Article

High-Dimensional Precision Matrix Estimation through GSOS with Application in the Foreign Exchange Market

Azam Kheyri [1] , Andriette Bekker [1,*] and Mohammad Arashi [1,2]

1   Department of Statistics, Faculty of Natural and Agricultural Sciences, University of Pretoria,
     Pretoria 0028, South Africa
2   Department of Statistics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad,
     Mashhad 9177948974, Iran
*   Correspondence: andriette.bekker@up.ac.za

Abstract: This article studies the estimation of the precision matrix of a high-dimensional Gaussian network. We investigate the graphical selector operator with shrinkage, GSOS for short, to maximize a penalized likelihood function where the elastic net-type penalty is considered as a combination of a norm-one penalty and a targeted Frobenius norm penalty. Numerical illustrations demonstrate that our proposed methodology is a competitive candidate for high-dimensional precision matrix estimation compared to some existing alternatives. We demonstrate the relevance and efficiency of GSOS using a foreign exchange markets dataset and estimate dependency networks for 32 different currencies from 2018 to 2021.

1. Introduction

In recent years, covariance and precision matrix estimates have been studied extensively under high-dimensional scenarios. The motivation for these investigations is the argument that traditional likelihood-based techniques perform inaccurately or do not even exist. Examples of these include the graphical lasso algorithm and its extensions ([1–7]), and the ridge regularization-driven approach for the log-likelihood function ([8–11]). In addition to considering single L_1 (graphical lasso) and Frobenius (graphical ridge) penalties for the precision matrix, some studies have proposed combining both, resulting in an elastic net-type penalty. For instance, the following articles can be mentioned [10,12–14].

Despite the vast number of reported studies in the literature on the estimation of precision matrices, there are few reported studies concerning generalization to allow for the incorporation of prior knowledge of a precision matrix. For the Frobenius penalized case we can mentioned [9–11]. The only proposals for incorporating target matrices (prior knowledge) for the L_1 and elastic net penalty are those of [10,15]. In this paper, we propose the graphical selector operator with shrinkage (GSOS) as a combination of the generalized ridge (ridge with target matrix) and L_1 penalty. The main aim of this paper is to present and compare the new precision matrix estimator and propose a novel algorithm for Gaussian graphical models. The numerical study shows that the GSOS estimator is comparable with other available estimators and performs better in most cases than others.

Estimation, Literature Review

In graphical models, Gaussian graphical models are frequently used for modeling conditional dependencies in multivariate data. Dependency structures are determined by

estimating the precision matrix through standard methodologies, for instance, a maximum likelihood or the regularized type of maximum likelihood in high-dimensional cases.

Consider random vector $z \sim \mathcal{N}_p(\mu, \Sigma)$; it follows a multivariate Gaussian distribution with mean vector $\mu \in \mathbb{R}^p$ and positive definite covariance matrix $\Sigma \in \mathbb{R}^p \times \mathbb{R}^p$ corresponding to conditional independence graph $\mathcal{G}$. In terms of mathematics, a pair $(z, \mathcal{G})$ is a Gaussian graphical model. In these models, the graph $\mathcal{G}$ and the precision matrix $\Theta := \Sigma^{-1}$ are intimately related. That means that the zeroes in the precision matrix correspond to pairs of conditionally independent features, given all the other variables. In other words, there is no edge between these two variables (nodes) in a graph $\mathcal{G}$. Furthermore, two coordinates z_k and z_k' are independent if, in the graph $\mathcal{G}$, there is no edge between them. Maximum likelihood estimation is a logical technique to estimate the precision matrix for the positive definite matrix Θ; this estimator is expressed as

$$\hat{\Theta} = \arg \min_{\Theta \succ 0} \{-\log \det(\Theta) + tr(S\Theta)\}, \tag{1}$$

where $S = z^T z / n$ is the sample covariance matrix and the maximum likelihood estimator Θ is equal to S^{-1}. However, two issues might emerge when utilizing this maximum likelihood technique to estimate Θ. First, in the high dimension case $p > n$, the empirical covariance matrix S is singular and hence cannot be inverted to provide an estimate of Θ. Even if p and n are almost equal and S is singular, the maximum likelihood estimate for Θ will have a relatively high variance. Second, it is frequently valuable to identify pairs of disconnected variables in a graphical model that are conditionally independent; these correspond to zeroes in Θ. However, in general, (1) will produce an estimate of Θ with no elements equal to zero.

The likelihood function is complemented with a penalty function in high-dimensional settings, which yields the maximum regularized likelihood estimator. Graphical lasso adds an L_1 penalty to log-likelihood and induces sparsity of Θ due to maximizing the penalized log-likelihood:

$$\hat{\Theta}_{Glasso}(\rho) = \arg \min_{\Theta \succ 0} \{-\log \det(\Theta) + tr(S\Theta) + \rho ||\Theta||_1\}, \tag{2}$$

where ρ is a non-negative tuning parameter and $||\Theta||_1$ denotes the sum of the absolute values of the elements of Θ. This approach was considered almost simultaneously by [1,16–18]. Due to the ensuing sparse answer, the graphical lasso estimate has gained significant interest and has become popular; it is still is an active area of research (cf. [6,19–21]). In contrast to circumstances where sparsity is advantageous, there are cases when more exact representations of the high-dimensional precision matrix are fundamentally desirable. Furthermore, the genuine (graphical) model does not have to be (very) sparse in terms of having many zero elements. In these situations, we may use a regularization approach that shrinks the estimated precision matrix elements instead of forcing them to be zero. The ridge precision estimator maximizes the log-likelihood augmented by a Frobenius norm, [9] in the most general form presented the ridge estimator as follows

$$\hat{\Theta}_{Ridge}(\lambda) = \arg \min_{\Theta \succ 0} \{-\log \det(\Theta) + tr(S\Theta) + \frac{1}{2}\lambda ||\Theta - T||_F^2\}, \tag{3}$$

where λ is the penalty parameter, T is a known symmetric and semi-positive definite target matrix, and $||A||_F^2$ denotes the Frobenius norm, the sum of the square values of the elements of the matrix A. Before estimating, the target matrix is set as an initial guess, toward which the precision estimate is shrunk. Estimator (3) is called the alternative Type I ridge precision estimator, and when the target matrix is equal to zero matrices, the alternative Type II ridge precision estimator. It should be mentioned that [11] considered the estimator mentioned in (3) independently and concurrently, and called it a ridge-type operator for precision matrix estimation (ROPE).

In a recent study in this area, Ref. [10] generalized the ridge inverse covariance estimator to allow for entry-wise penalization. Their proposed estimator shrinks toward a user-specified non-random target matrix and is shown to be positive definite and consistent. Furthermore, they obtained a generalization of the graphical lasso estimator and its elastic net counterpart. Recently, ref. [15] considered elastic net-type penalization (gelnet) for precision matrix estimation in the presence of a diagonal target matrix. They showed that it is possible to adopt the iterative procedure of [10] for the elastic net problem, but it is not computationally attractive.

Under the continuity property of the ridge estimator and apart from sparsity, estimations of the precision matrix with (3) can be far from the actual value Θ. Hence, the motivation of our approach to find a closer estimator. We use the Frobenius norm penalty to penalize the deviation between any primary or initial target estimator and Θ, and tune it to improve the accuracy of the final estimation. Furthermore, the L_1 penalty is added to account for sparsity, returning to the well-known elastic net with some added refinement. Therefore, in this paper, we consider combining the estimation approaches (2) and (3) to propose an elastic net-type estimator for the precision matrix: the GSOS.

2. The Proposed Method

Let $X_{n \times p}$ be the data matrix, including n observations of a p—dimensional Gaussian distribution with zero mean and positive definite covariance matrix Σ. Consider the following type of estimation problem for estimating the unknown precision matrix $\Theta = \Sigma^{-1}$ based on n observations:

$$\hat{\Theta}(\alpha, \lambda, T) = \arg\min_{\Theta \succ 0} \{ -\log \det(\Theta) + tr(S\Theta) + \lambda(\alpha||\Theta||_1 + \frac{(1-\alpha)}{2}||\Theta - T||_F^2) \}, \quad (4)$$

where $T_{p \times p}$ is a known symmetric and semi-positive definite target matrix. Additionally, $\lambda \geq 0, \alpha \in [0, 1]$ are tuning parameters. Note that if $\alpha = 0$, the precision matrix estimator (4) becomes the ridge estimator that is mentioned in (3).

