and $p = 100, 200$ and 300. This procedure is repeated 100 times.

2.5.2 Performance Evaluation

To compute the performance of a given estimator $\widehat{\Omega}$, we use the entropy loss function, also known as the Kullback-Leibler Loss (KLL) function (James and Stein 1961), defined as follows:

$$\text{KLL}(\widehat{\Omega}, \Omega) = \text{trace}(\Omega^{-1}\widehat{\Omega}) - \log\det(\Omega^{-1}\widehat{\Omega}) - p. \tag{2.9}$$

The KLL function is the simplified version of the Kullback-Leibler divergence (Kullback and Leibler 1951) for multivariate Gaussian distribution. This loss function has been used widely in the prior research on estimation of covariance and precision matrices (see, for instance, Yuan and Lin 2007; Rothman et al. 2008; Fan et al. 2009; Yin and Li 2013). Moreover, we also use the Mean Squared Error defined as:

$$\text{MSE}(\widehat{\Omega}, \Omega) = ||\widehat{\Omega} - \Omega||_2^2. \tag{2.10}$$

Regarding the sparsity pattern or GGM prediction performance, we compute specificity, sensitivity and Matthews Correlation Coefficient (MCC), defined as:

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}, \tag{2.11}$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{2.12}$$

and

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}}, \tag{2.13}$$

where TP, TN, FP and FN are the numbers of true positives (number of correctly estimated non-zero entries), true negatives (number of correctly estimated zero entries), false positives (number of incorrectly estimated non-zero entries) and false negatives (number of incorrectly estimated zero entries), respectively. Note that FP and FN can be seen as Type I and Type II errors, respectively. The MCC measure was introduced by Matthews (1975) and it is commonly used to measure the performance of binary classifiers. The MCC values are in [-1,1], and the closer the MCC to one is, the better the classification is.

We consider the GLASSO and the R-GLASSO procedures where the penalty parameters ν and ξ_k, as well as the k-root parameter, are estimated using the BIC criterion (2.8). We also focus on the square-root GLASSO procedure, i.e., $k = 2$, because of its good behaviour in practice. Finally, we include a comparison with the method CLIME[5] as it is one of the popular estimators for the precision matrix. Cai et al. (2011) proposed the CLIME estimator as a matrix which is obtained by symmetrizing the solution of the following

[5]For calculating GLASSO, R-GLASSO and CLIME estimators we use the **R** packages `glasso` and `clime`, available at http://cran.r-project.org/web/packages.

optimization problem:[6]

$$\min ||\Omega||_1 \tag{2.14}$$

$$\text{subject to } \ ||\Omega S - I||_\infty \leq \nu, \tag{2.15}$$

where ν is the associated regularization parameter. The parameters for GLASSO, CLIME and R-GLASSO for $k = 2$ are estimated using the BIC criterion (2.7).

2.5.3 Discussion of Results

We firstly compare the computational time of the considered methods. The computational time for each estimator represents the sum of the working time of the parameter selection process and the working time of the estimation using the selected parameters. For the proposed R-GLASSO method, the parameter selection process includes the estimation of both parameters ξ_k and k, where parameter k is selected from five values $k = 1, 2, ..., 5$. Finally, for selection of the penalty parameters, we consider the same grid size for all the methods. Table 2.1 provides the computational times of the three estimators for model 5.[7] We observe that CLIME method is very time consuming, especially when p is large. On the other hand, the difference between the time of the methods GLASSO and R-GLASSO is relatively small, even for large values of p. Hence, we do not sacrifice too much the computational cost for R-GLASSO method.

We provide the simulation results in the Appendix B to conserve space (see Tables B.1-B.5). Each table reports the averages over 100 replications and the standard deviations (SD) of the corresponding losses and prediction measures. We organize the discussion of our results as follows. We first compare

[6]Since in the problem (2.14) there is no symmetry condition on Ω, the solution $\hat{\Omega} = [\hat{\omega}_{ij}]_{1\leq i,j\leq p}$ is not symmetric in general. The CLIME estimator $\hat{\Omega}^o = [\hat{\omega}^o_{ij}]_{1\leq i,j\leq p}$ is obtained by symmetrizing $\hat{\Omega}$, i.e., $\hat{\omega}^o_{ij} = \hat{\omega}^o_{ji} = \hat{\omega}_{ij}\mathbb{I}_{|\hat{\omega}_{ij}|\leq|\hat{\omega}_{ji}|} + \hat{\omega}_{ji}\mathbb{I}_{|\hat{\omega}_{ij}|>|\hat{\omega}_{ji}|}$, for $1 \leq i, j \leq p$.

[7]The computational time differs for different models. However, the comparison results are roughly the same.

TABLE 2.1: Total computational time (in seconds) of the three estimators for model 5.

p	100	200	300
GLASSO	0.37	2.15	7.26
R-GLASSO	0.84	5.32	11.61
CLIME	31.50	458.29	2480.95

our proposed R-GLASSO estimator with the GLASSO estimator. We then compare the R-GLASSO estimator with the CLIME estimator.

We report the statistical losses in Tables B.1 and B.2. We observe that the proposed R-GLASSO method provides lower KLL and MSE than GLASSO for all the models. Therefore, R-GLASSO method outperforms GLASSO in terms of the statistical losses.

Tables B.3, B.4 and B.5 illustrate the results of the GGM prediction performances. From Tables B.3 and B.5, we observe that R-GLASSO outperforms GLASSO for all the models in terms of specificity and MCC.[8] Finally, R-GLASSO outperforms GLASSO in terms of sensitivity (see Table B.4) for models with deterministic sparsity patterns (models 6, 7). However, GLASSO performs better in terms of sensitivity for models with random sparsity patterns (models 1, 2, 3, 4). For model 5 all three methods provide the same sensitivity level.

When we compare the proposed estimator with CLIME, our proposed R-GLASSO provides better results for models 2, 3, 4, 5 and similar results for models 1, 6, 7 in terms of KLL. Moreover, R-GLASSO outperforms CLIME for models 2, 3, 4, 5, 6, 7 and provides similar results for model 1 in terms of MSE. In addition, the R-GLASSO estimator outperforms CLIME method in terms of MCC for models 1, 2, 3, 4, 7. Our proposed R-GLASSO method provides higher sensitivity for models 2, 3, 4, 6, 7 and higher specificity for models 1, 2, 3, 4, 7. On the other hand, CLIME provides better GGM prediction performances for model 5. However, we note that the computational

[8]Specificity and MCC are excluded for model 6, because these measurements are not defined for dense models.

cost of CLIME is considerably larger than that of R-GLASSO (see Table 2.1).

In sum, the proposed R-GLASSO estimation method provides better performance, including matrix losses and GGM predictions, than GLASSO and CLIME methods for most of the models. Note also that this conclusion holds if we use the square-root GLASSO method (i.e., $k = 2$). This finding allows us to simplify and "robustify" our framework without sacrificing too much the performance.

2.6 Real Data Applications

In this section, we conduct an empirical analysis of the proposed R-GLASSO method through three real-data applications. In particular, we use breast cancer and SRBC tumour datasets to predict the tumour behaviour using Linear Discriminant Analysis (LDA). Our last application aimed to select a large financial portfolios.

2.6.1 Breast Cancer Data

In this application, we focus on the problem of predicting breast cancer patients with pathological complete response (pCR). The literature has shown that the pCR state after the neoadjuvant chemotherapy strongly indicates a cancer-free life (Kuerer et al. 1999). Thus, it is important to select the patients with the pCR state correctly. In our application we use a dataset containing gene expression levels,[9] previously analysed by Hess et al. (2006). This dataset contains 22283 gene expression levels of 133 patients (subjects) with different stages of breast cancer. There are 34 patients with pCR and 99 patients with residual disease (RD).

First, we divide the data into a training set and a testing set with sizes 112 and 21, respectively. This process is repeated 100 times. We follow the same division scheme applied in Cai et al. (2011). The testing set randomly selects

[9]The dataset is available at http://bioinformatics.mdanderson.org/pubdata.html.

TABLE 2.2: Average pCR/RD classification measurements over 100 replications for $p = 113$ genes.

Method	Specificity	Sensitivity	MCC
GLASSO	0.726	0.580	0.281
R-GLASSO $k = 2$	0.633	0.840	0.413
R-GLASSO $k = 3$	0.618	0.856	0.414
R-GLASSO $k = 4$	0.611	0.868	0.419
CLIME	0.693	0.822	0.453

5 subjects with pCR and 16 subjects with RD. The training set contains the remaining subjects. Second, for the training set we apply two sample t-test between the two groups in order to select the most significant 113 genes with the smallest p-values. Finally, the precision matrix Ω is estimated with the methods GLASSO, R-GLASSO and CLIME, using the training set. The penalty parameters for all three methods are estimated using the BIC criterion (2.7). We analyse the performance of the R-GLASSO method when the parameter k is selected from a range 2 to 4.[10] The estimated precision matrix is used in the Linear Discriminant Analysis (LDA) score:

$$\delta_t(Y) = Y^T \widehat{\Omega} \widehat{\mu}_t - \frac{1}{2} \widehat{\mu}_t^T \widehat{\Omega} \widehat{\mu}_t, \tag{2.16}$$

where $t = 1, 2$ (i.e., $t = 1$ for pCR and $t = 2$ for RD) and $\widehat{\mu}_t = \frac{1}{n_t} \sum_{i \in class_t} x_i$ is the within group average, calculated using the training data. We use the LDA score $\delta_t(Y)$ to classify the subject Y from the testing set. The rule for the classification is $\widehat{t} = \arg\max \delta_t(Y)$ $(t = 1, 2)$.

To measure the prediction accuracy for the three methods, we use specificity, sensitivity and Matthews Correlation Coefficient (MCC), as defined in section 2.5.2. Moreover, we consider TP and TN as the number of correctly predicted pCR and RD, respectively, and FP and FN as the number of erroneously predicted pCR and RD, respectively. Table 2.2 reports the average measures over 100 replications.

We observe that the proposed R-GLASSO for different values of k has a higher MCC than the GLASSO one, which indicates a better classification

[10] For the sake of time, we do not estimate the parameter k. We choose different values for this parameter.

performance. Moreover, we find that the proposed R-GLASSO method outperforms GLASSO in terms of sensitivity. We also observe that R-GLASSO outperforms CLIME in terms of sensitivity. On the other hand, CLIME outperforms GLASSO and R-GLASSO estimators in terms of specificity and MCC. However, we note that CLIME is computationally time-consuming.

Additionally, we repeat the same application by considering the most significant 200 genes instead of 113. We provide the results in Table 2.3.

TABLE 2.3: Average pCR/RD classification measurements over 100 replications for $p = 200$ genes.

Method	Specificity	Sensitivity	MCC
GLASSO	0.750	0.606	0.328
R-GLASSO $k = 2$	0.700	0.836	0.470
R-GLASSO $k = 3$	0.690	0.844	0.476
R-GLASSO $k = 4$	0.689	0.856	0.476
CLIME	0.712	0.838	0.483

As can be observed, the results are roughly similar to those obtained with 113 genes.

2.6.2 SRBC Tumour Data

In this application, we consider the problem of predicting the type of the Small Round Blue Cell (SRBC) tumours. The accurate prediction and diagnosis of the SRBC tumours is a major challenge, because the associated therapy and the treatment highly depend on the diagnosis (Khan et al. 2001). We use a dataset analysed by Khan et al. (2001), which contains the expression levels of 2308 genes for 64 tissue samples.[11] In this dataset, there are four types of SRBC tumours: 12 tissues of Neuroblastoma (NB), 21 tissues of Rhabdomyosarcoma (RMS), 8 tissues of Burkit Lymphoma, a subset of non-Hodgkin Lymphoma (BL), and 23 tissues of Ewing family tumours (EWS).

[11]The dataset is available at http://www.bioinf.ucd.ie/people/aedin/R/full/_datasets/.

First, we divide the data into a training set and a testing set with sizes 50 and 14, respectively. This process is repeated 100 times. To ensure that in both sets there are tissues of all four types, we obtain the training set by randomly selecting 18 tissues from the EWS class, 6 tissues from BL class, 9 tissues from NB class and 17 tissues from RMS class (around 70% of the subjects from each class). The remaining 14 tissues form the testing set. Second, we select the most significant 100 genes according to their F-statistics values. We rank the genes in the training set using the F-statistics (Rothman et al. 2009), defined as

$$F = \frac{\frac{1}{m-1}\sum_{i=1}^{m} n_i(\bar{x}_i - \bar{x})^2}{\frac{1}{n-m}\sum_{i=1}^{m}(n_i - 1)s_i^2}, \tag{2.17}$$

where $m = 4$ is the number of tumour classes, $n = 50$ is the number of tissue samples, n_i is the number of tissue samples of class i, $\bar{x}$ is the overall mean, $\bar{x}_i$ and s_i^2 are the sample mean and the variance of the class i, respectively. Finally, using the training set, we estimate the precision matrix Ω by GLASSO, R-GLASSO and CLIME methods. The penalty parameters for all three methods are estimated using the BIC criterion (2.7). We analyse the performance of the R-GLASSO method when the parameter k is selected from a range 2 to 4.[12] The estimated precision matrix is used in the LDA score $\delta_t(Y)$, defined as (2.16), where $t = 1, 2, 3, 4$ is the index of tumour class. To measure the prediction accuracy, we use the average proportion of correctly classified tissues:

$$AP = \frac{1}{100}\sum_{i=1}^{100}\frac{NCC_i}{14}, \tag{2.18}$$

where NCC_i is the number of correctly classified tissues in the i-th replication. We also repeat the same application by considering the most significant 200 genes instead of 100. We report the results for both cases in Table 2.4.

[12]For the sake of time, we do not estimate the parameter k. We choose different values for this parameter.

TABLE 2.4: Average proportion of correctly classified tissues over 100 replications.

Method	$p = 100$	$p = 200$
GLASSO	0.949	0.874
R-GLASSO $k = 2$	0.988	0.983
R-GLASSO $k = 3$	0.991	0.983
R-GLASSO $k = 4$	0.990	0.983
CLIME	0.988	0.982

We highlight that the average prediction level is higher for the R-GLASSO estimator than that for the GLASSO one. Moreover, we observe that R-GLASSO and CLIME provide similar results.

2.6.3 S&P 500 Portfolio Stock Selection

In our last application, we focus on developing a stock portfolio with minimum risk (i.e., variance). The precision matrix estimation plays a fundamental role in computing this optimal portfolio (Stevens 1998). It is well known that the weights of the (global) minimum variance portfolio are defined as:

$$w_{MVP} = \frac{\Omega \mathbf{1}_p}{\mathbf{1}_p' \Omega \mathbf{1}_p}, \tag{2.19}$$

(see DeMiguel et al. 2009) where $\mathbf{1}_p$ denotes a $p \times 1$ vector of ones. As the minimum-variance portfolio depends directly on the estimation of the precision matrix, an accurate estimation of such matrix may lead to a decrease of the out-of-sample risk or variance of the portfolio.

Following the empirical analysis by Goto and Xu (2013), we use monthly returns of the stock constituents of $S\&P$ 500 index for a total of $n = 240$ months.[13] We consider three different portfolios: a *small* portfolio with $p = 80$ of the largest stocks in the S&P 500 index, a *medium* portfolio with $p = 200$ randomly selected stocks and a *large* portfolio with $p = 300$ randomly selected stocks. To compute the estimated precision matrices,

[13]The observations cover the period of April 1st 1994 - April 1st 2014.

we apply the R-GLASSO, GLASSO and CLIME methods, using a "rolling-horizon" procedure as in DeMiguel et al. (2009). In particular, the rolling window contains 100 months, leaving 140 months to compute the out-of-sample portfolio variances for each procedure.

In this particular application, we do not calibrate the penalty parameters using the BIC criterion because the sparsity pattern of the estimated precision matrix does not have an important role. To select the penalty parameters for the precision estimation methods, we propose the following methodology based on cross-validation. For each estimation window of 100 months, we select the first 80 months to compute the precision matrices and leave the last 20 observations to minimize the corresponding portfolio variance over the penalty parameter. Because this procedure is time consuming, we apply this procedure in the first estimation window and then we fix the selected parameter along the rest of the out-of-sample period, as in Goto and Xu (2013). We consider different versions of the R-GLASSO procedure where the root k is fixed from 1 to 5 with increment of 0.5.

Table 2.5 shows the out-of-sample variances for the different portfolios. The

TABLE 2.5: The out-of-sample variances for different portfolios.

Methods	$p = 80$	$p = 200$	$p = 300$
GLASSO	0.00203	0.00143	0.00106
R-GLASSO $k = 1.5$	0.00157	0.00101	0.00103
R-GLASSO $k = 2$	0.00142	0.00091	0.00088
R-GLASSO $k = 2.5$	0.00141	0.00088	0.00090
R-GLASSO $k = 3$	0.00138	0.00229	0.00110
R-GLASSO $k = 3.5$	0.00155	0.00116	0.00106
R-GLASSO $k = 4$	0.00158	0.00168	0.00103
R-GLASSO $k = 4.5$	0.00161	0.00282	0.00100
R-GLASSO $k = 5$	0.00165	0.00462	0.00108
CLIME	0.00162	0.00650	0.00210

results show that the R-GLASSO method provides lower out-of-sample portfolio risk than that of the GLASSO method, especially for values of k around 2. We observe the same insights when comparing R-GLASSO with CLIME.

