Precision Matrix Estimation with ROPE

Kuismin M. O.
Department of Mathematical Sciences, University of Oulu, Finland
and
Kemppainen J. T.
Department of Mathematical Sciences, University of Oulu, Finland
and
Sillanpää M. J.
Department of Mathematical Sciences, University of Oulu, Finland
Biocenter Oulu

November 15, 2016

Abstract

It is known that the accuracy of the maximum likelihood based covariance and precision matrix estimates can be improved by penalized log-likelihood estimation. In this article we propose a Ridge-type Operator for the Precision matrix Estimation, ROPE for short, to maximize a penalized likelihood function where the Frobenius norm is used as the penalty function. We show that there is an explicit closed form representation of a shrinkage estimator for the precision matrix when using a penalized log-likelihood, which is analogous to ridge regression in a regression context. The performance of the proposed method is illustrated by a simulation study and real data applications. Computer codes used in the example analyses as well as other supplementary materials for this article are available online.

Keywords: Frobenius norm, Penalized likelihood, Riccati equation, Ridge estimate, Shrinkage

1 Introduction

In recent years, there have been numerous studies which consider the estimation of a covariance matrix and its inverse, which is called the precision matrix (see e.g., Ledoit and Wolf (2004a,b); Friedman et al. (2008); Witten et al. (2011); Bien and Tibshirani (2011); Cai et al. (2011); Deng and Tsui (2013); Yuan and Wang (2013); Kuismin and Sillanpää (2016)). The drive behind this development has been the criticism that commonly-used likelihood-based methods produce inaccurate estimates for the covariance and precision matrices, even when there are more data points than variables (see e.g., Ledoit and Wolf (2004a)). The penalized log-likelihood functions and other constrained optimization techniques are used to gain better estimates for the matrices. Examples of these include the graphical Lasso algorithm and its extensions (Friedman et al. (2008); Fan et al. (2009); Witten et al. (2011); Bien and Tibshirani (2011)) and other regularization driven approaches for the log-likelihood function (see e.g., Won et al. (2009); Yuan and Wang (2013); Deng and Tsui (2013)). Examples of optimization-based approaches which do not deal with the likelihood-based inference are presented in Ledoit and Wolf (2004a,b) and Cai et al. (2011).

We divide the aforementioned approaches and other covariance and precision matrix estimation methods into sparse and non-sparse categories. Ignoring the exact theoretical justification, we intuitively parallel this division to regularized regression models; these models are polarized into two schools according to two of the most influential methods: Lasso and ridge regression.

Lasso (Tibshirani, 1996) permits a shrinkage-inducing model for regression coefficients, which shrinks unimportant coefficients toward zero. This has proved to be efficient when the true model is sparse. The precision matrix is closely related to the structure of the Gaussian graphical models. Thus, most precision matrix estimators are motivated on these grounds and, similar to Lasso, they can be called sparsity-inducing methods (the sparse category). Methods which exploit this standpoint are presented in Meinshausen and Bühlmann (2006); Friedman et al. (2008); Fan et al. (2009); Bien and Tibshirani (2011); Cai et al. (2011); Witten et al. (2011); Yuan and Wang (2013); Bühlmann et al. (2014) and Liu and Luo (2015).

Ridge regression is also a shrinkage-inducing method which shrinks all the regression coefficients of the model. However, unlike Lasso, it cannot set any of the coefficients exactly to zero. Ridge regression might be useful when the true model has many small non-zero elements or when there is substantial collinearity between the explanatory variables. Similarly, in the context of covariance and precision matrix estimation, there is an interest in examining non-sparse shrinkage methods which are more suitable to some applications, such as portfolio optimization, principal component analysis, linear discriminant analysis and genetic applications. Methods which are more convenient for these applications are presented in Ledoit and Wolf (2004a,b); Huang et al. (2006); Warton (2008); Won et al. (2009) and Deng and Tsui (2013) along with our approach presented in this article.

Huang et al. (2006) were among the first to parallel the penalized log-likelihood estimation with l_2 (and l_1) regularization with the common ridge (Lasso) regression. They shrunk the elements of the covariance matrix by examining the modified Cholesky decomposition. Using this decomposition, they transformed the estimation of a covariance matrix to a penalized regression problem. This approach uses an iterative procedure to obtain the estimated entries in the Cholesky decomposition matrices to gain covariance and precision matrix estimates in the original scale.

In the context of covariance matrix estimation, shrinkage/ridge-type estimators are linear combinations of a sample covariance matrix and a scaled identity matrix. This is equivalent to ridge regression in the sense that the estimate is not sparse but is able to correct the possible singularity of the sample covariance matrix and thus make it invertible (see e.g., Warton, 2008). This is more or less an ad hoc solution where the aim is to examine the precision matrix, since the inversion is always done with computer software causing some numerical error in the final estimate. There are just few methods available for direct precision matrix estimation which are computationally efficient when dealing with several hundreds or even thousands of variables. The starting point of our study is to find a simple and fast estimator which provides a symmetric and positive definite estimate directly for the precision matrix even in the high dimensional setting from the standpoint of common ridge regression. We propose a penalized maximum likelihood approach to obtain a shrinkage estimate of the precision matrix, thus avoiding the inversion of the estimate. This leads to a non-sparse, always positive definite and rotation-equivariant estimate for the precision matrix with shrunken elements, where the estimator is almost as lightweight to compute as other ridge-type estimators. We see our method as the "true" counterpart of ridge regression in the context of precision matrix estimation. This is due to subtle differences in the penalty functions between our approach and the common ridge-type estimator.

The valuable properties of our approach are:

- In the high-dimensional setting when $p > n$ one can determine a symmetric and positive definite estimate directly for the precision matrix which cannot be determined by inverting the singular sample covariance matrix.

- Ridge-type estimators are usually less complex to compute than estimators obtained by methods using the l_1-type penalty function. Moreover, ridge-type estimators do not force a rigid diagonal structure for the final estimate. There would be no need to use more complex or sparse methods in the precision or covariance matrix estimation if sparsity is not an essential property, like in risk minimization problems, principal component analysis, mixed model analysis and so on.

- Using "proper" ridge-penalty (the squared Frobenius norm) and deriving the estimate with our method one can gain even better estimates than with the commonly used ridge-type penalties.

- Our empirical observations also indicate that our method could be relatively robust for the choice of the tuning parameter.

The structure of the remaining article is as follows. In Section 2 we illustrate some properties of common ridge-type estimators. We show that one can calculate explicit closed form solution to the penalized maximum likelihood precision matrix estimate using the l_2-norm as a penalty function, which is consistent with using the Frobenius norm for matrices as the penalization function. In Section 3 we demonstrate the performance of our method with a basic simulation study using different precision matrix structures. In Section 4 we apply our method to the analysis of real data. We use different covariance and precision matrix estimates in linear discriminant analysis with an ionosphere data set and demonstrate how they affect the misclassification error. We also study whether the covariance and the precision matrix estimates can be efficiently used in place of the genomic relationship matrix as a part of linear mixed model formulation in plant breeding. We conclude the article with discussion in Section 5.

Deng and Tsui (2013) examined a similar penalized log-likelihood to us. They used the properties of the matrix-logarithm transform of the covariance matrix and expressed their penalized log-likelihood as the function of this transform. This means that their penalty-function can regularize both the small and large eigenvalues of the covariance matrix. This is a desirable penalization following the results of Ledoit and Wolf (2004b) but the ensuing penalized log-likelihood has to be minimized using the iterative quadratic programming algorithm, which could possibly be slow. We derive a closed form approach with an untransformed penalty function. Then we show that common matrix calculus can be used to derive a positive definite estimate of the precision matrix. The theories and methods described in this article are well-studied but we do not believe they have yet been applied to likelihood-based inference.

Finally, we note that our work was developed independently and concurrently by a recent paper of Wieringen and Peeters (2016). The main difference is that we use substantially different calculus to derive a positive definite estimate for the precision matrix. We also show that there is a connection between the penalized log-likelihood estimation and the Riccati equation. We have utilized their notion of a target matrix also here since it makes our method more flexible and interpretative. In contrast to Wieringen and Peeters (2016), we express the estimator in less complex form and provide more comprehensive numerical comparison between different precision and covariance matrix estimators using simulated and real data.

2 Penalized precision matrix estimation

2.1 Ridge-type Estimators

In ridge regression, one finds parameter estimates that minimize the sum of squared residuals with a regularization constraint of the parameters. This can be done by minimizing

$$Q(\boldsymbol{\beta}) = (\mathbf{Y} - X\boldsymbol{\beta})^T(\mathbf{Y} - X\boldsymbol{\beta}) + \rho \sum_{j=1}^{p} |\beta_j|^2, \tag{1}$$

where $\mathbf{Y} \in \mathbb{R}^n$ is the vector of response, $X \in \mathbb{R}^{n\times p}$ is a data matrix, $\boldsymbol{\beta} \in \mathbb{R}^p$ is the vector of parameter coefficients, the superscript T denotes the matrix transposition and $\rho > 0$ is a tuning parameter. This leads to so-called ridge solutions

$$\widehat{\boldsymbol{\beta}}_{ridge} = (X^T X + \rho I)^{-1} X^T \mathbf{Y}, \tag{2}$$

where I is the $p \times p$ identity matrix and $(\cdot)^{-1}$ denotes the inverse of a matrix. The aim of the penalization is to make $X^T X$ invertible by adding a small constant ρ to the diagonal entries of $X^T X$.