Using sub-gradient notation and rules from [22], we obtain the following optimal conditions for (4)

$$-\Theta^{-1} + S + \lambda\alpha\Gamma - \lambda(1-\alpha)(T - \Theta) = 0$$

$$\text{i.e.,} \quad -\Theta^{-1} + S + \lambda\alpha\Gamma - \lambda(1-\alpha)T + \lambda(1-\alpha)\Theta = 0$$

$$\text{i.e.,} \quad -\Theta^{-1} + A + \lambda\alpha\Gamma + \lambda(1-\alpha)\Theta = 0, \quad (5)$$

where $A := A(S, \alpha, \lambda, T) = S - \lambda(1-\alpha)T$, and the matrix $\Gamma = [\gamma_{kk'}]$ denotes the component-wise signs of Θ with $\gamma_{kk'} \in [-1, 1]$ for $\Theta_{kk'} = 0$. The solution of the normal Equation (5) is found by iteratively running over the columns/rows, considering the remaining ones fixed. The update requires that each matrix in (5) is partitioned as follows:

$$\Theta = \begin{pmatrix} \Theta_{11} & \theta_{12} \\ \theta_{21} & \theta_{22} \end{pmatrix}, \quad \Gamma = \begin{pmatrix} \Gamma_{11} & \gamma_{12} \\ \gamma_{21} & \gamma_{22} \end{pmatrix}, \quad A = \begin{pmatrix} A_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}, \quad (6)$$

where Θ_{11} is a square matrix, θ_{12} is a vector, and θ_{22} is a scalar. Γ and A are partitioned similarly. Consider $W = \Theta^{-1}$; using the properties of inverses of block-partitioned matrices, we have that

$$W = \begin{pmatrix} W_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix} = \begin{pmatrix} \Theta_{11}^{-1} + \frac{\Theta_{11}^{-1}\theta_{12}\theta_{21}\Theta_{11}^{-1}}{(\theta_{22} - \theta_{21}\Theta_{11}^{-1}\theta_{12})} & -\frac{\Theta_{11}^{-1}\theta_{12}}{\theta_{22} - \theta_{21}\Theta_{11}^{-1}\theta_{12}} \\ -\frac{\theta_{21}\Theta_{11}^{-1}}{\theta_{22} - \theta_{21}\Theta_{11}^{-1}\theta_{12}} & \frac{1}{\theta_{22} - \theta_{21}\Theta_{11}^{-1}\theta_{12}} \end{pmatrix}. \quad (7)$$

Considering the pth column of (5), we obtain

$$-\boldsymbol{w}_{12} + \boldsymbol{a}_{12} + \lambda\alpha\boldsymbol{\gamma}_{12} + \lambda(1-\alpha)\boldsymbol{\theta}_{12} = \mathbf{0}. \tag{8}$$

By substituting $\boldsymbol{w}_{12}$ and w_{22} into (7), we have that

$$\frac{\boldsymbol{\Theta}_{11}^{-1}\boldsymbol{\theta}_{12}}{\theta_{22} - \boldsymbol{\theta}_{21}\boldsymbol{\Theta}_{11}^{-1}\boldsymbol{\theta}_{12}} + \boldsymbol{a}_{12} + \lambda\alpha\boldsymbol{\gamma}_{12} + \lambda(1-\alpha)\boldsymbol{\theta}_{12} = \mathbf{0}$$

$$i.e., \ \boldsymbol{\Theta}_{11}^{-1}\boldsymbol{\theta}_{12}w_{22} + \boldsymbol{a}_{12} + \lambda\alpha\boldsymbol{\gamma}_{12} + \lambda(1-\alpha)\boldsymbol{\theta}_{12} = \mathbf{0}$$

$$i.e., \ (\boldsymbol{\Theta}_{11}^{-1}w_{22} + \lambda(1-\alpha)\boldsymbol{I}_{p-1})\boldsymbol{\theta}_{12} + \boldsymbol{a}_{12} + \lambda\alpha\boldsymbol{\gamma}_{12} = \mathbf{0}. \tag{9}$$

Consider $\tilde{\boldsymbol{q}}_{12} := abs(\boldsymbol{\theta}_{12})$ and $\tilde{\boldsymbol{\gamma}} := \lambda\alpha\boldsymbol{\gamma}_{12}$; then, (9) is equivalent to

$$(\boldsymbol{\Theta}_{11}^{-1}\frac{w_{22}}{\alpha\lambda} + \frac{1-\alpha}{\alpha}\boldsymbol{I}_{p-1})\tilde{\boldsymbol{q}}_{12} * \tilde{\boldsymbol{\gamma}} + \boldsymbol{a}_{12} + \tilde{\boldsymbol{\gamma}} = \mathbf{0},$$

$$\tilde{\boldsymbol{q}}_{12} * (abs(\tilde{\boldsymbol{\gamma}}) - \lambda\alpha\mathbf{1}_{p-1}) = \mathbf{0}, \tag{10}$$

$$||\tilde{\boldsymbol{\gamma}}||_{\infty} \leq \lambda\alpha,$$

where $*$ denotes the element-wise multiplication of two vectors. The conditions mentioned in (10) are KKT optimality conditions for the following box-constrained problem for $\gamma \in R^{p-1}$

$$\underset{||\tilde{\boldsymbol{\gamma}}||_{\infty}\leq\lambda\alpha}{minimize} \ \frac{1}{2}(\boldsymbol{a}_{12} + \tilde{\boldsymbol{\gamma}})^T(\boldsymbol{\Theta}_{11}^{-1}\frac{w_{22}}{\alpha\lambda} + \frac{1-\alpha}{\alpha}\boldsymbol{I}_{p-1})^{-1}(\boldsymbol{a}_{12} + \tilde{\boldsymbol{\gamma}}). \tag{11}$$

By solving the optimization problem (11) and finding the optimum point λ^*, $\boldsymbol{\theta}_{12}$ can be updated as follows for all $\alpha \in (0, 1]$

$$\hat{\boldsymbol{\theta}}_{12} = -(\boldsymbol{\Theta}_{11}^{-1}\frac{w_{22}}{\alpha\lambda} + \frac{1-\alpha}{\alpha}\boldsymbol{I}_{p-1})^{-1}(\boldsymbol{a}_{12} + \boldsymbol{\gamma}^*). \tag{12}$$

For the diagonal elements of the precision matrix θ_{22}, we consider (5) for diagonal elements

$$-w_{22} + \boldsymbol{a}_{22} + \lambda\alpha + \lambda(1-\alpha)\theta_{22} = 0. \tag{13}$$

From (7), we have that $w_{22} = 1/(\theta_{22} - \boldsymbol{\theta}_{21}\boldsymbol{\Theta}_{11}^{-1}\boldsymbol{\theta}_{12})$; this implies that (13) is equivalent to

$$1 + (\boldsymbol{a}_{22} + \lambda\alpha + \lambda(1-\alpha)\theta_{22})(\boldsymbol{\theta}_{21}\boldsymbol{\Theta}_{11}^{-1}\boldsymbol{\theta}_{12} - \theta_{22}) = 0. \tag{14}$$

Let $C := \boldsymbol{\theta}_{21}\boldsymbol{\Theta}_{11}^{-1}\boldsymbol{\theta}_{12}$; therefore, we have this quadratic equation

$$-\lambda(1-\alpha)\theta_{22}^2 + (\lambda(1-\alpha)C - \boldsymbol{a}_{22} - \lambda\alpha)\theta_{22} + C(\boldsymbol{a}_{22} + \lambda\alpha) + 1 = 0. \tag{15}$$

Quadratic Equation (15) has two distinct real roots: positive (acceptable) and negative (unacceptable). Hence, we can update the diagonal elements of the precision matrix with the positive root. Finally, $\boldsymbol{w}_{12}$ and w_{22} are updated using normal Equation (5). Our method here is similar to the dpglasso estimator proposed by [5]. Furthermore, we can follow the glasso approach to solve the problem. The fundamental distinction between glasso and dpglasso is that in glasso, $\boldsymbol{W}$ is not equal to the inverse of $\boldsymbol{\Theta}$. Furthermore, glasso deals with $\boldsymbol{W}$, whereas dpglasso considers its inverse $\boldsymbol{\Theta}$. Finally, Algorithm 1 shows the procedure for $\alpha \in (0, 1]$.

Algorithm 1 GSOS based on dpglasso approach

---

1. Initialize Θ, a diagonal matrix with the diagonal elements as follows, for $k = 1, \ldots, p$

$$\theta_{kk} = \frac{-(a_{kk} + \lambda\alpha) + \sqrt{(a_{kk} + \lambda\alpha)^2 + 4\lambda(1-\alpha)}}{2\lambda(1-\alpha)}$$

and $W = A + \lambda\alpha I_p + \lambda(1-\alpha)\Theta$,

2. Focus on the columns and repeatedly perform the following steps till convergence.

(A)   Rearrange the rows/columns so that the target column is last (according to (6));

(B)   Solve (11) and update

$$\hat{\theta}_{12} = -(\Theta_{11}^{-1}\frac{w_{22}}{\alpha\lambda} + \frac{1-\alpha}{\alpha}I_{p-1})^{-1}(a_{12} + \gamma^*);$$

(C)   Solve (15) to obtain $\hat{\theta}_{22}$;

(D)   Update $\hat{w}_{12} = a_{12} + \gamma^*$;

(E)   Update $\hat{w}_{22} = a_{22} + \lambda\alpha + \lambda(1-\alpha)\hat{\theta}_{22}$.

