Improving the Graphical Lasso Estimation for the Precision Matrix Through Roots of the Sample Covariance Matrix

Vahe Avagyan[a], Andrés M. Alonso[b], Francisco J. Nogales[c,*]

[a]*Department of Statistics, Universidad Carlos III de Madrid, c/ Madrid 126, 28903 Getafe (Madrid), Spain*
[b]*Department of Statistics and IFL, Universidad Carlos III de Madrid, c/ Madrid 126, 28903 Getafe (Madrid), Spain*
[c]*Department of Statistics, Universidad Carlos III de Madrid, Av. de la Universidad 30, 28911 Leganes (Madrid), Spain*

Abstract

In this paper, we focus on the estimation of a high-dimensional inverse covariance (precision) matrix. We propose a simple improvement of the 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.

Keywords: Gaussian Graphical Model, High-dimensionality, Penalized estimation, Gene expression, Portfolio selection.

1. Introduction

In recent years, there has been a growing interest in estimating the inverse covariance matrix (also known as precision or concentration matrix) in high dimensional settings. It is an important problem in many statistical methodologies and research fields. For instance, in finance an accurate precision matrix is required when computing optimal portfolios for a large number of assets (Stevens 1998; Frahm and Memmel 2010; Goto and Xu 2013). In machine or statistical learning methods, such as classification or clustering, a proper estimation of the precision matrix is fundamental

*Corresponding author. Tel:+34 91 6248773
Email addresses: vahe.avagyan@uc3m.es (Vahe Avagyan), andres.alonso@uc3m.es (Andrés M. Alonso), fcojavier.nogales@uc3m.es (Francisco J. Nogales)

Accepted at Journal of Computational and Graphical Statistics
https://doi.org/10.1080/10618600.2017.1340890

when dealing with a vast amount of predictor variables (Mardia et al. 1979; McLachlan 2004). The applications involving Gaussian Graphical Models (GGM) are particularly important, where the precision matrix is assumed to be sparse and its non-zero entries are related with the partial correlation coefficients (Dempster 1972; Lauritzen 1996). One notable application where the precision matrix is intrinsically sparse is the estimation of genetic regulatory networks through high dimensional microarray gene expression data (Stifanelli et al. 2013; Yin and Li 2013). Another application involving sparse precision matrices is the estimation of functional brain connectivity networks through neuroimaging techniques (Huang et al. 2010).

In this paper, we focus on the estimation of a high-dimensional precision matrix. There are several approaches that try to estimate efficiently such matrices. We assume that a $n \times p$ *centered* sample data matrix, $\mathbf{X}$, is observed, where each row $X_i = (X_{i1}, ..., X_{ip})$ is a realization of a p-variate random vector that is independent and identically distributed for $i = 1, ..., n$, and has a covariance matrix Σ with the corresponding precision matrix $\Omega = \Sigma^{-1}$.

We first classify the precision matrix estimation approaches by considering those that estimate it by inverting an estimator of the covariance matrix, and those that estimate the precision matrix directly. We refer the former approaches as *two-step* estimation procedures where a covariance matrix must be estimated in the first step. The classical estimator of the covariance matrix is the sample covariance matrix S. However, when the ratio between the number of variables p and the number of observations n is small but close to one, then the bias of the corresponding inverse of the classical estimator may be large, $E(S^{-1}) - \Omega = \frac{p+2}{n-p-2}\Omega$, and the associated precision matrix may be highly unstable. For instance, when $p = n/2 - 2$, then $E(S^{-1}) - \Omega = \Omega$, i.e., the bias has the same magnitude as Ω. Moreover, when $p/n > 1$, the classical estimator is not invertible. To overcome these difficulties, some techniques have been proposed to deal with the estimation of the covariance matrix when the dimension p is large compared with the number of observations n. In essence, all these approaches try to mitigate the effect of the smallest eigenvalues of the covariance matrix (see Chamberlain and Rothschild 1983; Bai 2003). One of the well accepted approaches is the shrinkage estimator proposed by Ledoit and Wolf (2004) and extended by Schafer and Strimmer (2005). This estimator shrinks the sample covariance matrix toward a target matrix using a linear combination. Even though this estimator presents good practical and theoretical properties, the associated inverse estimator may not inherit such properties. In particular, when the dimension of the problem is high, this inverse estimator may not be optimal and may amplify the estimation error

of the covariance matrix estimator (Ledoit and Wolf 2012). Moreover, these two-step approaches do not provide, in general, sparse precision matrix estimations. For these reasons, our proposed methodology is based on the second class of approaches that attempt to estimate the precision matrix directly.

Following the ideas of Ledoit and Wolf (2004), a shrinkage approach can also be applied directly to the precision matrix estimation. In this way, Frahm and Memmel (2010) proposed a precision matrix estimation by considering a *convex* linear combination between the inverse of the sample covariance matrix and a target matrix. A similar study by Kourtis et al. (2012) considers a *conical* combination between the inverse of the sample covariance matrix and a target matrix. However, these two studies focus on reducing the out-of-sample variance of the portfolio returns, rather than obtaining a better precision matrix estimator. Furthermore, these two methods rely on the assumption that the ratio between the number of variables and the number of observations is small enough ($p << n$).