Chapter 3

DT Estimator Using Adaptive LASSO Penalties

3.1 Introduction

As discussed in the Chapter 2, the GLASSO method has become a state-of-the-art estimator for the precision matrix and one of the most applied approaches for covariance selection. We saw that the loss function of the GLASSO method is the log-likelihood function of the Gaussian model. Although the Gaussian assumption of data is quite restrictive, the GLASSO framework still provides a consistent estimator for non-Gaussian data (Ravikumar et al. 2011). However, the log-likelihood function may not be a comprehensible loss function because of its complex nature. Recently, Zhang and Zou (2014) introduced a so-called *D-trace loss* which has a much simpler structure. The D-trace (DT) loss has the following definition:

$$f_{DT}(\Omega, \Sigma) = \frac{1}{2}\text{trace}(\Omega^2\Sigma) - \text{trace}(\Omega). \tag{3.1}$$

The function $f_{DT}(\Omega, \Sigma)$ is convex in Ω, has a positive-definite Hessian matrix, and a unique minimizer at Σ^{-1}. We provide detailed proof of these statements in the remark E.4 of Appendix E.

Through numerical simulations, Zhang and Zou (2014) show that the ℓ_1 norm penalized D-trace loss minimization approach outperforms the GLASSO method in terms of different performance measures.

In this chapter, we focus on the ℓ_1 norm penalized D-trace loss minimization method (hereafter, DT method). It is well known that ℓ_1 penalty produces significant biases because of its overestimation feature (see, for instance, Zou 2006; Fan et al. 2009; Bühlmann and van de Geer 2011). The contribution of this chapter aimed to mitigate those biases. Based on the adaptive framework, we propose two re-weighted versions of the DT method. We employ adaptive thresholding operators in our proposed extensions. Previously, the adaptive framework has been applied in other context, such as variable selection (see Zou 2006; Zhou et al. 2009), precision matrix estimation (see Fan et al. 2009) and covariance matrix estimation (see Rothman et al. 2009). The advantage of the adaptive LASSO framework in high-dimensional settings is that it provides a stable and sparse estimator, simultaneously corrects the bias and, moreover, it does not augment the computational time.

Through extensive numerical simulations we show that the methods based on the proposed extensions outperform the original DT method. In particular, for the simulation study we consider different models, including those used in the simulation experiments by Zhang and Zou (2014). To measure the statistical performance of the methods, we use the entropy loss, the Frobenius norm loss, the operator norm loss and the matrix ℓ_1 norm loss. Furthermore, we use the percentages of correctly estimated zeros and non-zeros, accuracy and Matthews Correlation Coefficient (MCC) to measure the GGM prediction performance. Finally, we investigate the performance of the estimators in discriminant analysis using real datasets.

The rest of the chapter is organized as follows. In Section 3.2, after introducing some notations, we describe two extensions of the DT precision matrix estimation based on the adaptive LASSO framework. We consider the statistical loss and GGM prediction performance of the proposed estimators in Section 3.3 through exhaustive numerical simulations. We compare our proposed estimators with the DT and GLASSO estimators. In Section 3.4, we

apply the proposed methodologies to two real-world applications: the prediction of breast cancer state and the prediction of the colon cancer state. We provide the simulation results in Appendix C.

3.2 Proposed Methodologies

Zhang and Zou (2014) have proposed precision matrix estimation method DT through minimizing the off-diagonal ℓ_1 norm penalized D-trace loss function (3.1). The DT estimator is the solution of the following optimization problem:

$$\widehat{\Omega}_{\text{DT}} = \arg\min_{\Omega \succeq \epsilon I} \frac{1}{2}\text{trace}(\Omega^2 S) - \text{trace}(\Omega) + \tau ||\Omega||_{1,\text{off}}, \tag{3.2}$$

where $\tau > 0$ is the associated penalty parameter and ϵ is a small positive value. Note that in problem (3.2), the regularization of the matrix Ω is considered through its off-diagonal entries. We have selected the off-diagonal $||\Omega||_{1,\text{off}}$ penalty term to be consistent with the original article. On the other hand, we note that $||\Omega||_1$ penalty can also be used in the problem (3.2).

In this way, for this chapter we employ the estimator GLASSO as the solution of the off-diagonal ℓ_1 norm penalized log-likelihood function (1.1), defined as follows:

$$\widehat{\Omega}_{\text{GLASSO}} = \arg\max_{\Omega} \log\det\Omega - \text{trace}(S\Omega) - \nu||\Omega||_{1,\text{off}}, \tag{3.3}$$

The choice of the term $||\Omega||_{1,\text{off}}$ enables us to achieve fair comparison with the proposed method and with the results obtained by Zhang and Zou (2014).

To solve the problem (3.2), Zhang and Zou (2014) developed an algorithm, based on the *alternating direction method*. Previously, other authors have applied this algorithm for solving convex optimization problems (see, for instance, Scheinberg et al. 2010; Xue et al. 2012; Cui et al. 2014).

One of the important steps in the algorithm, where the LASSO penalty appears, is the following optimization problem:

$$\min_{\Omega=\Omega^T} \frac{1}{2}\text{trace}(\Omega^2) - \text{trace}(\Omega A) + \tau||\Omega||_{1,\text{off}}, \tag{3.4}$$

where the matrix A is defined in the algorithm process. One can show that the optimization problem (3.4) is strongly related to the *soft thresholding operator*. We provide the proof of the equality between the problem (3.4) and the soft thresholding operator in the remark E.5 of Appendix E. The solution $\hat{\Omega} = [\hat{\omega}_{ij}]_{1\leq i,j\leq p}$ of problem (3.4) can be written as:

$$\hat{\Omega} = T(A, \tau), \tag{3.5}$$

where T is the soft thresholding operator defined as follows:

$$\begin{aligned}[T(A,\tau)]_{ij} &= \text{sign}(A_{ij})\max(|A_{ij}| - \tau, 0)\mathbb{I}_{i\neq j} + A_{ij}\mathbb{I}_{i=j} \\ &= \begin{cases} A_{ij}, & \text{if } i = j, \\ A_{ij} - \tau, & \text{if } i \neq j, A_{ij} > \tau, \\ A_{ij} + \tau, & \text{if } i \neq j, A_{ij} < -\tau, \\ 0, & \text{if } i \neq j, -\tau \leq A_{ij} \leq \tau \end{cases},\end{aligned} \tag{3.6}$$

for $1 \leq i, j \leq p$.

As discussed in the Section 3.1, this work addresses the bias problem of the LASSO. From the regularization point of view, the ℓ_1 penalty may not be the best choice because of this issue. In order to reduce the bias of the DT estimator, produced through the LASSO regularization in (3.4) (or through the soft thresholding operator (3.6)), we propose two adaptive extensions of the DT estimator.

We propose our first adaptive approach, motivated by the idea of the adaptive GLASSO method provided by Fan et al. (2009). First, for a specific weight matrix $W = [w_{ij}]_{1\leq i,j\leq p}$, we define the *Weighted Adaptive Thresholding operator* as:

$$[WAT(A,\tau)]_{ij} = \text{sign}(A_{ij})\max(|A_{ij}| - \frac{\tau}{|w_{ij}|}, 0)\mathbb{I}_{i\neq j} + A_{ij}\mathbb{I}_{i=j}, \tag{3.7}$$

for $1 \leq i, j \leq p$. One can straightforwardly verify the following property of the weighted adaptive thresholding operator (3.7):

$$w_{ij} = 0 \implies [WAT(A, \tau)]_{ij} = 0, \tag{3.8}$$

for $1 \leq i, j \leq p$. The small w_{ij} weights imply large penalties for the (i, j) entries, whereas the large w_{ij} weights imply small penalties for the (i, j) entries.

Next, we can write the Weighted Adaptive Thresholding operator (3.7) as the solution of the following convex optimization problem:

$$\min_{\Omega=\Omega^T} \frac{1}{2}\text{trace}(\Omega^2) - \text{trace}(\Omega A) + \tau \sum_{i=1}^{p} \sum_{j=1, j\neq i}^{p} \frac{|\omega_{ij}|}{|w_{ij}|}. \tag{3.9}$$

Finally, by replacing the problem in (3.4) with the problem in (3.9), we derive our proposed *Weighted Adaptive D-trace estimator*, defined as follows:

$$\widehat{\Omega}_{\text{WADT}} = \arg\min_{\Omega \succeq \epsilon I} \frac{1}{2}\text{trace}(\Omega^2 S) - \text{trace}(\Omega) + \tau \sum_{i=1}^{p} \sum_{j=1, j\neq i}^{p} \frac{|\omega_{ij}|}{|w_{ij}|}. \tag{3.10}$$

Essentially, the matrix W is a prior information about the precision matrix or any consistent, computationally cheap estimator (e.g., a well-defined two-step estimator) and, therefore, should be chosen properly.

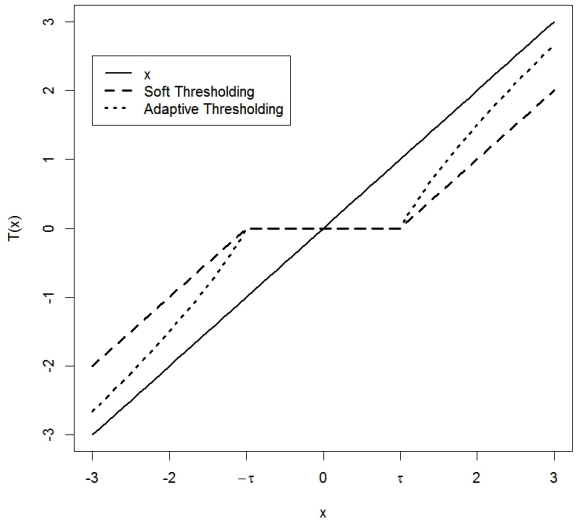
Our second adaptive approach is motivated by Rothman et al. (2009), where we use the *Adaptive Thresholding operator*, defined as follows:

$$[AT(A, \tau)]_{ij} = \text{sign}(A_{ij}) \max(|A_{ij}| - \frac{\tau}{|A_{ij}|}, 0)\mathbb{I}_{i\neq j} + A_{ij}\mathbb{I}_{i=j}, \tag{3.11}$$

for $1 \leq i, j \leq p$. The operator (3.11) can be considered as a special case of the operator (3.7), when $w_{ij} = A_{ij}$, $1 \leq i, j \leq p$. To illustrate the idea, Figure 3.1 depicts the soft and the adaptive thresholding operators for $\tau = 1$.

The main advantage of the operator (3.11) is the absence of a weight matrix. Through the Adaptive Thresholding operator (3.11), the large entries A_{ij} are penalized less and the small entries are penalized more. In other words, the

FIGURE 3.1: Soft and Adaptive thresholding functions for $\tau = 1$.



operator (3.11) overestimates less than the soft threshodling operator (3.6) since many smaller values will be discarded. Hence, the operator (3.11) provides smaller bias than the operator (3.6) (i.e., the LASSO penalization). As with the Weighted Adaptive D-trace estimator, one can derive formulations similar to (3.9) and (3.10) for the Adaptive Thresholding operator. However, in these formulations the weight matrix W can not be defined directly, since the matrix A appears in the solver and is not fixed. We can obtain the D-trace estimator through the Adaptive Thresholding operator (3.11) by simply replacing the soft thresholding operator (3.6) with the operator (3.11) in the algorithm (see the Algorithm 1 bellow for more details). We call the estimator obtained through the operator (3.11) the *Adaptive D-trace estimator* $\hat{\Omega}_{ADT}$.

For completeness, we present the algorithm for solving the DT method and the necessary modifications for solving WADT and ADT methods. We first provide definitions of some functions employed in the algorithm. Assume that $A = UVU^T = U\text{diag}(v_1, ..., v_p)U^T$ is the eigen-decomposition[1] of any

[1]For a vector $a = (a_1, ..., a_p)$ we set $\text{diag}(a_1, ..., a_p)$ a diagonal matrix with entries a_i. For a matrix A we set $\text{diag}(A)$ a diagonal matrix, which has the diagonal entries of A.

$p \times p$ symmetric matrix $A \succ 0$ and $v_1 \geq ... \geq v_p$ are its eigenvalues. For any $p \times p$ matrix B, define

$$G(A, B) = U\{(U^T BU) \circ C\}U^T, \tag{3.12}$$

where $C_{i,j} = \dfrac{2}{v_i + v_j}$ for $1 \leq i, j \leq p$ and $\circ$ denotes the Hadamard product of matrices. For any symmetric matrix A and any $\epsilon > 0$, define

$$[A]_+ = U\text{diag}\{\max(v_1, \epsilon), ..., \max(v_p, \epsilon)\}U^T. \tag{3.13}$$

Algorithm 1 provides the necessary steps for solving our proposed estimation methods:

Algorithm 1 Alternating direction method

Step 1. Initialization: $k = 0$, $\Lambda_0^0 = \Lambda_1^0$, $\Theta_0^0 = \Theta_1^0$.

Step 2. Repeat the following sub-steps until convergence:

(a) Set k=k+1.

(b) Compute the matrix $\Theta^{k+1} = G(S + 2\rho I, I + \rho\Theta_0^k + \rho\Theta_1^k - \Lambda_0^k - \Lambda_1^k)$, where function G is defined in (3.12).

(c) Set $\Theta_1^{k+1} = [\Theta^{k+1} + \Lambda_1^k/\rho]_+$. Compute $\Theta_0^{k+1} = T(\Theta^{k+1} + \Lambda_0^k/\rho, \tau/\rho)$ in case of DT estimator, $\Theta_0^{k+1} = WADT(\Theta^{k+1} + \Lambda_0^k/\rho, \tau/\rho)$ in case of WADT estimator and $\Theta_0^{k+1} = ADT(\Theta^{k+1} + \Lambda_0^k/\rho, \tau/\rho)$ in case of ADT estimator. The thresholding functions T, WADT and ADT are defined in (3.6), (3.7) and (3.11), respectively.

(d) Set $\Lambda_0^{k+1} = \Lambda_0^k + \rho(\Theta^{k+1} - \Theta_0^{k+1})$ and $\Lambda_1^{k+1} = \Lambda_1^k + \rho(\Theta^{k+1} - \Theta_1^{k+1})$.

It is important to note that we can significantly reduce the computational time of the Algorithm 1 by discarding the constraint $\Omega \succeq \epsilon I$ in the initial optimization problem (DT, WADT or ADT). This enables us to omit the function (3.13) from the step 2c, which is the most computationally expensive part of the algorithm. We call the optimization problem without the

constraint $\Omega \succeq \epsilon I$ as the secondary problem, defined as follows:

$$\tilde{\Omega} = \arg\min_{\Omega^T=\Omega} \frac{1}{2}\text{trace}(\Omega^2 S) - \text{trace}(\Omega) + \tau \text{PEN}(\Omega), \tag{3.14}$$

where PEN(Ω) term is defined according to the estimation method (DT, ADT or WADT). Following Zhang and Zou (2014), we also present the simplified version of the Algorithm 1.

Algorithm 2 Alternating direction method (simplified)

Step 1. Initialization: $k = 0$, Λ^0, $\Theta_0^0 = \text{diag}(S)^{-1}$.

Step 2. Repeat the following sub-steps until convergence:

(a) Set k=k+1.

(b) Compute the matrix $\Theta^{k+1} = G(S + \rho I, I + \rho\Theta_0^k - \Lambda^k)$.

Compute $\Theta_0^{k+1} = T(\Theta^{k+1} + \Lambda^k/\rho, \tau/\rho)$ in case of DT estimator, $\Theta_0^{k+1} = WADT(\Theta^{k+1} + \Lambda^k/\rho, \tau/\rho)$ in case of WADT estimator and $\Theta_0^{k+1} = ADT(\Theta^{k+1} + \Lambda^k/\rho, \tau/\rho)$ in case of ADT estimator.

(d) Set $\Lambda^{k+1} = \Lambda_0^k + \rho(\Theta^{k+1} - \Theta_0^{k+1})$.

Step 3. Consider the converged Θ^k as the solution of the secondary problem (3.14).

Step 4. If $\lambda_{\min}(\tilde{\Theta}) > \epsilon$, report $\tilde{\Theta}$ as the solution of the initial problem. Otherwise, use Algorithm 1 with $\tilde{\Theta}$ as the starting value for Θ_0^0 and Θ_1^0.

The algorithm stops if the following two conditions are satisfied:

$$\frac{||\Theta^{k+1} - \Theta^k||_2}{\max(1, ||\Theta^k||_2, ||\Theta^{k+1}||_2)} < 10^{-7}, \quad \frac{||\Theta_0^{k+1} - \Theta_0^k||_2}{\max(1, ||\Theta_0^k||_2, ||\Theta_0^{k+1}||_2)} < 10^{-7}.$$

Finally, in the algorithm we use $\rho = 1$ and $\epsilon = 10^{-8}$. For more details we refer to Zhang and Zou (2014).

