RANKING EDGES AND MODEL SELECTION IN HIGH-DIMENSIONAL GRAPHS

Ginette Lafit[a], Francisco J. Nogales[a] and Ruben H. Zamar[b]

Abstract

In this article we present an approach to rank edges in a network modeled through a Gaussian Graphical Model. We obtain a path of precision matrices such that, in each step of the procedure, an edge is added. We also guarantee that the matrices along the path are symmetric and positive definite. To select the edges, we estimate the covariates that have the largest absolute correlation with a node conditional to the set of edges estimated in previous iterations. Simulation studies show that the procedure is able to detect true edges until the sparsity level of the population network is recovered. Moreover, it can add efficiently true edges in the first iterations avoiding to enter false ones. We show that the top-rank edges are associated with the largest partial correlated variables. Finally, we compare the graph recovery performance with that of Glasso under different settings.

Keywords: High-dimensional statistics; Precision Matrix; Covariance selection; Gaussian Graphical Models; Edge Ranking; Least Angle Regression.

[a] Department of Statistics, Universidad Carlos III de Madrid.
[b] Department of Statistics, University of British Columbia.

Corresponding author: The first author.
Email address: glafit@est-econ.uc3m.es

Ranking Edges and Model Selection in High-Dimensional Graphs

Ginette Lafit[*,†] , Francisco J. Nogales[†] and Ruben H. Zamar[‡]

Department of Statistics, Universidad Carlos III de Madrid[†] and
Department of Statistics, University of British Columbia[‡]

Abstract: In this article we present an approach to rank edges in a network modeled through a Gaussian Graphical Model. We obtain a path of precision matrices such that, in each step of the procedure, an edge is added. We also guarantee that the matrices along the path are symmetric and positive definite. To select the edges, we estimate the covariates that have the largest absolute correlation with a node conditional to the set of edges estimated in previous iterations. Simulation studies show that the procedure is able to detect true edges until the sparsity level of the population network is recovered. Moreover, it can add efficiently true edges in the first iterations avoiding to enter false ones. We show that the top-rank edges are associated with the largest partial correlated variables. Finally, we compare the graph recovery performance with that of Glasso under different settings.

Keywords: High-dimensional statistics; Precision Matrix; Covariance Selection; Gaussian Graphical Models; Edge Ranking; Least Angle Regression.

1 Introduction

Consider the problem of estimating high-dimensional undirected graphs. Given n independent samples of a p-dimensional random vector $(X_1, \ldots, X_p)$, we can

represent the linear dependency between variables by an undirected graph. The undirected graph establishes that if the variables X_i and X_j are connected, they are adjacent (Lauritzen, 1996). Statistically, we can measure linear dependencies by estimating partial correlations to infer whether there is an association between a pair of variables, conditionally to the rest of them. Furthermore, we can relate the nonzero entries in the precision matrix, denoted by Ω, with the nonzero partial correlation coefficients (Edwards, 2000). This procedure is known as covariance selection and it is widely used to identify the linear relations that may appear in a set of random variables (Dempster, 1972). In particular, under a Gaussian distribution, the nonzero entries of the precision matrix imply that each pair of variables are conditional dependent when controlling for the rest of them (Lauritzen, 1996). This is known in the literature as Gaussian Graphical Models (GGM).

In a high-dimensional framework, the estimation of Ω is not straightforward because of the lack of a pivotal estimator like the empirical covariance matrix. Moreover, when the dimension p is larger than the number of available observations, the sample covariance matrix is not invertible. And even when the ratio p/n is approximately (but less than) one, the sample covariance matrix is bad conditioned and its inverse tends to amplify the estimation error, which can be observed by the presence of small eigenvalues (Ledoit and Wolf, 2004). From the asymptotic point of view, when both n and p are large (i.e. $p = O(n)$), the sample covariance matrix is not a consistent estimator (Karoui, 2008).

To deal with this problem, several covariance selection procedures have been proposed based on the assumption that Ω is mostly composed by zero elements. This suggests that even when $p = O(n)$, the dimension of the problem may be still tractable since the number of edges will grow slower than the number of observations (Peng et al., 2009). Meinshausen and Bühlmann (2006) propose the neighborhood selection procedure that consistently estimate sparse high-dimensional graphs, by estimating a lasso regression for each node in the graph. This method does not directly compute the precision matrix and imposes the same ℓ_1 penalty to each node. Peng et al. (2009) present a procedure that simultaneously perform neighborhood selection for all variables to estimate joint sparse regressions applying an active-shooting to solve lasso. Yuan (2010) replace the lasso regression by a Dantzig selector. Liu and Wang (2012) proposed an asymptotically tuning-free procedure that estimates the precision matrix in a column-by-column fashion.

ℓ_1 penalized likelihood methods to estimate GGM were also proposed (Yuan and Lin, 2007; Banerjee et al., 2008; Friedman et al., 2008). Friedman et al. (2008) proposed the Glasso procedure to estimate sparse precision matrices fitting a modified lasso regression to each variable and solving the problem by coordinate descent algorithm. Rothman et al. (2008) estimate convergence rates under the Frobeniuos norm and Yuan and Lin (2007) estimate convergence rates for subgaus-

sian distributions. Lam and Fan (2009) and Fan et al. (2009) propose methods to diminish the bias imposed by the ℓ_1 penalty by introducing a non-convex SCAD penalty. Cai et al. (2011) estimate precision matrices for both sparse and non-sparse matrices, without imposing a specific sparsity pattern solving the dual of an ℓ_1 penalized maximum likelihood problem.

In this article, we propose an approach to perform model selection in high-dimensional undirected graphs. Our main interest is in obtaining the ranking of edges that are most probable in a GGM. Thus, we are able to estimate a path of undirected graphs and identify which are the edges that will appear firstly to obtain a ranking of the top partially correlated variables. Hence, the procedure estimates a path of precision matrices such that in each step an edge is added. As a consequence, in each step we obtain a precision matrix with a pair of additional non-zero elements outside its principal diagonal.

For the variable associated with a node, an edge is selected by estimating the covariate that has the largest absolute correlation conditional to the set of edges obtained in the previous steps. To perform this task, we apply Least Angle Regression (LARS) proposed by Efron et al. (2004) which is a stylized version of Foward Stagewise Linear Regression, and is closely related with Foward Selection (see Weisberg, 2005) and Lasso (Tibshirani, 1996). LARS provides an ordering in which the covariates enter a regression model. It starts with all coefficients equal to zero, and finds the covariate most correlated with the response. In the next steps, the additional covariate is added by moving in an equiangular direction between the active predictors until a new variable has as much correlation with the current residual. In our procedure, we only estimate the first LARS step, this is equivalent to estimate a Foward Stagewise Linear Regression, in which the variable that enters the correlation set is the one that produces the highest absolute correlation with the residual.