Glasso Scenario

The following relationship between W and Θ is established in glasso

$$W = \begin{pmatrix} W_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix} = \begin{pmatrix} (\Theta_{11} - \frac{\theta_{12}\theta_{21}}{\theta_{22}})^{-1} & -W_{11}\frac{\theta_{12}}{\theta_{22}} \\ -\frac{\theta_{21}}{\theta_{22}}W_{11} & \frac{1}{\theta_{22}} + \frac{\theta_{21}W_{11}\theta_{12}}{\theta_{22}^2} \end{pmatrix}. \tag{16}$$

Consider the pth column of (5) and substitute w_{12} from (16):

$$-w_{12} + a_{12} + \lambda\alpha\gamma_{12} + \lambda(1-\alpha)\theta_{12} = 0,$$

$$W_{11}\frac{\theta_{12}}{\theta_{22}} + a_{12} + \lambda\alpha\gamma_{12} + \lambda(1-\alpha)\theta_{12} = 0.$$

Define $\beta = -\frac{\theta_{12}}{\theta_{22}}$; therefore

$$W_{11}\beta - a_{12} - \lambda\alpha\gamma_{12} + \lambda(1-\alpha)\theta_{22}\beta = 0. \tag{17}$$

Letting $W_{11} = V$, $a_{12} = u$ and $\gamma_{12} = \rho$, then, for every $i = 1, \ldots, p$, we have that

$$V_{ii}\beta_i + \sum_{k\neq i}^{p-1} V_{ki}\beta_k - u_i - \lambda\alpha\rho_i + \lambda(1-\alpha)\theta_{22}\beta_i = 0. \tag{18}$$

Subsequently, the update has the following form

$$\hat{\beta}_i \leftarrow \frac{S(u_i - \sum_{k\neq i}^{p-1} V_{ki}\beta_k, \alpha\lambda)}{V_{ii} + \lambda(1-\alpha)\theta_{22}}, \tag{19}$$

where S is the soft-threshold operator $S(x,t) = sign(x)(|x| - t)_+$. We cycle through the predictors until convergence. $\hat{\beta}$ is the optimum point of the following quadratic problem

$$\underset{\beta\in R^{p-1}}{minimize} \ \frac{1}{2}\beta^T W_{11}\beta - \beta^T a_{12} - \lambda\alpha||\beta||_1 + \frac{\lambda(1-\alpha)}{2}\theta_{22}\beta^T\beta, \tag{20}$$

which corresponds to the normal equations (17). After solving this quadratic problem and finding $\hat{\beta}$, we can update $\hat{w}_{12}$. From (5) for the diagonal elements of θ_{22} of the precision matrix, we have that

$$-w_{22} + a_{22} + \lambda\alpha + \lambda(1-\alpha)\theta_{22} = 0. \tag{21}$$

From (16) we have that $w_{22} = \frac{1}{\theta_{22}} + \frac{\theta_{21} W_{11} \theta_{12}}{\theta_{22}^2} = \frac{1}{\theta_{22}} + \boldsymbol{\beta}^T W_{11} \boldsymbol{\beta}$; hence, (21) is equivalent to

$$\lambda(1-\alpha)\theta_{22}^2 - (\boldsymbol{\beta}^T W_{11} \boldsymbol{\beta} - a_{22} - \alpha\lambda)\theta_{22} - 1 = 0. \tag{22}$$

Equation (22) has two distinct real roots with positive (acceptable) and negative (unacceptable) signs. Hence, we can update the diagonal elements of the precision matrix with the positive root. Finally, $\hat{\boldsymbol{\theta}}_{12}$ and $\hat{w}_{22}$ are updated by (5). Algorithm 2 briefly outlines these steps.

Algorithm 2 GSOS based on glasso approach

---

1. Initialize $\boldsymbol{\Theta}$, a diagonal matrix with the diagonal elements as follows, for $k = 1, \ldots, p$

$$\theta_{kk} = \frac{-(a_{kk} + \lambda\alpha) + \sqrt{(a_{kk} + \lambda\alpha)^2 + 4\lambda(1-\alpha)}}{2\lambda(1-\alpha)}$$

and $\boldsymbol{W} = \boldsymbol{A} + \lambda\alpha\boldsymbol{I}_p + \lambda(1-\alpha)\boldsymbol{\Theta}$.
2. Focus on the columns and repeatedly perform the following steps till convergence.

(A)   Rearrange the rows/columns so that the target column is last;
(B)   Solve (20) and update $\hat{w}_{12} = W_{11}\hat{\boldsymbol{\beta}}$;
(C)   Solve (22) to update $\hat{\theta}_{22}$;
(D)   Update $\hat{\boldsymbol{\theta}}_{12} = -\hat{\theta}_{22}\hat{\boldsymbol{\beta}}$;
(E)   Update $\hat{w}_{22} = a_{22} + \lambda\alpha + \lambda(1-\alpha)\hat{\theta}_{22}$.

---

3. Simulation

In this section, the statistical performance of GSOS is compared to some other popular estimators (glasso [1], ROPE or alternative ridge [9,11], and graphical elastic net [15]). We simulate the data from a multivariate Gaussian distribution $\mathcal{N}_p(\mathbf{0}, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma} = [\sigma_{k,k'}]$ and $\boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1} = [\theta_{k,k'}]$ are $p \times p$ positive definite matrices. Six different models are used to compare the methods (see Appendix A for precision matrices). We evaluate these estimators on the six network structures with $p = 20, 50, 100$ nodes and sample size of $n = 50$ by implementing Algorithm 2.

- Network 1: A model with compound symmetry structure where $\sigma_{k,k} = 1$ and $\sigma_{k,k'} = 0.6^2$ for $k \neq k'$. In this model, the covariance matrix is structured and non-sparse;
- Network 2. The prototype matrix $\boldsymbol{\Theta}_0$ is used to standardize the precision matrix $\boldsymbol{\Theta}$ to possess a unit diagonal. Let $\boldsymbol{\Theta}_0 = \boldsymbol{A} + a\boldsymbol{I}$, where each off-diagonal entry in $\boldsymbol{A}$ 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. Here we have an unstructured and sparse precision matrix;
- Network 3. The precision matrix is defined $\boldsymbol{\Theta} = \frac{1}{n}\boldsymbol{Y}^T\boldsymbol{Y}$, where $\boldsymbol{Y} = [y_{kk'}]$ is an $n \times p$ matrix with $n = 10,000$ and $y_{kk'}$ comes from $\mathcal{N}(0,1)$. The precision matrix of this model is unstructured and non-sparse;
- Network 4. A star model with $\theta_{k,k} = 1$, $\theta_{1,k} = \theta_{k,1} = 0.1$ and $\theta_{k,k'} = 0$ otherwise. This local area network has a structured and sparse precision matrix;
- Network 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;
- Network 6. A diagonally dominant model. Consider $\boldsymbol{B} = \frac{1}{2}(\boldsymbol{A} + \boldsymbol{A}^T)$, where $\boldsymbol{A}$ is a $p \times p$ matrix with zero diagonal elements. Each off-diagonal element of $\boldsymbol{A}$ is drawn from a standard uniform distribution. Compute a matrix $\boldsymbol{D} = \frac{1}{\gamma}\boldsymbol{B}$, where $\boldsymbol{D} = [d_{k,k'}]$ and γ is the largest row sum of the absolute values of the elements of the matrix $\boldsymbol{B}$. Finally, each off-diagonal element of $\boldsymbol{\Sigma}$ is chosen as $\sigma_{k,k'} = d_{k,k'}$ and $\sigma_{k,k} = 1 + e_i$, where e_i is drawn from uniform distribution with minimum 0 and maximum 0.1. This covariance matrix is unstructured and non-sparse.

To calculate the performance measures for all the methods, 100 independent simulations for each network are performed and the average of several loss functions is calculated. The optimal tuning parameter λ for each method is determined for each simulation run using five-fold cross-validation. For glasso, gelnet, and ROPE, we use the five-fold cross-validation found in the R-package "GLassoElnetFast" with a set value of λ of 50 elements varying from 0.01 to 10, and α with 20 elements varying from 0.01 to 0.99 for gelnet and our estimators. As a target matrix, assume identity matrix I and a scalar matrix νI, where $\nu = p^2/tr(S)$.

3.1. Performance Measures

To calculate the performance of a given estimator $\hat{\Theta}$, we consider four loss functions that have been used widely in other research in this area (see, e.g., [7,11,14,15]).

Loss Functions

- The Kullback–Leibler loss: $KL = tr(\Sigma\hat{\Theta}) - logdet(\Sigma\hat{\Theta}) - p$;
- The L_2 loss: $L_2 = ||\Theta - \hat{\Theta}||_F$;
- The quadratic loss: $QL = tr(\Sigma\hat{\Theta} - I_p)^2$;
- The spectral norm loss: $SP = d_1$, where d_1^2 is the largest eigenvalue of the matrix $(\Theta - \hat{\Theta})^2$.

To provide risk measures, the averages of these losses are determined for each method from 100 simulations. Figures 1–4 show the final findings. The plot layout is taken from [11]; the columns that go along the little black dots represent loss means. The bars at the top of each column represent standard errors (*mean* $\pm$ *SD*).

3.2. Simulation Results

The results for different networks are displayed in Figures 1–4. We summarize some observations based on the results as follows:

- In general, as mentioned in [15], it is advantageous to include a target in the methods, since each approach works better with the appropriate target;
- Compared to other alternatives, GSOS is often a considerable contender for high-dimensional precision matrix estimation;
- The question of which target is more effective remains. However, in most cases, our simulations suggest that the identity target matrix works better.
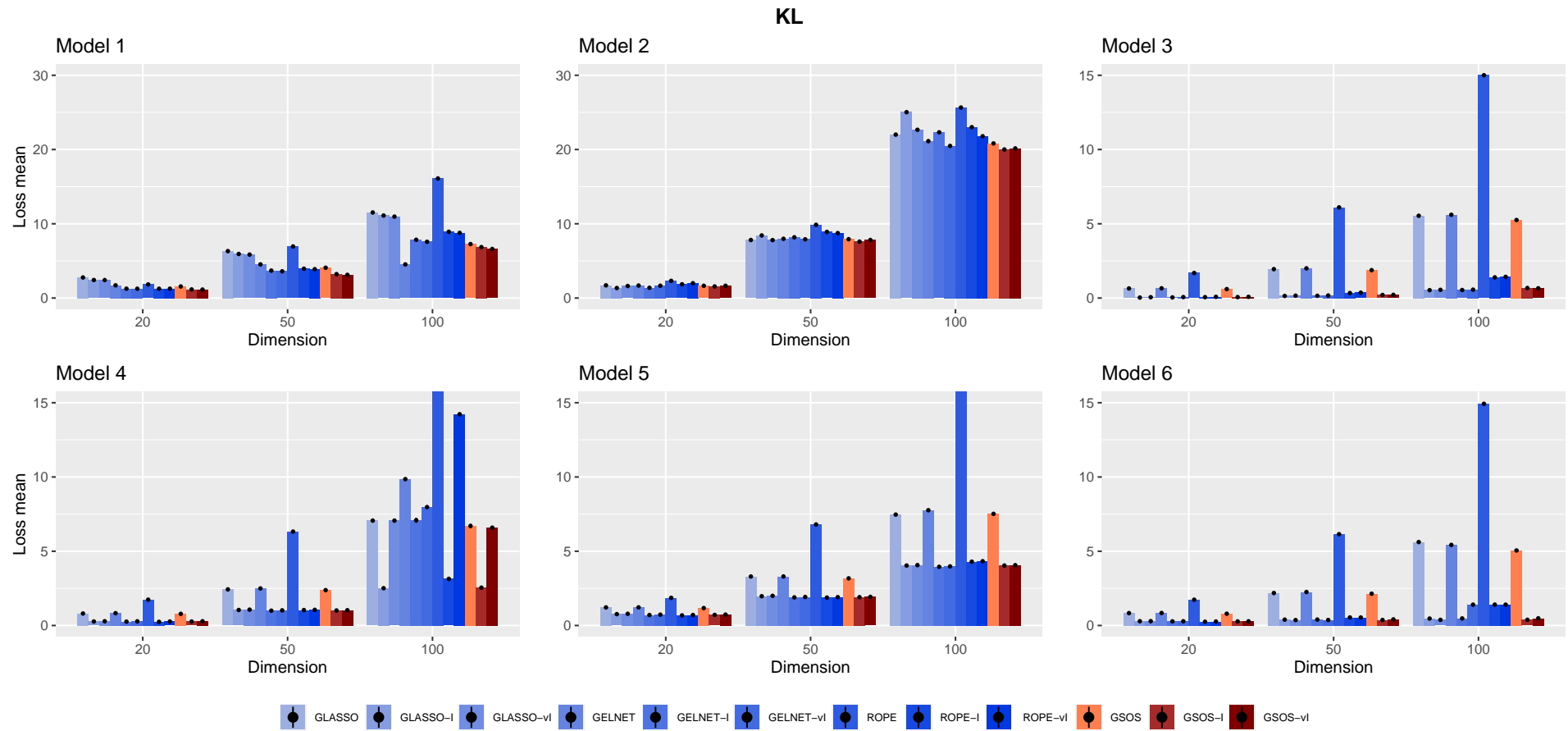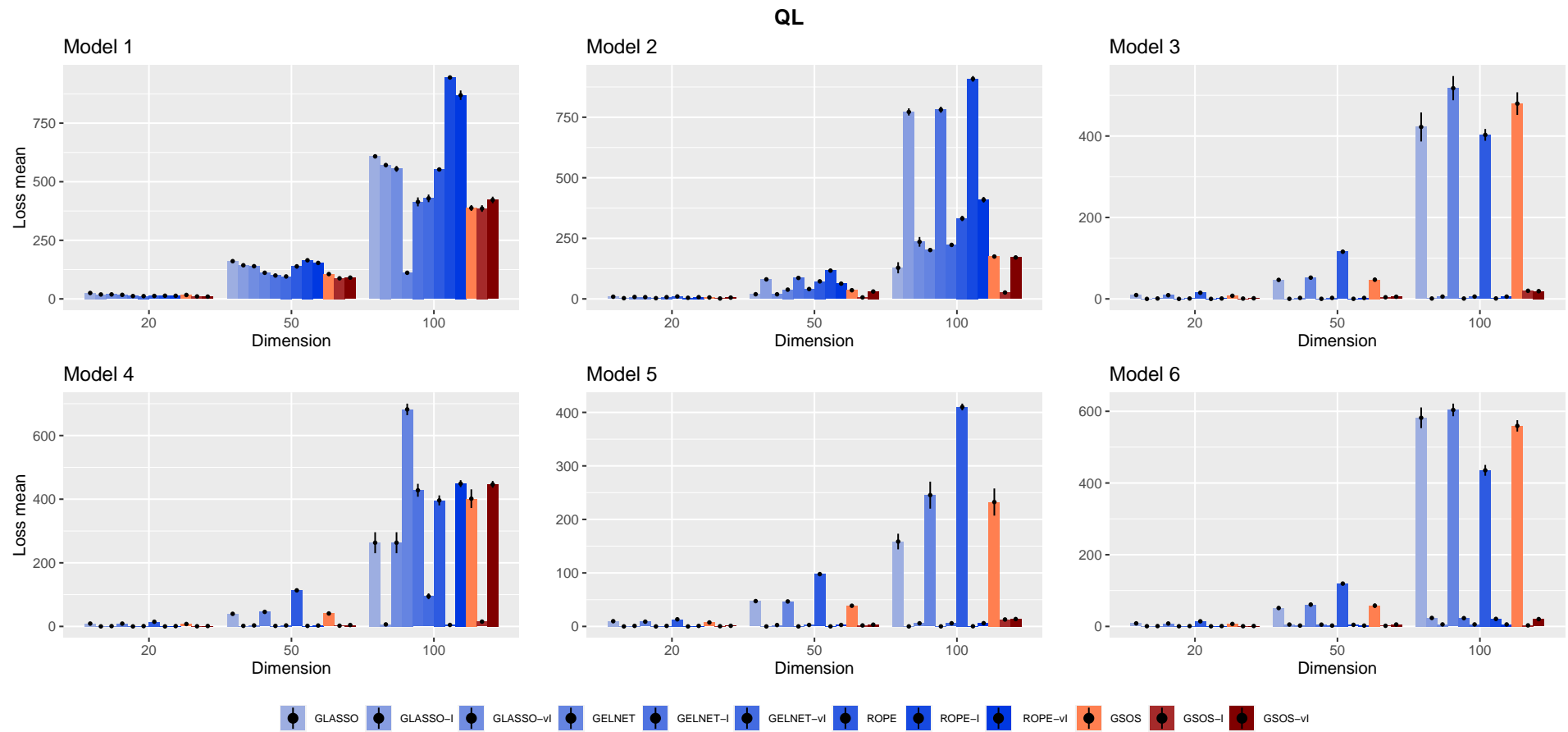
Figure 1. The mean of Kullback–Leibler loss for the different models and methods based on 100 replications.
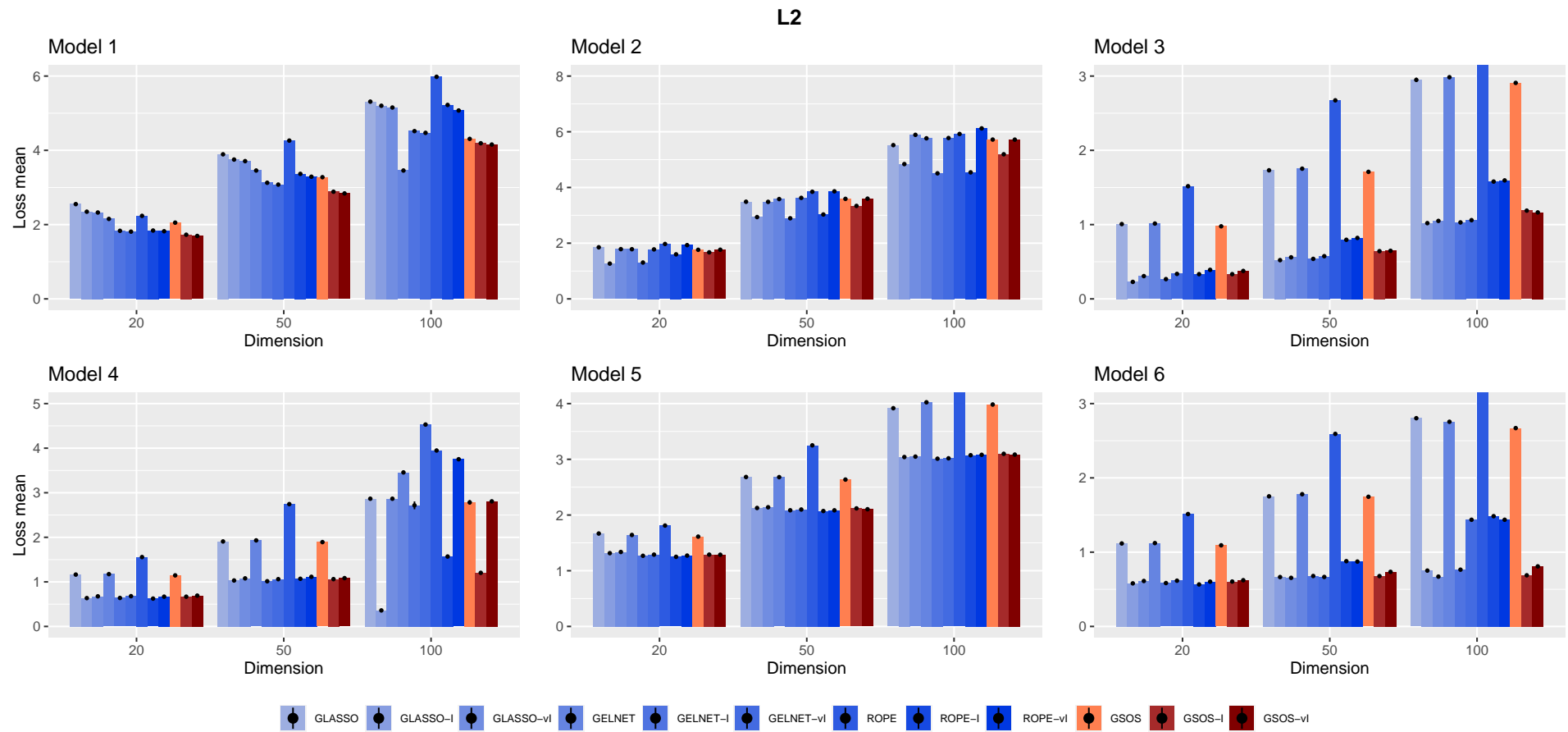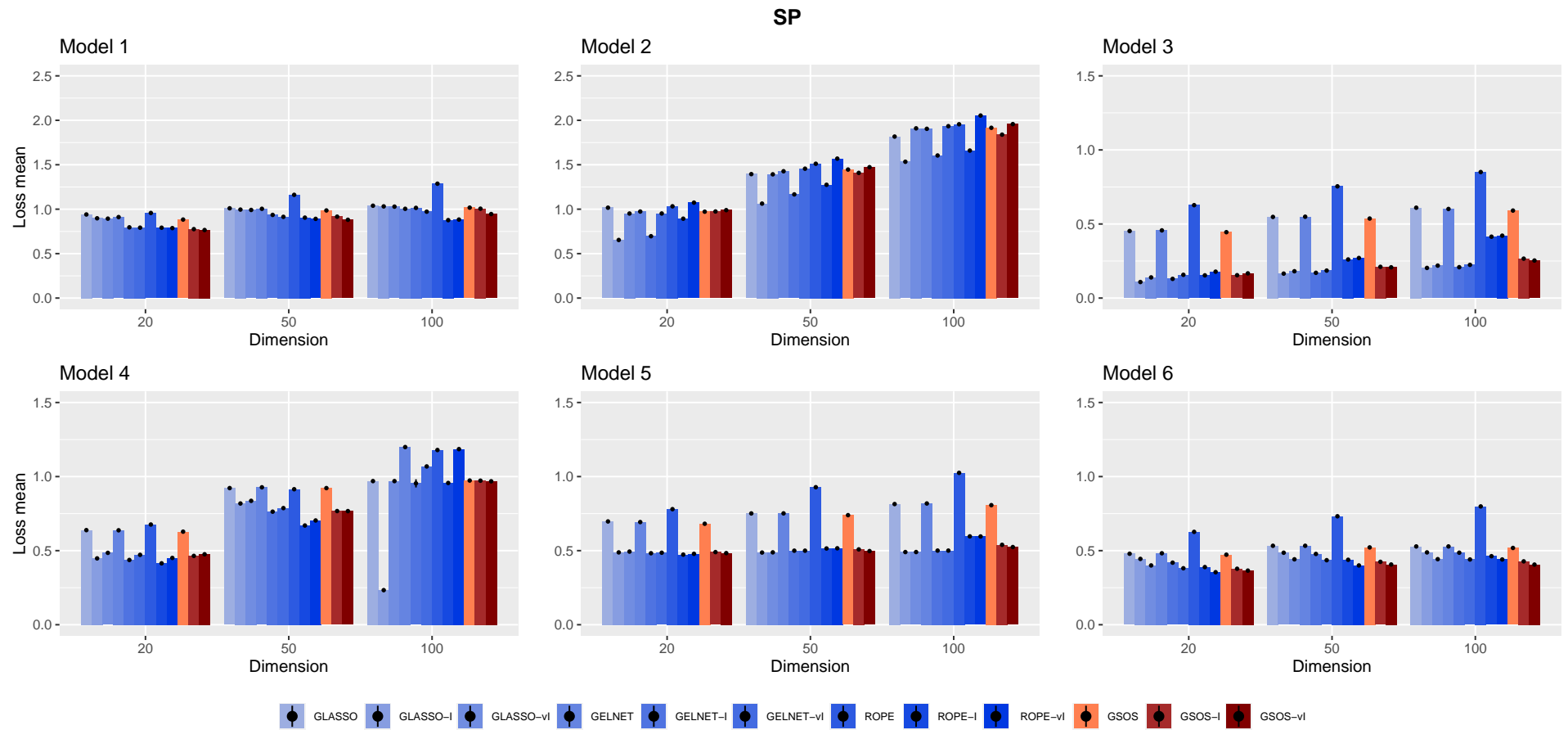
Figure 2. The mean of L_2 loss for the different models and methods based on 100 replications.