3.3 Simulation Study

In this section, we implement a simulation study to show the goodness of the proposed WADT and ADT estimators and to compare their associated performance with those of the DT estimator and the state-of-the-art estimator GLASSO. Particularly, in subsection 3.3.1, we introduce the models considered for the true precision matrix Ω. In subsection 3.3.2, we describe the performance evaluation. In subsection 3.3.3, we provide the discussion of the obtained results.

3.3.1 Considered Models

We perform an exhaustive numerical simulation study through eight different sparsity configurations for the precision matrix, including random and fixed patterns. The considered models for the true precision matrix Ω are the following:

- *Model* 1. AR(2) structure: $\omega_{i,i} = 1$, $\omega_{i,j} = 0.2$ for $1 \leq |i-j| \leq 2$, and zero otherwise.
- *Model* 2. AR(4) structure: $\omega_{i,i} = 1$, $\omega_{i,j} = 0.2$ for $1 \leq |i-j| \leq 4$, and zero otherwise.
- *Model* 3. A matrix with $\omega_{i,i} = 1$, $\omega_{i,i+1} = 0.2$ for $\text{mod}(i, p^{1/2}) \neq 0$, $\omega_{i,i+p^{1/2}} = 0.2$, and zero otherwise.
- *Model* 4. AR(1) structure: $\omega_{ii} = 1$, $\omega_{i,i-1} = \omega_{i-1,i} = 0.45$, and zero otherwise.
- *Model* 5. (Modified) AR(1) structure with different entries: $\Omega = D^{1/2}\Omega_{AR(1)}D^{1/2}$, where $D = \text{diag}(D_1, ..., D_p)$ with $D_i = \dfrac{4i+p-5}{5(p-1)}$ and $\Omega_{AR(1)}$ is a matrix with a structure defined in the model 4.
- *Model* 6. Decay structure: $\omega_{ij} = 0.6^{|i-j|}$.
- *Model* 7. A random positive-definite matrix, containing 5% of non-zero entries.

- *Model* 8. A random positive-definite matrix, containing 10% of non-zero entries.[2]

Our choice of these models is motivated as follows. To compare our proposed methods with Zhang and Zou (2014), we consider the models employed in their study (models 1, 2 and 3). In addition, we consider other models commonly used in the prior literature, such as AR(1) structure model (model 4 - Yuan and Lin (2007), Friedman et al. (2008)), its modified version (model 5) and decay structure model (model 6 - Cai et al. (2011), Fan et al. (2009)). Note that models 1-6 have deterministic patterns. We study the performance of the considered methods also using models with random patterns (models 7 and 8). This allows us to obtain more robust evaluation and to have better insight about the performance of the estimation methods.

Consistent with Zhang and Zou (2014), we simulate multivariate normal random samples with zero mean and sample size $n = 400$, for each of the models. For the number of variables, we choose $p = 484$ for model 3 and $p = 500$ for the other models.[3] These values allow us to examine the performance of the proposed estimators in high-dimensional settings and, especially, when $p > n$. Finally, we repeat this procedure 100 times.

3.3.2 Performance Evaluation

Similar to Zhang and Zou (2014), to evaluate the statistical performance of a given estimator $\widehat{\Omega}$, we consider the Frobenius norm ℓ_2, the spectral norm ℓ_{spec} and the matrix ℓ_1 norm, defined respectively as:

$$\ell_2(\widehat{\Omega}, \Omega) = ||\widehat{\Omega} - \Omega||_2, \tag{3.15}$$

$$\ell_{\text{spec}}(\widehat{\Omega}, \Omega) = ||\widehat{\Omega} - \Omega||_{\text{spec}}, \tag{3.16}$$

[2]Models 7 and 8 are generated using the Matlab command *sprandsym*.
[3]For model 3, $p^{1/2}$ is required to be an integer.

and

$$\ell_1(\widehat{\Omega}, \Omega) = ||\widehat{\Omega} - \Omega||_{\ell_1}. \tag{3.17}$$

Next, we consider the entropy loss function, also known as Kullback-Leibler Loss function (Kullback and Leibler 1951), defined as follows:

$$\text{KLL}(\widehat{\Omega}, \Omega) = \text{trace}(\Omega^{-1}\widehat{\Omega}) - \log\det(\Omega^{-1}\widehat{\Omega}) - p. \tag{3.18}$$

In order to evaluate the sparsity pattern or GGM estimation performance, we compute the percentages of correctly estimated non-zeros and zeros (also known as sensitivity and specificity, respectively) and the accuracy of classification, defined respectively as:

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100, \tag{3.19}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \times 100, \tag{3.20}$$

and

$$\text{Accuracy} = \frac{\text{TN+TP}}{p^2} \times 100. \tag{3.21}$$

We define TP, TN, FP and FN are defined in the Chapter 2. We also compute the Matthews Correlation Coefficient (MCC), which is defined as follows:

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}}. \tag{3.22}$$

In order to select the penalty parameters ν and τ, in line with Zhang and Zou (2014), we use *five-fold CV (cross-validation) technique*. This technique is defined as follows. We divide the sample data into five disjoint subgroups (i.e., folds). We denote the index of the observations in the k-th fold by T_k, for $k = 1, ..., 5$. We define five-fold CV score by

$$CV(\tau) = \sum_{k=1}^{5} \left(n_k \log\det(\hat{\Omega}_{-k}(\tau)) - \sum_{i \in T_k} X_i \hat{\Omega}_{-k}(\tau) X_i^T \right), \tag{3.23}$$

where n_k is the size of the k-th fold[4] and $\hat{\Omega}_{-k}(\tau)$ is the precision matrix estimate obtained using the sample $X \setminus T_k = \cup_{i=1}^{5} T_i \setminus T_k$ and the tuning parameter τ. Finally, we select the penalty parameter estimate by maximizing the score CV(τ) using a grid search.

For the WADT estimator, as a weight matrix we choose the inverse of the popular Ledoit-Wolf shrinkage covariance estimator, i.e., $W = \hat{\Sigma}_{LW}^{-1}$. Ledoit and Wolf (2004) proposed this covariance estimator as the following:

$$\hat{\Sigma}_{LW} = (1-\alpha)S + \alpha \frac{\text{trace}(S)}{p} I, \tag{3.24}$$

where $\alpha \in [0,1]$ is the shrinkage parameter. Finally, we use the Matlab code of Zhang and Zou (2014) to implement the algorithm for the DT method and the modification of their code for the WADT and ADT estimators.

3.3.3 Discussion of Results

We provide the simulation results in the Appendix C to conserve space. Tables C.1-C.8 report the averages of the corresponding losses and measurements over 100 replications. The standard deviations (SD) are given in parentheses. Tables C.5, C.6 and C.8 provide the measurements in percentages. We organize the discussion of our results as follows. We first compare our proposed estimators ADT and WADT with the DT estimator. We then compare our proposed estimators ADT and WADT with the GLASSO estimator. We finally compare the DT estimator with the GLASSO estimator.

We report the statistical losses in Tables C.1-C.4. We observe that for most of the models either the ADT or the WADT estimator provides the lowest losses versus the other methods (DT and GLASSO). More specifically, the ADT estimator provides the lowest KLL for models 1, 2, 6, the lowest Frobenius norm and spectral norm for models 2, 6 and the lowest matrix ℓ_1 norm for

[4] We set $n_k = \frac{n - \text{mod}(n,5)}{5}$, for all $k = 1, ..., 5$.

models 1, 3, 6. On the other hand, the WADT estimator provides the lowest KLL for models 3, 4, 5, 7, 8, the lowest Frobenius norm and spectral norm for models 1, 3, 4, 5, 7, 8 and the lowest matrix ℓ_1 norm for models 4, 5, 7, 8. The only exception when the ADT estimator fails to outperform the DT estimator is for models 2, 8 in terms of matrix ℓ_1 norm and for models 1, 3, 7 in terms of spectral norm. The only exception when the ADT estimator fails to outperform the GLASSO method is for model 3 in terms of KLL. The only exception when the WADT estimator fails to outperform the DT estimator is for models 1, 2 (only in terms of matrix ℓ_1 norm) and for model 6, which is precisely a dense model. The WADT method outperforms GLASSO method in all the models.

The comparison of the performances of DT versus GLASSO yields the following insights. In line with Zhang and Zou (2014), we find that DT outperforms GLASSO for all the models in terms of Frobenius norm, spectral norm, and ℓ_1 norm. However, in their work, Zhang and Zou (2014) did not compare DT and GLASSO in terms of KLL. We find mixed results in comparative performance of DT versus GLASSO. We observe that DT outperforms GLASSO for models 1, 2, 5, 6, 7, 8 in terms of the KLL. In contrast to Zhang and Zou (2014), we find that DT fails to outperform GLASSO for models 3 and 4 in terms of KLL.

We report the GGM prediction performance in Tables C.5-C.8.[5] We observe that for most of the models either the ADT or the WADT estimator provides better GGM prediction performance than the other methods (DT and GLASSO). More specifically, the ADT estimator provides the highest specificity for models 1, 3, the highest sensitivity for model 2 and the highest MCC and accuracy for models 1, 2, 3. On the other hand, the WADT estimator provides the highest specificity for models 2, 4, 5, 7, 8 and the highest MCC and accuracy for models 4, 5, 7, 8. All the estimators provide the same sensitivity for models 4 and 5. The only exception when our proposed estimators (ADT and WADT) fail to outperform the DT estimator is for models 1, 3, 6, 7, 8 only in terms of sensitivity. Note that the weak performance in

[5]Specificity, MCC and accuracy are excluded for model 6 because these measures are defined only for sparse models.

terms of sensitivity is due to the adaptive framework. As mentioned earlier, this framework omits small values, which leads to sparser precision matrix estimator. However, for those models the DT estimator fails to outperform the estimators ADT and WADT in terms of the overall GGM prediction measures MCC and accuracy. In addition, the only exception when our proposed estimators fail to outperform the GLASSO estimator is for models 3, 6, 7, 8 in terms of sensitivity. However, the GLASSO estimator fails to outperform the proposed estimators in terms of the overall GGM prediction measures MCC and accuracy for those models. Comparing the DT estimator with the GLASSO estimator our findings show that the later outperforms the DT estimator for models 3, 6, 7, 8 in terms of sensitivity and for model 3 in terms of specificity. In terms of the overall GGM prediction measures the DT estimator outperforms the GLASSO estimator for all the models except for model 3, where the GLASSO provides slightly higher accuracy and MCC than DT.

As a summary, our proposed adaptive approaches ADT and WADT outperform DT and GLASSO for overwhelming majority of the considered models. In spite of few exceptions, the proposed methods provide better performance in terms of the statistical losses and GGM prediction measures, than the competitive methods. In addition, our findings show that the WADT method provides relatively better results than the ADT method when the required weight matrix is the inverse of an estimated covariance matrix.

3.3.4 Comparison with R-GLASSO method

In this subsection, we provide a simulation study to compare the precision estimation measures of the estimators ADT and WADT with those of the estimator R-GLASSO (see Chapter 2). Thus, for an accurate and equitable comparison we consider the models and evaluation measures employed in numerical study of the Chapter 2 (see subsections 2.5.1 and 2.5.2, respectively). Moreover, we select the penalty parameter τ using the BIC criterion, proposed by Yuan and Lin (2007) and defined in (2.7). We recall that the penalty term of the R-GLASSO estimator is applied to all the entries of the

estimated matrix (including the diagonal elements). Therefore, in the simulations below we employ DT, ADT and WADT methods with a penalization term, which includes all the entries of the estimated matrix.[6]

We provide the simulation results in the Appendix D to conserve space. Tables D.1-D.5 report the averages of the corresponding losses and measurements over 100 replications. The standard deviations (SD) are given in parentheses. We organize the discussion of our results as follows. We first compare our proposed estimators ADT and WADT with the DT estimator as an addition to the numerical study of the subsection 3.3.3. We then compare the obtained measures of ADT and WADT with those obtained for the R-GLASSO estimator.

We report the statistical losses in Tables D.1-D.2. We observe that the ADT estimator provides the lowest KLL for model 7. On the other hand, the WADT estimator provides the lowest KLL for models 1, 2, 3, 4, 5, 6 and the lowest MSE for all the models. The only exception when the ADT estimator fails to outperform the DT estimator is for model 1 in terms of KLL (only when $p = 300$).

We report the GGM prediction performance in Tables D.3-D.5.[7] We observe that the ADT estimator provides the highest MCC for model 7. On the other hand, the WADT estimator provides the highest specificity for models 1, 2, 3, 4, 5, 7, the highest sensitivity for models 1, 2, 3, 4 and the highest MCC for models 1, 2, 3, 4, 5. All the estimators provide the same sensitivity for model 5. However, the ADT fails to outperform the DT estimator for models 1, 2, 3, 4, 6 in terms of sensitivity, for models 1, 2, 3, 4 and 5 (only when $p = 300$) in terms of specificity and MCC. The WADT fails to outperform the DT estimator for models for model 6 and 7 in terms of sensitivity.

Now we compare the measures of ADT and WADT from Tables D.1-D.5 with their corresponding measures of R-GLASSO from Tables B.1-B.5. We

[6] In this particular subsection, we consider the penalty term $||\Omega||_1$ instead of $||\Omega||_{1,\text{off}}$ in the optimization problem (3.2).

[7] Specificity, MCC and accuracy are excluded for model 6 because these measures are defined only for sparse models.

observe that both ADT and WADT estimators outperform R-GLASSO estimator for all the models in terms of KLL. The ADT estimator outperforms R-GLASSO for model 5 in terms of MSE and provides similar results for models 1, 2, 3, 4. On the other hand, WADT estimator outperforms R-GLASSO for models 1, 2, 3, 4, 5 in terms of MSE. However, R-GLASSO outperforms ADT and WADT for models 6 and 7 in terms of MSE.

We obtain the following comparison regarding the GGM prediction performance. Firsly, we observe that both ADT and WADT estimators outperform R-GLASSO estimator for models 5, 7 in terms of the specificity. However, R-GLASSO outperforms ADT and WADT for models 1, 2, 3, 4 in terms of specificity. Secondly, the ADT estimator outperforms R-GLASSO estimator for models 4 and provides similar results for models 2, 3 in terms of sensitivity. On the other hand, WADT estimator outperforms R-GLASSO estimator for models 1, 2, 3, 4 in terms of sensitivity. However, R-GLASSO outperforms ADT estimator for models 1, 6, 7 and WADT estimator for models 6, 7 in terms of sensitivity. All the estimators provide the same sensitivity for model 5. Finally, we observe that the ADT estimator outperforms R-GLASSO estimator for model 5 in terms of MCC. On the other hand, the WADT estimator outperforms R-GLASSO for models 1, 2, 3, 5. However, R-GLASSO outperforms ADT estimator for models 1, 2, 3, 4, 7 and WADT estimator for models 4, 7 in terms of MCC.

3.4 Real Data Applications

In this section, we perform an empirical analysis of the proposed adaptive approaches through real-data examples. In particular, we use breast cancer and colon cancer datasets to predict the tumour behaviour using Linear Discriminant Analysis (LDA). All applied datasets are available in the web site of the *National Center for Biotechnology Information*.[8]

[8]Available at http://www.ncbi.nlm.nih.gov/.

3.4.1 Breast Cancer Data

In the first application, we focus on the problem of predicting breast cancer patients (subjects) with pathological complete response (pCR). This is an important issue because after the neoadjuvant chemotherapy, according to Kuerer et al. (1999), the pCR indicates a cancer-free life with high probability. For this application we use a dataset (see Shi et al. 2010) containing gene expression levels of subjects with different stages of breast cancer. The dataset consists of 22,283 gene expression levels of 271 subjects. There are 58 subjects with pCR and 213 subjects with residual disease (RD).

First, we divide the data into a training set and a testing set with sizes 227 (almost 5/6 of the observations) and 44 (almost 1/6 of the observations), respectively, and repeat this process 100 times. For the testing set, we randomly select 9 subjects with pCR and 35 subjects with RD (roughly proportional to the number of the subjects in each group). The training set contains the remaining subjects. Second, based on the training set we perform two sample t-tests between the two groups in order to select the most significant 100 genes with the smallest p-values. Third, using the training set, we estimate the precision matrix Ω with the DT, ADT, WADT and GLASSO methods. We obtain the penalty parameters for these methods using five-fold cross-validation technique. Finally, we use the estimated precision matrix in the LDA score, defined as follows:

$$\delta_t(Y) = Y^T \widehat{\Omega} \widehat{\mu}_t - \frac{1}{2} \widehat{\mu}_t^T \widehat{\Omega} \widehat{\mu}_t, \tag{3.25}$$

where $t = 1, 2$ ($t = 1$ for pCR and $t = 2$ for RD) and $\widehat{\mu}_t = \frac{1}{n_t} \sum_{i \in class_t} x_i$ is the within group average, calculated using the training data. We use the LDA score $\delta_t(Y)$ to classify the subject Y from the testing set. The rule for the classification is $\widehat{t} = \arg\max \delta_t(Y)$. To measure the prediction accuracy for all the methods, we use the specificity, sensitivity and Matthews Correlation Coefficient (MCC), as defined in Section 3.3.2. We consider TP and TN as the number of correctly predicted RD and pCR, respectively, and FP and FN as the number of erroneously predicted RD and pCR, respectively. We report the average measurements over 100 replications in Table 3.1.