Ledoit and Wolf (2004a,b) proposed a linear combination of the form

$$\widehat{\Sigma} = \alpha_1 I + \alpha_2 S, \tag{3}$$

where $\widehat{\Sigma}$ is an estimator of the covariance Σ and S is the sample covariance matrix. With α_1 and α_2 properly chosen, this always leads to a positive definite estimate for Σ and Σ^{-1} even when S is singular.

All estimators which resemble (3) can be considered shrinkage/ridge-type estimators and they leave the eigenvectors of S intact, only shrinking the eigenvalues of S. Warton (2008) set α_2 to 1 and showed that the ridge estimator $\widehat{\Sigma}_\rho = S + \rho I$ is the maximum penalized normal likelihood estimator when log-likelihood is maximized in terms of Σ with a tuning term proportional to $-tr(\Sigma^{-1}) = \sum_{i=1}^{p} \lambda_i$, where $tr(\cdot)$ denotes the matrix trace and λ_i is the ith eigenvalue of the precision matrix. Compared to equation (1), this type of penalization does not follow the original (squared) ridge-penalization. Instead, ridge-type covariance matrix estimation relies on an expression resembling the form $X^T X + \rho I$ in the formula (2). Even the graphical Lasso (hereafter Glasso) algorithm (Friedman et al., 2008) uses the matrix $S + \rho I$ as an initial covariance matrix estimate to derive a positive definite estimate. For more information about shrinkage/ridge-type estimators, see Pourahmadi (2013) pp. 13–15 and 99–105.

2.2 The Proposed Method

We are interested in estimating the precision matrix Θ which is defined as the inverse of the covariance matrix Σ, $\Theta = \Sigma^{-1}$. Both matrices $\Theta, \Sigma \in \mathbb{R}^{p \times p}$ are symmetric and positive definite. Consider a data matrix Y comprising of n sample realizations from a p-dimensional, independent and identically distributed random vectors $\mathbf{Y}_i$ that follow a multivariate Gaussian distribution with the mean vector $\boldsymbol{\mu} \in \mathbb{R}^p$ and the covariance matrix Σ, that is

$$\mathbf{Y}_i \sim N(\boldsymbol{\mu}, \Theta^{-1}), \quad i = 1, \ldots, n, \quad \mathbf{Y}_i \in \mathbb{R}^p.$$

Without loss of generality, assume that $\boldsymbol{\mu} = \mathbf{0}$. The log-likelihood can be expressed as a function of the $n \times p$ data matrix $Y = (\mathbf{Y}_1, \ldots, \mathbf{Y}_n)^T$,

$$\log p(Y|\Theta) = \log \Big[\prod_{i=1}^{n} p(\mathbf{Y}_i|\Theta) \Big] \propto \log |\Theta| - tr(S\Theta), \tag{4}$$

where $p(\mathbf{Y}_i|\Theta)$ denotes the multivariate Gaussian probability density function of a single data vector, $|\Theta|$ is the determinant of the matrix Θ and $S = Y^T Y/n$, which is also the maximum likelihood (ML) estimate of the covariance matrix. Glasso penalizes the log-likelihood (4) with an l_1-norm to induce sparsity of Θ and maximizes the penalized log-likelihood

$$\log |\Theta| - tr(S\Theta) - \rho ||\Theta||_1, \quad ||\Theta||_1 = \sum\nolimits_{k,k'}^{p} |\theta_{k,k'}|, \quad \rho > 0, \tag{5}$$

over all non-singular matrices Θ.

Motivated by expression (1), we penalize (4) with the squared Frobenius norm, which is a convex penalty function consistent with the (squared) ridge penalty, and which does not substantially complicate the maximization of the expression

$$\log |\Theta| - tr(S\Theta) - \rho ||\Theta||_F^2, \quad ||\Theta||_F^2 = \sum\nolimits_{k,k'}^{p} |\theta_{k,k'}|^2 = tr(\Theta^2) = \sum_{i=1}^{p} \lambda_i^2, \tag{6}$$

where $\Theta^2 = \Theta\Theta$.

By the Karush-Kuhn-Tucker conditions, Θ maximizes the penalized log-likelihood (6) if the sub-gradient of (6) is a zero matrix (see e.g., Boyd and Vandenberghe, 2004). The sub-gradient equation for maximization of (6) is

$$\mathit{\Theta}^{-1} - S - 2\rho\mathit{\Theta} = \mathit{0}, \tag{7}$$

where $\mathit{0} = [0_{k,k'}]$ is a zero matrix.

Multiplying the resulting equation by $\mathit{\Theta}$ from the right, a solution to the maximization problem can be found by solving the equation

$$D(\mathit{\Theta}) = 2\rho\mathit{\Theta}^2 + S\mathit{\Theta} - I = \mathit{0}, \tag{8}$$

There is no unambiguous method to solve this quadratic matrix equation (see e.g., Highman and Kim, 2000 and Larin, 2014) and the solution might not be symmetric. However, we will use the known properties of the Riccati equations (see e.g., Dym, 2007 pp. 390–398) to prove that the symmetric solution of (8) is unique if it exists.

The equation (8) is closely related to the Riccati equation. Here we will use a special case of the Riccati equation, which is of the form (Laub, 1979)

$$A^T X + XA - XRX + Q = \mathit{0}, \tag{9}$$

where $A, R, Q \in \mathbb{R}^{p\times p}$, $R = R^T \geq 0$ and $Q = Q^T \geq 0$, where $M \geq 0$ means that the matrix M is positive-semidefinite.

Let us assume that (8) has a symmetric solution. Adding (8) together with its transpose, we derive the equation

$$-D(\mathit{\Theta}) - D(\mathit{\Theta})^T = -S\mathit{\Theta} - \mathit{\Theta}S - \mathit{\Theta}4\rho I\mathit{\Theta} + 2I = \mathit{0}, \tag{10}$$

which is a special case of the Riccati equation (9) and where the left hand side of (10) is clearly symmetric. By finding a method to solve the equation (10), we find the unique symmetric solution of (8), provided it exists.

The Riccati equation (10) is connected with the invariant subspaces of the Hamiltonian matrix H defined as

$$H = \begin{pmatrix} -S & -4\rho I \\ -2I & S \end{pmatrix}. \tag{11}$$

The following property of the eigenvalues of the Hamiltonian matrix (11) plays a central role in the following (Dym, 2007, pp. 390).

Lemma 1. *The eigenvalues of H are symmetrically distributed with respect to the imaginary axis $i\mathbb{R}$ and $\sigma(H) \cap i\mathbb{R} = \emptyset$, where $\sigma(H)$ denotes the spectrum of H.*

Lemma 1 guarantees that H admits a Jordan decomposition of the form

$$H = U \begin{pmatrix} J_1 & \mathit{0} \\ \mathit{0} & J_2 \end{pmatrix} U^{-1}, \tag{12}$$

where $J_1, J_2 \in \mathbb{R}^{p\times p}$, $\sigma(J_1) \subset \Pi_-$, $\sigma(J_2) \subset \Pi_+$, where Π_- (Π_+) denotes the left (right) half-plane $\{z \in \mathbb{C} : Re(z) < 0\}$ ($\{z \in \mathbb{C} : Re(z) > 0\}$). U is a square matrix and we divide it to equal sized blocks as follows

$$U = \begin{pmatrix} X_1 & X_3 \\ X_2 & X_4 \end{pmatrix}. \tag{13}$$

We can use the following results shown by Wonham (1968) and Laub (1979) to determine the unique positive-semidefinite solution to the Riccati equation (10).

Theorem 1. *The equation (10) has a unique positive-semidefinite and symmetric solution. Moreover, the $p \times p$ block X_1 defined in (13) is invertible and the solution of (10) is given by the matrix $X = X_2X_1^{-1}$.*

Proof. See (Laub, 1979, Theorem 5). See also Wonham (1968). □

Thus, we can derive the unique positive-semidefinite and symmetric solution to the matrix equation (8), assuming it exists. Hence, we have a solution which maximizes (6), where the solution is

$$\widehat{\Theta} = X_2 X_1^{-1}. \tag{14}$$

It is clear that $\widehat{\Theta}$ has to be a positive definite matrix but this is hard to verify from the form (14). It can be shown that (14) is not only a positive-semidefinite solution but a positive definite solution by using another method similar to Larin (2014):

Property 1. *The eigenvalues $\widehat{\lambda}_i$ of the solution (14) are*

$$\widehat{\lambda}_i = \frac{2}{l_i + \sqrt{l_i^2 + 8\rho}}, \quad i = 1, \dots, p,$$

where l_is are the eigenvalues of the sample covariance matrix S and ρ is a tuning parameter.

It is not difficult to see that the solution (14) also has the following property.

Property 2. *The eigenvectors of the solution (14) are the same as the eigenvectors of the sample covariance matrix S.*

Proof. From the sub-gradient (7) equation for maximization of (6) we derive that $\widehat{\Theta}$ commutes with the sample covariance matrix S. Then, we apply the spectral theorem for commuting Hermitian matrices (see e.g., Theorem 9.6 in Dym, 2007, pp. 188), which completes the proof. □

The technical proofs can be found in the Supplementary materials.