Moreover, as explained previously, recent applications require the estimation of GGMs where conditional dependencies between the variables are estimated through the off-diagonal and nonzero entries of the precision matrix, which is assumed intrinsically sparse. To attain a sparsity pattern in the estimated precision matrix and deal with the case $p/n > 1$, the ℓ_1 or Lasso (Least Absolute Shrinkage and Selection Operator) regularization framework can be applied. This approach was proposed by Tibshirani (1996) in the regression framework. Banerjee et al. (2006) proposed the precision matrix estimation by maximizing the ℓ_1-penalized log-likelihood function to attain sparse solution[1].

This approach has been extensively analysed by other authors (e.g., Yuan and Lin 2007; d'Aspremont et al. 2008; Banerjee et al. 2008; Rothman et al. 2008; Yin and Li 2013). Several efficient algorithms have been developed to solve the problem efficiently, such as the Graphical Lasso (Friedman et al. 2008), a Project Sub-gradient Method (Duchi et al. 2008), an Alternating Linear Minimization (Scheinberg et al. 2010), and an Interior Point method (Li and Toh 2010), among others.

Regarding non-likelihood approaches, Meinshausen and Bühlmann (2006) proposed a neighborhood selection framework based on lasso regressions. Yuan (2010) proposed the use of the Dantzig

[1]Other penalty functions have been proposed to regularize the log-likelihood, see Fan et al. (2009).

selector to replace the lasso regression in this framework. Finally, Cai et al. (2011) introduced the constrained ℓ_1 - minimization based on constraining the ℓ_1 norm of the precision matrix (also known as clime estimator).

In this paper, we focus on the ℓ_1 penalized log-likelihood maximization approach and propose a simple modification that is able to attain a better statistical performance without sacrificing too much the computational cost. One of the most efficient algorithms to compute numerically the ℓ_1 penalized log-likelihood estimator is the *glasso* framework. This framework allows a fast, efficient and stable solution for high-dimensional problems. The glasso algorithm 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 more spread 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 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 spread eigenvalues and therefore, obtain a more accurate estimation of $\Omega^{1/k}$ and also Ω.

The proposed k-root glasso algorithm is a simple modification of the glasso one. Similar to the original glasso method, 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 performance measures. In particular, we use the entropy loss and the mean squared error to measure the statistical performance. 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 associated tuning (penalty) parameter that regularizes the log-likelihood. Finally, for the proposed k-root glasso method, we establish the convergence rate in the Frobenius norm.

The rest of the manuscript is organized as follows. Section 2 describes the proposed k-root glasso (or simply r-glasso) methodology to estimate large precision matrices. Section 3 proposes a different approach for selecting both the k-root of the sample covariance matrix and the associated penalty

parameter that regularizes the log-likelihood. Section 4 exhaustively evaluates the statistical loss and GGM performance of the proposed methodology and compares with that of the glasso one. Section 5 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. Section 6 provides the conclusions. Finally, Appendix A develops the analytical convergence rate of the proposed method and Appendix B contains the tables of the numerical results.

2. Proposed *k*-root glasso framework

Before proposing the k-root glasso methodology, we introduce the following notations. For any vector $\mathbf{a} = (a_1, ..., a_p)^T \in \mathbb{R}^p$, the ℓ_1 or Manhattan norm is denoted by $||\mathbf{a}||_1 = \sum_{j=1}^{p} |a_j|$, the ℓ_2 or Euclidean norm by $||\mathbf{a}||_2 = \sqrt{\sum_{j=1}^{p} a_j^2}$, and the ℓ_∞ or Maximum norm by $||\mathbf{a}||_\infty = \max(|a_1|, ..., |a_p|)$. For any symmetric matrix $\mathbf{A} = [a_{ij}]_{1 \leq i,j \leq p}$, the componentwise ℓ_1 norm is denoted by $||\mathbf{A}||_1 = \sum_{i=1}^{p} \sum_{j=1}^{p} |a_{ij}|$, the componentwise ℓ_2 or Frobenius norm by $||\mathbf{A}||_2 = \sqrt{\sum_{i=1}^{p} \sum_{j=1}^{p} a_{ij}^2}$, the componentwise ℓ_∞ norm by $||\mathbf{A}||_\infty = \max_{1 \leq i,j \leq p} |a_{ij}|$, and the spectral norm by $||\mathbf{A}||_{\text{spec}} = \sup_{||x||_2 \leq 1} ||Ax||_2$. For any positive definite symmetric matrix A, $\lambda(A)$ will denote the vector of eigenvalues of matrix A, where $\lambda_{\max}(A) = \max \lambda_i(A) = ||A||_{\text{spec}}$ and $\lambda_{\min}(A) = \min \lambda_i(A) = ||A||_{\min}$ denote the maximum and minimum eigenvalues, respectively. Finally, we assume that a centered sample data matrix, $\mathbf{X}$, is observed with dimension $n \times p$, where each row $X_i = (X_{i1}, ..., X_{ip})$ 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 $\Omega = \Sigma^{-1}$.