TABLE 3.1: Average pCR/RD classification measurements over 100 replications for $p = 100$ genes.

Methods	Specificity	Sensitivity	MCC
GLASSO	0.4800	0.7751	0.2333
DT	0.6556	0.7537	0.3572
ADT	0.6989	0.7409	0.3782
WADT	0.7211	0.7334	0.3889

Our findings show that the GLASSO provides the highest sensitivity, but it attains the lowest specificity and MCC. On the other hand, the adaptive approach WADT provides the highest specificity and dominates all the other estimators in terms of MCC. Furthermore, the ADT and WADT estimators show similar results, the latter being slightly better.

To check the robustness of the obtained results, we repeat the same application by considering the most significant 200 genes instead of 100. Table 3.2 reports the results. Our findings show that the results are roughly similar to those obtained with 100 genes. The methods ADT and WADT outperform DT and GLASSO methods in terms of the overall measurement MCC.

TABLE 3.2: Average pCR/RD classification measurements over 100 replications for $p = 200$ genes.

Methods	Specificity	Sensitivity	MCC
GLASSO	0.4600	0.7891	0.2310
DT	0.6333	0.7620	0.3459
ADT	0.7033	0.7394	0.3793
WADT	0.7089	0.7414	0.3860

3.4.2 Colon Cancer Data

In the second application, we consider the problem of classifying the colorectal cancer patients with Microsatellite Stability (MSS) state and Microsatellite Instability (MSI) state. The dataset (see Jorissen et al. 2008) contains the expression levels of 54,675 genes for 155 colorectal cancer samples. There are 77 MSS and 78 MSI specimens in the dataset.

As with the first application, we divide the data into a training set and a testing set with sizes 130 (almost 5/6 of the observations) and 25 (almost

1/6 of the observations), respectively, and repeat this process 100 times. We randomly select 12 MSS and 13 MSI specimens (roughly proportional to the number of the subjects in each group), respectively, for the testing set and the training set contains the remaining subjects. Again, we select the 100 most-significant genes and estimate the precision matrix Ω with the DT, ADT, WADT and GLASSO methods. We obtain the penalty parameters for these methods using five-fold cross-validation technique. Finally, we use the estimated precision matrix in the LDA score (3.25), where $t = 1$ is for MSS specimens and $t = 2$ is for MSI specimens.

Table 3.3 shows the average performance measures over the 100 replicates. We observe that GLASSO provides the lowest performance measures while the WADT estimator provides the highest ones. The DT and ADT estimators provide relatively similar results.

TABLE 3.3: Average MSI/MSS classification measurements over 100 replications for $p = 100$ genes.

Methods	Specificity	Sensitivity	MCC
GLASSO	0.9258	0.8961	0.8262
DT	1	0.8977	0.9020
ADT	1	0.8915	0.8966
WADT	1	0.9208	0.9235

We repeat the same application by considering the most significant 200 genes instead of 100. Table 3.4 provides the results. We observe that the results are similar to those obtained using 100 genes.

TABLE 3.4: Average MSI/MSS classification measurements over 100 replications for $p = 200$ genes.

Methods	Specificity	Sensitivity	MCC
GLASSO	0.8558	0.8330	0.6956
DT	1	0.9015	0.9050
ADT	1	0.9054	0.9086
WADT	1	0.9238	0.9258

In sum, our findings show that in the considered applications the proposed WADT and ADT methods are able to provide better classification performance than DT and GLASSO estimators.

Chapter 4

Conclusions and Future Research

4.1 Conclusions

Accurate estimation of the precision matrix is an important and attractive problem because it has an essential role in various methodologies and research fields. This problem is quite challenging when the dimensionality has the same order as the sample size or is much larger. The main goal of this thesis is to develop new estimation methods for the precision matrix in high-dimensional statistical settings. Moreover, the proposed estimators should provide competitive performance comparing with existing prominent estimators.

In this thesis, we propose and analyse two novel approaches, which provide proper precision matrix estimator for high-dimensional problems. The numerical results show that our proposed precision estimators are found to compare favourably with the state-of-the-art estimators (e.g., GLASSO, CLIME, etc.) in terms of several measures, even when the number of the variables exceeds the sample size. Moreover, our proposed estimators provide advantageous numerical properties in terms of GGM prediction.

In Chapter 2, we provide a new approach for estimating high-dimensional precision matrices, using ℓ_1 penalization framework. The proposed method is a simple modification of the popular GLASSO approach based on performing a k-root transformation of the sample covariance matrix which allows to reduce the spread of the corresponding eigenvalues. Through an extensive analysis, using both simulated and real data sets, we show numerically that the proposed improvement helps to achieve better performance without having to increase considerably the computational burden. In particular, the proposed R-GLASSO method provides lower statistical losses and higher accuracy for the prediction of GGM, than those for the GLASSO method. Moreover, the proposed procedure attains better results to CLIME, being computationally less demanding. Our proposed method requires the calibration of an additional parameter k associated with the root transformation. We propose a calibration procedure based on the BIC criterion. However, our results show that the square root transformation (e.g., $k = 2$) can be a reasonable choice in practice. Finally, we establish the convergence rate of the proposed R-GLASSO estimator in the Frobenius norm, under certain conditions.

In Chapter 3, we develop two novel approaches for estimating the precision matrix, based on the adaptive ℓ_1 regularization framework. We extend the recently introduced D-trace estimator to Weighted Adaptive D-trace (WADT) and Adaptive D-trace (ADT) estimators to correct the bias of the estimated precision matrix produced by the ℓ_1 penalty. In our proposed methodologies, we use the adaptive thresholding operators. We conduct an extensive numerical analysis, applying both simulated and real data sets. For the WADT estimator we use the two-step precision matrix estimator as a weight matrix. Our findings show that it is a practical choice. We use different loss functions and prediction performance measures for the evaluation. The results show that the proposed estimators outperform the DT and GLASSO estimators. In particular, the WADT and ADT estimators provide lower statistical losses and higher GGM prediction measures than those for the DT and GLASSO methods.

4.2 Future Research Directions

In this thesis, we have proposed effective methods to estimate the precision matrix in high-dimensional settings. However, we note that other interesting approaches could be employed for this purpose. In this subsection, we present several directions for the future research.

1. As discussed in this thesis, the ℓ_1 norm penalty guarantees the sparsity pattern of the precision matrix estimator for a well-selected penalty parameter. However, it does not control the eigenvalues of the estimator. Maurya (2014) showed that imposing an additional penalty of sum of the eigenvalues (i.e., the trace) on the objective function of the GLASSO method can improve the performance of the precision estimator.

Our analysis shows that an additional *negative trace penalization* of the DT method significantly improves the performance of the precision estimator in terms of the norm losses. We propose the following estimator:

$$\begin{aligned}\hat{\Omega} &= \arg\min_{\Omega} \frac{1}{2}\text{trace}(\Omega^2 S) - \text{trace}(\Omega) + \tau||\Omega||_{1,\text{off}} - \gamma\text{trace}(\Omega) \\ &= \arg\min_{\Omega} \frac{1}{2}\text{trace}(\Omega^2 S) - (1+\gamma)\text{trace}(\Omega) + \tau||\Omega||_{1,\text{off}}.\end{aligned} \tag{4.1}$$

Note that we can solve problem (4.1) using the same algorithm employed for the DT method.

2. Most of the methods in the literature assume that the data are *independent and identically distributed*. In other words, we assume that the precision matrix and, therefore, the corresponding GGM or the network are time-invariant. However, the recent research shows that in the real-world applications the time-invariance of the graphical structure often fails. In this way, the graphical structure *evolves over time* (i.e., is time-varying) and the data are not identically distributed. For example, fMRI research shows that brain connectivity networks are often not stable over time and an additional study is required to examine the dynamic changes of those networks over time. In genetic studies, there is also a need for learning the time-varying gene interaction network structure. On the other hand, the time-varying networks are effective way for representing the interactions

between varying stocks over time. Therefore, they are of high importance for clearly understanding the stock markets. As discussed in this thesis, a large amount of literature is dedicated to precision matrix estimation and non-evolving graphical models using independent and identically distributed data. To the best of our knowledge, a very small amount of research studies the time-varying networks. Among the notable studies are by Zhou et al. (2010); Chen et al. (2013); Monti et al. (2014). We note that all the existing methods are based on the popular GLASSO method. In other words, the methods use the following log-likelihood function at a given time t:

$$\ell(\mathbf{X}^t, \Omega, t) = \log\det\Omega - \text{trace}(\Omega S(t)), \tag{4.2}$$

where $S(t)$ is the kernel estimate of the sample covariance matrix and has the following definition:

$$S(t) = \frac{\sum\limits_i w_{it} X_i X_i^T}{\sum\limits_i w_{it}}. \tag{4.3}$$

The weights are given as $w_{it} = K\left(\frac{|i-t|}{h}\right)$, where K is a symmetric, non-negative kernel function with the bandwidth parameter h.

We suggest to consider the following D-trace function at a given time t:

$$f_{DT}(S(t), \Omega, t) = \frac{1}{2}\text{trace}(\Omega^2 S(t)) - \text{trace}(\Omega), \tag{4.4}$$

where $S(t)$ is defined in (4.3). In this way, by penalizing the function $f_{DT}(S(t), \Omega, t)$, we propose the methods DT, ADT and WADT for time-varying network estimation. The numerical study of Chapter 3 shows that for time-invariant data these methods outperform the GLASSO method for most of the cases (especially, our proposed ADT and WADT methods). Therefore, we expect that for time-varying data the methods DT, ADT and WADT will outperform GLASSO method.

3. The inverse covariance shrinkage estimator is very popular in portfolio selection (see, for instance, Frahm and Memmel 2010; Kourtis et al. 2012;

DeMiguel et al. 2013), assuming that $n > p$. Precision shrinkage estimator has the following form:

$$\hat{\Omega}_{\text{shrink}} = (1-\alpha)S^{-1} + \alpha F_1, \tag{4.5}$$

where $\alpha \in [0,1]$ is the shrinkage intensity parameter and F_1 is a target matrix.

Based on the shrinkage concept, we propose the following precision shrinkage estimators which can be used also under high-dimensional settings:

$$\hat{\Omega}_1 = (1-\alpha)\left((1-\alpha)S + \alpha F_2\right)^{-1} + \alpha F_3, \tag{4.6}$$

$$\hat{\Omega}_2 = (1-\alpha)\left(S + \alpha F_2\right)^{-1} + \alpha F_3, \tag{4.7}$$

where F_2 and F_3 are target matrices. In contrast to the estimator $\hat{\Omega}_{\text{shrink}}$, the estimators $\hat{\Omega}_1$ and $\hat{\Omega}_2$ are defined when $n < p$. Moreover, our numerical analysis showed that the proposed estimators $\hat{\Omega}_1$ and $\hat{\Omega}_2$ outperform $\hat{\Omega}_{\text{shrink}}$ in terms of the MSE.[1]

[1] In the numerical analysis, we compare the minimum possible MSE of three shrinkage estimators. We set $n = 200$, $p = 100$ and we consider target matrices $F_1 = F_2 = F_3 = I$.

Appendix A

Proofs of Chapter 2

Proof of Theorem 2.2: We define our proposed R-GLASSO estimator as $\widehat{\Omega}_{\text{R-GLASSO}} = \widehat{\Gamma}^k$, where $\widehat{\Gamma}$ is the solution of the problem (2.5). We note that the solution $\widehat{\Gamma}$ can be considered as the GLASSO estimator for the matrix $\Omega^{1/k}$. Therefore, before proceeding with the convergence rate of the estimator $\widehat{\Omega}_{\text{R-GLASSO}}$, we provide the convergence rate of estimator $\widehat{\Gamma}$. First, consider the following conditions for the true model:

$$\text{B1} \ : \ \lambda_{\min}(\Omega^{1/k}) \geq \underline{\beta} > 0,$$

$$\text{B2} \ : \ \lambda_{\max}(\Omega^{1/k}) \leq \bar{\beta},$$

for some positive values $\bar{\beta}$ and $\underline{\beta}$. Note that the conditions $A1, A2$ imply the conditions $B1, B2$ and vice versa. We prove that under the assumptions of the Theorem 2.2

$$||\widehat{\Gamma} - \Omega^{1/k}||_2 = O_P\left(\sqrt{\frac{(p+s)\log p}{n}}\right). \tag{A.1}$$

The proof of (A.1) is inspired by Rothman et al. (2008). First, consider the following function

$$\begin{aligned} Q(\Theta) &= \text{trace}(\Theta S^{1/k}) - \log\det(\Theta) + \xi_k||\Theta||_1 - \text{trace}(\Omega^{1/k}S^{1/k}) - \log\det(\Omega^{1/k}) \\ &- \xi_k||\Omega^{1/k}||_1 = \text{trace}\left((\Theta - \Omega^{1/k})(S^{1/k} - \Sigma^{1/k})\right) - \left(\log\det(\Theta) - \log\det(\Omega^{1/k})\right) \\ &+ \text{trace}\left((\Theta - \Omega^{1/k})\Sigma^{1/k}\right) + \xi_k\left(||\Theta||_1 - ||\Omega^{1/k}||_1\right). \end{aligned} \tag{A.2}$$

It can be seen that the estimator $\widehat{\Gamma}$ minimizes the function $Q(\Theta)$, and therefore $\widehat{\Delta} = \widehat{\Gamma} - \Omega^{1/k}$ minimizes the function $G(\Delta) = Q(\Omega^{1/k} + \Delta)$. Consider the following set:

$$\Phi_n(M) = \{\Delta : \Delta = \Delta^T, ||\Delta||_2 = Mr_n\}, \tag{A.3}$$

where

$$r_n = \sqrt{\frac{(p+s)\log p}{n}} \to 0. \tag{A.4}$$

Note that $G(\Delta) = Q(\Omega^{1/k} + \Delta)$ is a convex function, and $G(\widehat{\Delta}) \leq G(0) = 0$. Then, if we show that

$$\inf\{G(\Delta) : \Delta \in \Phi_n(M)\} > 0, \tag{A.5}$$

the minimizer $\widehat{\Delta}$ must be inside the set defined by $\Phi_n(M)$, and therefore $||\widehat{\Delta}||_2 \leq Mr_n$.

We have

$$\begin{aligned} G(\Delta) &= \text{trace}\left(\Delta(S^{1/k} - \Sigma^{1/k})\right) - \left(\log\det(\Omega^{1/k} + \Delta) - \log\det(\Omega^{1/k})\right) \\ &+ \text{trace}\left(\Delta\Sigma^{1/k}\right) + \xi_k\left(||\Omega^{1/k} + \Delta||_1 - ||\Omega^{1/k}||_1\right). \end{aligned} \tag{A.6}$$

For the logarithm term in the equation (A.6), doing the Taylor expansion of the function $f(t) = \log\det(\Theta + t\Delta)$, we get

$$\begin{aligned}\log\det(\Omega^{1/k} + \Delta) - \log\det(\Omega^{1/k}) &= \text{trace}(\Sigma^{1/k}\Delta) \\ - \tilde{\Delta}^T \left[\int_0^1 (1-\nu)(\Omega^{1/k} + \mu\Delta)^{-1} \otimes (\Omega^{1/k} + \mu\Delta)^{-1} d\nu\right] \tilde{\Delta},\end{aligned} \tag{A.7}$$

where $\otimes$ is the Kronecker product and $\tilde{\Delta}$ is a vectorization of Δ. The equation (A.6) can be rewritten in the following form

$$\begin{aligned}G(\Delta) &= \text{trace}\left(\Delta(S^{1/k} - \Sigma^{1/k})\right) \\ &+ \tilde{\Delta}^T \left[\int_0^1 (1-\nu)(\Omega^{1/k} + \mu\Delta)^{-1} \otimes (\Omega^{1/k} + \mu\Delta)^{-1} d\nu\right] \tilde{\Delta} \\ &+ \xi_k \left(||\Omega^{1/k} + \Delta||_1 - ||\Omega^{1/k}||_1\right) = T_1 + T_2 + T_3.\end{aligned} \tag{A.8}$$

For an index set U and a matrix $A = [a_{ij}]$, denote $A_U = [a_{ij}\mathbb{I}_{(i,j)\in U}]$. Recall $Z = \{(i,j) : \Omega_{ij}^{(1/k)} \neq 0\}$ and $\bar{Z}$ is its complement. Note that $||\Omega^{1/k} + \Delta||_1 = ||\Omega_Z^{1/k} + \Delta_Z||_1 + ||\Delta_{\bar{Z}}||_1$ and $||\Omega^{1/k}||_1 = ||\Omega_Z^{1/k}||_1$. From the triangular inequality we have

$$T_3 = \xi_k \left(||\Omega^{1/k} + \Delta||_1 - ||\Omega^{1/k}||_1\right) \geq \xi_k \left(||\Delta_{\bar{Z}}||_1 - ||\Delta_Z||_1\right). \tag{A.9}$$

Next, consider the term T_1

$$\begin{aligned}|T_1| = \left|\text{trace}\left(\Delta(S^{1/k} - \Sigma^{1/k})\right)\right| &\leq \left|\sum_{i\neq j} (S^{1/k} - \Sigma^{1/k})_{ij}\Delta_{ij}\right| \\ + \left|\sum_i (S^{1/k} - \Sigma^{1/k})_{ii}\Delta_{ii}\right| &= T_{11} + T_{12}.\end{aligned} \tag{A.10}$$

To bound the terms T_{11} and T_{12}, we use the following result (Bickel and Levina 2008)

$$||S - \Sigma||_\infty = \max_{ij} |(S-\Sigma)_{ij}| = O_P\left(\sqrt{\frac{\log p}{n}}\right), \tag{A.11}$$

which holds under the assumptions of the Theorem 2.2 and $\frac{\log p}{n} = o(1)$. On the other hand, the assumption in Theorem 2.2 implies

$$\max_{ij} |(S^{1/k} - \Sigma^{1/k})_{ij}| = O_P\left(\sqrt{\frac{\log p}{n}}\right). \tag{A.12}$$

Therefore, using the sum inequality we can have the bound of the term T_{11}, with probability tending to 1,

$$T_{11} \leq C_1\sqrt{\frac{\log p}{n}}||\Delta^-||_1 \leq C_1\sqrt{\frac{\log p}{n}}||\Delta||_1. \tag{A.13}$$

From the Cauchy-Schwartz inequality we get

$$\begin{aligned} T_{12} &\leq \left[\sum_{i=1}^{p}(S^{1/k} - \Sigma^{1/k})_{ii}^2\right]^{1/2} ||\Delta^+||_2 \leq \sqrt{p}\max_{1\leq i\leq p}|(S^{1/k} - \Sigma^{1/k})_{ii}|||\Delta^+||_2 \\ &\leq C_2\sqrt{\frac{p\log p}{n}}||\Delta^+||_2 \leq C_2\sqrt{\frac{(p+s)\log p}{n}}||\Delta||_2, \end{aligned} \tag{A.14}$$

also with probability tending to 1.