Using these properties we can derive the maximizing solution of (6) in the following form:

$$\widehat{\Theta} = M \mathit{\Lambda} M^T, \tag{15}$$

where $\mathit{\Lambda}$ is a $p \times p$ diagonal matrix whose diagonal elements are those presented in Property 1, sorted in ascending order beginning from the upper left corner of $\mathit{\Lambda}$, i.e. $\mathit{\Lambda} = diag(\widehat{\lambda}_1, \dots, \widehat{\lambda}_p)$, $\widehat{\lambda}_1 \geq \dots, \geq \widehat{\lambda}_p$. M is an orthogonal $p \times p$ matrix whose columns correspond to the eigenvectors of S, which are ordered in the same order in which the sample covariance matrix eigenvalues appear in the matrix $\mathit{\Lambda}$. We call this solution a *Ridge-type Operator for the Precision matrix Estimation*, shortly *ROPE.* We note that ROPE clearly shares the complexity of principal component analysis (PCA) as in both PCA and ROPE one has to determine the eigenvalue decomposition of the sample covariance matrix S.

Wieringen and Peeters (2016) suggested a penalty $\rho||\Theta - T||_F^2$, where T is a $p \times p$ symmetric and positive definite *target matrix* in the expression (6). Clearly the penalty $\rho||\Theta||_F^2$ is a special case of this penalty when the target matrix T is a $p \times p$ zero matrix. Our equation (15) can still be used to find the maximizing solution of

$$\log|\Theta| - tr(S\Theta) - \rho||\Theta - T||_F^2 \tag{16}$$

by replacing the sample covariance matrix S with $S^* = S - 2\rho T$ in the above mentioned equations and using the eigenvalues and eigenvectors of the matrix S^* in the equation (15). From Property 1 it can be seen by using an elementary algebra that when the target matrix T is a diagonal matrix or a zero matrix the eigenvalues of the solution (15) actually go towards the eigenvalues of the target matrix T when the tuning parameter ρ approaches infinity. In addition, because in this case the solution (15) is also a rotation equivariant estimator (Property 2), the solution (15) approaches the target matrix T as ρ approaches infinity. In this paper we consider only the cases of T being either a zero matrix or a diagonal matrix. Wieringen and Peeters (2016) have shown that the solution (15) in fact approaches any symmetric and positive definite target matrix T when ρ approaches infinity (see their Proposition 1). This target matrix has valuable properties in risk minimization which we will demonstrate in the next section.

From Property 1 it can be seen that ROPE induces a nonlinear shrinkage in the sample covariance matrix eigenvalues. The shrinkage is nonlinear in the sense that the relationship between the eigenvalues $1/\widehat{\lambda}_i$ and the eigenvalues l_i is not affine. Here, it differs from other ridge-type estimators of the form (3), since they shrink the sample covariance matrix eigenvalues linearly, where the eigenvalues are of the form $\alpha_1 + \alpha_2 l_i$ for the covariance matrix estimate $\widehat{\Sigma}$. When the sample covariance matrix is singular ($l_i = 0$), corresponding eigenvalue of ROPE is $\sqrt{2\rho}$ for the estimated covariance matrix (α_1 for the common ridge-type estimator) and $1/\sqrt{2\rho}$ ($1/\alpha_1$, respectively) for the estimated precision matrix.

Choosing $\alpha_2 = 1$ as in Warton (2008) and using the inequality $a^2 + b^2 \leq (a+b)^2$ we get the inequality

$$\widehat{\lambda}_i \geq \frac{1}{l_i + \sqrt{2\rho}},$$

which shows that the shrinkage of the eigenvalues of S^{-1} by using the estimator (3) with $\alpha_2 = 1$ and $\alpha_1 = \sqrt{2\rho}$ is at least as heavy as the shrinkage given by ROPE (see also Proposition 4 in Wieringen and Peeters, 2016). At least for large values of ρ this is a desirable property, since in the limit $\rho \to \infty$ all the eigenvalues of the estimators are zero, which naturally is not desirable.

In general, it is not a simple task to compare the shrinkage of the eigenvalues since there may be neither simple formulae nor simple estimates for the eigenvalues of the estimator (see e.g. Formula (4.3) in Ledoit and Wolf, 2012).

In Figure 1, we demonstrate how the different methods shrink the eigenvalues of the sample covariance matrix (hereafter sample eigenvalues) compared to the real eigenvalues (hereafter population eigenvalues). For a more convenient interpretation, we illustrate the methods with the sample eigenvalues and not with the eigenvalues of the precision matrix estimate.

As studied in (Ledoit and Wolf, 2004b) and (Won et al., 2009), the eigenstructure of the sample eigenvalues tend to be biased in that the small population eigenvalues are underestimated and the large population eigenvalues are overestimated. The ridge estimator which has the form of a convex combination $\rho v I + (1 - \rho)S$ (Ledoit and Wolf, 2004b) tries to overcome this problem by setting a "fulcrum" at the point v which is the mean of the sample eigenvalues determined by $v = tr(S)/p$. This can be seen in the left-hand plot of Figure 1 since all the sample eigenvalues under v are increased and the values greater than v are decreased, whereas the ridge estimator of the form $\rho I + S$ with the same level of regularization can even increase the bias between the large population eigenvalues and the shrunken eigenvalues. In the right-hand plot of Figure 1 we have illustrated a special case where most of the population eigenvalues are underestimated by the sample covariance matrix but the large population eigenvalues are fairly close to the sample eigenvalues. With a small regularization, the ridge estimator is quite close to the population eigenvalues but it still underestimates the small ones and overestimates the large ones. With the same regularization, ROPE is able to reduce this bias by actually setting a fulcrum at the point

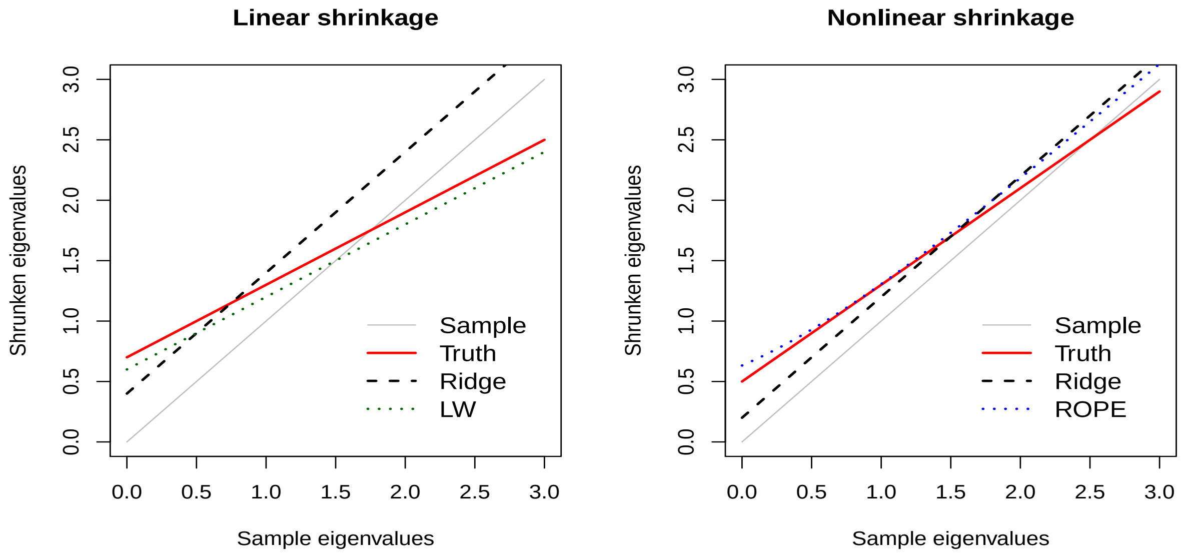

Figure 1: A rough illustration of how different ridge estimators shrink the eigenvalues compared to the hypothetical population eigenvalues as a function of the sample eigenvalues. The gray solid lines correspond to the eigenvalues of S (Sample). The red solid lines illustrate the assumed population eigenvalues (Truth). The black dashed lines correspond to the ridge-type estimator 1 (Ridge). The green dotted line illustrates how the ridge-type estimator 2 (LW) linearly shrinks the real eigenvalues in a case close to the optimal shrinkage. The blue dotted line demonstrates how ROPE would non-linearly shrink the eigenvalues in an ideal shrinkage scenario.

$2 - \rho$ with the ridge estimator $\rho I + S$. This can be seen by setting $1/\widehat{\lambda}_i = l_i$.

However, our main goal is still to estimate the precision matrix elements. Like its regression counterpart, ROPE does not lead to a sparse estimate of Θ. For graphical illustration of the differences between l_2- and l_1-type penalties, see Bühlmann et al. (2014). Figure 2 illustrates the shrinkage of five random off-diagonal elements of the ROPE estimate, Glasso, the ridge-estimate of the form $S + \rho I$, and a linear convex combination resembling the estimator presented in Ledoit and Wolf (2004b) as a function of ρ. The data matrix is drawn from $N(\mathbf{0}, \Theta^{-1})$, where Θ is a sparse precision matrix and $p > n$. ROPE shrinks the off-diagonals much faster than the other two ridge estimates and, furthermore, there is some resemblance between the solution paths of Glasso and ROPE.