To illustrate the idea, consider the following example from Sachs et al. (2005) that contains data of a flow cytometry with $p = 11$ proteins and $n = 7466$ cells. The authors estimated a directed acyclic graph (DAG) to the data and found active 17 edges. Figure 1 shows the result of applying the proposed iterative procedure to rank edges. In the initial step, Step 0, all partial correlations are equal to zero. Then, at each iteration, the procedure estimates an edge which is incorporated to a set of active edges. Only $\frac{p \times (p-1)}{2}$ iterations are necessary to obtain a full connected graph. The first edges that enter the graph evidence the largest partial correlations. As shown in Figure 1, at Step 17, there are 10 edges that agree with the directed acyclic graph estimated by Sachs et. al. (Sachs et al., 2005).
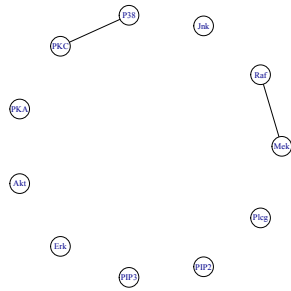
The rest of the article is organized as follows. In the next section we present the iterative procedure to estimate the GGM, which exploits the relation between the elements of the precision matrix and the partial correlations. We also present
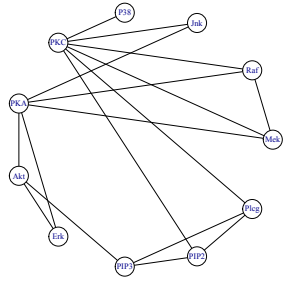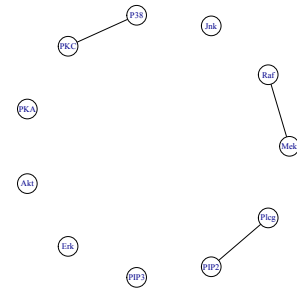
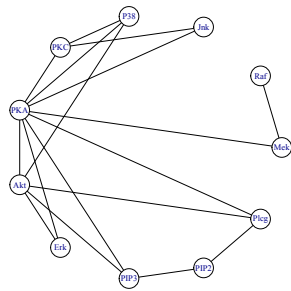(a) Directed acylic graph from cell-signaling data (Sachs et al., 2005)
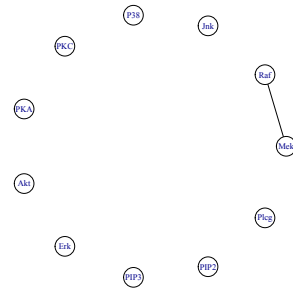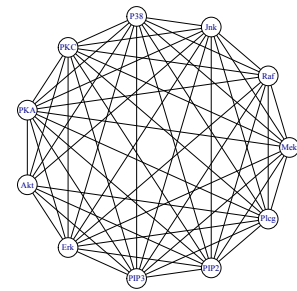
(b) Step 1

(c) Step 2

(d) Step 3

(e) Step 17

(f) Step 55 (Full connected undirected graph)

Figure 1: Cell-signaling data: undirected graphs. Steps corresponds to the number of active edges in each iteration of the procedure.

modifications to ensure that, in each step, the associated estimation of the precision matrix is symmetric and definite positive. In section 3, we show simulation results and introduce measures to evaluate the ranking performance of the procedure, and we also compare the classification performance of the method with that of Glasso. Finally, we provide further discussions on the connections and differences between our results and existing methods and possible future directions.

2 The proposed method

In this section, we describe an iterative method to rank edges in a Gaussian Graphical Model (GGM). We use the following notation. Suppose that $\mathbf{X} = (X_1, \ldots, X_p)^T \in \mathbb{R}^p$ has a joint normal distribution with mean 0 and covariance Σ_p. The conditional independence structure of the distribution can be represented by a graphical model $\mathcal{G} = (\Gamma, E)$ where $\Gamma = \{1, \ldots, p\}$ represents the set of nodes, and E the set of edges in $\Gamma \times \Gamma$. A pair (i, j) is contained in E if and only if X_i is conditionally dependent to X_j, given all the remaining variables $X_{-(i,j)} = \{X_r : 1 \leq r \neq i, j \leq p\}$. Let $\Omega = \Sigma^{-1}$ denotes the precision matrix. If a pair of variables is not contained in E, then they are conditionally independent, given all the remaining variables, and corresponds to a zero entry in Ω (Lauritzen, 1996; Meinshausen and Bühlmann, 2006). Finally, we denote by ρ_{ij} ($1 \leq i < j \leq p$) the partial correlation between X_i and X_j, which is defined as the correlation between ϵ_i and ϵ_j, where ϵ_i and ϵ_j are the prediction errors of the best linear predictors of X_i and X_j based on $X_{-(i,j)}$ respectively.

The following Lemma (see Peng et al., 2009) relates the estimation of the elements of the precision matrix to a regression problem.

Lemma 1. *For $1 \leq i \leq p$, X_i is expressed as $X_i = \sum_{j \neq i} \beta_{ij} X_j + \epsilon_i$, such that ϵ_i is uncorrelated with $X_{-i} = \{X_r : 1 \leq r \neq i \leq p\}$ if and only if*

$$\beta_{ij} = -\frac{\omega_{ij}}{\omega_{ii}} = \rho_{ij} \sqrt{\frac{\omega_{jj}}{\omega_{ii}}}, \tag{2.1}$$

where $Var(\epsilon_i) = \frac{1}{\omega_{ii}}$, $Cov(\epsilon_i, \epsilon_j) = \frac{\omega_{ij}}{\omega_{ii}\omega_{jj}}$, and $\rho_{ij} = -\frac{\omega_{ij}}{\sqrt{\omega_{ii}\omega_{jj}}}$.

The proposed iterative procedure to rank the edges of the GGM is as follows. Given an independently and identically distributed distributed random sample $\mathbf{X} = \{\mathbf{X}_1, \ldots, \mathbf{X}_n\}$ from the distribution of $\mathbf{X}$. Define the empirical covariance matrix as

$$S = \frac{1}{n} \sum_{t=1}^{n} \mathbf{X}_t \mathbf{X}_t^T \tag{2.2}$$

5

Let $k \in \{0, 1, \ldots, K\}$ be the number of possible edges of a GGM with $K \leq \frac{p \times (p-1)}{2}$, and let $\Omega^{(k)}$ be the corresponding proposed precision matrix at iteration k. The procedure starts with a graph where the set of edges E is empty and hence, the initial precision matrix ($\Omega^{(0)}$), is a diagonal matrix where its elements are the inverse of the diagonal elements of S.

In the next step, $k = 1$, the procedure estimates an undirected graph with $k + 1$ edges. The optimal edge is selected by estimating p matrices, denoted by $\Omega_i^{(k+1)}$ ($1 \leq i \leq p$), each of them having an additional pair of non-zero elements outside its principal diagonal corresponding to variable X_i. These non-zero elements are estimated applying the results in Lemma 1. The additional covariate, X_j, is selected in a way that it has the largest absolute correlation with variable X_i conditional to the set of edges selected in the previous step k. Thus, in iteration $k + 1$, the estimated p matrices satisfy the following recursive expression:

$$\Omega_i^{(k+1)} = \hat{\Omega}^{(k)} + \alpha_k D_i^{(k+1)}, \quad i = 1, \ldots, p \tag{2.3}$$

where $D_i^{(k+1)}$ contains the pair of additional non-zero elements, ω_{ij} and ω_{ji}, and α_k is a parameter that makes $\Omega_i^{(k+1)}$ positive definite (see Subsection 2.3).

Given the previous set of p positive-definite matrices, the proposed precision matrix at iteration k, $\hat{\Omega}^{(k)}$, is selected by minimizing a negative log-likelihood function, i.e.

$$\hat{\Omega}^{(k)} = \underset{\Omega_i^{(k)} \succ 0}{\operatorname{argmin}} \left\{ -\log(\det(\Omega_i^{(k)})) + \operatorname{tr}(\Omega_i^{(k)} S). \right\} \tag{2.4}$$

As an illustration, consider an undirected graph with $p = 4$ variables and the following AR(1) structure for Ω: $\omega_{ii} = 1$, $\omega_{i,i+1} = \omega_{i-1,i} = 0.5$ and 0 otherwise. We apply the iterative procedure to simulated data with $n = 100$. Equation 2.5 shows two consecutive iterations. In step $k = 2$, we estimate a positive definite matrix, given by $\hat{\Omega}^{(k=2)}$, that represents a GGM with two edges. In the next step (i.e. $k = 3$) an additional edge is added and $\hat{\Omega}^{(k=3)}$ will have a new pair of non-zero elements outside its principal diagonal, following the expression in 2.3, we obtain a positive definite matrix that represents a GGM with three edges.

$$\hat{\Omega}^{(k=3)} = \begin{pmatrix} 1.03 & 0.64 & 0.00 & 0.00 \\ 0.64 & 1.23 & 0.51 & 0.00 \\ 0.00 & 0.51 & 1.09 & \mathbf{0.44} \\ 0.00 & 0.00 & \mathbf{0.44} & 1.10 \end{pmatrix} = \tag{2.5}$$

$$\begin{pmatrix} 1.04 & 0.64 & 0.00 & 0.00 \\ 0.64 & 1.23 & 0.58 & 0.00 \\ 0.00 & 0.58 & 0.91 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.74 \end{pmatrix} + \begin{pmatrix} 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & \mathbf{0.44} \\ 0.00 & 0.00 & \mathbf{0.44} & 0.00 \end{pmatrix}$$

In the next subsections, we explain in detail how we perform the estimation of the additional edge that enters the precision matrix in step $k+1$, and how we guarantee that, in each step, the proposed precision-matrix estimates are symmetric and positive definite.