The glasso estimator is defined as the solution of the following optimization problem:

$$\widehat{\Omega}_{\text{glasso}} = \arg\max_{\Omega} \ \log\det\Omega - \text{trace}(S\Omega) - \nu||\Omega||_1, \tag{1}$$

where $S = (1/n) \sum_{i=1}^{n} X_i X_i^T$ is the sample covariance matrix and $\nu > 0$ is a penalty parameter. This parameter controls the sparsity pattern of the glasso estimation.

Note that problem (1) is convex, and its dual problem (2) is defined as (Banerjee et al. 2008):

$$\begin{aligned} &\widehat{\Omega}_{\text{glasso}} = \arg\min_{\Omega} \ \log\det\Omega \\ &\text{subject to} \ \ ||\Omega^{-1} - S||_\infty \leq \nu. \end{aligned} \tag{2}$$

As discussed in Section 1, the glasso estimation 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 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 k-root glasso estimator:

$$\begin{aligned} \widehat{\Omega}_{\text{r-glasso}} &= \arg\min_{\Omega} \ \log\det\Omega \\ &\text{subject to} \ \ ||\Omega^{-1/k} - S^{1/k}||_\infty \leq \xi_k, \end{aligned} \tag{3}$$

where $\xi_k > 0$ is the associated penalty parameter. The problem (3) can be rewritten as

$$\begin{aligned} \widehat{\Gamma} &= \arg\min_{\Gamma} \ \log\det\Gamma \\ &\text{subject to} \ \ ||\Gamma^{-1} - S^{1/k}||_\infty \leq \xi_k, \end{aligned} \tag{4}$$

and we define the k-root glasso estimator as $\widehat{\Omega}_{\text{r-glasso}} = \widehat{\Gamma}^k$, for a given k and ξ_k. Note that the primal problem of the optimization problem (4) can be written as the following:

$$\widehat{\Gamma} = \arg\max_{\Gamma} \ \log\det\Gamma - \text{trace}(S^{1/k}\Gamma) - \xi_k||\Gamma||_1. \tag{5}$$

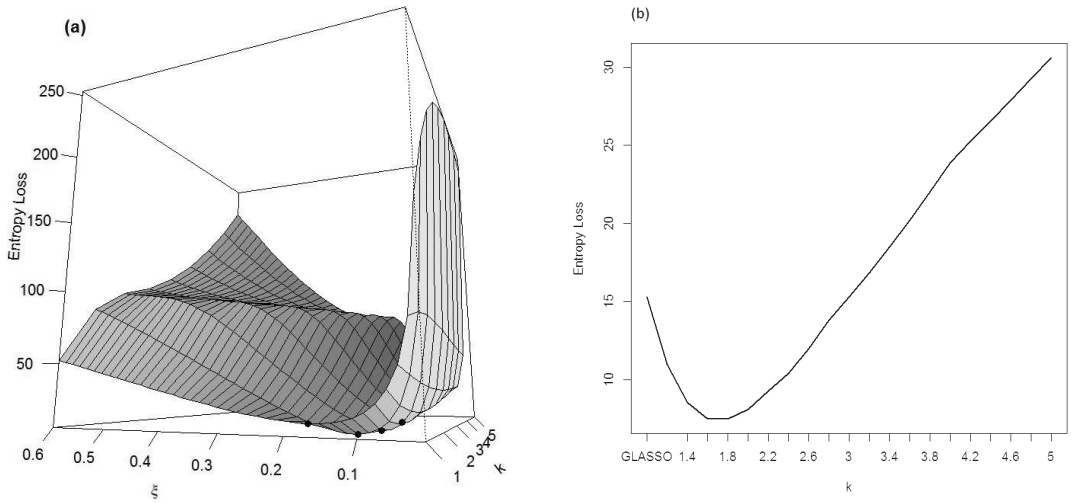
Therefore, we can obtain the proposed estimator $\widehat{\Omega}_{\text{r-glasso}} = \widehat{\Gamma}^k$ by solving the problem (5) using the same algorithm as for the problem (2) without any additional cost. Finally, we note that when $k = 1$, the k-root glasso estimator reduces to the original one, and, moreover, when $\xi_k = 0$, we obtain the classical naive estimator S^{-1} for any value of k.

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 other elements are 0. For this example we specify the values $p = 200$ and $n = 200$.

In Figure 1(a), the entropy loss (see Section 4.2 for a formal definition) 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 the k-root 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. That is, the minimum possible error of the glasso estimator along the ν path is larger than the minimum possible error of the proposed k-root glasso estimator along the ξ_k path, for some values of k. This improvement can be observed more clearly in Figure 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 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 4, through an exhaustive empirical analysis including several sparsity patterns for the precision matrix, we show how the proposed k-root glasso estimator can outperform the glasso one under other statistical performance measures covering those for graphical models.