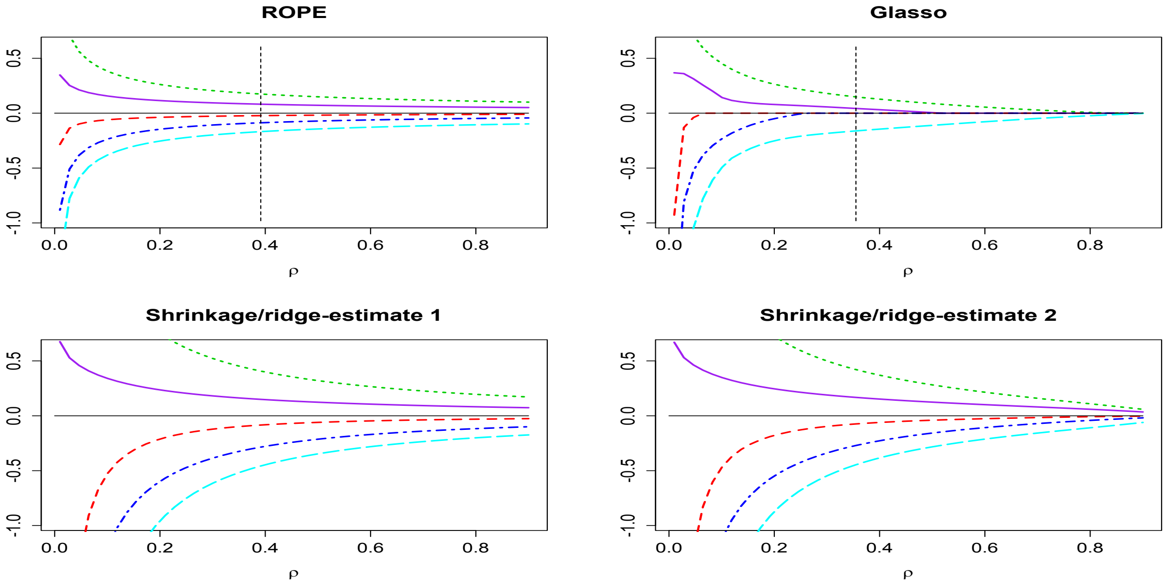

Figure 2: The solution paths of five random off-diagonal elements of $\widehat{\Theta}$ for ROPE, graphical Lasso (Glasso), common ridge-type estimate $\widehat{\Theta} = (\rho I + S)^{-1}$ (Shrinkage/ridge estimate 1), and a linear convex combination $\widehat{\Theta} = (\rho v I + (1-\rho)S)^{-1}$ (Shrinkage/ridge estimate 2) as a function of ρ. The vertical dashed lines represent the value of ρ selected by five-fold cross-validation ($\widehat{\rho} = 0.373$ for ROPE and $\widehat{\rho} = 0.355$ for Glasso) when the log-likelihood (4) is used as a loss-function.

Due to the closed form of the solution (15) to the penalized maximum likelihood problem, this method is easy to implement and computationally fast even with problems which consider a cross-validation setting, because the computational speed does not depend on the value of the tuning parameter or the structure of the sample covariance matrix. The computational speed of Glasso and SCIO (Liu and Luo, 2015) depends on the value of the tuning parameter and on the structure of the sample covariance matrix; when ρ is smaller than the absolute value of the off-diagonal elements in the sample covariance matrix, Glasso and SCIO slow down considerably. For example, when $p = 1000$, $\rho = 0.01$ and the data is a sample of size $n = 5000$ from a multivariate Gaussian distribution with mean vector $\mathbf{0}$ and covariance matrix I, the computation of the precision matrix estimate takes under three seconds with ROPE, about 14 seconds with SCIO and over a minute with Glasso with R version 3.1.3 running on a standard desktop computer (64-bit operating system, 3.1 GHz CPU and 8Gb RAM). When $\rho = 0.9$, Glasso and SCIO speed up and finish in under one second. With a more complex (sample) covariance matrix structure and a smaller sample size, Glasso and SCIO slow down substantially and they can take several minutes to converge unless ρ is very large, which usually causes the final estimate to be a simple diagonal matrix. The computational performance of ROPE depends on the method which is used to compute the eigenvalue decomposition of the sample covariance S. A time taken to find the optimal value for the tuning parameter ρ makes ROPE somewhat slower to use than the LW-estimator, which derives the optimal regularization from the data. The tuning parameter ρ for ROPE can be chosen either by cross-validation with a suitable loss-function (e.g., Bien and Tibshirani, 2011), an information criterion (e.g., Bayesian information criterion, like in Yuan and Wang, 2013), or an independent validation method (Deng and Tsui, 2013).

3 Simulation study

There are numerous estimators for the precision matrix and the covariance matrix. We have chosen four other methods for comparison with ROPE. Three of the methods are developed for precision matrix estimation like ours and one is developed for the estimation a better-conditioned covariance matrix;

- *Glasso* is discussed in Friedman et al. (2008); Witten et al. (2011). Glasso can be used either for the covariance or for the precision matrix estimation without any need to invert the final estimate to have one or the other.

- *Ledoit and Wolf estimator (LW)* presented in Ledoit and Wolf (2004b) leads to a better-conditioned, non-sparse estimate for the covariance matrix, and we invert the estimate to obtain an estimate for the precision matrix. Note that the matrix inversion is not an issue with the dimensions in our model problems. The inversion is needed to compare the results.

- *Constrained l_1-minimization for Inverse Matrix Estimation (CLIME)* presented in Cai et al. (2011); Pang et al. (2014).

- *Sparse Column-wise Inverse Operator (SCIO)* discussed in Liu and Luo (2015).

We consider cases where the data is a sample from a multivariate Gaussian distribution $N(\mathbf{0}, \Theta^{-1})$, where $\Theta = [\theta_{k,k'}]$ and $\Theta^{-1} = \Sigma = [\sigma_{k,k'}]$ are $p \times p$ positive definite matrices. Six different models are used to compare the methods. We have described below the matrices used in our simulation study. We have also derived the condition numbers of the matrices (three corresponding to each of the dimensions) which can be calculated by dividing the largest eigenvalue with the smallest eigenvalue of the corresponding matrix. We examined three settings with dimension p set to 20, 50 and 100, respectively.

- *Model 1.* A compound symmetry model with $\sigma_{k,k} = 1$ and $\sigma_{k,k'} = 0.6^2$ for $k \neq k^{'}$. This covariance matrix is structured and non-sparse. The condition numbers of this matrix are 12.25, 29.13 and 57.25.

- *Model 2.* The second model comes from Cai et al. (2011) and Liu and Luo (2015). Let the prototype $\Theta_0 = A + aI$, where each off-diagonal entry in A is generated independently and equals 0.5 with probability 0.1 or 0 with probability 0.9. a is chosen such that the condition number of the matrix is equal to p. Finally, the matrix is standardized to have unit diagonal. This precision matrix is unstructured and sparse.

- *Model 3.* $\Theta = \frac{1}{n} Y^T Y$ where $Y = [y_{i,j}]$ is a $n \times p$ matrix with $n = 10000$ and each $y_{i,j}$ is drawn from $N(0, 1)$. This precision matrix is unstructured and non-sparse. The condition numbers of this matrix are 1.16, 1.32 and 1.47.

- *Model 4.* A star model with $\theta_{k,k} = 1$, $\theta_{1,k} = \theta_{k,1} = 0.1$ and $\theta_{k,k'} = 0$ otherwise. This precision matrix is structured and sparse. The condition numbers of this matrix are 2.55, 5.67 and 398.00.

- *Model 5.* A moving average (MA) model with $\sigma_{k,k'} = 1$, $\sigma_{k,k-1} = \sigma_{k-1,k} = 0.2$ and $\sigma_{k,k-2} = \sigma_{k-2,k} = 0.2^2$. This covariance matrix is structured and sparse. The condition numbers of this matrix are 2.16, 2.17 and 2.18.

- *Model 6.* A diagonally dominant model. Let $B = \frac{1}{2}(A + A^T)$, where $A = [a_{k,k'}]$ is a $p \times p$ matrix. Each $a_{k,k'}$ for $k \neq k^{'}$ is drawn from $U(0, 1)$ and $a_{k,k} = 0$. Compute a matrix $D = \frac{1}{\gamma}B$, where $D = [d_{k,k'}]$ and γ is the largest row sum of the absolute values of the elements of the matrix B. Finally, each off-diagonal elements of Σ are chosen as $\sigma_{k,k'} = d_{k,k'}$ and $\sigma_{k,k} = 1 + e_i$, where e_i is drawn from $U(0, 0.1)$. This covariance matrix is unstructured and non-sparse. The condition numbers of this matrix are 2.04, 2.12 and 2.06.

An independent sample of size 50 is generated from a multivariate Gaussian distribution using each of the models. All the methods, except LW, need to pre-specify a restriction/tuning parameter ρ. A five-fold cross-validation described by Bien and Tibshirani (2011) is used to choose the parameter for ROPE and Glasso such that the optimal value of ρ minimizes the log-likelihood (4). For CLIME, we use the five-fold cross-validation found in the R-package "clime" (version 0.41) along with the CLIME estimator; the value of ρ, in the context of minimization of $||\Theta||_1$ subject to $|S\Theta - I|_\infty \leq \rho$, is chosen to be such that it minimizes the likelihood-based loss-function $tr(\Sigma\Theta) - \log|\Theta| - p$ as presented in the R reference manual. For SCIO, we use the cross-validation portrayed by Liu and Luo (2015) and implemented in the R-package "scio" (version 0.6.1). For ROPE and Glasso, we use a candidate set for ρ with 50 elements varying from 0.01 to 10. For ROPE we used two target matrices for T: an identity matrix I and a scalar matrix vI, where $v = p/tr(S)$, shortly ROPE I and ROPE vI. For CLIME, we consider the sequence of tuning parameters to 20 elements since CLIME tended to suffer from slow computations with some of the models; we note that there are faster implementations for CLIME in R such as "flare" and "fastclime" but, in practice, we found the "clime" package to be the most stable one. ROPE is based on our own implementations with R. The R-package "glasso" (version 1.8) is used to solve the Glasso problem (5) and the LW-estimator is implemented with our own R code based on the MATLAB code freely available at the web page of Michael Wolf (`www.econ.uzh.ch/faculty/wolf/publications.html`).