2.1 Estimation of the additional edge

In this subsection we show how we compute the elements of matrix $D_i^{(k+1)}$, that contains the pair of additional non-zero elements ω_{ij} and ω_{ji}.

Assuming we are at iteration $k+1$, an additional edge is added to $\hat{\Omega}^{(k)}$. For a variable X_i $(1 \leq i \leq p)$ we select the covariate applying a foward stagewise linear regression. This is equivalent to estimate the first LARS step, in which the variable that enters the correlation set is the one that produces the largest absolute correlation with the current residual (see Efron et al., 2004).

Given variable X_i, we define the active set of indexes $\mathcal{A}_i^{(k)} \subseteq \{1, 2, \ldots, i-1, \ldots, i+1, p\}$ corresponding to covariates with the largest absolute correlations and the matrix that contains these covariates

$$\mathbf{X}_{\mathcal{A}_i^{(k)}} = (\cdots X_r \cdots)_{r \in \mathcal{A}_i^{(k)}}$$

To select the index of the covariate that will enter the active set $\mathcal{A}_i^{(k)}$, we select the variable that has the largest absolute correlation with the current residual $\hat{c}_i^{(k)}$, given by

$$\hat{c}_i^{(k)} = X_i - \sum_{r \in \mathcal{A}_i^{(k)}} \hat{\beta}_{ir} X_r \tag{2.6}$$

At iteration $k+1$, let $j \notin \mathcal{A}_i^{(k)}$ be the index of the new variable X_j that enters the active set . We update $\mathcal{A}_i^{(k+1)} = \mathcal{A}_i^{(k)} \cup \{j\}$ and the variable X_j is added to matrix $\mathbf{X}_{\mathcal{A}_i^{(k)}}$. To estimate the new non-zero element of $\Omega_i^{(k+1)}$, we apply Lemma 1

$$X_i = \sum_{r \in \mathcal{A}_i^{(k+1)}} \beta_{ir} X_r + \epsilon_i, \tag{2.7}$$

We compute the additional non-zero element as $\omega_{ij} = -\frac{\beta_{ij}}{\text{Var}(\epsilon_i)}$ and update the diagonal element $\omega_{ii} = \frac{1}{\text{Var}(\epsilon_i)}$.

We summarize the main result the following proposition.

Proposition 1. *At iteration k of the proposed framework, if $\Omega^{(k)}$ denotes the corresponding GGM with k edges, the for every $i = 1, \ldots, p$, the position of the additional non-zero elements outside its principal diagonal corresponds to the variable*

that has the largest absolute correlation with the current residual $\hat{c}_i^{(k)}$ of variable X_i.

2.2 Symmetrization

For the new variable that enters the equicorrelation set X_j, we set $\omega_{ji} = \omega_{ij}$ and update $\mathcal{A}_j^{(k+1)} = \mathcal{A}_j^{(k)} \cup \{i\}$. Then, we estimate a regression where the covariates are given by the variables that belong to the equicorrelation set of X_j in step k and variable X_i

$$X_j = \sum_{r \in \mathcal{A}_j^{(k+1)}} \beta_{jr} X_r + \epsilon_j \tag{2.8}$$

the element that corresponds to the diagonal of X_j is computed as the inverse of the variance of the regression residuals $\omega_{jj} = \frac{1}{\text{Var}(\epsilon_j)}$.

2.3 Positive Definiteness

The iteration law of the proposed framework has the following form

$$\Omega_i^{(k+1)} = \Omega^{(k)} + \alpha_k D_i^{(k+1)},$$

where α_k is a positive parameter that guarantees $\Omega_i^{(k+1)}$ is positive definite for all $i = 1, \ldots, p$. We propose the following approach to guarantee this property.

Given $\Omega_i^{(k+1)} = \Omega^{(k)} + \alpha_k D_i^{(k+1)}$ that contains the new pair of non-zero elements ω_{ij} and ω_{ji}. Define $\lambda_{min}^{(k+1)}$ to be the minimum eigenvalue of

$$\Omega^{(k)} + D_i^{(k+1)} \tag{2.9}$$

Given a tolerance $\varepsilon > 0$, if $\lambda_{min}^{(k+1)} < \varepsilon$ we have to correct equation 2.9 to be positive definite.

We propose to add a scalar $\alpha_k > 0$ to the elements in the principal diagonal of $\Omega_i^{(k+1)}$. We select α_k such that the minimum eigenvalue of $\Omega_i^{(k)}$ is at least equal to the tolerance ε.

$$\alpha_k = \varepsilon - \lambda_{min}^{(k+1)} \tag{2.10}$$

As a result $\lambda_{min}^{(k+1)} \geq \varepsilon$.

The following proposition summarizes these result.

Proposition 2. *In each step a positive definite matrix is estimated. Given a matrix $\Omega_i^{(k+1)}$ and a positive tolerance ε. It is possible to apply a penalization $\alpha_k > 0$ to the elements of the principal diagonal such that the minimum eigenvalue $\lambda_{min}^{(k+1)} \geq \varepsilon$.*

2.4 Summary of proposed framework

We finish the Section 2 summarizing the algorithm to estimate the precision-matrix path:

Algorithm

1. Start with $k = 0$ edges and set $\Omega^{(0)} = \text{diag}(\frac{1}{s_{11}}, \ldots, \frac{1}{s_{pp}})$, where s_{ii} denotes the corresponding diagonal values of the empirical covariance matrix, S.

2. Estimate p matrices, $\Omega_i^{(k+1)}$ $(1 \leq i \leq p)$, each of them associated to variable X_i, respectively, and containing an additional pair of non-zero elements outside its principal diagonal. The computation for each matrix satisfies the following recursive expression:

$$\Omega_i^{(k+1)} = \Omega^{(k)} + \alpha_k D_i^{(k+1)}$$

where $D_i^{(k+1)}$ contains the pair of additional elements ω_{ij} and ω_{ji} and α_k is the positive parameter that makes $\Omega_i^{(k+1)}$ positive definite (see equation 2.10).

The additional non-zero element ω_{ij} associated with variable X_i is estimated as follows,

2.1 Given the active set of indexes $\mathcal{A}_i^{(k)}$ associated with variable X_i and the matrix of active covariates $\mathbf{X}_{\mathcal{A}_i^{(k)}}$. let $j \notin \mathcal{A}_i^{(k)}$ be the index of the new variable X_j that has the largest absolute correlation with the current residual $\hat{c}_i^{(k)} = X_i - \sum_{r \in \mathcal{A}_i^{(k)}} \hat{\beta}_{ir} X_r$.

2.2 Update the active set of indexes $\mathcal{A}_i^{(k+1)} = \mathcal{A}_i^{(k)} \cup \{j\}$ and add the variable X_j to matrix $\mathbf{X}_{\mathcal{A}_i^{(k)}}$

2.3 Apply Lemma 1 to estimate the new non-zero element as $\omega_{ij} = -\frac{\beta_{ij}}{\text{Var}(\epsilon_i)}$ and update the diagonal element $\omega_{ii} = \frac{1}{\text{Var}(\epsilon_i)}$.

3. Among all the p matrices, $\Omega_i^{(k+1)}$ $(1 \leq i \leq p)$, select the optimal one that minimizes the negative log-likelihood function in equation 2.4.