3. 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 and especially its sparsity level. To account for this sparsity level, we suggest the use of the BIC-type criterion.[2] The

[2]In one of the empirical applications in Section 5, we use a cross-validation procedure to calibrate the penalty parameter, since in this application the sparsity pattern is not important.

original BIC criterion is proposed by Yuan and Lin (2007), which has the following formulation:

$$\mathrm{BIC}(\xi) = n\left(-\log\det\widehat{\Omega}(\xi) + \mathrm{trace}(S\widehat{\Omega}(\xi))\right) + \log n \times \mathrm{NZ}, \tag{6}$$

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

$$\mathrm{BIC}(\xi_k, k) = n\left(-\log\det\widehat{\Omega}(\xi_k, k) + \mathrm{trace}(S\widehat{\Omega}(\xi_k, k))\right) + \log n \times \mathrm{NZ}, \tag{7}$$

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 $\mathrm{BIC}(\xi_k, k)$ using a two-dimensional grid search.

4. 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 one $\widehat{\Omega}_{\text{glasso}}$. Specifically, in subsection 4.1 we detail the considered models for the precision matrix Ω, and in subsection 4.2 we describe the performance evaluation. Finally, in subsection 4.3 we provide the discussion of the results.

4.1. 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 (where the sparsity pattern and the elements are not fixed across replications) and non-random (with fixed sparsity pattern and deterministic elements). The considered models for the precision matrix Ω are the following:

(i) Random models[3]

- *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,
- *Model* 3. A random p.d. matrix, containing 20% of non-zero entries,

[3]The random models are generated using the MATLAB command *sprandsym*.

- *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 other values are 0 (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.

4.2. Performance evaluation

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

$$\mathrm{KLL}(\widehat{\Omega}, \Omega) = \mathrm{trace}(\Omega^{-1}\widehat{\Omega}) - \log\det(\Omega^{-1}\widehat{\Omega}) - p. \tag{8}$$

The KL 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:

$$\mathrm{MSE}(\widehat{\Omega}, \Omega) = ||\widehat{\Omega} - \Omega||_2^2. \tag{9}$$

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

$$\text{Specificity} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}, \tag{10}$$

$$\text{Sensitivity} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{11}$$

$$\mathrm{MCC} = \frac{\mathrm{TP} \times \mathrm{TN} - \mathrm{FP} \times \mathrm{FN}}{\sqrt{(\mathrm{TP} + \mathrm{FP})(\mathrm{TP} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})(\mathrm{TN} + \mathrm{FN})}}, \tag{12}$$

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 (7). We also focus on the square-root glasso procedure, $k = 2$, because of its good behaviour in practice. Finally, we include a comparison with the method clime[4] (see Cai et al. 2011) as it is one of the popular estimators for the precision matrix.[5]

4.3. 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 number of steps for all the methods. Table 1 provides the computational times of the three estimators for model 5[6]. 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.

The simulation results are provided in the Appendix B to conserve space (see Tables B.6-B.10). Each table reports the averages over 100 replications and the standard deviations (SD) of the corresponding losses and prediction measurements.

[4]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`

[5]The penalty parameters for glasso and clime methods are estimated using the BIC criterion (6).

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

Table 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

As it can be seen from Tables B.6 and B.7, the proposed r-glasso method provides lower KL loss and MSE than glasso for all the models. Therefore, r-glasso method outperforms glasso in terms of the statistical losses.

Tables B.8, B.9 and B.10 illustrate the results of the GGM prediction performances. From Tables B.8 and B.10, we observe that in terms of specificity and MCC[7] r-glasso outperforms glasso for all the models. Finally, r-glasso outperforms glasso in terms of sensitivity (see Table B.9) 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, r-glasso provides better results for models 2, 3, 4, 5 and similar results for models 1, 6, 7 in terms of KL loss. 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 cost of clime is considerably larger than that of r-glasso (see Table 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.

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

5. Real Data Analysis

In this section, we conduct an empirical analysis of the proposed r-glasso method through three real-data applications: the first two aimed at predicting tumours while the last one aimed at selecting a large financial portfolio.

5.1. 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,[8] analysed previously 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 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 the methods are estimated using the BIC criterion (6). We analyse the performance of the r-glasso method when the parameter k is selected from a range 2 to 4[9]. The estimated precision matrix is used in the Linear Discriminant Analysis (LDA) score:

$$\delta_t(Y) = Y^T \widehat{\Omega}\widehat{\mu}_t - \frac{1}{2}\widehat{\mu}_t^T \widehat{\Omega}\widehat{\mu}_t, \tag{13}$$

where $t = 1, 2$ (i.e., $t = 1$ for pCR and $t = 2$ for RD) and $\widehat{\mu}_t = \dfrac{1}{n_t}\sum_{i \in class_t} x_i$ 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 4.2. Moreover, we consider TP

[8] Available at `http://bioinformatics.mdanderson.org/pubdata.html`.