To compare the performance of the methods, four loss-functions are used:

- The Kullback-Leibler loss $KL = tr(\Sigma\widehat{\Theta}) - \log(|\Sigma\widehat{\Theta}|) - p$.
- The L2 loss $L2 = ||\Theta - \widehat{\Theta}||_F$.
- The quadratic loss $QL = tr(\Sigma\widehat{\Theta} - I)^2$.

- The spectral (a.k.a operator) norm loss $SP = ||\Theta - \widehat{\Theta}||_{2,2} = d_1$, where d_1^2 is the largest eigenvalue of the matrix $(\Theta - \widehat{\Theta})^2$.

Averages of these losses are calculated for each of the methods from 100 simulations to gain risk measures. The results are displayed in Figures 3, 4, 5 and 6.

3.1 Results

The risk measures give a comprehensive look at the benefits of ROPE. It is quite clear that ROPE is a very efficient estimator when Θ and Σ have a structure as described in these Models. Particularly this can be seen in Figures 3 and 4. In all Models, ROPE is the top ranking method compared to all other methods. With the exception of few risk estimates, like high averaged quadratic losses in Figure 5, ROPE produces a lower risk than LW, CLIME and SCIO in all Model matrices. Only in Model 2 Glasso is able to give competitive risk measures compared to ROPE.

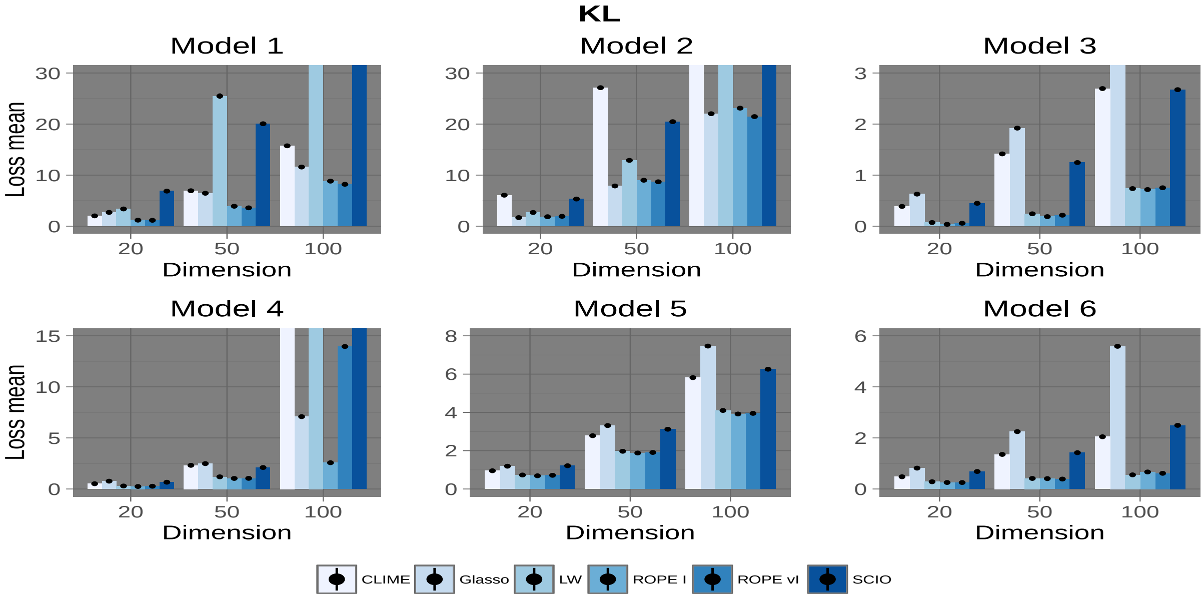

Figure 3: Summary of Kullback-Leibler loss for the different Models and different methods based on 100 replications. The columns along the small black dots indicate loss means. The bars on the top of each column show standard errors (mean ± SE).

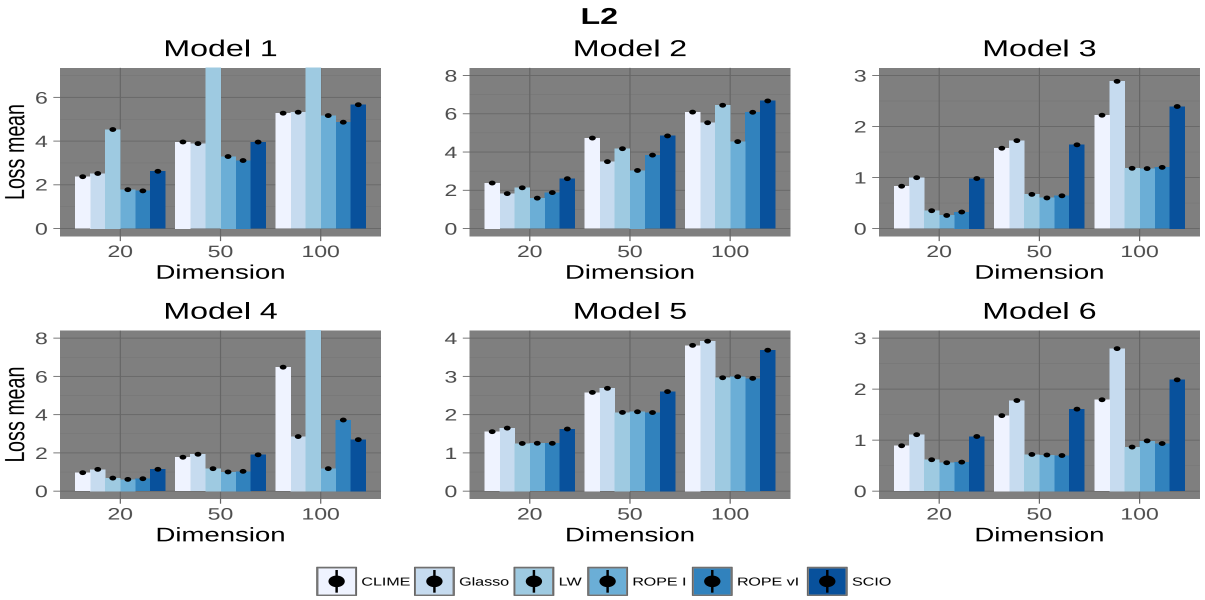

Figure 4: Summary of L2 loss for the different Models and different methods based on 100 replications. The columns along the small black dots indicate loss means. The bars on the top of each column show standard errors (mean ± SE).

In Models 4 and 5 ROPE performs better than Glasso, CLIME and SCIO although these methods should have the advantage in graph structure estimation. In particular this can be seen by comparing averaged spectral norm losses in Figure 6. When the precision matrix is unstructured and non-sparse like the precision matrix in Models 3 and 6, ROPE gives almost unequivocally the lowest risk measures compared to the other methods. Also, the risk associated with the quadratic loss is quite low in Models 3 and 5, indicating that ROPE is able to estimate the underlying eigenvalues associated with this model quite efficiently overall.

SCIO performed quite poorly in this simulation study, but this is in line with the results of Liu and Luo (2015). In their study, SCIO with a cross-validation scheme showed better performance when the dimension p was quite large ($p \geq 800$) and no Model we used was a block diagonal matrix. We also note that when the precision matrix Θ had the structure described in the Models 1, 2, 4, some methods produced remarkably high risk estimates.

We also computed the risk estimates using the penalization (6) (a zero matrix as the

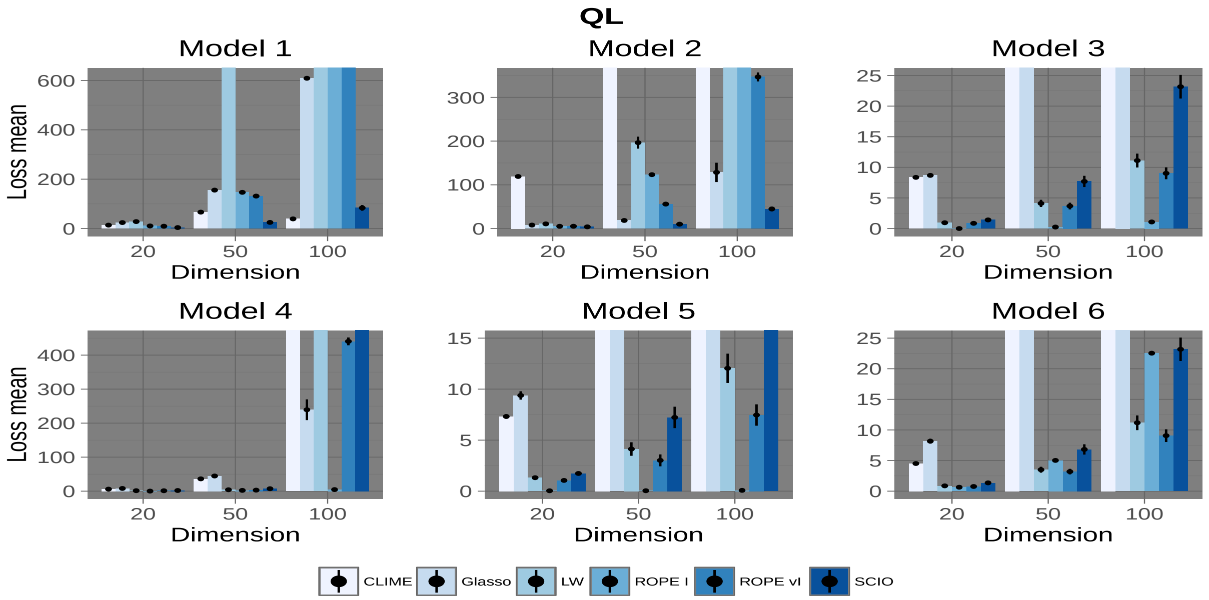

Figure 5: Summary of quadratic loss for the different Models and different methods based on 100 replications. The columns along the small black dots indicate loss means. The bars on the top of each column show standard errors (mean ± SE).