Finally, it remains to check the bound of the second term T_2. For $\Delta \in \Phi_n(M)$

$$\begin{aligned} T_2 &\geq \lambda_{\min}\left(\int_0^1 (1-\nu)(\Omega^{1/k} + \mu\Delta)^{-1} \otimes (\Omega^{1/k} + \mu\Delta)^{-1} d\nu\right)||\Delta||_2^2 \\ &\geq \int_0^1 (1-\nu)\lambda_{\min}^2(\Omega^{1/k} + \mu\Delta)^{-1} d\nu ||\Delta||_2^2 \geq \frac{1}{2}\min_{0\leq\nu\leq 1}\lambda_{\min}^2(\Omega^{1/k} + \Delta)^{-1}||\Delta||_2^2 \\ &\geq \frac{1}{2}\min\{\lambda_{\min}^2(\Omega^{1/k} + \Delta)^{-1}, ||\Delta||_2 \leq Mr_n\}||\Delta||_2^2. \end{aligned} \tag{A.15}$$

On the other hand,

$$\lambda_{\min}^2(\Omega^{1/k} + \mu\Delta)^{-1} = \lambda_{\max}^{-2}(\Omega^{1/k} + \Delta) \geq (||\Omega^{1/k}|| + ||\Delta||)^{-2} \geq (\bar{\beta} + o(1))^{-2}, \tag{A.16}$$

since $||\Delta|| \leq ||\Delta||_2 = o(1)$, with probability tending to 1. Thus, we get

$$T_2 \geq \frac{1}{2}||\Delta||_2^2(\bar{\beta} + o(1))^{-2} = \frac{1}{2}||\Delta||_2^2\gamma, \tag{A.17}$$

where $\gamma = (\bar{\beta} + o(1))^{-2}$.

By our assumption in Theorem 2.2, $\xi_k \asymp \sqrt{\frac{\log p}{n}}$. Taking $\xi_k = \frac{C_1}{\epsilon}\sqrt{\frac{\log p}{n}}$ and using the obtained bounds (A.9), (A.10), (A.17), we get

$$\begin{aligned} G(\Delta) &\geq \frac{1}{2}||\Delta||_2^2\gamma - C_1\sqrt{\frac{\log p}{n}}||\Delta||_1 - C_2\sqrt{\frac{(p+s)\log p}{n}}||\Delta||_2 \\ &+ \xi_k\left(||\Delta_{\bar{Z}}||_1 - ||\Delta_Z||_1\right) = \frac{1}{2}||\Delta||_2^2\gamma - C_1\sqrt{\frac{\log p}{n}}(1-\frac{1}{\epsilon})||\Delta_{\bar{Z}}||_1 \\ &- C_1\sqrt{\frac{\log p}{n}}(1+\frac{1}{\epsilon})||\Delta_Z||_1 - C_2\sqrt{\frac{(p+s)\log p}{n}}||\Delta||_2. \end{aligned} \tag{A.18}$$

Since the second term is always positive, we can omit it for the lower bound. Note that

$$||\Delta_Z||_1 \leq \sqrt{s}||\Delta_Z||_2 \leq \sqrt{s}||\Delta||_2 \leq \sqrt{s+p}||\Delta||_2. \tag{A.19}$$

Hence, we have

$$\begin{aligned} &G(\Delta) \geq \frac{1}{2}||\Delta||_2^2\gamma - C_1\sqrt{\frac{(p+s)\log p}{n}}(1+\frac{1}{\epsilon})||\Delta||_2 \\ &- C_2\sqrt{\frac{(p+s)\log p}{n}}||\Delta||_2 \geq ||\Delta||_2^2\left[\frac{1}{4}\gamma - C_1\sqrt{\frac{(p+s)\log p}{n}}(1+\frac{1}{\epsilon})||\Delta||_2^{-1}\right] \\ &+ ||\Delta||_2^2\left[\frac{1}{4}\gamma - C_2\sqrt{\frac{(p+s)\log p}{n}}||\Delta||_2^{-1}\right] = ||\Delta||_2^2\left[\frac{1}{4}\gamma - \frac{C_1}{M}(1+\frac{1}{\epsilon})\right] \\ &+ ||\Delta||_2^2\left[\frac{1}{4}\gamma - \frac{C_2}{M}\right] > 0. \end{aligned} \tag{A.20}$$

for M sufficiently large. This establishes the convergence rate (A.1).

To obtain the convergence rate of our estimator, we prove the following lemma:

Lemma A.1. *For any symmetric, p.d. matrices A and B and for any finite $q \in \mathbb{N}$, if $||A||_{spec} = O_P(1)$, $||B||_{spec} = O_P(1)$, $||A||_{\min} = O_P(1)$ and $||B||_{\min} = O_P(1)$, then*

$$||A^q - B^q||_2 \overset{P}{\asymp} ||A - B||_2.. \tag{A.21}$$

Proof of Lemma 1: For any matrices A and B, we have that

$$A^q - B^q = \sum_{i=1}^{q} A^{q-i}(A - B)B^{i-1}. \tag{A.22}$$

Therefore, we can write the following:

$$\begin{aligned}
&||A^q - B^q||_2^2 = \text{trace}\left((A^q - B^q)(A^q - B^q)^T\right) \\
&= \text{trace}\left(\left(\sum_{i=1}^{q} A^{q-i}(A - B)B^{i-1}\right)\left(\sum_{i=1}^{q} B^{i-1}(A - B)A^{q-i}\right)\right) \\
&= \text{trace}\left(\sum_{i=1}^{q}\sum_{j=1}^{q} A^{q-i}(A - B)B^{i-1}B^{j-1}(A - B)A^{q-j}\right) \\
&= \text{trace}\left(\sum_{i=1}^{q}\sum_{j=1}^{q} A^{2q-i-j}(A - B)B^{i+j-2}(A - B)\right) \\
&= \sum_{i=1}^{q}\sum_{j=1}^{q} \text{trace}\left(A^{2q-i-j}(A - B)B^{i+j-2}(A - B)\right).
\end{aligned} \tag{A.23}$$

Next, for any symmetric matrices X, Y and Z consider trace$(XYZY)$. For any matrix A, we denote $A_{\cdot i}$ and $A_{i\cdot}$ as the i-th row and the i-th column of matrix A, respectively. We can write

$$\begin{aligned}
&\text{trace}(XYZY) = \sum_{i=1}^{p}(XY)_{i\cdot}ZY_{\cdot i} \leq \lambda_{\max}(Z)\sum_{i=1}^{p}(XY)_{i\cdot}Y_{\cdot i} = \lambda_{\max}(Z)\text{trace}(XYY) \\
&= \lambda_{\max}(Z)\sum_{i=1}^{p} Y_{i\cdot}X(Y_{\cdot i}) \leq \lambda_{\max}(Z)\lambda_{\max}(X)\sum_{i=1}^{p} Y_{i\cdot}(Y_{\cdot i}) \\
&= \lambda_{\max}(Z)\lambda_{\max}(X)\text{trace}(YY^T) = \lambda_{\max}(Z)\lambda_{\max}(X)||Y||_2^2.
\end{aligned} \tag{A.24}$$

Similarly, we can write

$$\begin{aligned}\text{trace}(XYZY) &= \sum_{i=1}^{p}(XY)_{i\cdot}ZY_{\cdot i} \geq \lambda_{\min}(Z)\sum_{i=1}^{p}(XY)_{i\cdot}Y_{\cdot i} = \lambda_{\min}(Z)\text{trace}(XYY)\\ &= \lambda_{\min}(Z)\sum_{i=1}^{p}Y_{i\cdot}X(Y_{\cdot i}) \geq \lambda_{\min}(Z)\lambda_{\min}(X)\sum_{i=1}^{p}Y_{i\cdot}(Y_{\cdot i})\\ &= \lambda_{\min}(Z)\lambda_{\min}(X)\text{trace}(YY^T) = \lambda_{\min}(Z)\lambda_{\min}(X)||Y||_2^2.\end{aligned} \tag{A.25}$$

We can summarize the expressions (A.24) and (A.25) as the following:

$$\lambda_{\min}(Z)\lambda_{\min}(X)||Y||_2^2 \leq \text{trace}(XYZY) \leq \lambda_{\max}(Z)\lambda_{\max}(X)||Y||_2^2. \tag{A.26}$$

We can apply the inequalities in (A.26) on the trace of the equality (A.23). Thus, we can write the following two inequalities:

$$\begin{aligned}&\sum_{i=1}^{q}\sum_{j=1}^{q}\text{trace}\left(A^{2q-i-j}(A-B)B^{i+j-2}(A-B)\right)\\ &\geq ||A-B||_2^2\sum_{i=1}^{q}\sum_{j=1}^{q}\lambda_{\min}(A^{2q-i-j})\lambda_{\min}(B^{i+j-2}),\end{aligned} \tag{A.27}$$

$$\begin{aligned}&\sum_{i=1}^{q}\sum_{j=1}^{q}\text{trace}\left(A^{2q-i-j}(A-B)B^{i+j-2}(A-B)\right)\\ &\leq ||A-B||_2^2\sum_{i=1}^{q}\sum_{j=1}^{q}\lambda_{\max}(A^{2q-i-j})\lambda_{\max}(B^{i+j-2}).\end{aligned} \tag{A.28}$$

From the inequalities (A.27), (A.28) and the equality (A.23) it follows that

$$||A^q - B^q||_2 \geq ||A-B||_2\left(\sum_{i=1}^{q}\sum_{j=1}^{q}\lambda_{\min}(A^{2q-i-j})\lambda_{\min}(B^{i+j-2})\right)^{\frac{1}{2}}, \tag{A.29}$$

$$||A^q - B^q||_2 \leq ||A-B||_2\left(\sum_{i=1}^{q}\sum_{j=1}^{q}\lambda_{\max}(A^{2q-i-j})\lambda_{\max}(B^{i+j-2})\right)^{\frac{1}{2}}. \tag{A.30}$$

Since q is finite, the assumptions $\lambda_{\max}(A) = ||A||_{\text{spec}} = O_P(1)$, $\lambda_{\max}(B) = ||B||_{\text{spec}} = O_P(1)$, $\lambda_{\min}(A) = ||A||_{\min} = O_P(1)$, $\lambda_{\min}(B) = ||B||_{\min} = O_P(1)$ imply that the following rates:

$$\left(\sum_{i=1}^{q}\sum_{j=1}^{q}\lambda_{\min}(A^{2q-i-j})\lambda_{\min}(B^{i+j-2})\right)^{\frac{1}{2}} = O_P(1), \tag{A.31}$$

$$\left(\sum_{i=1}^{q}\sum_{j=1}^{q}\lambda_{\max}(A^{2k-i-j})\lambda_{\max}(B^{i+j-2})\right)^{\frac{1}{2}} = O_P(1). \tag{A.32}$$

From the inequalities (A.29), (A.30), (A.31) and (A.32) it follows that

$$||A^q - B^q||_2 \overset{P}{\asymp} ||A - B||_2, \tag{A.33}$$

which concludes the proof of Lemma A.1.

From the assumptions B1 and B2 it follows that $||\Omega^{\frac{1}{k}}||_{\min} = O(1)$ and $||\Omega^{\frac{1}{k}}||_{\text{spec}} = O(1)$, respectively. Assuming that n grows faster than p, the rate (A.1) implies that $||\hat{\Gamma}||_{\min} = O_P(1)$ and $||\hat{\Gamma}||_{\text{spec}} = O_P(1)$. Now, if we consider $q = k$, $A = \hat{\Gamma}$, $B = \Omega^{\frac{1}{k}}$, we will have $A^q = \hat{\Gamma}^k = \hat{\Omega}_{\text{R-GLASSO}}$ and $B^q = \Omega$. Therefore, Lemma A.1 implies that

$$||\widehat{\Omega}_{\text{R-GLASSO}} - \Omega||_2 \overset{P}{\asymp} ||\widehat{\Gamma} - \Omega^{\frac{1}{k}}||_2, \tag{A.34}$$

which concludes the proof of the theorem for $k \in \mathbb{N}$.

We can prove the Theorem 2.2 under assumption that k is a rational number. We express k as a fraction $\dfrac{r}{m}$, where $r, m \in \mathbb{N}$. In this case we have that $\widehat{\Omega}_{\text{R-GLASSO}} = \widehat{\Gamma}^{\frac{r}{m}}$. If we consider $q = r$, $A = \hat{\Gamma}$, $B = \Omega^{\frac{m}{r}}$, we will have $A^q = \hat{\Gamma}^r$ and $B^q = \Omega^m$. Since r and m are finite, we can use the Lemma (A.1), which implies that

$$||\hat{\Gamma}^r - \Omega^m||_2 \overset{P}{\asymp} ||\hat{\Gamma} - \Omega^{\frac{m}{r}}||_2. \tag{A.35}$$

On the other hand, if we consider $q = m$, $A = \hat{\Gamma}^{\frac{r}{m}}$, $B = \Omega$, we will have $A^q = \hat{\Gamma}^r$ and $B^q = \Omega^m$. Therefore, as previously, Lemma (A.1) implies that

$$||\hat{\Gamma}^r - \Omega^m||_2 \overset{P}{\asymp} ||\hat{\Gamma}^{\frac{r}{m}} - \Omega||_2. \tag{A.36}$$

Summarizing (A.35) and (A.36), we will have the following:

$$||\hat{\Gamma} - \Omega^{\frac{m}{r}}||_2 \overset{P}{\asymp} ||\hat{\Gamma}^{\frac{r}{m}} - \Omega||_2, \tag{A.37}$$

Finally, (A.1) and (A.37) establish the rate (2.6) for rational $k = \dfrac{r}{m}$.

Appendix B

Numerical Results of Chapter 2

TABLE B.1: Average KLL (with standard deviations) over 100 replications.

	Model 1			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	12.225 (0.832)	5.049 (0.462)	7.8280 (0.231)	9.0541 (1.328)
200	34.760 (1.469)	19.770 (1.063)	18.970 (0.397)	21.015 (0.481)
300	62.975 (1.927)	41.488 (0.667)	41.488 (0.667)	40.036 (2.648)
	Model 2			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	14.382 (0.902)	8.336 (0.548)	10.244 (0.684)	12.556 (1.743)
200	40.423 (1.634)	28.555 (0.718)	28.511 (0.542)	30.094 (0.507)
300	69.625 (1.704)	52.375 (0.961)	52.375 (0.961)	56.741 (4.129)
	Model 3			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	15.572 (0.959)	10.883 (0.965)	12.985 (0.959)	18.316 1.6251
200	44.006 (1.672)	33.932 (0.803)	33.932 (0.803)	38.444 1.1220
300	73.999 (2.026)	57.472 (0.761)	57.472 (0.761)	62.256 0.6433
	Model 4			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	16.076 (0.798)	12.073 (1.227)	13.102 (0.963)	18.019 (2.497)
200	45.844 (1.786)	34.595 (0.756)	34.554 (0.581)	37.908 (2.629)
300	78.341 (2.003)	65.810 (1.822)	65.810 (1.822)	76.770 (2.498)
	Model 5			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	11.134 (0.936)	10.413 (0.447)	10.399 (0.433)	13.145 1.920
200	28.082 (0.989)	16.684 (2.571)	16.684 (2.571)	21.429 1.510
300	49.287 (0.486)	34.198 (1.421)	34.198 (1.421)	35.856 4.437
	Model 6			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	17.553 (0.483)	12.735 (0.247)	12.735 (0.247)	13.457 (0.274)
200	38.697 (0.450)	26.778 (0.859)	26.778 (0.859)	28.413 (0.420)
300	58.169 (0.386)	46.054 (1.179)	46.054 (1.179)	41.965 (0.536)
	Model 7			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	17.194 (0.528)	12.363 (0.365)	12.434 (0.334)	12.983 (0.308)
200	38.163 (0.932)	26.409 (0.743)	26.409 (0.743)	27.850 (0.378)
300	57.904 (0.399)	45.602 (1.051)	45.602 (1.051)	41.531 (0.555)

TABLE B.2: MSE (with standard deviations) over 100 replications.