Figure 3. The mean of quadratic loss for the different models and methods based on 100 replications.

Figure 4. The mean of spectral norm loss for the different models and methods based on 100 replications.

4. Real Data

In the following section, we study market data provided by the Pacific Exchange Rate Service dataset (PERS), which is available on https://fx.sauder.ubc.ca/, accessed on 13 September 2022. PERS provides daily values of the currencies and commodities priced in various base currencies, known as numeraire. We consider 32 different currencies for the four years between 2018 and 2021. The names of these 32 currencies and their abbreviations are listed in Table 1.

We gather the data initially with the US dollar (USD) as numeraire and then, according to the initial data, calculate all currency exchange rates (for more detail, see [23,24]). After determining all exchange rates, using each of the 32 currencies as a numeraire, we compute the daily log returns of the remaining 31 currencies. We estimate the related covariance and precision matrices based on all possible exchange currencies (496 different exchanges). We apply our proposed estimator GSOS, with the scalar matrix as a target for four-year and annual data separately (the five-fold cross-validation is considered in selecting the tuning parameters). The estimated networks are presented in Figures 5–9; it should be mentioned that we consider 50 of the strongest partial correlations.
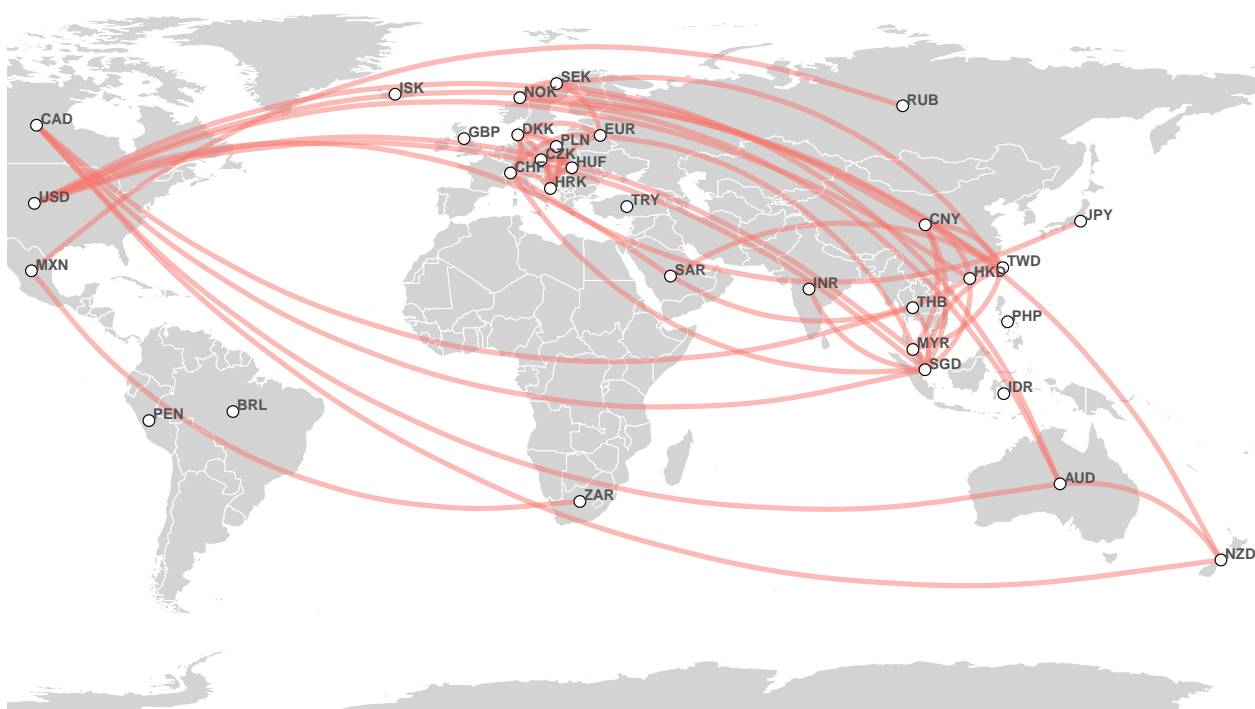