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

Table 2. Average pCR 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

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 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 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 estiamtors in terms of specificity and MCC. However, we note that clime is computationally time-consuming.

As a robustness check, we repeat the same application by considering the most significant 200 genes instead of 113. We provide the results in Table 3.

Table 3. Average pCR 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. We observe that r-glasso and clime provide very similar results.

5.2. 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 the same dataset analysed by Khan et al. (2001), which contains the expression levels of 2308 genes for 64 tissue samples.[10] 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).

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 by the level of the information that they provide using the F-statistics (Rothman et al. 2009), defined as

$$F = \frac{\frac{1}{m-1}\sum_{i=1}^{m} n_i(\bar{x}_i - \bar{x})^2}{\frac{1}{n-m}\sum_{i=1}^{m}(n_i - 1)s_i^2}, \tag{14}$$

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 the methods are estimated using the BIC criterion (6). We analyse the performance of the r-glasso method when the parameter k is selected from a range 2 to 4[11]. The estimated precision matrix is used in the LDA score $\delta_t(Y)$, defined as (13), where $t = 1, 2, 3, 4$ is the index of tumour class. To measure the

[10] Available at `http://www.bioinf.ucd.ie/people/aedin/R/full/_datasets/`.

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

prediction accuracy, we use the average proportion of correctly classified tissues:

$$AP = \frac{1}{100}\sum_{i=1}^{100}\frac{NCC_i}{14}, \tag{15}$$

where NCC_i is the number of correctly classified tissues in the i-the 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 4.

Table 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.

5.3. 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 (see DeMiguel et al. 2009):

$$w_{MVP} = \frac{\Omega \mathbf{1}_p}{\mathbf{1}_p'\Omega\mathbf{1}_p}, \tag{16}$$

where $\mathbf{1}_p$ denotes a $p \times 1$ vector. As the minimum-variance portfolio depends directly on the estimation of the precision matrix, an accurate estimation of such a 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.[12] 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, 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.

To estimate the penalty parameters for the precision estimation methods, we propose the following methodology based on cross-validation.[13] 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 5 shows the out-of-sample variances for the different portfolios.

The 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.

6. Conclusions

In this paper, we provide a new approach for estimating high-dimensional precision matrices, using the 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.

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

[13] In this application, we do not calibrate the parameters using the BIC criterion because the sparsity pattern of the precision matrix does not have an important role.

Table 5. The out-of-sample variances for different portfolios.

Method	$p = 80$	$p = 200$	$p = 300$
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

In particular, the proposed r-glasso method provides lower statistical losses and higher accuracy for covariance selection (e.g., prediction of Gaussian Graphical Models), 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 estimator in the Frobenius norm, under certain conditions.

Appendix A. Analytical Results

We analyse the convergence rate of the proposed estimator $\widehat{\Omega}_{\text{r-glasso}}$ for rational values of k. Before proceeding with our results, we state the following main assumptions on the precision matrix Ω:

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

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

for some positive values $\bar{\alpha}$ and $\underline{\alpha}$.

Note that the assumptions A1 and A2 guarantee the existence of the matrix Ω. Also let the set $Z = \{(i,j) : \Omega_{ij}^{(1/k)} \neq 0\}$ and $card(Z) \leq s$. The following theorem presents the convergence rate of the proposed r-glasso estimator.

Theorem 1. *Suppose $\widehat{\Omega}_{r\text{-}glasso}$ is the solution of problem (3) and $k \in \boldsymbol{N}$. Under the assumptions A1, A2, if $||\Sigma^{1/k} - S^{1/k}||_\infty = O_P(||\Sigma - S||_\infty)$ and $\xi_k \asymp \sqrt{\frac{\log p}{n}}$,*

$$||\widehat{\Omega}_{r\text{-}glasso} - \Omega||_2 = O_P\left(\sqrt{\frac{(p+s)\log p}{n}}\right). \tag{A.1}$$

Proof of Theorem 1: 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 (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:

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

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 $B1, B2$ and vice versa. We prove that under the assumptions of the Theorem 1

$$||\widehat{\Gamma} - \Omega^{1/k}||_2 = O_P\left(\sqrt{\frac{(p+s)\log p}{n}}\right). \tag{A.2}$$

The proof of (A.2) is inspired by Rothman et al. (2008). First, consider the following function

$$\begin{aligned} Q(\Theta) = \text{trace}(\Theta S^{1/k}) - \log\det(\Theta) + \xi_k||\Theta||_1 - \text{trace}(\Omega^{1/k}S^{1/k}) - \log\det(\Omega^{1/k}) - \\ \xi_k||\Omega^{1/k}||_1 = \text{trace}\left((\Theta - \Omega^{1/k})(S^{1/k} - \Sigma^{1/k})\right) - \left(\log\det(\Theta) - \log\det(\Omega^{1/k})\right) + \\ \text{trace}\left((\Theta - \Omega^{1/k})\Sigma^{1/k}\right) + \xi_k\left(||\Theta||_1 - ||\Omega^{1/k}||_1\right). \end{aligned} \tag{A.3}$$