Table 1: Comparison of average (sd) computational times for different methods over 100 replications when $p = 100$ and $n = 50$.

Method	ROPE	LW	Glasso	CLIME	SCIO
Time (seconds)	1.45 (0.05)	0.00 (0.01)	21.72 (5.44)	639.93 (4.36)	0.01 (0.01)

target matrix). The additional results with all averaged risk measures are given in the supplementary materials. Table 1 shows how long it takes on average for each method to compute one of the replicated simulation analyses through in Model 1 with dimension $p = 100$ and sample size $n = 50$ for standardized data.

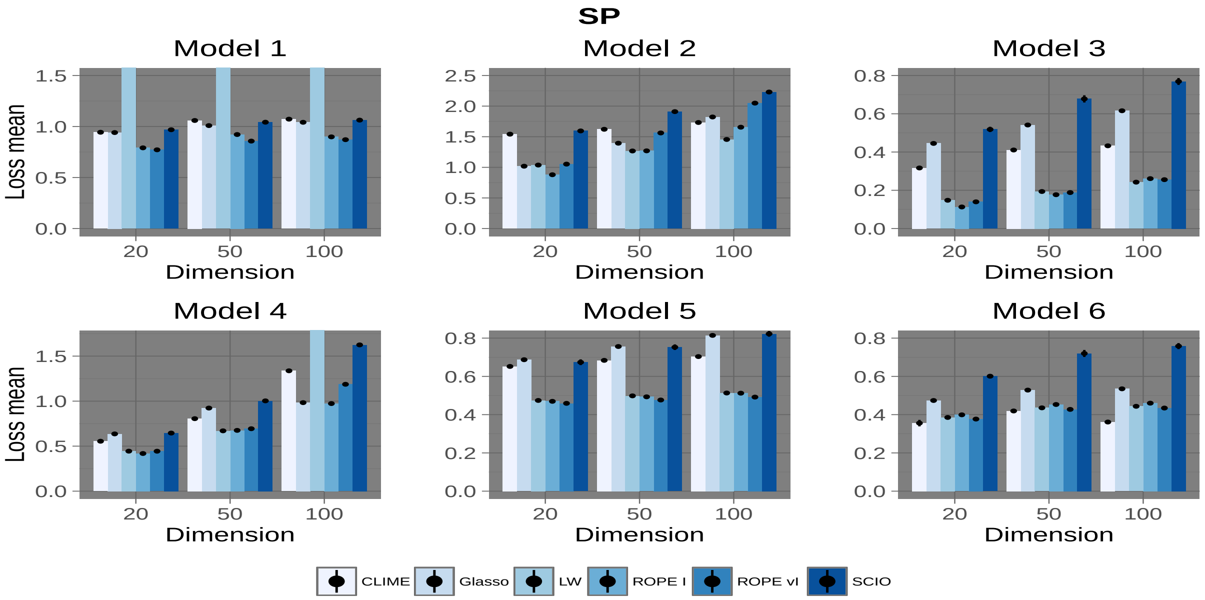

Figure 6: Summary of spectral norm loss for the different Models and different methods based on 100 replications. The columns along the small black dots indicate loss means. The bars on the top of each column show standard errors (mean ± SE).

4 Application to real data

4.1 Linear Discriminant Analysis With Ionosphere Data

In this section, we calculate the misclassification rates of linear discriminant analysis (LDA) by using different covariance and precision matrix estimates and the Mahalanobis distance to allocate samples to two different groups using a so called ionosphere data set. The data set is freely available at `http://archive.ics.uci.edu/ml/datasets/Ionosphere`.

The data was collected by a system in Goose Bay, Labrador. The targets were free electrons in the ionosphere. ”Good” radar returns are those showing evidence of some type of structure in the ionosphere. ”Bad” returns are those that do not show any evidence; their signals pass through the ionosphere. The data contains 351 observations with 35 variables; 34 continuous and one nominal with two values g(“good”) or b(“bad”).

Here we illustrate how different covariance matrix and precision matrix estimates can affect the classification performance of LDA. Let $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ be the population means and

$\Theta = \Sigma^{-1}$ the population precision matrix. It is reasonable to assume that the observations are divided into two groups, G_1 (good radar returns) and G_2 (bad radar returns). Suppose G_1 is the $N(\boldsymbol{\mu}_1, \Sigma)$ distribution and G_2 is the $N(\boldsymbol{\mu}_2, \Sigma)$ distribution. To estimate $\boldsymbol{\mu}_1$, $\boldsymbol{\mu}_2$ and Θ, we split the data into two separate sets: a training set of 40 observations is chosen randomly and a test set contains the rest of the 311 observations. We chose the size of the training set advisedly close to the dimension $p = 32$. The misclassification rate was calculated by classifying observations in the test set using the rule determined from the training set. We use the squared Mahalanobis distance $\mathbf{a}^T(\mathbf{x} - \boldsymbol{\mu})$, where $\mathbf{a} = \Theta(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$ and $\boldsymbol{\mu} = 0.5(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2)$ (see e.g., Mardia et al., 1979, pp. 300-315), to determine which group an observation $\mathbf{x}$ belongs to. We use $\overline{\mathbf{x}}_1$ and $\overline{\mathbf{x}}_2$ to estimate $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_1$, and different estimators including the common sample covariance matrix S to estimate Θ. We use ROPE, LW, Glasso, CLIME and SCIO as the alternative estimates of Θ and allocate $\mathbf{x}$ to group G_1 if $\widehat{\mathbf{a}}^T(\mathbf{x} - \widehat{\boldsymbol{\mu}}) > 0$, where $\widehat{\mathbf{a}} = \widehat{\Theta}(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)$ and $\widehat{\boldsymbol{\mu}} = .5(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)$. With ROPE, we use three different target matrices: the zero matrix (ROPE 0), identity matrix (ROPE I) and scalar matrix vI, where $v = p/tr(S)$ (ROPE vI). The corresponding misclassification rates are the averages based on 100 repetitions of the aforementioned setting. We derive the misclassification rates for ROPE, Glasso, CLIME and SCIO for each value of the tuning parameter ρ from a sequence of length 50 varying from 0.1 to 0.9. The results are presented in Figure 7**A**. For additional analysis we randomly divide the data into three separate sets: a training set of 40 observations, a validation set of also 40 observations and a test set containing the rest of the observations. The validation set is used to choose the tuning parameter through the cross-validation for ROPE, Glasso, CLIME and SCIO. Then we derive the precision matrix estimates from the training set based on this tuning parameter value. Finally we compute the misclassification errors based on the test set. We repeat this procedure 100 times. The boxplots of the 100 misclassification errors produced with this procedure are presented in Figure 7**B**.

For each value of ρ, ROPE gives a lower misclassification rate than Glasso, CLIME, SCIO and the sample covariance matrix. ROPE even shows some robustness to the tuning parameter selection with quite a low misclassification rate. With a small value of ρ, the precision matrix estimate determined with ROPE gives a slightly smaller misclassification

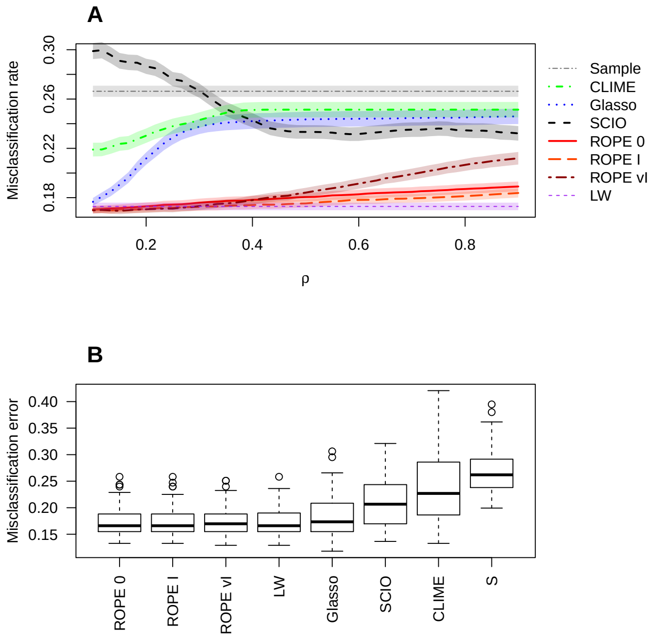

Figure 7: The misclassification rates of LDA and standard errors (shaded areas) averaged over 100 replications as the solution path of the tuning parameter ρ (**A**) and boxplots of test misclassification errors from 100 replications when the tuning parameter is chosen with the cross-validation from a validation set (**B**). All standard errors in **A** are smaller than 0.007.

rate than the LW-estimator; ROPE has rate of 0.170 (zero target matrix), 0.170 (identity matrix) and 0.170 (scalar matrix vI). LW estimator gives a rate of 0.173. It seems that with this data set, ROPE gives the smallest misclassification rate when the tuning parameter is small. We tested this by setting the values of ρ almost at zero, but this only marginally reduced the misclassification rates compared to the aforementioned value (results not shown). When the tuning parameter is chosen with cross-validation from the validation set (Figure 7**B**), ROPE also gives the lowest median calculated for the misclassification error: 0.166 (ROPE 0), 0.166 (ROPE I), 0.170 (ROPE vI). LW estimator gives a rate of 0.166.