4. Obtain the corresponding representation of the GGM with $k+1$ edges, $\Omega^{(k+1)}$, set $k := k+1$, and repeat Steps 2 and 3 until $K \leq \frac{p \times (p-1)}{2}$ edges are estimated.

3 Simulation experiments

In this section we present simulation experiments to examine the performance of the proposed method to rank edges in high-dimensional GGMs. We first investigate the ranking performance of our procedure. Then, we compare the classification performance with that of Glasso (Friedman et al., 2008).

We consider four different specifications for the population precision-matrix Ω:

- AR(1) Model: $\omega_{ii} = 1$, $\omega_{i,i+1} = \omega_{i-1,i} = 0.5$ and 0 otherwise.

- AR(2) Model: $\omega_{ii} = 1$, $\omega_{i,i+1} = \omega_{i-1,i} = 0.4$, $\omega_{i,i+2} = \omega_{i-2,i} = 0.2$ and 0 otherwise

- AR(3) Model: $\omega_{ii} = 1$, $\omega_{i,i+1} = \omega_{i-1,i} = 0.4$, $\omega_{i,i+2} = \omega_{i-2,i} = 0.2$, $\omega_{i,i+3} = \omega_{i-3,i} = 0.2$ and 0 otherwise

- Hub Model: 16 hub nodes are randomly choose, each of them connects with 5 distinct nodes with $w_{ij} = 0.2$. Nodes not associated with the hub nodes are set are not considered. The diagonal is $\omega_{ii} = 1$.

Table 1 shows the sparsity level for each model, given as the proportion of non-zero elements outside the diagonal over the total number of elements of the network. We observe that the Hub structure increases its sparsity level more rapidly as p increases compared with the others models. This is due to the fact that the number of edges do not depend on the size of the network. We also report in Table 1 the associated condition numbers for each model. Note these numbers are moderate except for the AR(1) model.

Next, we draw n independent samples from a multivariate normal distribution with mean 0 and covariance matrix $\Sigma = \Omega^{-1}$. We fix $n = 100$ and consider different values of $p = \{30, 60, 90, 120\}$. We replicate each simulation experiment 100 times. To make the precision matrix positive definite we add in each iteration a positive scalar in order to make the minimum eigenvalue greater than a given tolerance. The tolerance was defined in a decreasing logarithm scale.

The values of the maximum iteration K were set to account for the sparse structure of the four proposed models. For $p = 30$, $p = 60$, $p = 90$ and $p = 120$, K was set recover a sparsity level of 45%, 30%, 20% and 10% respectively.

In the next section we introduce performance measures to obtain the ranking of most probable edges and to evaluate the classification assessment of our procedure.

3.1 Performance measures

The aim of the simulation experiment is to analyze whether the edges that appear in the first iterations are related with true edges in the population network, and

Table 1: Sparsity level and Conditioned Number for the considered networks

	n	Sparsity Level	Condition Number
AR(1) Model	30	0.067	605.368
	60	0.033	2481.008
	90	0.022	5678.965
	120	0.017	10220.410
AR(2) Model	30	0.131	5.549
	60	0.066	5.668
	90	0.044	5.706
	120	0.033	5.725
AR(3) Model	30	0.193	12.056
	60	0.098	12.561
	90	0.066	12.714
	120	0.050	12.788
Hub Model	30	0.163	16.432
	60	0.044	8.815
	90	0.020	11.932
	120	0.011	5.983

evaluate if they are associated with the largest partial correlations. To evaluate the performance of the method to rank edges in a GGM, we study assessment metrics specific to evaluate ranking procedures (see Tomal et al., 2013). For the maximum iteration $K \leq \frac{p \times (p-1)}{2}$, let $M \leq K$ be the iteration that recovers the true active edges in Ω.

Assuming we are in iteration k, we define $0 \leq H(k) \leq M$ to be the number of edges of the GGM decoded by $\hat{\Omega}^{(k)}$. We consider the following performance metrics.

Hit Curve. The hit curve is a plot of $H(k)$ versus k or equivalently a plot of the proportion of true edges, $H(k)/M$ versus the proportion of true edges in Ω, $p(k) = k/K$. The hit curve shows the ranking performance of a procedure at all possible iterations, k.

Average Precision. A summary of the hit curve can be obtained computing the average precision (AveP), which gives a single summary for a hit curve. For iteration $k \leq K$, we define the hit rate of precision for the top k ranked edges,

$$h(k) = \frac{H(k)}{k} \in [0, 1] \tag{3.1}$$

Let $1 \leq t_1 \leq t_2 \leq \ldots \leq t_M \leq K$ be the position of the M edges in the ranked list. AveP is defined as the average of the hit rates at the points of the hit curve where true edges are found:

11

$$\text{AveP} = \frac{1}{M}[h(t_1) + h(t_1) + \ldots + h(t_M)] \tag{3.2}$$

When all true edges are ranked before all the non-true edges, AveP reaches the maximum value 1.

Heat maps. We are also interested in determine the frequency of detected active edges in the iteration that recovers the true sparsity level of the population precision matrix, given by M. To do that we build heat maps that counts the average number of times an edge is hit. Therefore, we are able to determine which edges are estimated more frequently.

ROC curves. To evaluate the classification performance of the procedure we compute the ROC curves. Let TP be the true non-zero elements and TN be the true zero elements estimated by $\hat{\Omega}$ and FP be the false non-zero elements and and FN be the false zero elements estimated by $\hat{\Omega}$. The classification performance measures are given by

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{3.3}$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3.4}$$

The ROC curve is a plot where the x-axis has the proportion of the false detected non-zero elements, given by $1 - \text{Specificity}$, and in the y-axis the proportion of the correctly true positive edges given by Sensitivity.

For each $\hat{\Omega}^{(k)}$ ($1 \leq k \leq K$) we obtain the Specificity and Sensitivity and we plot the ROC curves.

We now compare the classification performance of our method with Glasso. Friedman et al. (2008) proposed an estimator of Ω^{Glasso} by solving the following ℓ_1 penalized-likelihood problem

$$\min_{\Omega \succ 0} -\log(\det(\Omega)) + \text{tr}(S\Omega) + \theta \parallel \Omega \parallel_1 \tag{3.5}$$

where $\theta \geq 0$ is the regularization parameter and $\parallel \Omega \parallel_1 = \sum_{j \neq i} \mid \omega_{ij} \mid$ the element-wise ℓ_1 norm.

We estimate $\Omega^{\text{Glasso}}(\theta)$ for values of the ℓ_1 penalty θ, that recover undirected graphs under different levels of sparsity, and compute the Specificity and Sensitivity to plot the ROC curve.

3.2 Simulation results

The simulation results of the ranking performance are summarized in Figures 2-5, where the hit curves for each model are shown. For the AR(1) Model, the hit curves

show that for small p (i.e. $p = 30$ and $p = 60$) the proportion of true detected edges increases rapidly up to the iteration that corresponds with the number of true edges, M. At iteration M, most of the true edges were ranked before the false ones. As expected, this performance is deteriorated when p increases, before reaching M. There is an increment in the proportion of false edges when $k \leq M$ (see Panel (c) and (d) from Figure 2), but the procedure is able to satisfactory detect the true edges before the false ones. This is confirmed in Table 2, where the AveP averaged over 100 replications is shown.

Figure 7 shows the heat maps of the AR(1) Model associated to the frequency of non-zero edges up to iteration M over 100 replications. For moderate dimensions, $p = 30$ and $p = 60$, the procedure estimates true edges with high frequency, and this frequency decreases for larger dimensions.

For the second simulated model, AR(2), Figure 3 presents the hit curves for different values of p. Similarly to the previous model, the proposed methodology shows a good ranking performance, even for high dimensions. The heat maps, in Figure 8, shows that the largest partial correlated nodes have the highest frequency over the 100 replications. Moreover, up to iteration M, the procedure is able to recover the majority of the largest partial correlated variables (see Figures 8 panel (c) and (d)).