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:

$$\Phi_n(M) = \{\Delta : \Delta = \Delta^T, ||\Delta||_2 = Mr_n\}, \tag{A.4}$$

where

$$r_n = \sqrt{\frac{(p+s)\log p}{n}} \to 0. \tag{A.5}$$

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

$$\inf\{G(\Delta) : \Delta \in \Phi_n(M)\} > 0, \tag{A.6}$$

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

$$G(\Delta) = \text{trace}\left(\Delta(S^{1/k} - \Sigma^{1/k})\right) - \left(\log\det(\Omega^{1/k} + \Delta) - \log\det(\Omega^{1/k})\right) + \text{trace}\left(\Delta\Sigma^{1/k}\right) + \xi_k\left(||\Omega^{1/k} + \Delta||_1 - ||\Omega^{1/k}||_1\right). \tag{A.7}$$

For the logarithm term in the equation $(A.7)$, doing the Taylor expansion of the function $f(t) = \log\det(\Theta + t\Delta)$, we get

$$\log\det(\Omega^{1/k} + \Delta) - \log\det(\Omega^{1/k}) = \text{trace}(\Sigma^{1/k}\Delta) - \tilde{\Delta}^T\left[\int_0^1 (1-\nu)(\Omega^{1/k} + \mu\Delta)^{-1} \otimes (\Omega^{1/k} + \mu\Delta)^{-1} d\nu\right]\tilde{\Delta}, \tag{A.8}$$

where $\otimes$ is the Kronecker product and $\tilde{\Delta}$ is a vectorization of Δ. The equation $(A.7)$ can be rewritten in the following form

$$G(\Delta) = \text{trace}\left(\Delta(S^{1/k} - \Sigma^{1/k})\right) + \tilde{\Delta}^T\left[\int_0^1 (1-\nu)(\Omega^{1/k} + \mu\Delta)^{-1} \otimes (\Omega^{1/k} + \mu\Delta)^{-1} d\nu\right]\tilde{\Delta} + \xi_k\left(||\Omega^{1/k} + \Delta||_1 - ||\Omega^{1/k}||_1\right) = T_1 + T_2 + T_3. \tag{A.9}$$

For an index set U and a matrix $A = [a_{ij}]$, denote $A_U = [a_{ij} I((i,j) \in U)]$, where $I(\cdot)$ is an indicator function. 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

$$T_3 = \xi_k\left(||\Omega^{1/k} + \Delta||_1 - ||\Omega^{1/k}||_1\right) \geq \xi_k\left(||\Delta_{\bar{Z}}||_1 - ||\Delta_Z||_1\right). \tag{A.10}$$

Next, consider the term T_1

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

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

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

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

$$\max_{ij}|(S^{1/k} - \Sigma^{1/k})_{ij}| = O_P\left(\sqrt{\frac{\log p}{n}}\right). \tag{A.13}$$

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

$$T_{11} \leq C_1\sqrt{\frac{\log p}{n}}||\Delta^-||_1 \leq C_1\sqrt{\frac{\log p}{n}}||\Delta||_1. \tag{A.14}$$

From the Cauchy-Schwartz inequality we get

$$T_{12} \leq \left[\sum_{i=1}^{p}(S^{1/k} - \Sigma^{1/k})_{ii}^2\right]^{1/2} ||\Delta^+||_2 \leq \sqrt{p}\max_{1\leq i\leq p}|(S^{1/k} - \Sigma^{1/k})_{ii}|||\Delta^+||_2 \leq$$
$$C_2\sqrt{\frac{p\log p}{n}}||\Delta^+||_2 \leq C_2\sqrt{\frac{(p+s)\log p}{n}}||\Delta||_2, \tag{A.15}$$

also with probability tending to 1.

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

$$T_2 \geq \lambda_{\min}\left(\int_0^1 (1-\nu)(\Omega^{1/k} + \mu\Delta)^{-1} \otimes (\Omega^{1/k} + \mu\Delta)^{-1}d\nu\right)||\Delta||_2^2 \geq$$
$$\int_0^1 (1-\nu)\lambda_{\min}^2(\Omega^{1/k} + \mu\Delta)^{-1}d\nu||\Delta||_2^2 \geq \frac{1}{2}\min_{0\leq\nu\leq 1}\lambda_{\min}^2(\Omega^{1/k} + \Delta)^{-1}||\Delta||_2^2 \geq$$
$$\frac{1}{2}\min\{\lambda_{\min}^2(\Omega^{1/k} + \Delta)^{-1}, ||\Delta||_2 \leq Mr_n\}||\Delta||_2^2. \tag{A.16}$$

On the other hand,

$$\lambda_{\min}^2(\Omega^{1/k}+\mu\Delta)^{-1} = \lambda_{\max}^{-2}(\Omega^{1/k}+\Delta) \geq (||\Omega^{1/k}|| + ||\Delta||)^{-2} \geq (\bar{\beta}+o(1))^{-2}, \tag{A.17}$$