		Model 1		
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	4.609 (0.256)	0.722 (0.094)	2.433 (0.079)	1.997 (0.304)
200	11.383 (0.324)	3.665 (0.514)	4.064 (0.110)	3.991 (0.161)
300	19.325 (0.353)	7.394 (0.099)	7.394 (0.099)	7.105 (0.732)
		Model 2		
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	5.275 (0.239)	1.686 (0.106)	3.281 (0.207)	3.216 (0.400)
200	12.858 (0.301)	6.979 (0.182)	6.995 (0.091)	6.575 (0.137)
300	20.531 (0.297)	9.658 (0.206)	9.658 (0.206)	11.249 (1.174)
		Model 3		
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	5.565 (0.229)	1.756 (0.145)	3.903 (0.233)	3.709 (0.338)
200	13.207 (0.301)	7.523 (0.191)	7.523 (0.191)	7.319 (0.267)
300	21.214 (0.331)	10.750 (0.100)	10.750 (0.100)	13.185 (0.159)
		Model 4		
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	6.324 (0.212)	2.467 (0.596)	4.387 (0.252)	4.188 (0.510)
200	13.989 (0.331)	7.787 (0.217)	7.807 (0.105)	7.766 (0.640)
300	22.708 (0.303)	12.286 (0.593)	12.286 (0.593)	15.210 (0.684)
		Model 5		
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	31.298 (2.435)	21.720 (1.365)	21.612 (0.773)	29.116 (4.448)
200	75.484 (1.984)	22.586 (5.487)	22.586 (5.487)	47.712 (3.624)
300	127.591 (0.710)	23.393 (1.448)	23.393 (1.448)	78.381 (11.235)
		Model 6		
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	117.466 (1.621)	97.470 (0.890)	97.470 (0.890)	100.700 (1.218)
200	247.054 (1.251)	184.370 (4.257)	184.370 (4.257)	209.921 (1.616)
300	371.535 (0.966)	217.055 (2.138)	217.055 (2.138)	311.636 (2.055)
		Model 7		
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	111.704 (1.797)	91.816 (1.870)	92.229 (1.240)	94.703 (1.257)
200	240.704 (2.739)	179.012 (3.692)	179.012 (3.692)	203.530 (1.398)
300	365.989 (0.970)	212.590 (1.954)	212.590 (1.954)	305.770 (2.050)

TABLE B.3: Average specificity (with standard deviations) over 100 replications.

	Model 1			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.941 (0.009)	0.998 (0.001)	0.987 (0.004)	0.977 (0.008)
200	0.973 (0.003)	0.998 (0.0009)	0.997 (0.0008)	0.994 (0.0008)
300	0.983 (0.002)	0.999 (0.0001)	0.999 (0.0001)	0.993 (0.001)
	Model 2			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.947 (0.009)	0.994 (0.002)	0.986 (0.005)	0.983 (0.009)
200	0.972 (0.004)	0.996 (0.0008)	0.996 (0.0008)	0.992 (0.001)
300	0.983 (0.001)	0.999 (0.0003)	0.999 (0.0003)	0.997 (0.002)
	Model 3			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.938 (0.010)	0.992 (0.003)	0.986 (0.005)	0.978 (0.011)
200	0.972 (0.004)	0.997 (0.001)	0.997 (0.001)	0.990 (0.001)
300	0.984 (0.002)	0.999 (0.0001)	0.999 (0.0001)	0.999 (0.0002)
	Model 4			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.938 (0.007)	0.993 (0.003)	0.984 (0.005)	0.972 (0.014)
200	0.974 (0.003)	0.997 (0.0008)	0.997 (0.0008)	0.990 (0.003)
300	0.984 (0.001)	0.999 (0.0006)	0.999 (0.0006)	0.998 (0.001)
	Model 5			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.932 (0.010)	0.947 (0.004)	0.947 (0.004)	0.965 (0.016)
200	0.967 (0.002)	0.972 (0.005)	0.972 (0.005)	0.985 (0.002)
300	0.981 (0.0009)	0.989 (0.001)	0.989 (0.001)	0.994 (0.004)
	Model 6			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	NA (NA)	NA (NA)	NA (NA)	NA (NA)
200	NA (NA)	NA (NA)	NA (NA)	NA (NA)
300	NA (NA)	NA (NA)	NA (NA)	NA (NA)
	Model 7			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.980 (0.005)	0.992 (0.003)	0.992 (0.003)	0.963 (0.006)
200	0.992 (0.002)	0.998 (0.001)	0.998 (0.001)	0.993 (0.001)
300	0.992 (0.0006)	0.997 (0.0006)	0.997 (0.0006)	0.994 (0.0005)

TABLE B.4: Average sensitivity (with standard deviations) over 100 replications.

	Model 1			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.501 (0.020)	0.485 (0.032)	0.498 (0.021)	0.430 (0.031)
200	0.225 (0.010)	0.196 (0.009)	0.202 (0.007)	0.211 (0.006)
300	0.163 (0.006)	0.138 (0.005)	0.138 (0.005)	0.164 (0.010)
	Model 2			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.290 (0.017)	0.283 (0.023)	0.274 (0.019)	0.243 (0.036)
200	0.148 (0.008)	0.145 (0.006)	0.145 (0.005)	0.146 (0.005)
300	0.100 (0.004)	0.081 (0.004)	0.081 (0.004)	0.078 (0.011)
	Model 3			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.203 (0.015)	0.186 (0.020)	0.189 (0.016)	0.163 (0.021)
200	0.096 (0.006)	0.080 (0.005)	0.080 (0.005)	0.084 (0.005)
300	0.062 (0.004)	0.042 (0.001)	0.042 (0.001)	0.036 (0.001)
	Model 4			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.290 (0.013)	0.353 (0.056)	0.345 (0.029)	0.246 (0.041)
200	0.147 (0.010)	0.139 (0.007)	0.140 (0.007)	0.132 (0.016)
300	0.100 (0.004)	0.092 (0.010)	0.092 (0.010)	0.066 (0.006)
	Model 5			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	1 (0)	1 (0)	1 (0)	1 (0)
200	1 (0)	1 (0)	1 (0)	1 (0)
300	1 (0)	1 (0)	1 (0)	1 (0)
	Model 6			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.049 (0.004)	0.060 (0.003)	0.060 (0.003)	0.084 (0.006)
200	0.022 (0.001)	0.027 (0.001)	0.027 (0.001)	0.029 (0.001
300	0.017 (0.0005)	0.019 (0.0005)	0.019 (0.0005)	0.019 (0.0004)
	Model 7			
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.136 (0.006)	0.207 (0.014)	0.204 (0.006)	0.219 (0.009)
200	0.066 (0.002)	0.100 (0.001)	0.100 (0.001)	0.096 (0.002)
300	0.046 (0.0009)	0.068 (0.001)	0.068 (0.001)	0.061 (0.001)

TABLE B.5: Average MCC (with standard deviations) over 100 replications.

		Model 1		
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.350 (0.020)	0.660 (0.021)	0.563 (0.033)	0.443 (0.042)
200	0.230 (0.010)	0.408 (0.013)	0.394 (0.013)	0.364 (0.013)
300	0.205 (0.007)	0.349 (0.007)	0.349 (0.007)	0.291 (0.014)
		Model 2		
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.269 (0.016)	0.463 (0.015)	0.400 (0.027)	0.356 (0.027)
200	0.186 (0.008)	0.326 (0.009)	0.326 (0.010)	0.283 (0.010)
300	0.159 (0.005)	0.258 (0.005)	0.258 (0.005)	0.234 (0.012)
		Model 3		
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.199 (0.013)	0.352 (0.015)	0.325 (0.018)	0.262 (0.021)
200	0.138 (0.006)	0.230 (0.007)	0.230 (0.007)	0.196 (0.007)
300	0.121 (0.005)	0.180 (0.004)	0.180 (0.004)	0.161 (0.003)
		Model 4		
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.263 (0.013)	0.527 (0.039)	0.473 (0.027)	0.326 (0.024)
200	0.204 (0.006)	0.330 (0.011)	0.330 (0.011)	0.265 (0.008)
300	0.175 (0.004)	0.275 (0.009)	0.275 (0.009)	0.223 (0.006)
		Model 5		
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.543 (0.030)	0.592 (0.017)	0.592 (0.017)	0.686 (0.075)
200	0.555 (0.019)	0.593 (0.039)	0.593 (0.039)	0.707 (0.045)
300	0.593 (0.010)	0.696 (0.016)	0.696 (0.016)	0.826 (0.105)
		Model 6		
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	NA (NA)	NA (NA)	NA (NA)	NA (NA)
200	NA (NA)	NA (NA)	NA (NA)	NA (NA)
300	NA (NA)	NA (NA)	NA (NA)	NA (NA)
		Model 7		
p	GLASSO	R-GLASSO $k = k_{BIC}$	R-GLASSO $k = 2$	CLIME
100	0.237 (0.014)	0.372 (0.022)	0.368 (0.014)	0.290 (0.014)
200	0.173 (0.008)	0.265 (0.006)	0.265 (0.006)	0.229 (0.006)
300	0.131 (0.004)	0.207 (0.005)	0.207 (0.005)	0.178 (0.003)

Appendix C

Numerical Results of Chapter 3 (Part I)

TABLE C.1: Average KLL (with standard deviations) over 100 replications.

Methods	Model 1	Model 2	Model 3	Model 4
GLASSO	21.680 (1.388)	38.497 (0.342)	16.792 (0.436)	18.201 (0.685)
DT	20.598 (0.384)	37.335 (0.331)	19.317 (0.494)	19.381 (0.454)
ADT	19.171 (0.483)	34.318 (0.434)	17.511 (0.595)	6.727 (0.285)
WADT	19.536 (0.970)	35.512 (0.496)	12.860 (0.513)	4.652 (0.214)
Methods	Model 5	Model 6	Model 7	Model 8
GLASSO	23.336 (0.389)	53.028 (0.270)	41.149 (0.333)	47.480 (0.333)
DT	18.518 (0.517)	30.733 (0.433)	29.248 (0.423)	33.168 (0.407)
ADT	10.642 (0.416)	21.219 (0.344)	28.342 (0.390)	31.611 (0.916)
WADT	4.9425 (0.247)	48.439 (0.337)	21.880 (0.475)	26.031 (0.483)

TABLE C.2: Average Frobenius norm losses (with standard deviations) over 100 replications.

Methods	Model 1	Model 2	Model 3	Model 4
GLASSO	7.402 (0.314)	12.042 (0.030)	5.398 (0.060)	6.931 (0.314)
DT	6.953 (0.066)	11.467 (0.041)	4.898 (0.063)	4.890 (0.082)
ADT	6.685 (0.094)	10.681 (0.068)	4.803 (0.081)	2.741 (0.076)
WADT	6.563 (0.396)	10.949 (0.123)	4.068 (0.080)	2.366 (0.066)
Methods	Model 5	Model 6	Model 7	Model 8
GLASSO	5.512 (0.042)	20.782 (0.032)	3.307 (0.013)	3.774 (0.011)
DT	2.668 (0.070)	16.057 (0.080)	2.296 (0.027)	2.765 (0.025)
ADT	1.938 (0.063)	13.478 (0.119)	2.205 (0.030)	2.627 (0.068)
WADT	1.562 (0.058)	18.106 (0.066)	1.960 (0.024)	2.307 (0.026)

TABLE C.3: Average operator norm losses (with standard deviations) over 100 replications.

Methods	Model 1	Model 2	Model 3	Model 4
GLASSO	0.774 (0.028)	1.630 (0.007)	0.589 (0.013)	0.663 (0.026)
DT	0.741 (0.018)	1.556 (0.012)	0.535 (0.016)	0.539 (0.024)
ADT	0.750 (0.023)	1.454 (0.020)	0.544 (0.022)	0.388 (0.029)
WADT	0.704 (0.054)	1.474 (0.024)	0.451 (0.021)	0.348 (0.038)
Methods	Model 5	Model 6	Model 7	Model 8
GLASSO	0.797 (0.022)	2.980 (0.005)	0.613 (0.009)	0.736 (0.007)
DT	0.412 (0.030)	2.474 (0.017)	0.544 (0.009)	0.644 (0.009)
ADT	0.317 (0.031)	2.198 (0.032)	0.551 (0.009)	0.644 (0.012)
WADT	0.292 (0.032)	2.691 (0.012)	0.509 (0.010)	0.593 (0.012)

TABLE C.4: Average matrix ℓ_1 norm losses (with standard deviations) over 100 replications.

Methods	Model 1	Model 2	Model 3	Model 4
GLASSO	1.329 (0.138)	2.032 (0.042)	0.992 (0.050)	0.970 (0.038)
DT	1.109 (0.047)	1.939 (0.034)	0.924 (0.043)	0.680 (0.034)
ADT	1.051 (0.045)	1.953 (0.052)	0.840 (0.045)	0.505 (0.038)
WADT	1.138 (0.121)	1.955 (0.052)	0.846 (0.053)	0.477 (0.048)
Methods	Model 5	Model 6	Model 7	Model 8
GLASSO	0.923 (0.030)	3.390 (0.039)	1.242 (0.014)	1.659 (0.015)
DT	0.590 (0.045)	2.900 (0.042)	1.077 (0.033)	1.571 (0.028)
ADT	0.426 (0.047)	2.612 (0.054)	1.077 (0.034)	1.575 (0.039)
WADT	0.385 (0.046)	2.916 (0.026)	0.997 (0.044)	1.522 (0.038)

TABLE C.5: Average specificity (with standard deviations) over 100 replications.

Methods	Model 1	Model 2	Model 3	Model 4
GLASSO	97.01 (1.21)	98.14 (0.05)	98.18 (0.05)	96.30 (0.72)
DT	98.26 (0.03)	98.18 (0.04)	98.03 (0.04)	99.33 (0.02)
ADT	99.49 (0.02)	98.68 (0.03)	99.73 (0.01)	99.63 (0.01)
WADT	98.96 (0.68)	98.87 (0.15)	99.05 (0.08)	99.66 (0.02)
Methods	Model 5	Model 6	Model 7	Model 8
GLASSO	95.31 (0.07)	NA (NA)	94.78 (0.07)	94.73 (0.07)
DT	97.40 (0.05)	NA (NA)	97.86 (0.05)	98.02 (0.04)
ADT	99.17 (0.02)	NA (NA)	98.45 (0.03)	98.38 (0.38)
WADT	99.70 (0.02)	NA (NA)	99.63 (0.02)	99.46 (0.02)

TABLE C.6: Average sensitivity (with standard deviations) over 100 replications.

Methods	Model 1	Model 2	Model 3	Model 4
GLASSO	88.94 (5.04)	61.23 (0.77)	99.62 (0.15)	100 (0)
DT	91.17 (0.82)	67.53 (0.81)	99.54 (0.20)	100 (0)
ADT	84.13 (1.26)	68.18 (0.91)	97.20 (0.53)	100 (0)
WADT	84.55 (5.20)	62.01 (1.62)	98.97 (0.33)	100 (0)
Methods	Model 5	Model 6	Model 7	Model 8
GLASSO	100 (0)	4.86 (0.08)	20.25 (0.30)	12.92 (0.19)
DT	100 (0)	3.88 (0.04)	19.37 (0.26)	11.36 (0.15)
ADT	100 (0)	1.77 (0.02)	17.87 (0.24)	10.72 (0.67)
WADT	100 (0)	0.68 (0.09)	16.65 (0.23)	9.80 (0.13)

TABLE C.7: Average MCC (with standard deviations) over 100 replications.

Methods	Model 1	Model 2	Model 3	Model 4
GLASSO	0.467 (0.082)	0.467 (0.005)	0.589 (0.006)	0.372 (0.036)
DT	0.555 (0.005)	0.511 (0.005)	0.573 (0.004)	0.685 (0.006)
ADT	0.720 (0.009)	0.565 (0.006)	0.872 (0.005)	0.785 (0.007)
WADT	0.639 (0.080)	0.549 (0.010)	0.708 (0.016)	0.800 (0.009)
Methods	Model 5	Model 6	Model 7	Model 8
GLASSO	0.329 (0.002)	NA (NA)	0.136 (0.002)	0.094 (0.002)
DT	0.427 (0.003)	NA (NA)	0.216 (0.003)	0.164 (0.002)
ADT	0.646 (0.006)	NA (NA)	0.230 (0.003)	0.172 (0.007)
WADT	0.816 (0.010)	NA (NA)	0.325 (0.004)	0.229 (0.003)

TABLE C.8: Average accuracy (with standard deviations) over 100 replications.