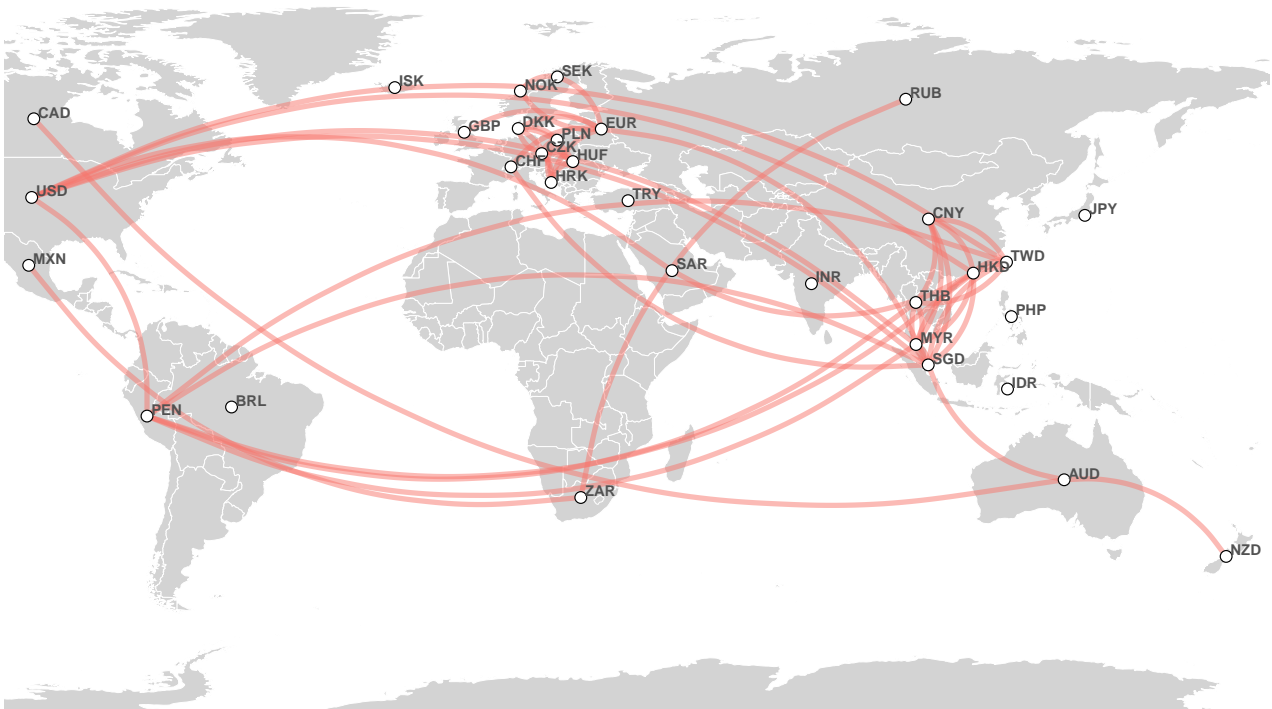**Figure 5.** Estimated currency network based on 2018 to 2021 foreign exchange markets data.

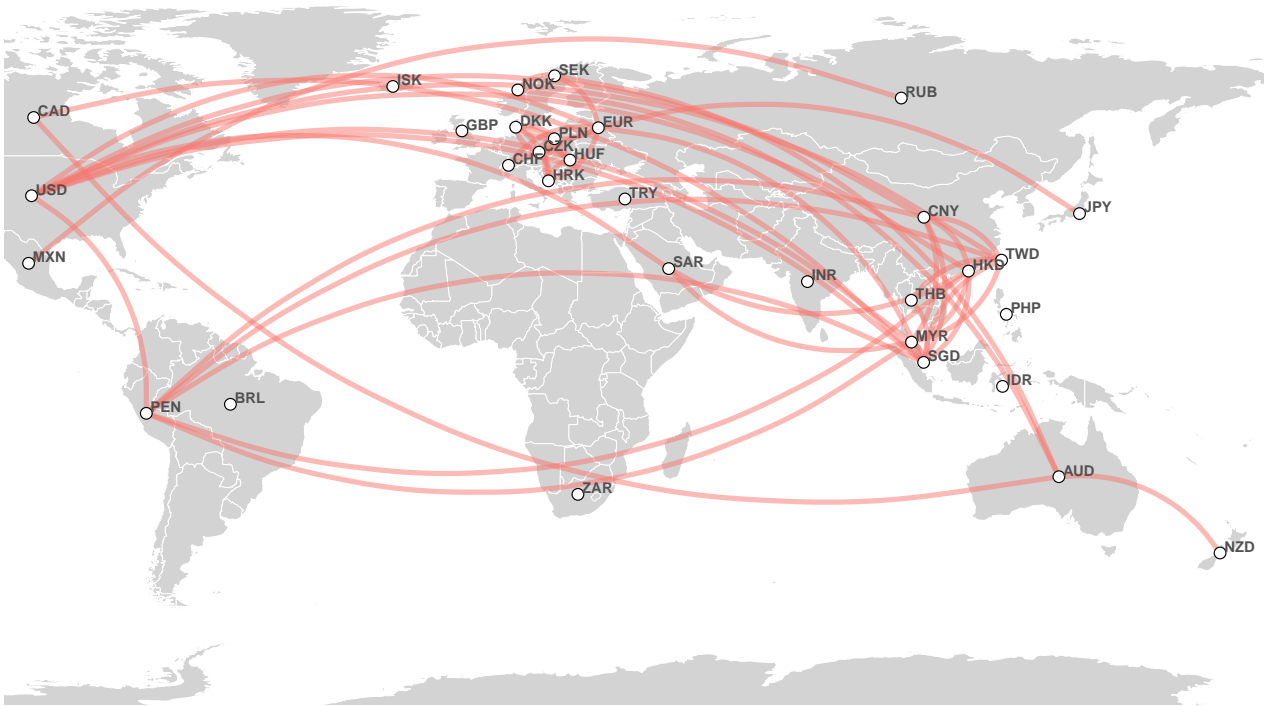**Figure 6.** Estimated currency network based on 2018 foreign exchange markets data.

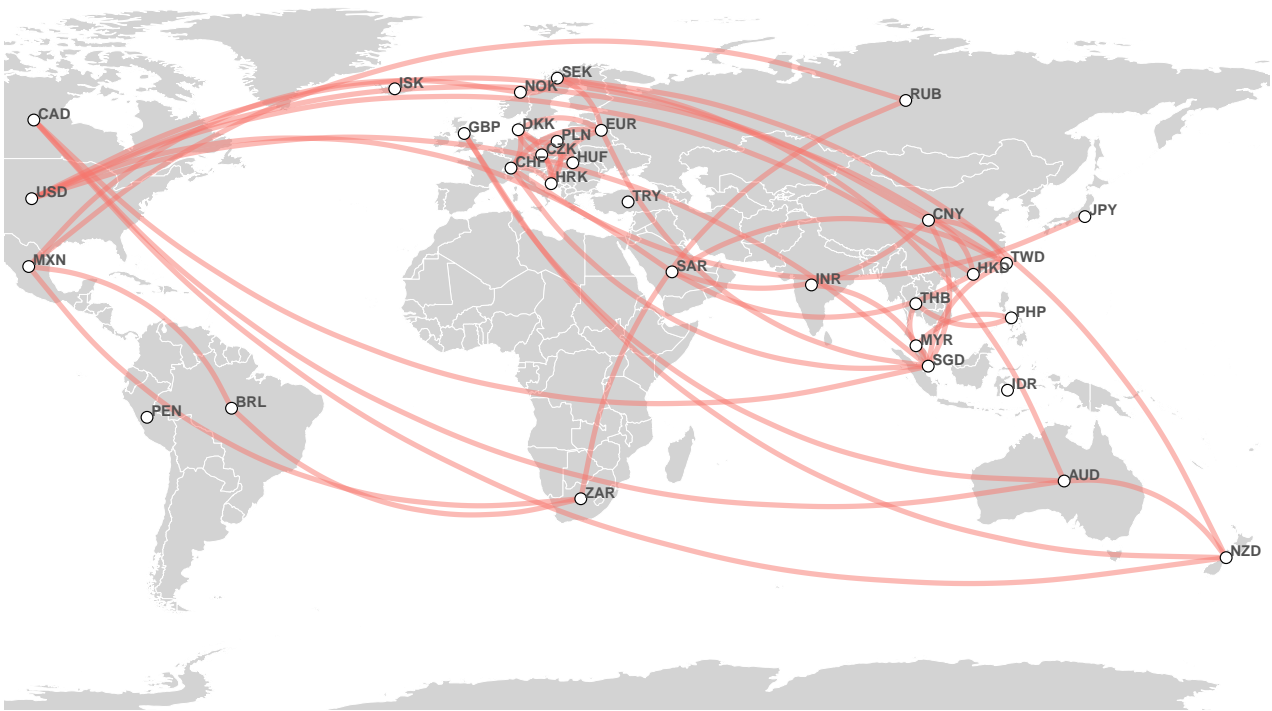**Figure 7.** Estimated currency network based on 2019 foreign exchange markets data.