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

$$T_2 \geq \frac{1}{2}||\Delta||_2^2(\bar{\beta}+o(1))^{-2} = \frac{1}{2}||\Delta||_2^2\gamma, \tag{A.18}$$

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

By our assumption in Theorem 1, $\xi_k \asymp \sqrt{\frac{\log p}{n}}$. Taking $\xi_k = \frac{C_1}{\epsilon}\sqrt{\frac{\log p}{n}}$ and using the obtained bounds $(A.10)$, $(A.11)$, $(A.18)$, we get

$$G(\Delta) \geq \frac{1}{2}||\Delta||_2^2\gamma - C_1\sqrt{\frac{\log p}{n}}||\Delta||_1 - C_2\sqrt{\frac{(p+s)\log p}{n}}||\Delta||_2 + \xi_k\left(||\Delta_{\bar{Z}}||_1 - ||\Delta_Z||_1\right) =$$
$$\frac{1}{2}||\Delta||_2^2\gamma - C_1\sqrt{\frac{\log p}{n}}(1-\frac{1}{\epsilon})||\Delta_{\bar{Z}}||_1 - C_1\sqrt{\frac{\log p}{n}}(1+\frac{1}{\epsilon})||\Delta_Z||_1 - C_2\sqrt{\frac{(p+s)\log p}{n}}||\Delta||_2. \tag{A.19}$$

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

$$||\Delta_Z||_1 \leq \sqrt{s}||\Delta_Z||_2 \leq \sqrt{s}||\Delta||_2 \leq \sqrt{s+p}||\Delta||_2. \tag{A.20}$$

Hence, we have

$$G(\Delta) \geq \frac{1}{2}||\Delta||_2^2\gamma - C_1\sqrt{\frac{(p+s)\log p}{n}}(1+\frac{1}{\epsilon})||\Delta||_2 - C_2\sqrt{\frac{(p+s)\log p}{n}}||\Delta||_2 \geq$$
$$||\Delta||_2^2\left[\frac{1}{4}\gamma - C_1\sqrt{\frac{(p+s)\log p}{n}}(1+\frac{1}{\epsilon})||\Delta||_2^{-1}\right] + ||\Delta||_2^2\left[\frac{1}{4}\gamma - C_2\sqrt{\frac{(p+s)\log p}{n}}||\Delta||_2^{-1}\right] =$$
$$||\Delta||_2^2\left[\frac{1}{4}\gamma - \frac{C_1}{M}(1+\frac{1}{\epsilon})\right] + ||\Delta||_2^2\left[\frac{1}{4}\gamma - \frac{C_2}{M}\right] > 0. \tag{A.21}$$

for M sufficiently large. This establishes the convergence rate $(A.2)$.

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

Lemma 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*

$$||A^q - B^q||_2 \overset{P}{\asymp} ||A-B||_2.. \tag{A.22}$$

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

$$A^q - B^q = \sum_{i=1}^{q} A^{q-i}(A-B)B^{i-1}. \tag{A.23}$$

Therefore, we can write the following:

$$\begin{aligned}
||A^q - B^q||_2^2 = \text{trace}\left((A^q - B^q)(A^q - B^q)^T\right) = \\
\text{trace}\left(\left(\sum_{i=1}^{q} A^{q-i}(A-B)B^{i-1}\right)\left(\sum_{i=1}^{q} B^{i-1}(A-B)A^{q-i}\right)\right) = \\
\text{trace}\left(\sum_{i=1}^{q}\sum_{j=1}^{q} A^{q-i}(A-B)B^{i-1}B^{j-1}(A-B)A^{q-j}\right) = \\
\text{trace}\left(\sum_{i=1}^{q}\sum_{j=1}^{q} A^{2q-i-j}(A-B)B^{i+j-2}(A-B)\right) = \\
\sum_{i=1}^{q}\sum_{j=1}^{q} \text{trace}\left(A^{2q-i-j}(A-B)B^{i+j-2}(A-B)\right).
\end{aligned} \tag{A.24}$$

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

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

Similarly, we can write

$$\begin{aligned}
\text{trace}(XYZY) = \sum_{i=1}^{p}(XY)_{i\cdot}ZY_{\cdot i} \geq \lambda_{\min}(Z)\sum_{i=1}^{p}(XY)_{i\cdot}Y_{\cdot i} = \lambda_{\min}(Z)\text{trace}(XYY) = \\
\lambda_{\min}(Z)\sum_{i=1}^{p} Y_{i\cdot}X(Y_{\cdot i}) \geq \lambda_{\min}(Z)\lambda_{\min}(X)\sum_{i=1}^{p} Y_{i\cdot}(Y_{\cdot i}) = \\
\lambda_{\min}(Z)\lambda_{\min}(X)\text{trace}(YY^T) = \lambda_{\min}(Z)\lambda_{\min}(X)||Y||_2^2.
\end{aligned} \tag{A.26}$$