Methods	Model 1	Model 2	Model 3	Model 4
GLASSO	96.93 (1.15)	97.48 (0.04)	98.19 (0.05)	96.33 (0.71)
DT	98.19 (0.03)	97.63 (0.04)	98.05 (0.03)	99.33 (0.02)
ADT	99.33 (0.02)	98.13 (0.03)	99.71 (0.01)	99.63 (0.01)
WADT	98.82 (0.62)	98.21 (0.12)	99.05 (0.08)	99.67 (0.02)
Methods	Model 5	Model 6	Model 7	Model 8
GLASSO	95.34 (0.07)	NA (NA)	91.21 (0.06)	86.96 (0.06)
DT	97.41 (0.05)	NA (NA)	94.10 (0.04)	89.79 (0.04)
ADT	99.18 (0.02)	NA (NA)	94.59 (0.03)	90.05 (0.28)
WADT	99.70 (0.02)	NA (NA)	95.65 (0.02)	90.94 (0.02)

Appendix D

Numerical Results of Chapter 3 (Part II)

TABLE D.1: Average KLL (with standard deviations) over 100 replications.

	Model 1		
p	DT	ADT	WADT
100	7.416 (0.948)	6.540 (0.588)	3.011 (0.359)
200	19.752 (0.710)	19.144 (1.359)	9.269 (0.749)
300	36.848 (1.097)	38.560 (3.307)	19.554 (1.700)
	Model 2		
p	DT	ADT	WADT
100	10.165 (1.013)	9.703 (0.846)	4.817 (0.535)
200	27.398 (0.771)	24.880 (1.783)	13.509 (0.787)
300	45.939 (0.670)	43.043 (4.538)	25.399 (1.294)
	Model 3		
p	DT	ADT	WADT
100	10.600 (0.305)	9.453 (0.855)	5.355 (0.569)
200	30.606 (2.968)	28.022 (1.582)	17.669 (0.924)
300	52.170 (0.563)	48.650 (3.311)	31.463 (2.044)
	Model 4		
p	DT	ADT	WADT
100	10.954 (0.722)	10.213 (0.925)	6.286 (0.496)
200	33.509 (2.580)	30.912 (1.855)	19.306 (1.102)
300	57.717 (2.737)	52.566 (3.242)	33.564 (1.234)
	Model 5		
p	DT	ADT	WADT
100	8.103 (0.410)	3.306 (0.441)	2.406 (0.279)
200	16.542 (1.146)	7.794 (0.406)	4.724 (0.374)
300	33.946 (1.267)	11.801 (0.772)	8.027 (0.491)
	Model 6		
p	DT	ADT	WADT
100	13.410 (0.287)	9.378 (0.819)	10.028 (0.236)
200	27.019 (0.334)	20.076 (1.208)	20.855 (0.302)
300	40.580 (0.508)	32.452 (2.506)	31.840 (0.541)
	Model 7		
p	DT	ADT	WADT
100	12.937 (0.309)	8.931 (0.793)	9.618 (0.305)
200	26.473 (0.431)	19.565 (1.337)	20.388 (0.273)
300	40.044 (0.461)	31.189 (2.553)	31.248 (0.327)

TABLE D.2: MSE (with standard deviations) over 100 replications.

	Model 1		
p	DT	ADT	WADT
100	1.565 (0.209)	0.975 (0.108)	0.371 (0.058)
200	4.139 (0.178)	3.050 (0.250)	1.383 (0.141)
300	8.980 (0.233)	7.866 (0.617)	4.098 (0.360)
	Model 2		
p	DT	ADT	WADT
100	2.757 (0.209)	2.299 (0.162)	1.210 (0.147)
200	6.122 (0.167)	4.605 (0.349)	2.469 (0.158)
300	11.695 (0.177)	9.874 (0.914)	5.681 (0.290)
	Model 3		
p	DT	ADT	WADT
100	2.942 (0.102)	2.242 (0.187)	0.986 (0.142)
200	7.643 (0.615)	6.168 (0.329)	3.707 (0.176)
300	13.068 (0.151)	11.220 (0.634)	7.403 (0.493)
	Model 4		
p	DT	ADT	WADT
100	3.385 (0.172)	2.664 (0.239)	1.544 (0.127)
200	7.939 (0.517)	6.377 (0.350)	3.861 (0.237)
300	14.970 (0.543)	12.653 (0.625)	8.013 (0.293)
	Model 5		
p	DT	ADT	WADT
100	19.542 (0.959)	5.700 (0.965)	3.408 (0.474)
200	39.495 (2.686)	13.826 (0.885)	6.228 (0.632)
300	80.105 (2.803)	20.819 (1.653)	10.457 (0.755)
	Model 6		
p	DT	ADT	WADT
100	102.051 (1.222)	74.978 (5.855)	72.381 (2.744)
200	206.092 (1.511)	159.585 (7.894)	150.432 (3.138)
300	309.819 (2.095)	254.779 (16.547)	229.845 (5.492)
	Model 7		
p	DT	ADT	WADT
100	95.905 (1.259)	69.252 (5.515)	66.876 (3.198)
200	199.773 (1.882)	153.710 (8.626)	145.100 (2.934)
300	303.596 (1.978)	244.170 (16.740)	223.509 (3.236)

TABLE D.3: Average specificity (with standard deviations) over 100 replications.

	Model 1		
p	DT	ADT	WADT
100	0.990 (0.003)	0.989 (0.002)	0.999 (0.0007)
200	0.994 (0.0006)	0.994 (0.001)	0.999 (0.0004)
300	0.996 (0.0004)	0.996 (0.001)	0.999 (0.0005)
	Model 2		
p	DT	ADT	WADT
100	0.989 (0.003)	0.989 (0.002)	0.998 (0.001)
200	0.996 (0.0006)	0.994 (0.001)	0.998 (0.0003)
300	0.997 (0.0002)	0.993 (0.013)	0.999 (0.0002)
	Model 3		
p	DT	ADT	WADT
100	0.991 (0.001)	0.989 (0.002)	0.998 (0.001)
200	0.995 (0.002)	0.994 (0.001)	0.998 (0.0003)
300	0.997 (0.0002)	0.995 (0.001)	0.999 (0.0004)
	Model 4		
p	DT	ADT	WADT
100	0.990 (0.002)	0.990 (0.003)	0.998 (0.0009)
200	0.996 (0.001)	0.994 (0.001)	0.999 (0.0003)
300	0.997 (0.0008)	0.996 (0.001)	0.999 (0.0002)
	Model 5		
p	DT	ADT	WADT
100	0.990 (0.001)	0.995 (0.001)	0.997 (0.001)
200	0.992 (0.001)	0.998 (0.0003)	0.998 (0.0004)
300	0.998 (0.0005)	0.998 (0.0003)	0.999 (0.0001)
	Model 6		
p	DT	ADT	WADT
100	NA (NA)	NA (NA)	NA (NA)
200	NA (NA)	NA (NA)	NA (NA)
300	NA (NA)	NA (NA)	NA (NA)
	Model 7		
p	DT	ADT	WADT
100	0.997 (0.0008)	0.997 (0.001)	0.999 (0.0006)
200	0.997 (0.0004)	0.998 (0.0005)	0.999 (0.0002)
300	0.997 (0.0003)	0.999 (0.0005)	0.999 (0.0001)

TABLE D.4: Average sensitivity (with standard deviations) over 100 replications.

	Model 1		
p	DT	ADT	WADT
100	0.390 (0.016)	0.378 (0.011)	0.399 (0.018)
200	0.206 (0.005)	0.192 (0.007)	0.217 (0.009)
300	0.152 (0.003)	0.136 (0.008)	0.158 (0.009)
	Model 2		
p	DT	ADT	WADT
100	0.237 (0.016)	0.226 (0.015)	0.263 (0.014)
200	0.135 (0.004)	0.133 (0.008)	0.146 (0.005)
300	0.083 (0.001)	0.085 (0.018)	0.090 (0.004)
	Model 3		
p	DT	ADT	WADT
100	0.150 (0.005)	0.148 (0.009)	0.146 (0.0090)
200	0.076 (0.008)	0.075 (0.004)	0.081 (0.0032)
300	0.044 (0.0008)	0.045 (0.004)	0.050 (0.0040)
	Model 4		
p	DT	ADT	WADT
100	0.234 (0.011)	0.2258 (0.014)	0.227 (0.013)
200	0.113 (0.011)	0.1118 (0.008)	0.128 (0.007)
300	0.078 (0.005)	0.0810 (0.006)	0.092 (0.003)
	Model 5		
p	DT	ADT	WADT
100	1 (0)	1 (0)	1 (0)
200	1 (0)	1 (0)	1 (0)
300	1 (0)	1 (0)	1 (0)
	Model 6		
p	DT	ADT	WADT
100	0.036 (0.001)	0.040 (0.004)	0.031 (0.0012)
200	0.019 (0.0005)	0.019 (0.001)	0.015 (0.0003)
300	0.014 (0.0002)	0.012 (0.001)	0.010 (0.0003)
	Model 7		
p	DT	ADT	WADT
100	0.135 (0.002)	0.151 (0.011)	0.124 (0.004)
200	0.070 (0.001)	0.073 (0.004)	0.060 (0.0007)
300	0.047 (0.0008)	0.047 (0.003)	0.040 (0.0003)

TABLE D.5: Average MCC (with standard deviations) over 100 replications.

	Model 1		
p	DT	ADT	WADT
100	0.498 (0.024)	0.478 (0.019)	0.608 (0.010)
200	0.356 (0.008)	0.337 (0.009)	0.439 (0.006)
300	0.303 (0.006)	0.287 (0.010)	0.371 (0.005)
	Model 2		
p	DT	ADT	WADT
100	0.381 (0.013)	0.366 (0.013)	0.475 (0.010)
200	0.304 (0.007)	0.282 (0.007)	0.349 (0.006)
300	0.230 (0.004)	0.208 (0.020)	0.275 (0.004)
	Model 3		
p	DT	ADT	WADT
100	0.299 (0.010)	0.288 (0.011)	0.337 (0.008)
200	0.215 (0.005)	0.204 (0.005)	0.247 (0.003)
300	0.160 (0.003)	0.147 (0.004)	0.191 (0.004)
	Model 4		
p	DT	ADT	WADT
100	0.392 (0.011)	0.381 (0.012)	0.442 (0.010)
200	0.278 (0.006)	0.260 (0.007)	0.327 (0.007)
300	0.236 (0.004)	0.223 (0.004)	0.273 (0.004)
	Model 5		
p	DT	ADT	WADT
100	0.872 (0.016)	0.936 (0.024)	0.960 (0.016)
200	0.817 (0.027)	0.943 (0.009)	0.940 (0.012)
300	0.940 (0.018)	0.921 (0.013)	0.955 (0.007)
	Model 6		
p	DT	ADT	WADT
100	NA (NA)	NA (NA)	NA (NA)
200	NA (NA)	NA (NA)	NA (NA)
300	NA (NA)	NA (NA)	NA (NA)
	Model 7		
p	DT	ADT	WADT
100	0.307 (0.006)	0.331 (0.008)	0.307 (0.004)
200	0.209 (0.003)	0.227 (0.005)	0.211 (0.001)
300	0.165 (0.003)	0.180 (0.003)	0.170 (0.001)

Appendix E

Proofs of Background Statements

The following material provides detailed derivation of some statements and formulas employed in this thesis.

Remark E.1. The probability density function of a mean-centered multivariate Normally distributed variable $x \in \mathbb{R}^p$ is given by

$$p(x|\Sigma) = (2\pi)^{-p/2} \det(\Sigma)^{-1/2} \exp\left(-\frac{1}{2} x^T \Sigma^{-1} x\right). \tag{E.1}$$

We can express the function (E.1) in terms of the precision matrix Ω as follows:

$$p(x|\Omega) = (2\pi)^{-p/2} \det(\Omega)^{1/2} \exp\left(-\frac{1}{2} x^T \Omega x\right). \tag{E.2}$$

We write the log-likelihood function of a mean-centered sample dataset X as

$$\begin{aligned} \ell(\mathbf{X}, \Omega) &= \log \prod_{i=1}^{n} p(X_i|\Omega) = -\frac{p}{2}\log(2\pi) + \frac{n}{2}\log\det(\Omega) - \frac{1}{2}\sum_{i=1}^{n}\left(X_i^T \Omega^{-1} X_i\right) \\ &= \frac{n}{2}\log\det(\Omega) - \frac{1}{2}\sum_{i=1}^{n}\operatorname{trace}\left(X_i^T \Omega X_i\right) - \frac{p}{2}\log(2\pi). \end{aligned} \tag{E.3}$$

Since the trace is a linear map in the space of square matrices, we can rewrite the expression (E.3) as

$$\begin{aligned}\ell(\mathbf{X},\Omega) &= \frac{n}{2}\log\det(\Omega) - \frac{1}{2}\text{trace}\left(\Omega\sum_{i=1}^{n}X_i^TX_i\right) - \frac{p}{2}\log(2\pi)\\ &= \frac{n}{2}\log\det(\Omega) - \frac{n}{2}\text{trace}\,(\Omega S) - C.\end{aligned} \tag{E.4}$$

Note that the final expression of the equation (E.4) is proportionally equivalent to the function (1.1) excluding the constant.

Remark E.2. In order to obtain the MLE of the matrix Ω, we calculate the partial derivative of the log-likelihood function $\ell(X,\Omega)$ with respect to Ω.

$$\frac{\partial\ell(\mathbf{X},\Omega)}{\partial\Omega} = \Omega^{-1} - S. \tag{E.5}$$

By setting the partial derivative to zero, we get $\hat{\Omega}_{\text{MLE}} = S^{-1}$.

Remark E.3. We can write the optimization problem (2.1) as follows:

$$\begin{aligned}&\max_{\Omega}\min_{||\Theta||_\infty\leq\nu}\ \log\det\Omega - \text{trace}(\Omega(S+\Theta))\\ &= \min_{||\Theta||_\infty\leq\nu}\max_{\Omega}\ \log\det\Omega - \text{trace}(\Omega(S+\Theta)).\end{aligned} \tag{E.6}$$

Consider the derivative of the objective function of the inner maximization problem with respect to Ω. By setting the derivative to zero and solving the resulting equation for Ω, we obtain $\Omega = (S+\Theta)^{-1}$. Similarly, we can have $\Theta = \Omega^{-1} - S$. By employing the obtained expression of Ω in the problem (E.6), we obtain

$$\min_{||\Theta||_\infty\leq\nu}\max_{\Omega}\ -\log\det(S+\Theta) - p. \tag{E.7}$$

Finally, by writing the problem (E.6) in more accurate way, using the expression of Θ, we obtain the dual problem (2.2)

$$\min_{\Omega} \ \log\det\Omega \tag{E.8}$$

$$\text{subject to} \ \ ||\Omega^{-1} - S||_\infty \leq \nu. \tag{E.9}$$

Remark E.4. In order to show that $f_{DT}(\Omega, \Sigma)$ is convex function of Ω, we have to prove that for any $\eta \in [0,1]$ and $\Omega_1,\ \Omega_2 \succ 0$ the following inequality holds:

$$\eta f_{DT}(\Omega_1, \Sigma) + (1-\eta) f_{DT}(\Omega_2, \Sigma) \geq f_{DT}(\eta\Omega_1 + (1-\eta)\Omega_2, \Sigma), \tag{E.10}$$

or equivalently

$$\eta f_{DT}(\Omega_1, \Sigma) + (1-\eta) f_{DT}(\Omega_2, \Sigma) - f_{DT}(\eta\Omega_1 + (1-\eta)\Omega_2, \Sigma) \geq 0. \tag{E.11}$$

Using the definition of $f_{DT}(\Omega, \Sigma)$ given in (3.1), we can write the left-hand side of inequality (E.11) as follows:

$$\begin{aligned}
&\eta\left(\frac{1}{2}\text{trace}(\Omega_1^2\Sigma) - \text{trace}(\Omega_1)\right) + (1-\eta)\left(\frac{1}{2}\text{trace}(\Omega_2^2\Sigma) - \text{trace}(\Omega_2)\right) \\
&- \frac{1}{2}\text{trace}((\eta\Omega_1 + (1-\eta)\Omega_2)^2\Sigma) + \text{trace}(\eta\Omega_1 + (1-\eta)\Omega_2) \\
&= \eta\frac{1}{2}\text{trace}(\Omega_1^2\Sigma) + (1-\eta)\frac{1}{2}\text{trace}(\Omega_2^2\Sigma) - \eta^2\frac{1}{2}\text{trace}(\Omega_1^2\Sigma) - (1-\eta)^2\frac{1}{2}\text{trace}(\Omega_2^2\Sigma) \\
&- \eta(1-\eta)\frac{1}{2}\text{trace}(\Omega_1\Omega_2\Sigma) - \eta(1-\eta)\frac{1}{2}\text{trace}(\Omega_2\Omega_1\Sigma) = \eta(1-\eta)\frac{1}{2}\text{trace}(\Omega_1^2\Sigma) \\
&+ \eta(1-\eta)\frac{1}{2}\text{trace}(\Omega_2^2\Sigma) - \eta(1-\eta)\frac{1}{2}\text{trace}(\Omega_1\Omega_2\Sigma) - \eta(1-\eta)\frac{1}{2}\text{trace}(\Omega_2\Omega_1\Sigma) \\
&= \eta(1-\eta)\frac{1}{2}\text{trace}((\Omega_1 - \Omega_2)^2\Sigma),
\end{aligned} \tag{E.12}$$

which is always positive for any $\Omega_1,\ \Omega_2 \succ 0$ and $\eta \in [0,1]$.