Figure 8. Estimated currency network based on 2020 foreign exchange markets data.
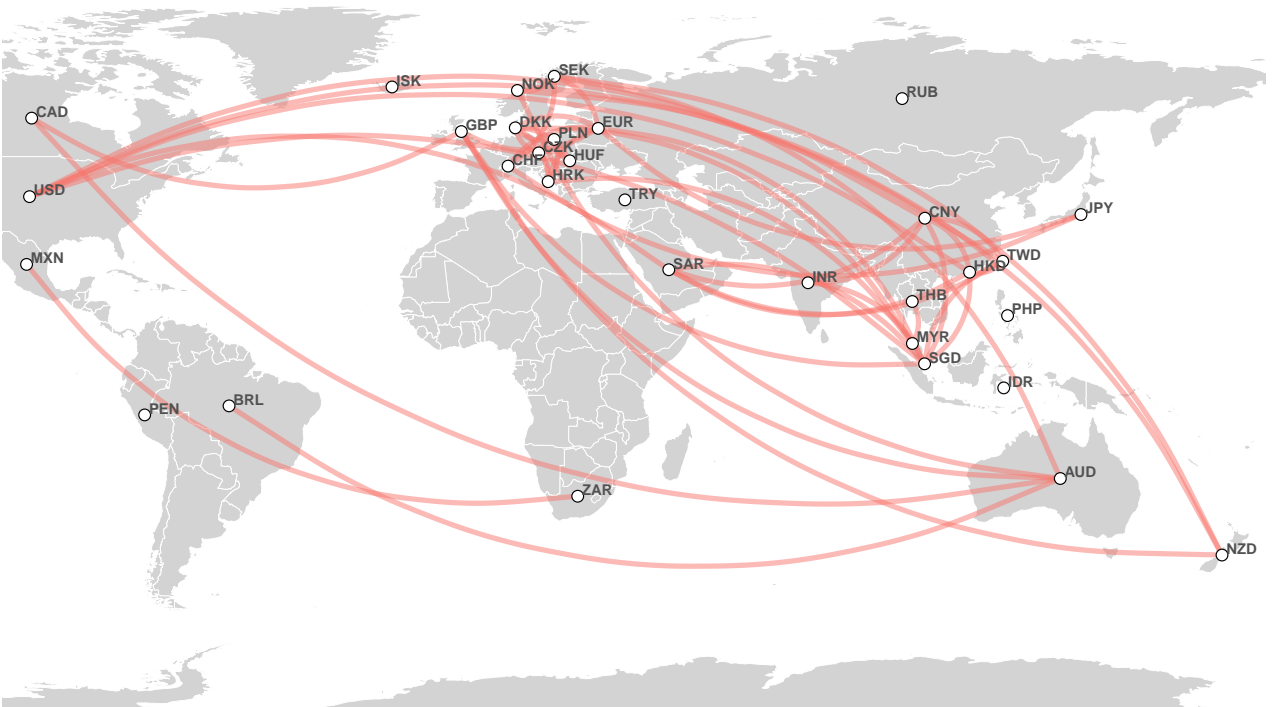

Figure 9. Estimated currency network based on 2021 foreign exchange markets data.

Table 1. List of abbreviations of the considered currencies.

Currencies			
AUD	Australian dollar	MXN	Mexican peso
BRL	Brazilian real	MYR	Malaysian ringgit
CAD	Canadian dollar	NOK	Norwegian krone
CHF	Swiss franc	NZD	New Zealand dollar
CNY	Chinese renminbi	PEN	Peruvian nuevo sol
CZK	Czech koruna	PHP	Philippines peso
DKK	Danish krone	PLN	Polish zloty
EUR	Euro	RUB	Russian ruble
GBP	British pound	SAR	Saudi Arabian riyal
HKD	Hong Kong dollar	SEK	Swedish krona
HRK	Croatian kuna	SGD	Singapore dollar
HUF	Hungarian forint	THB	Thailand baht
IDR	Indonesian rupiah	TRY	Turkish lira
ISK	Icelandic krona	TWD	Taiwanese dollar
INR	Indian rupee	USD	US dollar
JPY	Japanese yen	ZAR	South African rand

Real Data Results

To report the analytical results, we consider the different thresholds for partial correlation to explain the most important relations between considered currencies. We apply filtering by considering only ones that exceed those thresholds to explain how these networks are formed.

- 2018–2021 network: First, strong partial correlations appear between European currencies: DKK, HRK, CZK, PLN, HUF, CHF, and EUR. Additionally, we find a cluster that consists of HKD, SAR, CNY, TWD, SGD, and USD. Considering a smaller threshold, one notices the partial correlation between the Oceania couple of AUD and NZD and CAD, and the joining of EUR to the USD-based cluster, which connects European currencies to this cluster. Finally, a rather interesting cluster centered around a Latin American currency, MXN, appears, which consists of BRL, ZAR, and RUB;

- 2018 network: We observe strong correlations between European currencies: DKK, HRK, CZK, PLN, HUF, and EUR. The USD-based cluster here consists of HKD, PEN, TWD, SGD, CNY, THB, and MYR. The EUR currency connects these two major clusters. We find the relation between the Oceania couple AUD and NZD and CAD for a smaller threshold. In addition, a cluster between the Latin American, African, and Asian currencies MXN, ZAR, and RUB is noteworthy;

- 2019 network: Similar to the 2018 network, we have strong and connected European currencies: DKK, HRK, CZK, PLN, and HUF. The USD-based cluster is formed by HKD, SAR, TWD, SGD, CNY, and MYR at first; by considering a smaller threshold, we find other currencies such as PEN and THB. We observe that EUR connects these two clusters, and another cluster consists of AUD, NZD, NOK, and SEK;

- 2020 network: This year, we observe the USD cluster as follows: HKD, SAR, TWD, SGD, and CNY. We can see MYR, PHP, and THB as a disjointed clique (every two distinct vertices are adjacent). Regarding European currencies, EUR and CHF have a connecting role between this cluster and the USD-based cluster. The European currency cluster consists of DKK, HRK, CZK, PLN, and HUF. In addition, we observe two other clusters: the Oceania couple AUD and NZD with CAD and GBP, and MXN, BRL, ZAR, and RUB. These two clusters have a connection to European currencies by SEK;

- 2021 network: As previously, we observe that the USD-based cluster consists of HKD, SAR, TWD, CNY, and SGD. DKK, HRK, CZK, HUF, PLN, CHF, SEK, and EUR form a European currency cluster, which is connected to the USD cluster by EUR. In addition, we discover an interesting cluster centered around AUD consisting of NZD, CAD, NOK, and GBP. Considering a smaller threshold, we find more relations in the last

cluster and considerable partial correlations between AUD, NZD, NOK, and SEK. This year, we see a strong partial correlation between GBP and SGD, which has never been observed before. Finally, we have a small cluster consisting of three Latin American, Asian, and African currencies: MXN, RUB, and ZAR.

5. Discussion

This section provides a summary of the paper and highlights the results. Additionally, we address some limitations of the research and gaps for future research. Finally, we review the findings and the results of the proposed method: GSOS.

5.1. Summary

In this paper, we proposed two methods to estimate a precision matrix in multivariate Gaussian settings. Estimating a precision matrix is one of the most critical tools for reconstructing an undirected graphical model expressing conditional dependencies between variables. Furthermore, obtaining an estimated network of partial correlations with a suitable rescaling of the precision matrix is possible.

We focused on the high-dimensional case to obtain regularized and sparse estimates considering the elastic net-type penalty. This penalty is a combination of the Frobenius norm when a target matrix is taken into account and the L_1 norm penalty to create sparsity. The first method, Algorithm 1, employs the dpglasso technique proposed by [5] to find penalized log-likelihood estimation. The second method, Algorithm 2, follows a similar approach to glasso, as studied by [1], and solves optimization problems by turning them into quadratic problems.

We conducted a simulation study to test the proposed algorithms using three different sample sizes and six common network structures reflecting various forms of conditional dependence. To calculate the performance measures for all the methods, we considered 100 independent simulations for each network; we presented the average of the Kullback–Leibler, L_2, quadratic, and spectral norm loss functions. The optimal tuning parameters for each method were determined using five-fold cross-validation for each simulation run.

Lastly, we presented an empirical study of the network between 32 currencies for the years between 2018 and 2021. We estimated annual and four-year networks according to these years. We evaluated the high-dimensional precision matrix in every yearly network in this real data study.

5.2. Contributions

We added sparsity to the alternative ridge estimator by considering the L_1 norm as an additional penalty. In this area, [15] also focused on the elastic net-type penalties for a Gaussian log-likelihood-based precision matrix estimation called gelnet. However, they simultaneously considered the target matrix for the L_1 and Frobenius norm. Therefore, we cannot obtain our estimator from gelnet when considering nonzero target matrices. These relations encouraged us to choose glasso, alternative ridge, and gelnet as competitor estimators to compare with our proposed estimator, GSOS.

We used the R programming language [25] and the "GLassoElnetFast" package for the simulation study; the codes are available on https://github.com/Azamkheyri/GSOS.git, accessed on 27 October 2022. In terms of simulation results, GSOS outperformed alternative ridge with most of the underlying structures in the high-dimensional case for all considered sample sizes. GSOS and gelnet, in most cases, behaved almost similarly, but in Network 4 for the Kullback–Leibler and L_2 risk measures, GSOS significantly outperformed gelnet.

On glasso: according to the Kullback–Leibler risk measure, GSOS and glasso performed similarly, except in Networks 1 and 2, where GSOS performed better. For the L_2 measure, GSOS performed better in Network 1, while for the quadratic risk measure, their behaviors are different; in three networks, GSOS outperformed glasso and for the remainder, glasso was better. Therefore, our proposed estimator is an efficient way to estimate precision matrices for the high-dimension Gaussian graphical models.

Finally, we added a real data example, the PERS dataset, and estimated four annual and one four-year dependency networks from 2018 to 2021. Since the log returns must be calculated using data from two consecutive days, we removed from our research those currencies for which there were at least ten missing values for exchange rates with the USD. Therefore, we estimated high-dimension precision matrices with GSOS for 32 currencies worldwide.

5.3. Strengths and Limitations

As we mentioned before, we considered the elastic net-type penalty for the Gaussian log-likelihood problem to estimate the precision matrix. The most important strength of our proposal is that this kind of penalty could help simultaneously obtain the advantages of the Frobenius norm and L_1-penalized estimations. We could not prove that our proposed estimator mathematically outperforms the known methods; this might be considered as a limitation. However, we illustrated the performance of the GSOS estimator with a simulation study under six different frequently used dependency networks in the literature.

Our simulation results are related to the considered network's structure and performance measures; hence, they are not general. Furthermore, despite our proposed estimator's considerable statistical performance, there is no overall winner.

Finally, because of the presence of missing data in the foreign exchange markets and the fact that log returns of the data have to be calculated from two consecutive days, only those currencies in which there were less than ten missing values for exchange rates based on USD were considered.