We summarize the expressions (A.25) and (A.26) as the following:

$$\lambda_{\min}(Z)\lambda_{\min}(X)||Y||_2^2 \leq \text{trace}(XYZY) \leq \lambda_{\max}(Z)\lambda_{\max}(X)||Y||_2^2. \tag{A.27}$$

We can apply the inequalities in (A.27) on the trace of the equality (A.24). Thus, we can write the following two inequalities:

$$\sum_{i=1}^{q}\sum_{j=1}^{q}\text{trace}\left(A^{2q-i-j}(A-B)B^{i+j-2}(A-B)\right)\geq$$
$$||A-B||_2^2\sum_{i=1}^{q}\sum_{j=1}^{q}\lambda_{\min}(A^{2q-i-j})\lambda_{\min}(B^{i+j-2}), \quad \text{(A.28)}$$

$$\sum_{i=1}^{q}\sum_{j=1}^{q}\text{trace}\left(A^{2q-i-j}(A-B)B^{i+j-2}(A-B)\right)\leq$$
$$||A-B||_2^2\sum_{i=1}^{q}\sum_{j=1}^{q}\lambda_{\max}(A^{2q-i-j})\lambda_{\max}(B^{i+j-2}). \quad \text{(A.29)}$$

From the inequalities (A.28), (A.29) and the equality (A.24) it follows that

$$||A^q-B^q||_2\geq||A-B||_2\left(\sum_{i=1}^{q}\sum_{j=1}^{q}\lambda_{\min}(A^{2q-i-j})\lambda_{\min}(B^{i+j-2})\right)^{\frac{1}{2}}, \quad \text{(A.30)}$$

$$||A^q-B^q||_2\leq||A-B||_2\left(\sum_{i=1}^{q}\sum_{j=1}^{q}\lambda_{\max}(A^{2q-i-j})\lambda_{\max}(B^{i+j-2})\right)^{\frac{1}{2}}. \quad \text{(A.31)}$$

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

$$\left(\sum_{i=1}^{q}\sum_{j=1}^{q}\lambda_{\min}(A^{2q-i-j})\lambda_{\min}(B^{i+j-2})\right)^{\frac{1}{2}}=O_P(1), \quad \text{(A.32)}$$

$$\left(\sum_{i=1}^{q}\sum_{j=1}^{q}\lambda_{\max}(A^{2k-i-j})\lambda_{\max}(B^{i+j-2})\right)^{\frac{1}{2}}=O_P(1). \quad \text{(A.33)}$$

From the inequalities (A.30), (A.31), (A.32) and (A.33) it follows that

$$||A^q-B^q||_2\overset{P}{\asymp}||A-B||_2, \quad \text{(A.34)}$$

which concludes the proof of Lemma 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 $(A.2)$ 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 1 implies that

$$||\widehat{\Omega}_{\text{r-glasso}} - \Omega||_2 \overset{P}{\asymp} ||\widehat{\Gamma} - \Omega^{\frac{1}{k}}||_2, \tag{A.35}$$

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

We can prove the Theorem 1 under assumption that k is a rational number. We express k as a fraction $\frac{r}{m}$, where $r, m \in \mathbf{N}$. 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 (1), which implies that

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

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 (1) implies that

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

Summarizing (A.36) and (A.37), we will have the following:

$$||\hat{\Gamma} - \Omega^{\frac{m}{r}}||_2 \overset{P}{\asymp} ||\hat{\Gamma}^{\frac{r}{m}} - \Omega||_2, \tag{A.38}$$

Finally, (A.2) and (A.38) establish the rate (A.1) for rational $k = \frac{r}{m}$.

Appendix B. Performance Measures for Simulation Study

Table B.6. Average KL losses (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.7. 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.8. 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 [14]			
p	glasso	r-glasso $k = k_{BIC}$	r-glasso $k = 2$	clime
100	N/A	N/A	N/A	N/A
200	N/A	N/A	N/A	N/A
300	N/A	N/A	N/A	N/A
	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)

[14] Specificity and MCC (see Table B.10) are not considered for model 6, because these measurements are not defined for dense models.

Table B.9. 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)
hline	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.10. 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	N/A	N/A	N/A	N/A
200	N/A	N/A	N/A	N/A
300	N/A	N/A	N/A	N/A
	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)

Acknowledgements

Andrés M. Alonso gratefully acknowledges financial support from the Spanish Ministry of Science and Innovation grants ECO2011-25706 and ECO2012-38442.

Francisco J. Nogales and Vahe Avagyan are supported by the Spanish Government through project MTM2013-44902-P.

References

Bai, J. (2003). Inferential theory for factor models of large dimensions. *Econometrica*, 71(1):135–172.

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.

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

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.

Chamberlain, G. and Rothschild, M. (1983). Arbitrage, factor structure and mean-variance analysis in large asset markets. *Econometrica*, 51:1305–1324.

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.

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

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*.

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)*.

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.

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

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.

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.

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.

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.

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*.

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.

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.