4.2 Genomic Prediction with Wheat Data

In this section, we introduce how different precision and covariance matrix estimators could be used to increase the cross-validation prediction accuracy and decrease the prediction error in genomic selection as part of genomic-enabled best linear unbiased prediction (G-BLUP). We compare the results with the commonly used genomic relationship matrix of VanRaden (VanRaden, 2008).

We want to estimate the parameters of a mixed model of the form

$$\mathbf{y} = \mathbf{1}\beta + Z\mathbf{u} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim N(\mathbf{0}, I\sigma_\epsilon^2), \quad \mathbf{u} \sim N(\mathbf{0}, K\sigma_u^2), \tag{17}$$

where $\mathbf{1}$ is a vector of ones, β is a value of fixed effects, $\mathbf{u}$ is a vector of random effects, $\boldsymbol{\epsilon}$ is a vector of random errors and K is a known positive semidefinite matrix. The design matrix $Z = [I\ 0]$ for the genomic values has been partitioned into lines which have their own observation (I) and lines which do not. The variance components σ_ϵ^2 and σ_u^2 can be estimated with a restricted maximum likelihood (REML) algorithm (Patterson and Thompson, 1971). The best linear unbiased prediction (BLUP) estimate of vector $\mathbf{u}$ is defined as a solution to the mixed model equations given by Henderson (1975).

When one wants to predict the genomic breeding values with the model (17), the genomic additive relationship matrix G is used in place of the covariance matrix K, where G is derived from the genotype matrix $X = [x_{i,k}]$, $x_{i,k} \in \{-1, 1\}$ as follows: $G = WW^T/c$, where $W = [w_{i,k}] = [x_{i,k} + 1 - 2p_k]$, p_k is the frequency of allele 1 at the marker k and $c = 2\sum_k p_k(1 - p_k)$. This is known as VanRaden's method 1 (VanRaden, 2008).

For comparison, we derive the marker-based matrix K from the covariance and precision matrix estimates and examine whether they improve the prediction ability and the accuracy of the G-BLUP model (17) over VanRaden's method. We follow the data analyses described in Endelman (2011) and Li and Sillanpää (2012). We use the R package "rrBLUP" (version 4.4) for the parameter estimation (Endelman, 2011), which allows us to use a specific covariance structure for K.

We calculate the five-fold cross-validation error (CVE) described in Li and Sillanpää (2012) based on the mean squared prediction error

$$P_{CV} = \frac{1}{V}\sum_{v=1}^{V} P(\mathbf{y}_v, \tilde{\mathbf{u}}_v),$$

where $V = 5$ for five-fold cross-validation and $P(\mathbf{y}_v, \tilde{\mathbf{u}}_v)$ is the mean squared error (MSE)

$$P(\mathbf{y}_v, \tilde{\mathbf{u}}_v) = \frac{1}{m_v}(\mathbf{y}_v - \tilde{\mathbf{u}}_v)^T(\mathbf{y}_v - \tilde{\mathbf{u}}_v).$$

Here m_v is the size of the partition v, $v = 1, \ldots, 5$, and $\tilde{\mathbf{u}}_v$ is the vector of prediction values. We also calculate the cross-validation accuracies (ACC) based on the means of the correlations $cor(\tilde{\mathbf{u}}_v, \mathbf{y}_v)$, i.e.

$$ACC = \frac{1}{V}\sum_{v=1}^{V} cor(\tilde{\mathbf{u}}_v, \mathbf{y}_v).$$

The data is divided into 5 roughly equal sized partitions and each partition is used in turns as the prediction set while the rest of the partitions are used as the training set. We standardize the marker data following the technique of Patterson et al. (2006) to binary data in the following manner: Recode X such that $x_{i,k} \in \{0, 1\}$. Then derive the standardized data matrix $V = [v_{i,k}]$, $v_{i,k} = (x_{i,k} - p_k)/\sqrt{p_k(1 - p_k)}$, where p_k is determined as above. Finally, derive the covariance matrix estimates from the transpose V^T.

We analyze the set of $n = 599$ wheat lines (Crossa et al., 2010). The data set consists of 599 historical CIMMYT wheat lines and the data can be found from the R package "BLR" (Pérez et al., 2010). The trait was the 2-year average grain yield in four environments. Phenotypes are standardized to unit variance for each environment. We evaluate each environment independently.

We use three of the fastest and most stable estimators along with ROPE: Glasso, SCIO and LW-estimator, out of which the LW is distribution-free estimator (Ledoit and Wolf, 2004b). We compare their results with the common "VanRaden" estimator G. We repeat the cross-validation 50 times and for ROPE, Glasso and SCIO, we use a tuning parameter from a sequence of length 40 varying from 0.1 to 0.9 and calculate accuracies for each of these values. Because we calculate the estimators for each value of ρ, we use the Glasso algorithm implemented in the R-package "huge" (version 1.2.7), which we found to be faster in this data study compared to the R-package used in Sections 3 and 4.1. Again we used three different target matrices T for ROPE: 0 (ROPE 0), I (ROPE I) and a scalar matrix for each partition $b_v I$, where $b_v = n/c_v$, $c_v = tr(V_v^T V_v)/r$ and r is the number of markers (ROPE vI). The results are presented in Figure 8.

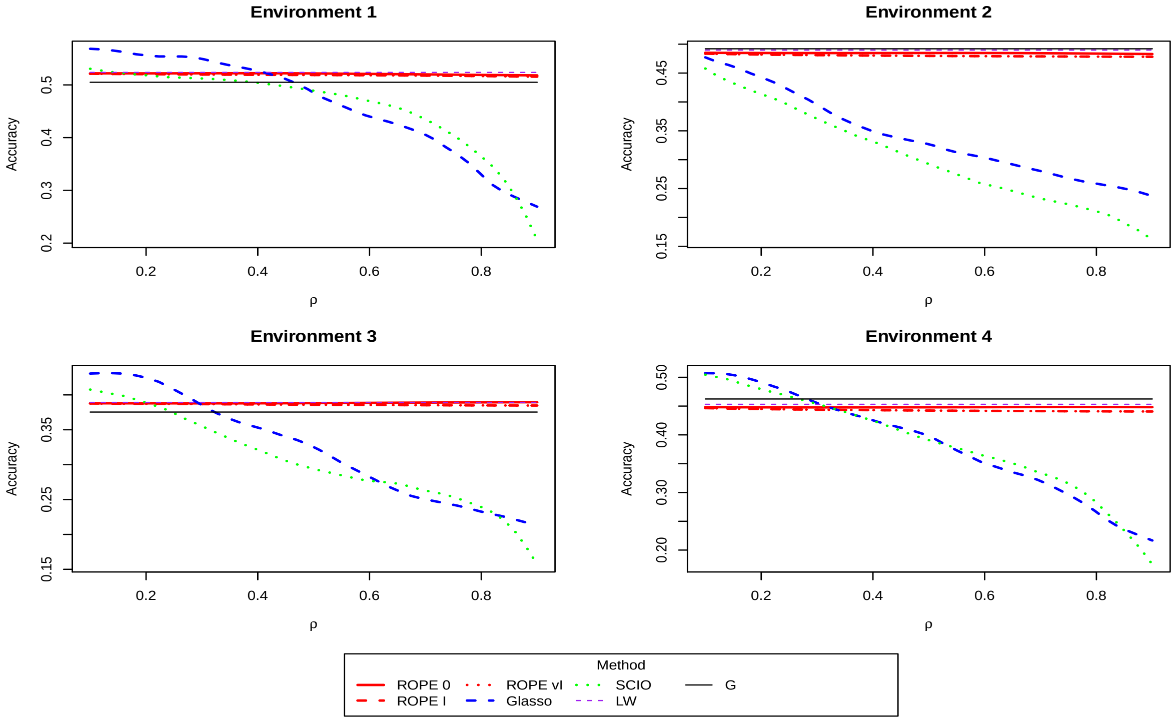

Figure 8: The accuracy for wheat traits for different precision and covariance matrix estimators. All standard errors were smaller than 0.005 and do not have a visible difference in the averaged accuracy diagnostics.

In environments 1 and 3, ROPE is able to improve the accuracy compared to the VanRaden matrix G. Other methods seem to perform even better than ROPE in environments 1,3 and 4 but they tend to produce a very high CV-error (see Figure in the supplementary materials), whereas, compared to the VanRaden matrix, ROPE is able to greatly reduce errors. SCIO gives a higher accuracy and a lower CV-error in environments 3 and 4 when the tuning parameter ρ is chosen to be very small. When the marker-based matrix K is estimated with ROPE, accuracies and CV-errors seem to differ very little regardless of the ρ value, indicating that ROPE is highly robust compared to Glasso and SCIO in this particular case. Still, a closer look shows that the curves are not strictly monotonic.