The ranking performance results of the third model, AR(3), in Figure 4, show that when an additional partial correlated covariate is added to a node, the procedure diminishes its ability to detect true edges for $k \leq M$. As p increases, the procedure estimates first the true edges associated with the largest partial correlations, although the number of non-true edges also increases in the first iterations, as Table 2 shows. In any case, from the heat maps (see Figure 9), we observe that the largest partial correlated nodes have the highest frequency over 100 replications.

Figure 5 presents the hit curves of the Hub model. This is a particularly difficult model to estimate due to the random nature of the network structure. But even in this case, the proposed methodology is able to recover satisfactory the true edges in the the first iterations, although the general performance is worse than previous models as expected, see Figure 9 and Table 2.

Table 2: AveP averaged over 100 replications.

	$p = 30$	$p = 60$	$p = 90$	$p = 120$
AR(1) Model	0.927	0.832	0.706	0.594
AR(2) Model	0.754	0.667	0.606	0.312
AR(3) Model	0.685	0.551	0.495	0.454
Hub Model	0.606	0.429	0.361	0.243

Finally, regarding the GGM recovery performance, Figures 10-13 show the

ROC curves for each of the four considered network configurations, respectively, associated with the proposed methodology and Glasso. We can observe that our procedure obtain a better performance than Glasso, especially in high dimensions, except for the Hub random network configuration where Glasso outperforms the proposed methodology.

4   Conclusions

In this article we presented an approach to rank edges in a high dimensional Gaussian Graphical Model (GGM) to perform model selection. To do this, we proposed and algorithm to estimate an iterative path of precision matrices. Thus, in each step a symmetric and positive matrix with an additional edge was added to the path. The position of the new non-zero elements in Ω corresponded to the covariate that has the largest absolute correlation with the node conditional to the set of edges estimated in the previous step. For each step, the optimal graph is selected by minimizing a negative log-likelihood function. Our approach did not directly imposed any type of penalty into the likelihood function when estimated the precision matrix and was able to estimate a set of undirected graph that recover different levels of sparsity. We presented measures to asses the ranking performance of our procedure and we were able to obtain the ranking of edges that appear first.

The novelty of the approach is that we can obtain a ranking of more probable edges in a GGM. Thus, the first edges that enters in the algorithm are related with the largest partial correlated variables. In applications to real data, it is not possible to know the true sparsity level of the graph. Thus, knowing which are the most probable edges that will appear in a network is a relevant result to understand the undirected relations between variables. Also, the procedure is able to obtained the last rank edges in a GGM. These are the edges that will appear in the last positions of a ranking. Knowing in which iteration this last-ranked edges are recovered gave information of the sparsity level necessary to recover the true structure of the network.

Simulation studies show that the procedure is able to detect true edges in each iteration until the true sparsity level is recovered. Also, when the true precision matrix has an order structure, the procedure is able to add efficiently true edges in the first iterations while avoiding to add false edges. We also show that the edges that are first identified, are the ones associated with the largest partial correlated variables. The ranking performance shows that in the neighborhood of the true sparsity level, the procedure is able to add, in average, one true edge. Finally, our approach also has satisfactory GGM recovery performance compared with Glasso.
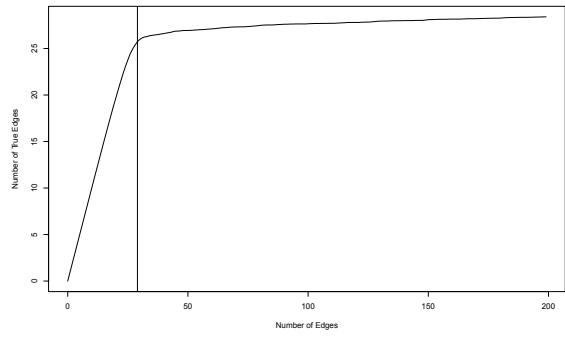
(a) $p = 30$
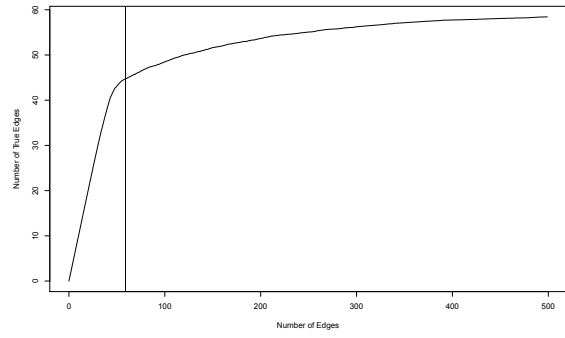
(b) $p = 60$
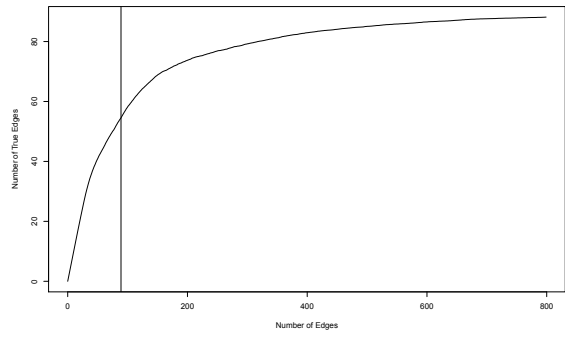
(c) $p = 90$
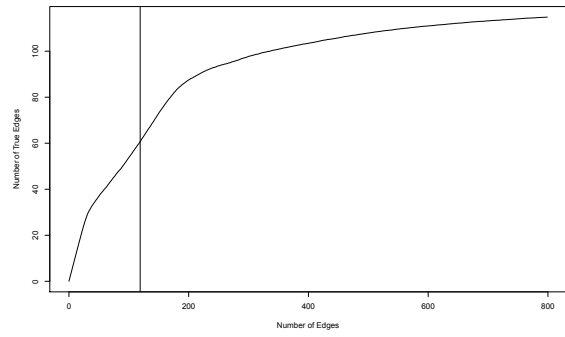
(d) $p = 120$