5.4. Future Work

For future research, we will consider studying the asymptotic behavior of our suggested estimators. In addition, while all the recommended estimators depend on the multivariate Gaussian assumption, we might also consider ways to do away with it by proposing distribution-free approaches or looking at other distributions to capture the data's properties better. The relationship between the values of two tuning parameters—α and λ—might be worthwhile to investigate.

Moreover, the considered real data example has more potential for further study, for instance, considering long-term studies and interpreting the results based on geographic, economic, or political factors.

6. Conclusions

The proposed precision estimator is based on the L_1 penalty for sparsity on Gaussian log-likelihood and penalization with Frobenius norm shrinkage to an arbitrary non-random target value. We proposed two algorithms based on gradient descent (glasso) and the box-constrained quadratic program (dpglasso). Our approach is similar to gelnet by [15], but with some refinement; the estimator proposed here cannot be obtained from gelnet. Compared to other alternatives, the simulation study illustrated that our proposed strategy is a good competitor for high-dimensional precision matrix estimation. Additionally, we presented an empirical analysis of a network of 32 popular currencies and estimated yearly and four-year dependency networks.

Abbreviations

The following abbreviations are used in this manuscript:

GLASSO	Graphical lasso with zero target matrix
GLASSO-I	Graphical lasso with identity target matrix
GLASSO-vI	Graphical lasso with scalar target matrix
GELNET	Graphical elastic net with zero target matrix
GELNET-I	Graphical elastic net with identity target matrix
GELNET-vI	Graphical elastic net with scalar target matrix
ROPE	ROPE with zero target matrix
ROPE-I	ROPE with identity target matrix
ROPE-vI	ROPE with scalar target matrix
GSOS	GSOS with zero target matrix
GSOS-I	GSOS with identity target matrix
GSOS-vI	GSOS with scalar target matrix

Appendix A

Figure A1 shows the precision matrices of the considered networks in the simulation study.
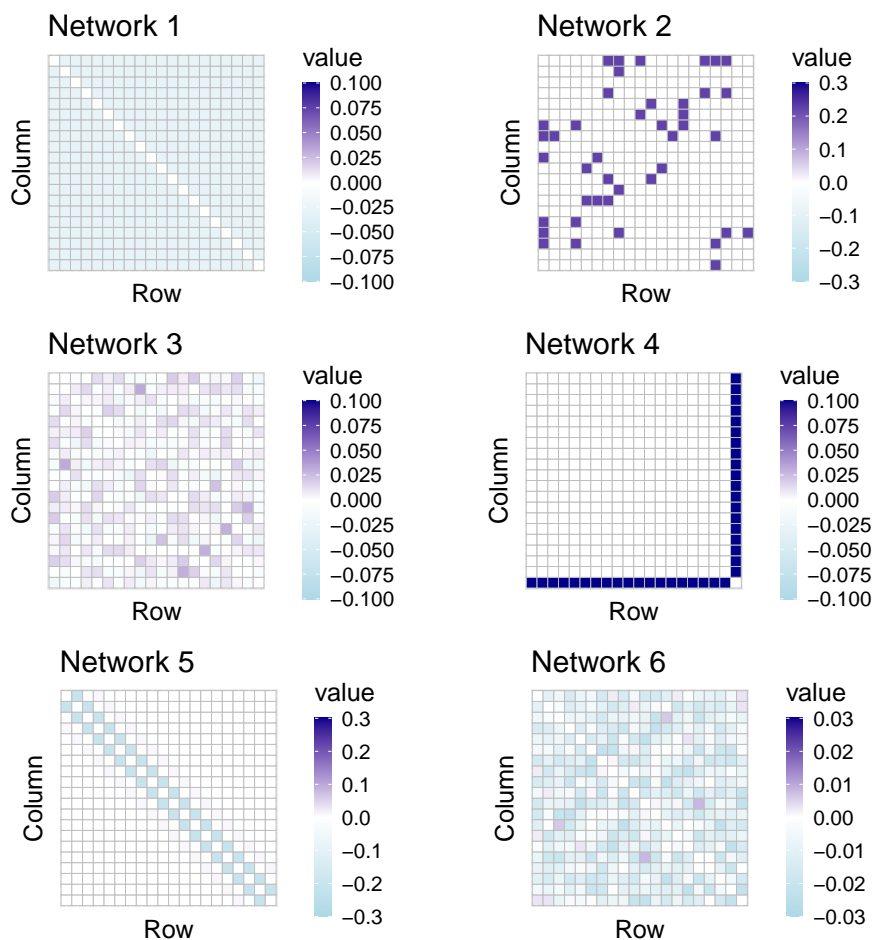
Figure A1. Precision matrices (diagonal elements ignored).

References

1. Friedman, J.; Hastie, T.; Tibshirani, R. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* **2008**, *9*, 432–441. [CrossRef] [PubMed]
2. Fan, J.; Feng, Y.; Wu, Y. Network exploration via the adaptive LASSO and SCAD penalties. *Ann. Appl. Stat.* **2009**, *3*, 521. [CrossRef] [PubMed]
3. Bien, J.; Tibshirani, R.J. Sparse estimation of a covariance matrix. *Biometrika* **2011**, *98*, 807–820. [CrossRef] [PubMed]
4. Witten, D.M.; Friedman, J.H.; Simon, N. New insights and faster computations for the graphical lasso. *J. Comput. Gr. Stat.* **2011**, *20*, 892–900. [CrossRef]
5. Mazumder, R.; Hastie, T. The graphical lasso: New insights and alternatives. *Electron. J. Stat.* **2012**, *6*, 2125. [CrossRef]
6. Danaher, P.; Wang, P.; Witten, D.M. The joint graphical lasso for inverse covariance estimation across multiple classes. *J. R. Stat. Soc. Ser. B (Statist. Methodol.)* **2014**, *76*, 373–397. [CrossRef]
7. Avagyan, V.; Alonso, A.M.; Nogales, F.J. Improving the graphical lasso estimation for the precision matrix through roots of the sample covariance matrix. *J. Comput. Graph. Stat.* **2017**, *26*, 865–872. [CrossRef]
8. Warton, D.I. Penalized normal likelihood and ridge regularization of correlation and covariance matrices. *J. Am. Stat. Assoc.* **2008**, *103*, 340–349. [CrossRef]
9. Van Wieringen, W.N.; Peeters, C.F. Ridge estimation of inverse covariance matrices from high-dimensional data. *Comput. Stat. Data Anal.* **2016**, *103*, 284–303. [CrossRef]
10. van Wieringen, W.N. The generalized ridge estimator of the inverse covariance matrix. *J. Comput. Graph. Stat.* **2019**, *28*, 932–942. [CrossRef]
11. Kuismin, M.; Kemppainen, J.; Sillanpää, M. Precision matrix estimation with ROPE. *J. Comput. Graph. Stat.* **2017**, *26*, 682–694. [CrossRef]
12. Rothman, A.J. Positive definite estimators of large covariance matrices. *Biometrika* **2012**, *99*, 733–740. [CrossRef]
13. Atchadé, Y.F.; Mazumder, R.; Chen, J. Scalable computation of regularized precision matrices via stochastic optimization. *arXiv* **2015**, arXiv:1509.00426.
14. Bernardini, D.; Paterlini, S.; Taufer, E. New estimation approaches for graphical models with elastic net penalty. *arXiv* **2021**, arXiv:2102.01053.
15. Kovács, S.; Ruckstuhl, T.; Obrist, H.; Bühlmann, P. Graphical Elastic Net and Target Matrices: Fast Algorithms and Software for Sparse Precision Matrix Estimation. *arXiv* **2021**, arXiv:2101.02148.
16. Yuan, M.; Lin, Y. Model selection and estimation in the Gaussian graphical model. *Biometrika* **2007**, *94*, 19–35. [CrossRef]
17. Banerjee, O.; El Ghaoui, L.; d'Aspremont, A. Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data. *J. Mach. Learn. Res.* **2008**, *9*, 485–516.
18. Yuan, M. Efficient computation of $\ell 1$ regularized estimates in Gaussian graphical models. *J. Comput. Graph. Stat.* **2008**, *17*, 809–826. [CrossRef]
19. Guo, J.; Levina, E.; Michailidis, G.; Zhu, J. Joint estimation of multiple graphical models. *Biometrika* **2011**, *98*, 1–15. [CrossRef]
20. Shan, L.; Kim, I. Joint estimation of multiple Gaussian graphical models across unbalanced classes. *Comput. Stat. Data Anal.* **2018**, *121*, 89–103. [CrossRef]
21. Londschien, M.; Kovács, S.; Bühlmann, P. Change-point detection for graphical models in the presence of missing values. *J. Comput. Graph. Stat.* **2021**, *30*, 768–779. [CrossRef]
22. Boyd, S.; Boyd, S.P.; Vandenberghe, L. *Convex Optimization*; Cambridge University Press: Cambridge, UK, 2004.
23. Basnarkov, L.; Stojkoski, V.; Utkovski, Z.; Kocarev, L. Correlation patterns in foreign exchange markets. *Phys. A Stat. Mech. Its Appl.* **2019**, *525*, 1026–1037. [CrossRef]
24. Fenn, D.J.; Porter, M.A.; Mucha, P.J.; McDonald, M.; Williams, S.; Johnson, N.F.; Jones, N.S. Dynamical clustering of exchange rates. *Quant. Financ.* **2012**, *12*, 1493–1520. [CrossRef]
25. R Core Team. *R: A Language and Environment for Statistical Computing*; R Foundation for Statistical Computing: Vienna, Austria, 2022.