Increased accuracy and decreased error in environments 1 and 3 indicate that using ROPE for the covariance structure of the random effects $\mathbf{u}$ in model (17) could improve the performance of G-BLUP over that of the VanRaden G matrix with certain data sets regardless the choice of the target matrix and the value of the tuning parameter. In addition to our incremental findings, benefits of shrinkage based methods to improve prediction accuracy (ACC) for small number of markers have been observed by Müller et al. (2015). When the G-BLUP model (17) is used for predictions, that is, fitted using mixed model equations, the inverse of the matrix K has to be determined. This can be problematic since the VanRaden matrix G can be singular and, particularly in high dimensions, we would like to avoid unnecessarily inverting the matrix K. With ROPE, Glasso and SCIO, one can always determine a positive definite estimate directly for the inverse of K, which is a valuable property in practice. However, this possibility is not fully utilized in the current genetics oriented literature.

5 Discussion

We have shown that, similar to graphical Lasso, there is an analogue to the ridge-type penalization in the likelihood-based precision matrix inference. This analogue even shares similar properties with its penalized regression counterpart. We call this method ROPE. ROPE is easy to use and will always lead to a positive definite estimate even when one cannot derive the precision matrix from the maximum likelihood estimate of the covariance matrix. In the real data applications ROPE was found to show quite robust performance despite the choice of the target matrix and the value of the tuning parameter.

In some applications it might be convenient to use a special weight matrix P to emphasize the effect of the tuning parameter ρ on the precision matrix elements in the regularized log-likelihood function. This leads to a maximization problem for the expression $\log|\Theta| - tr(S\Theta) - \rho||P \circ \Theta||_F^2$. The weight matrix would extend the regularized likelihood inference to an adaptive setting similar to the adaptive LASSO penalty and SCAD, as demonstrated by Fan et al. (2009). This optimization problem needs further studying.

Arguably ROPE could offer some worthwhile properties in applications when combined with other methods. In data studies it would be more useful to fuse the favorable properties of different methods to strengthen the overall performance. For example, ROPE could potentially be used as an initial covariance matrix estimate in the Glasso algorithm to gain more accurate covariance or precision matrix estimates. How this would work in practice is an open research question.

ROPE has also a Bayesian interpretation when the prior is placed to the eigenvalues of the precision matrix $\varTheta$ independently for each individual element and the target matrix is a zero matrix. Suppose that the eigenvalues of the precision matrix $\varTheta$ have been assigned the generalized gamma distribution priors $(\beta/\theta^{\kappa})\lambda^{\kappa-1}e^{-(\lambda/\theta)^{\beta}}$ (see e.g., Stacy, 1962) with parameters β, θ and κ set to $\beta = 2$, $\theta = 1/\sqrt{\rho}$ and $\kappa = 1$ respectively. It can be easily shown that the mode of the logarithm of the posterior $p(\varTheta | Y)$ is identical to that of the penalized log-likelihood (6), up to a multiplicative constant of proportionality. This choice of prior is related to Huang et al. (2006), where independent normal priors are used for the generalized autoregressive parameters in the Cholesky decomposition when the normalizing constant is omitted.

Though ROPE is primarily a precision matrix estimator, it is interesting to examine its properties in the context of principal component analysis (PCA). As mentioned in recent studies (see e.g., Ledoit and Wolf, 2015), the variation explained by a principal component is not equal to the population eigenvalue. Because of this, alternative estimators for the covariance matrix could improve the accuracy of PCA. Motivated by this, we tested how ROPE, along with other precision and covariance matrix estimators, can improve the accuracy of PCA in a small simulation study similar to the one described in Ledoit and Wolf (2015). The results were promising, and in some cases ROPE clearly outperformed all the other methods (results not shown).

SUPPLEMENTARY MATERIALS

ROPE_codes.zip: This .zip file contains various R codes used in the simulation studies and real data applications along with alternative procedures to derive the ROPE solution, and a README.txt file.

Proof.pdf: This .pdf file contains the detailed proofs of Lemma 1 and Property 1.

Table1.pdf and Table2.pdf: These .pdf files contain tables of detailed and additional risk measures of the simulation study described in section 3.

Fig_S1.pdf, Fig_S2.pdf, Fig_S3.pdf and Fig_S4.pdf: These .pdf files contain a graphical illustration of the information presented in the supplementary Tables 1 and 2.

WheatCVE.pdf: This .pdf file contains a plot of the cross-validation errors as a function of the tuning parameter ρ concerning the genomic prediction study described in Section 4.2.

ACKNOWLEDGMENTS

We would like to thank the Associate Editor, three anonymous referees, Ilkka Launonen and Ashley Last for their valuable comments, which helped us to improve the presentation of this paper. This work was supported by the University of Oulu's Exactus Doctoral Programme.

References

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

Boyd, S. and L. Vandenberghe (2004). *Convex Optimization.* Cambridge, U.K.: Cambridge University Press.

Bühlmann, P., M. Kalisch, and L. Meier (2014). High-dimensional statistics with a view toward applications in biology. *Annual Review of Statistics and Its Applications 1*, 255–278.

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

Crossa, J., C. G. DeLos, P. Pérez, D. Gianola, J. Burgueño, J. L. Araus, D. Makumbi, R. P. Singh, S. Dreisigacker, J. Yan, V. Arief, M. Banziger, and H.-J. Braun (2010). Prediction of genetic values of quantitative traits in plant breeding using pedigree and molecular markers. *Genetics 186*, 713–724.

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

Dym, H. (2007). *Linear Algebra in Action*. Providence, Rhode Island: American Mathematical Society.

Endelman, J. B. (2011). Ridge regression and other kernels for genomic selection with R package rrBLUP. *The Plant Genome 4*(3), 250–255.

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

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

Henderson, C. R. (1975). Best linear unbiased estimation and prediction under a selection model. *Biometrics 31*(2), 423–447.

Highman, N. J. and H.-M. Kim (2000). Numerical analysis of a quadratic matrix equation. *IMA Journal of Numerical Analysis 20*, 499–519.

Huang, J. Z., N. Liu, M. Pourahmadi, and L. Liu (2006). Covariance selection and estimation via penalised normal likelihood. *Biometrika 93*(1), 85–98.

Kuismin, M. and M. J. Sillanpää (2016). Use of Wishart prior and simple extensions for sparse precision matrix estimation. *PLOS ONE 11*(2), e0148171.

Larin, V. B. (2014). Algorithms for solving a unilateral quadratic matrix equation and the model updating problem. *International Applied Mechanics 50*(3), 321–334.

Laub, A. J. (1979). A Schur method for solving algebraic Riccati equations. *IEEE Transactions on Automatic Control 24*(6), 913–921.

Ledoit, O. and M. Wolf (2004a). Honey, I shrunk the sample covariance matrix. *The Journal of Portfolio Management 30*(4), 110–119.

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

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

Ledoit, O. and M. Wolf (2015). Spectrum etimation: A unified framework for covariance matrix estimation and PCA in large dimensions. *Journal of Multivariate Analysis 139*, 360–384.

Li, Z. and M. J. Sillanpää (2012). Overview of LASSO-related penalized regression methods for quantitative trait mapping and genomic selection. *Theoretical and Applied Genetics 125*, 419–435.

Liu, W. and X. Luo (2015). Fast and adaptive sparse precision matrix estimation in high dimensions. *Journal of Multivariate Analysis 135*, 153–162.

Mardia, K., J. Kent, and J. Bibby (1979). *Multivariate Analysis*. London; New York : Academic Press.

Meinshausen, N. and P. Bühlmann (2006). High dimensional graphs and variable selection with the LASSO. *The Annals of Statistics 34*(3), 1436–1462.

Müller, D., F. Technow, and A. E. Melchinger (2015). Shrinkage estimation of the genomic relationship matrix can improve genomic estimated breeding values in the training set. *Theoretical and Applied Genetics 128*, 693–703.

Pang, H., H. Liu, and R. Vanderbei (2014). The FASTCLIME package for linear programming and large-scale precision matrix estimation in R. *Journal of Machine Learning Research 15*, 489–493.

Patterson, H. D. and R. Thompson (1971). Recovery of inter-block information when block sizes are unequal. *Biometrika 58*(3), 545–554.

Patterson, N., A. L. Price, and D. Reich (2006). Population structure and eigenanalysis. *PLoS Genet 2*(12), 1–20.

Pérez, P., C. G. DeLos, J. Crossa, and D. Gianola (2010). Genomic-enabled prediction based on molecular markers and pedigree using the BLR package in R. *Plant Genome 3*, 106–116.

Pourahmadi, M. (2013). *High-Dimensional Covariance Estimation*. New York: John Wiley & Sons.

Stacy, E. W. (1962). A generalization of the Gamma distribution. *The Annals of Mathematical Statistics 33*(3), 1187–1192.

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

VanRaden, P. M. (2008). Efficient methods to compute genomic predictions. *Journal of Dairy Science 91*(11), 4414–4423.

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.

Wieringen, W. N. and C. F. W. Peeters (2016). Ridge estimation of inverse covariance matrices from high-dimensional data. *Computational Statistics and Data Analysis 103*, 284–303.

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

Won, J.-H., J. Lim, S.-J. Kim, and B. Rajaratnam (2009). Maximum likelihood covariance estimation with the condition number constraint. Technical report, Stanford University, Department of Statistics.

Wonham, W. M. (1968). On a matrix Riccati equation of stochastic control. *SIAM J. Contr. 6*(4), 681–697.

Yuan, T. and J. Wang (2013). A coordinate descent algorithm for sparse positive definite matrix estimation. *Statisticl Analysis and Data Mining 6*, 431–442.