Figure 2: Hit curves for AR(1) Model with $n = 100$. x-axis: the number of total detected edges y-axis: the number of correctly identified edges. The vertical line corresponds to the number of true edges M.
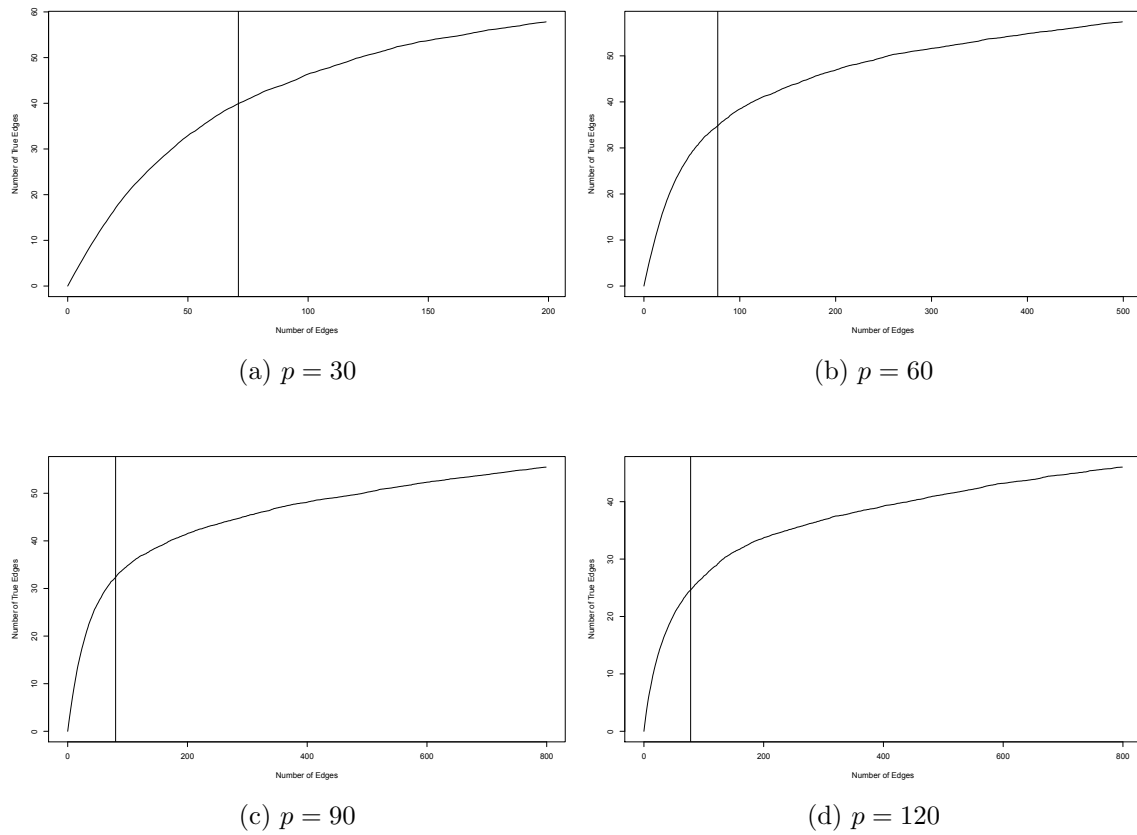
15
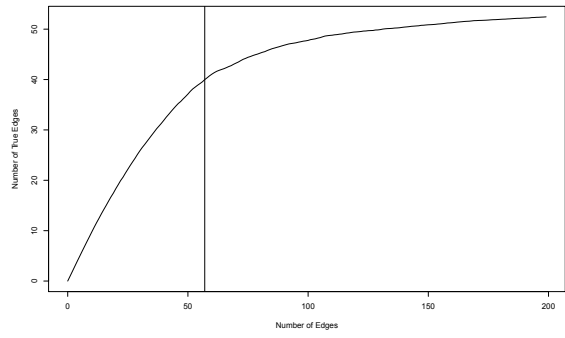
(a) $p = 30$
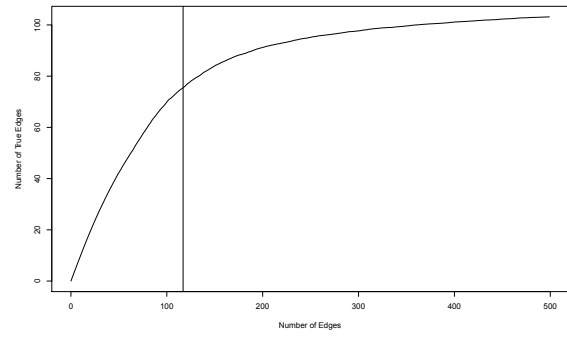
(b) $p = 60$
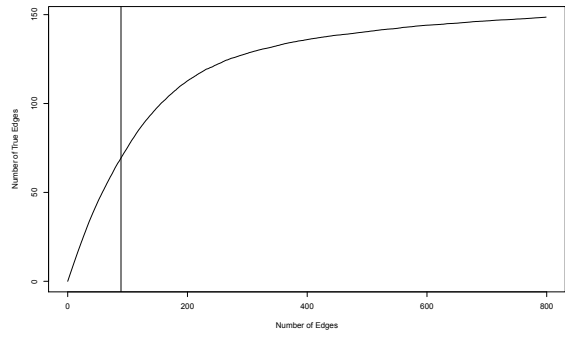
(c) $p = 90$
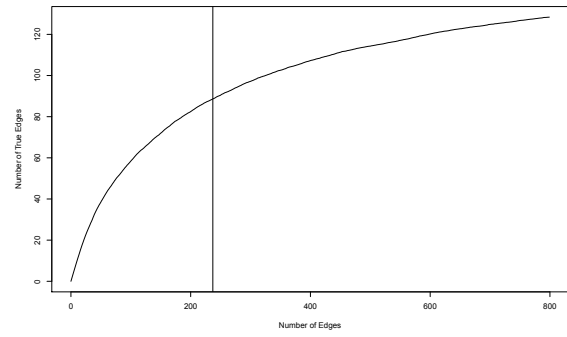
(d) $p = 120$

Figure 3: Hit curves for AR(2) Model with $n = 100$. x-axis: the number of total detected edges y-axis: the number of correctly identified edges. The vertical line corresponds to the number of true edges M.
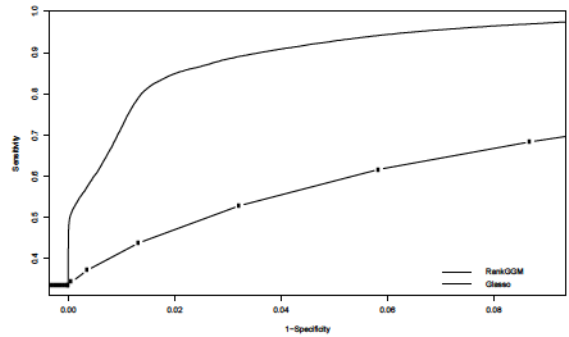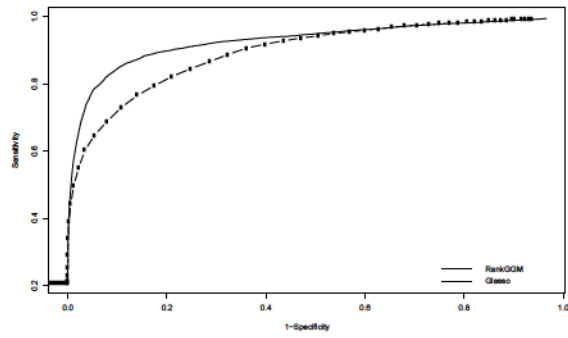
(a) $p = 30$

(b) $p = 60$

(c) $p = 90$

(d) $p = 120$
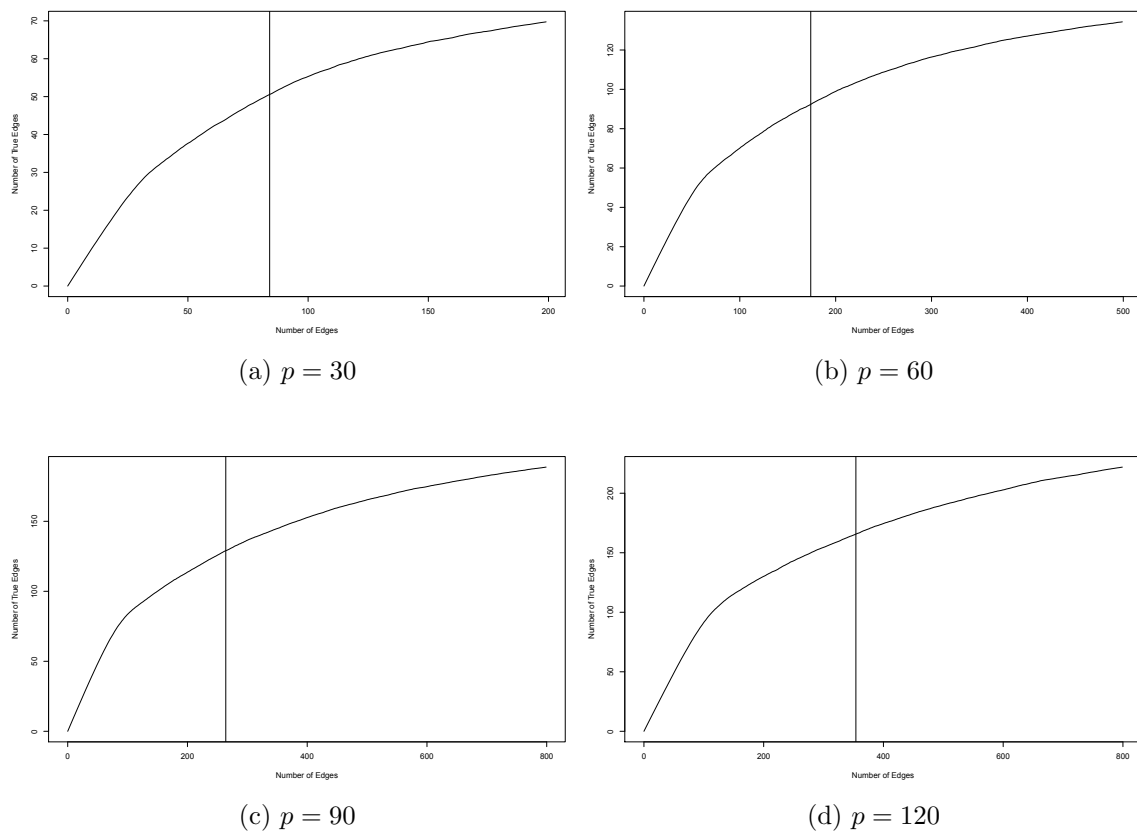
Figure 4: Hit curves for AR(3) Model with $n = 100$. x-axis: the number of total detected edges y-axis: the number of correctly identified edges. The vertical line corresponds to the number of true edges M.