Next, we show that the convex function $f_{DT}(\Omega, \Sigma)$ has a unique minimizer in Σ^{-1}. To check verify this statement, we consider the derivative of $f_{DT}(\Omega, \Sigma)$

with respect to Ω and set it to zero:

$$\frac{\partial f_{DT}(\Omega,\Sigma)}{\partial \Omega} = \frac{\partial}{\partial \Omega}\left(\frac{1}{2}\text{trace}(\Omega^2\Sigma) - \text{trace}(\Omega)\right) = \frac{\Sigma\Omega + \Omega\Sigma}{2} - I = 0. \tag{E.13}$$

The matrix equation (E.13) is known as the *Lyapunov equation*. We verify that the (E.13) holds for $\Omega = \Sigma^{-1}$. Firstly, we consider the eigen-decomposition of the matrix Σ as $\Sigma = PUP^T$. Matrix $U = \text{diag}\{u_i, ..., u_p\}$ is diagonal and contains the eigenvalues of Σ. Secondly, we pre-multiply the last expression in (E.13) by P^T and post-multiply by P.

$$\frac{P^T(PUP^T\Omega + \Omega PUP^T)P}{2} - I = \frac{UP^T\Omega P + P^T\Omega PU}{2} - I = 0. \tag{E.14}$$

We denote $\bar{\Omega} = [\bar{\omega}_{ij}]_{1\leq i,j\leq p} = P^T\Omega P$. We can write (E.14) in terms of the matrix entries as follows:

$$(u_i\bar{\omega}_{ii} + \bar{\omega}_{ii}u_i)/2 - 1 = 0, \quad \text{for} \quad 1 \leq i \leq p, \tag{E.15}$$

and

$$(u_i\bar{\omega}_{ik} + \bar{\omega}_{ik}u_k)/2 = 0, \quad \text{for} \quad 1 \leq i,k \leq p, \ i \neq k. \tag{E.16}$$

From the equation (E.15) we obtain $\bar{\omega}_{ii} = u_i^{-1}$ for $1 \leq i \leq p$. On the other hand, from the equation (E.16) we obtain $\bar{\omega}_{ik} = 0$ for $1 \leq i,k \leq p$ and $i \neq k$. Therefore, $\bar{\Omega} = \text{diag}\{u_i^{-1}, ..., u_p^{-1}\}$ is a diagonal matrix. Finally, using the definition of $\bar{\Omega}$, we obtain $\Omega = P\bar{\Omega}P^T = P\text{diag}\{u_i^{-1}, ..., u_p^{-1}\}P^T = PU^{-1}P^T = \Sigma^{-1}$.

Finally, we consider the Hessian matrix of function $f_{DT}(\Omega,\Sigma)$, which can be written as follows:

$$\frac{\partial^2 f_{DT}(\Omega,\Sigma)}{\partial \Omega^2} = \frac{\Sigma \otimes I + I \otimes \Sigma}{2}, \tag{E.17}$$

where $\otimes$ is the Kronecker product. Note that the Hessian matrix (E.17) is positive-definite because the matrix Σ is positive-definite.

Remark E.5. Consider the following optimization problem

$$\min_{\Omega=\Omega^T} \frac{1}{2}\text{trace}(\Omega^2) - \text{trace}(\Omega A) + \tau||\Omega||_{1,\text{off}}. \tag{E.18}$$

Here we show that we can represent the solution of the problem (E.18) through the soft thresholding operator.

In order to solve the problem (E.18), we set the partial derivative of its objective function to zero.

$$\begin{aligned} 0 &\in \frac{\partial}{\partial\Omega}\left(\frac{1}{2}\text{trace}(\Omega^2) - \text{trace}(\Omega A) + \tau||\Omega||_{1,\text{off}}\right) \\ &= \Omega - A + \tau\partial||\Omega||_{1,\text{off}}. \end{aligned} \tag{E.19}$$

The entries of the matrix $\partial||\Omega||_{1,\text{off}}$ depend on whether the corresponding entries ω_{ij} are positive, negative or equal to zero. Below we consider each of the cases.

When $\omega_{ij} > 0$, for $1 \leq i, j \leq p$ and $i \neq j$, then $\omega_{ij} - A_{ij} + \tau = 0$. Therefore, $\omega_{ij} = A_{ij} - \tau$. Since $\omega_{ij} > 0$, we have that $A_{ij} > \tau$.

When $\omega_{ij} < 0$, for $1 \leq i, j \leq p$ and $i \neq j$, then $\omega_{ij} - A_{ij} - \tau = 0$. Therefore, $\omega_{ij} = A_{ij} + \tau$. Since $\omega_{ij} < 0$, we have that $A_{ij} < -\tau$.

When $\omega_{ij} = 0$, for $1 \leq i, j \leq p$ and $i \neq j$, then $\omega_{ij} - A_{ij} \in [-\tau, \tau]$. Therefore, $\omega_{ij} \in [A_{ij} - \tau, A_{ij} + \tau]$. Since $\omega_{ij} = 0$, we have that $-\tau \leq A_{ij} < \tau$.

Finally, since we consider the off-diagonal ℓ_1 norm penalization of Ω, we can write $\omega_{ii} - A_{ii} = 0$, for $1 \leq i \leq p$. Summarizing all the cases, we obtain the expression given in (3.6). Similarly, we can write the entries ω_{ij} of the solution in terms of the soft thresholding operator, given as:

$$\omega_{ij} = \text{sign}(A_{ij})\max(|A_{ij}| - \tau, 0)\mathbb{I}_{i\neq j} + A_{ij}\mathbb{I}_{i=j}. \tag{E.20}$$

Straightforwardly, we can prove the equality between the problem (3.9) and the weighted adaptive thresholding operator (3.7), simply by substituting τ with $\frac{\tau}{|w_{ij}|}$ in the proof given above.

Bibliography

Anderson, T., W. (2003). *An Introduction to Multivariate Statistical Analysis.* Wiley-Interscience.

Banerjee, O., El Ghaoui, L., and d'Aspremont, A. (2008). Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *The Journal of Machine Learning Research*, 9:485–516.

Banerjee, O., El Ghaoui, L., d'Aspremont, A., and Natsoulis, G. (2006). Convex optimization techniques for fitting sparse gaussian graphical models. Pittsburg. Proceedings of the 23^{rd} International Conference on Machine Learning.

Banerjee, S. and Ghosal, S. (2015). Bayesian structure learning in graphical models. *The Journal of Multivariate Analysis*, 136:147–162.

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

Bühlmann, P. and van de Geer, S. (2011). *Statistics for High-Dimensional Data*. Springer-Verlag GmbH.

Cai, T., Liu, W., and Luo, X. (2011). A constrained ℓ_1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association*, 106(494):594–607.

Cai, T. and Yuan, M. (2012). Adaptive covariance matrix estimation through block thresholding. *The Annals of Statistics*, 40(4):2014–2042.

Chen, X., Xu, M., and Wu, W. B. (2013). Covariance and precision matrix estimation for high-dimensional time series. *The Annals of Statistics*, 41(6):2994–3021.

Cui, Y., Leng, C., and Sun, D. (2014). Sparse estimation of high-dimensional correlation matrices. *Computational Statistics and Data Analysis*, Preprint avilable at http://dx.doi.org/10.1016/j.csda.2014.10.001.

d'Aspremont, A., Banerjee, O., and Ghaoui, L. (2008). First-order methods for sparse covariance selection. *SIAM J. Matrix Anal Appl.*, 30:56–66.

DeMiguel, V., Garlappi, L., Nogales, F. J., and Uppal, R. (2009). A generalized approach to portfolio optimization: Improving performance by constraining portfolio norms. *Management Science*, 55(5):798–812.

DeMiguel, V., Martin-Utrera, A., and Nogales, F. J. (2013). Size matters: Optimal calibration of shrinkage estimators for portfolio selection. *Journal of Banking & Finance*, 37(8):3018–3034.

Dempster, A. (1972). Covariance selection. *Biometrics*, 28(1):157–175.

Deng, X. and Tsui, K. (2013). Penalized covariance matrix estimation using a matrix-logarithm transformation. *Journal of Computational and Graphical Statistics*, 22(2):494–512.

Duchi, J., Gould, S., and Koller, D. (2008). Projected subgradient methods for learning sparse gaussians. In *Proceeding of the 24th Conference on Uncertainty in Artificial Intelligence*.

El Karoui, N. (2008). Operator norm consistent estimation of large-dimensional sparse covariance matrices. *The Annals of Applied Statistics*, 36(6):2717–2756.

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

Frahm, G. and Memmel, C. (2010). Dominating estimator for minimum-variance portfolios. *Journal of Econometrics*, 159:289–302.

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

Goto, S. and Xu, Y. (2013). Improving mean variance optimization through sparse hedging restrictions. *Journal of Financial and Quantitative Analysis, (Forthcoming)*.

Haff, L. R. (1980). Estimation of the inverse covariance matrix: Random mixtures of the inverse wishart matrix and the identity. *The Annals of Statistics*, 8(3):586–597.

Hess, L., Anderson, K., Symmans, W. F., Valero, V., Ibrahim, N., Mejia, J. A., B. D., Theriault, R. L., Buzdar, A. U., Dempsey, P. J., Rouzier, R., Sneige, N., Ross, J. S., Vidaurre, R., Gomez, H. L., Hortobagyi, G. N., and Pusztai, L. (2006). Pharmacogenomic predictor of sensitivity to preoperative chemotherapy with paclitaxel and fluorouracil, doxorubicin, and cyclophosphamide in breast cancer. *Journal of Clinical Oncology*, 24:4236–4244.

Huang, S., Li, J., Sun, L., Ye, J., Fleisher, A., Wu, T., Chen, K., and Reiman, E. (2010). Learning brain connectivity of alzheimer's disease by sparse inverse covariance estimation. *NeuroImage*, 50:935–949.

James, W. and Stein, C. (1961). *Estimation with quadratic loss.* Univ. of Calif. Press, Proc. Fourth Berkeley Symp. Math. Statist. Probab.

Johnstone, I. M. (2001). On the distribution of the largest eigenvalue in principal component analysis. *The Annals of Statistics*, 29(3):295–327.

Jorissen, R. N., Lipton, L., Gibbs, P., Chapman, M., Desai, J., Jones, I. T., Yeatman, T. J., East, P., Tomlinson, I. P., Verspaget, H. W., Aaltonen, L. A., Kruhøffer, M., Orntoft, T. F., Andersen, C. L., and Sieber, O. M. (2008). DNA copy-number alterations underlie gene expression differences between microsatellite stable and unstable colorectal cancers. *Clinical Cancer Research*, 14(24):8061–8069.

Khan, J., Wei, J. S., Ringner, M., Saal, L. H., Ladanyi, M., Westermann, F., Berthold, F., Schwab, M., Antonescu, R., A., Peterson, C., and Meltzer,

P. S. (2001). Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature Medicine*, 7:673–679.

Kourtis, A., Dotsis, G., and Markellos, N. (2012). Parameter uncertainty in portfolio selection: Shrinking the inverse covariance matrix. *Journal of Banking & Finance*, 36:2522–2531.

Kuerer, H. M., Newman, L. A., Smith, T. L., Ames, F. C., Hunt, K. K., Dhingra, K., Theriault, R. L., Singh, G., Binkley, S. M., Sneige, N., Buchholz, T. A., Ross, M. I., McNeese, M. D., Buzdar, A. U., Hortobagyi, G. N., and Singletary, S. E. (1999). Clinical course of breast cancer patients with complete pathologic primary tumor and axillary lymph node response to doxorubicin-based neoadjuvant chemotherapy. *Journal of Clinical Oncology*, 17(2):460–469.

Kullback, S. and Leibler, R. A. (1951). *The Annals of Mathematical Statistics*, 22(1):79–86.

Lauritzen, S. (1996). *Graphical Models.* Clarendon Press. Oxford.

Ledoit, O. and Wolf, M. (2004). A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88:365–411.

Ledoit, O. and Wolf, M. (2012). Nonlinear shrinkage estimation of large-dimensional covariance matrices. *The Annals of Statistics*, 40(2):1024–1060.

Li, L. and Toh, K. (2010). An inexact interior point method for ℓ_1-regurized sparse covariance selection. *Mathematical Programming Computation*, 2:291–315.

Mardia, K. V., Kent, J. T., and Bibby, J. M. (1979). *Multivariate Analysis.* Academic Press, New York.

Matthews, B. W. (1975). Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochimica et Biophysica Acta*, 405:442–451.

Maurya, A. (2014). A joint convex penalty for inverse covariance matrix estimation. *Computational Statistics and Data Analysis*, 75:15–27.

McLachlan, S. (2004). *Discriminant Analysis and Statistical Pattern Recognition.* Willey Interscience.

Meinshausen, N. and Bühlmann, P. (2006). High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, 34(2):1436–1462.

Monti, R. P., Hellyer, P., Sharp, D., Leech, R., Anagnostopoulos, C., and Montana, G. (2014). Estimating time-varying brain connectivity networks from functional mri time series. *Neuroimage*, 103:427–443.

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

Ravikumar, P., Wainwright, M., Raskutti, G., and Yu, B. (2011). High-dimensional covariance estimation by minimizing ℓ_1-penalized log-determinant divergence. *Electronic Journal of Statistics*, 5:935–980.

Rothman, A., Bickel, P., and Levina, E. (2009). Generalized thresholding of large covariance matrices. *Journal of the American Statistical Association*, 104(485):177–186.

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

Rothman, A. J. (2012). Positive definite estimators of large covariance matrices. *Biometrika*, 99(2):733–740.

Ryali, S., Chen, T., Supekar, K., and Menon, V. (2012). Estimation of functional connectivity in fMRI data using stability selection-based sparse partial correlation with elastic net penalty. *NeuroImage*, 59(4):3852–3861.

Schafer, J. and Strimmer, K. (2005). A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics.

Statistical Applications in Genetics and Molecular Biology, 4(1):Article 32.

Scheinberg, K., Ma, S., and Goldfarb, D. (2010). Sparse inverse covariance selection via alternating linearization methods. In *Advances in Neural Information Processing Systems*.

Shi, L., Reid, L. H., Jones, W. D., Shippy, R., Warrington, J. A., Baker, S. C., Collins, P. J., deLongueville, F., Kawasaki, E. S., Lee, K. Y., Luo, Y., Sun, Y. A., Willey, J. C., Setterquist, R. A. Fischer, G. M., Tong, W., Dragan, Y. P., Dix, D. J., Frueh, F. W., Goodsaid, F. M., Herman, D., Jensen, R. V., Johnson, C. D., Lobenhofer, E. K., Puri, R. K., Scherf, U., Thierry-Mieg, J., Wang, C., Wilson, M., and Wolber, P. K. (2010). The microarray quality control (MAQC)-II study of common practices for the development and validation of microarray-based predictive models. *Nature Biotechnology*, 28(8):827–838.

Stevens, G. V. G. (1998). On the inverse of the covariance matrix in portfolio analysis. *The Journal of Finance*, 53(5):1821–1827.

Stifanelli, P. F., Creanza, T. M., Anglani, R., Liuzzi, V. C., Mukherjee, S., Schena, F. P., and Ancona, N. (2013). A comparative study of covariance selection models for the inference of gene regulatory networks. *Journal of Biomedical Informatics*, 46:894–904.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society*, 58(1):267–288.

Touloumis, A. (2015). Nonparametric Stein-type shrnikage covariance matrix estimators in high-dimensional settings. *Computational Statistics and Data Analysis*, 83:251–261.

Wang, Y. and Daniels, M. J. (2014). Computationally efficient banding of large covariance matrices for ordered data and connections to banding the inverse Cholesky factor. *Journal of Multivariate Analysis*, 130:21–26.

Warton, D. I. (2008). Penalized normal likelihood and ridge regularization of correlation and covariance matrices. *Journal of the American Statistical Association*, 103(481):340–349.

Whittaker, J. (1990). *Graphical Models in Applied Multivariate Statistics*. Chichester: Wiley.

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

Xue, L., Ma, S., and Zou, H. (2012). Positive-definite ℓ_1-penalized estimation of large covariance matrices. *Journal of the American Statistical Association*, 107(500):1480–1491.

Yin, J. and Li, J. (2013). Adjusting for high-dimensional covariates in sparse precision matrix estimation by ℓ_1-penalization. *Journal of Multivariate Analysis*, 116:365–381.

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

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

Zerenner, T., Friederichs, P., Lehnertz, K., and Hense, A. (2014). A gaussian graphical model approach to climate networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 24(2):023103.

Zhang, T. and Zou, H. (2014). Sparse precision matrix estimation via lasso penalized d-trace loss. *Biometrika*, 88:1–18.

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

Zhou, S., van de Geer, S., and Bühlmann, P. (2009). Adaptive lasso for high dimensional regression and gaussian graphical modeling. Available at http://arxiv.org/abs/0903.2515.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429.