(a) $p = 30$

(b) $p = 60$

(c) $p = 90$

(d) $p = 120$

Figure 5: Hit curves for Hub Model with $n = 100$. x-axis: the number of total detected edges y-axis: the number of correctly identified edges. The vertical line corresponds to the number of true edges M.
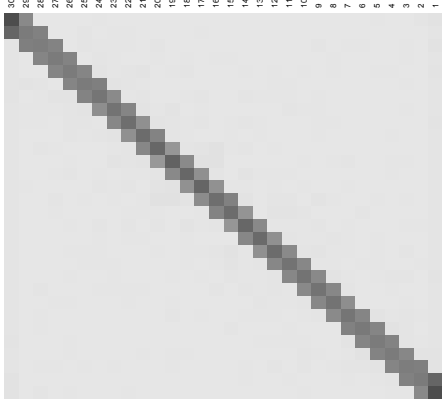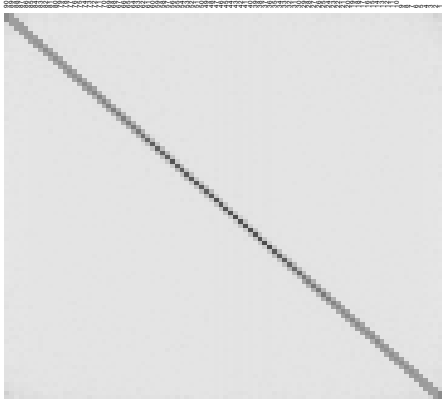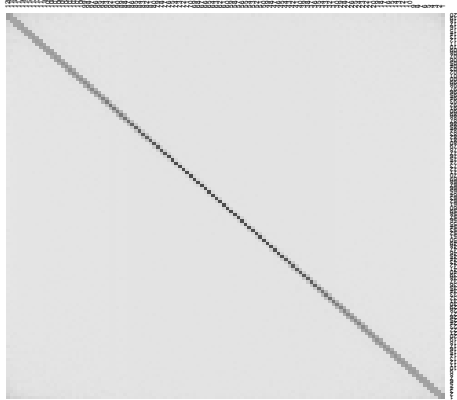
(a) $p = 30$

(b) $p = 60$

(c) $p = 90$

(d) $p = 120$

Figure 6: Heat maps of the frequency of non-zero edges up to iteration M over 100 replications for AR(1) Model with $n = 100$.

19
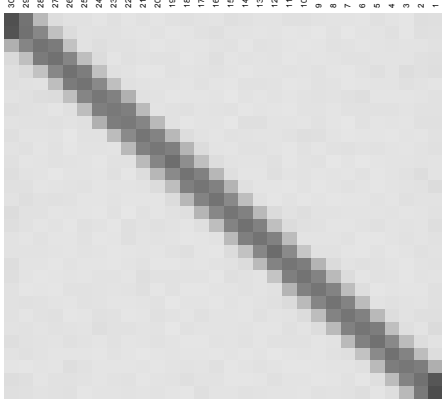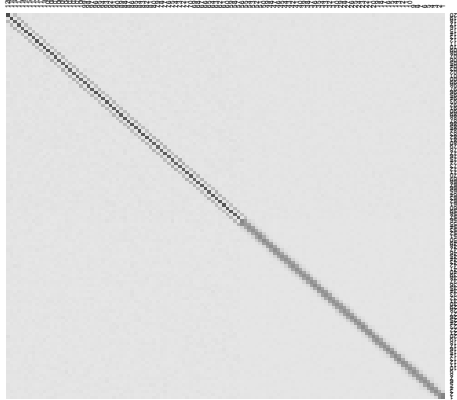
(a) $p = 30$

(b) $p = 60$

(c) $p = 90$

(d) $p = 120$

Figure 7: Heat maps of the frequency of non-zero edges up to iteration M over 100 replications for AR(2) Model with $n = 100$.

20
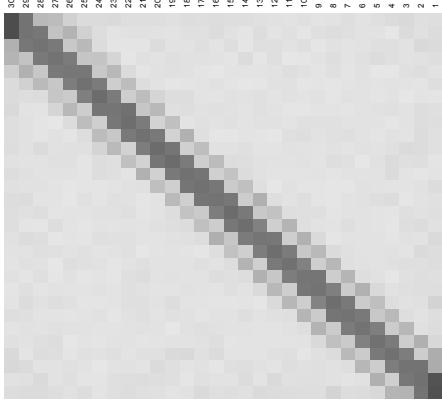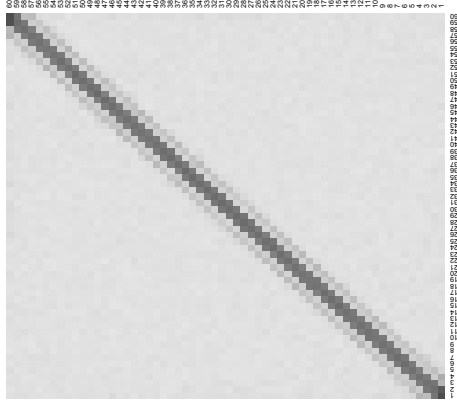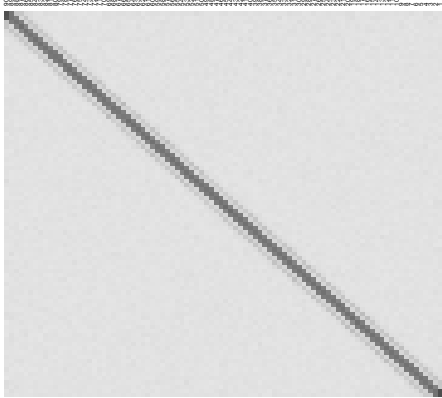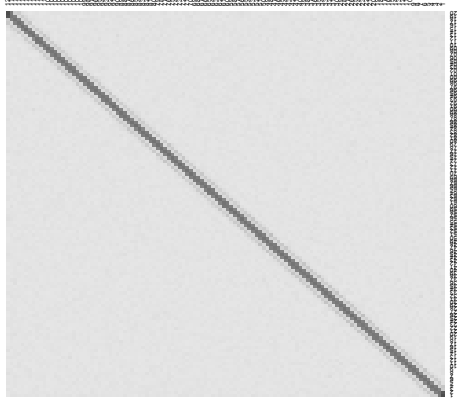
(a) $p = 30$

(b) $p = 60$

(c) $p = 90$

(d) $p = 120$

Figure 8: Heat maps of the frequency of non-zero edges up to iteration M over 100 replications for AR(3) Model with $n = 100$.
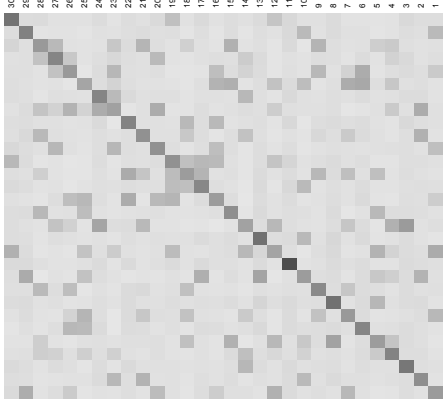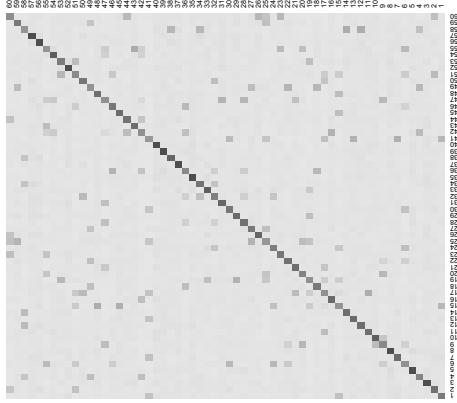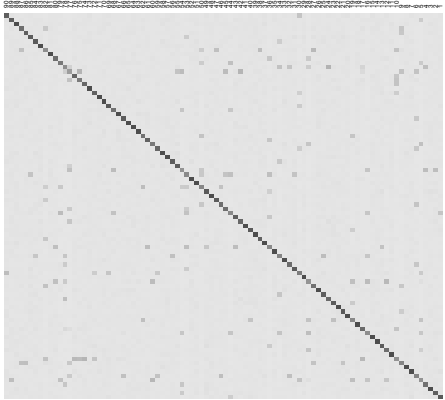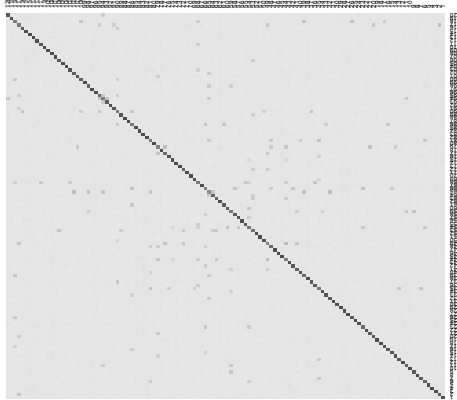
(a) $p = 30$

(b) $p = 60$

(c) $p = 90$

(d) $p = 120$

Figure 9: Heat maps of the frequency of non-zero edges up to iteration M over 100 replications for Hub Model with $n = 100$.
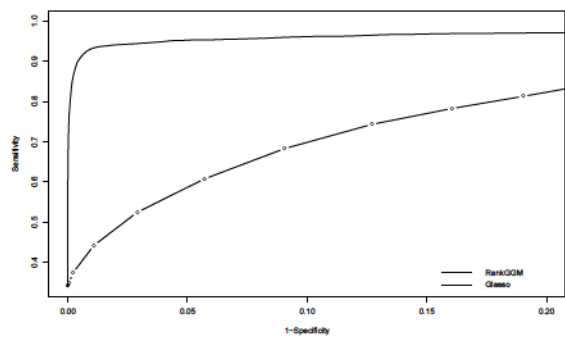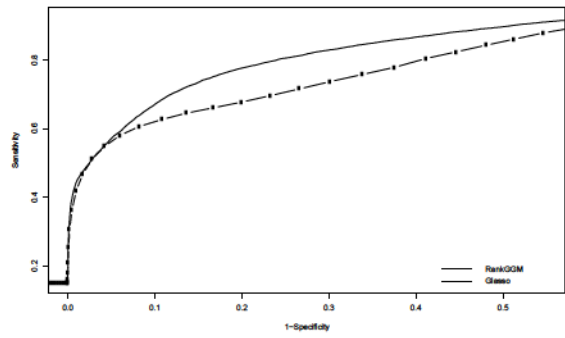
(a) $p = 30$
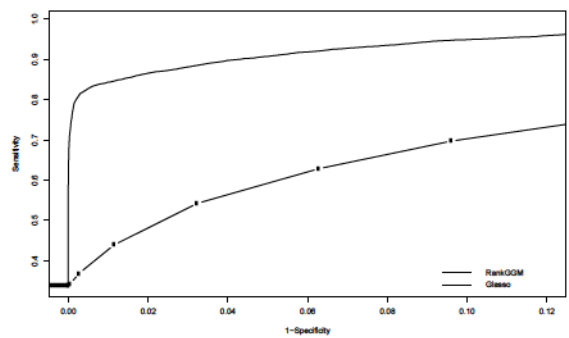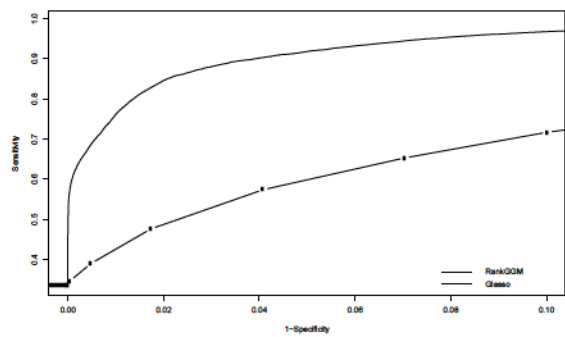
(b) $p = 60$

(c) $p = 90$

(d) $p = 120$

Figure 10: ROC curves for AR(1) Model with $n = 100$.
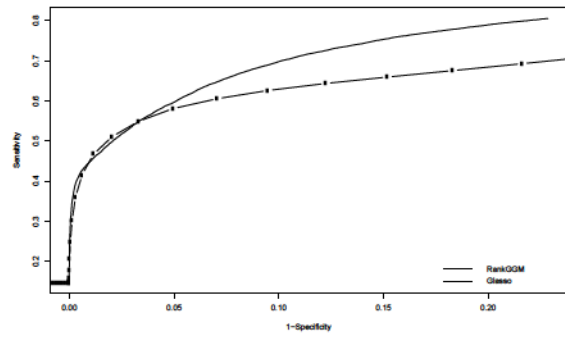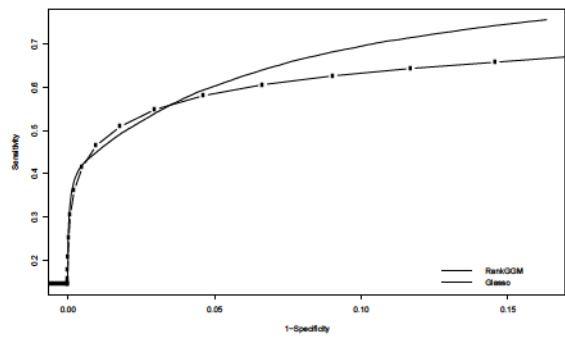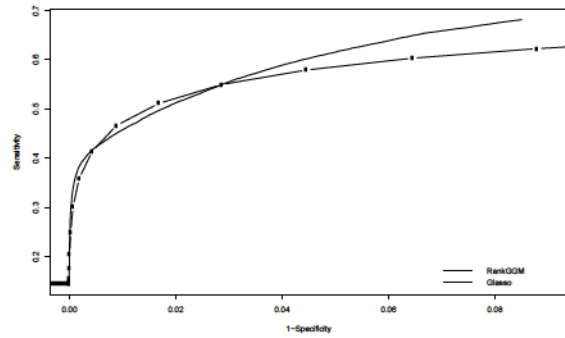
(a) $p = 30$
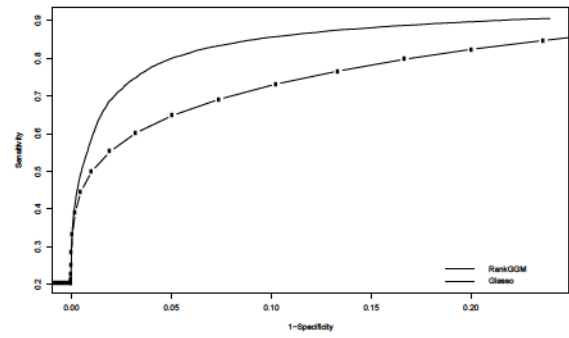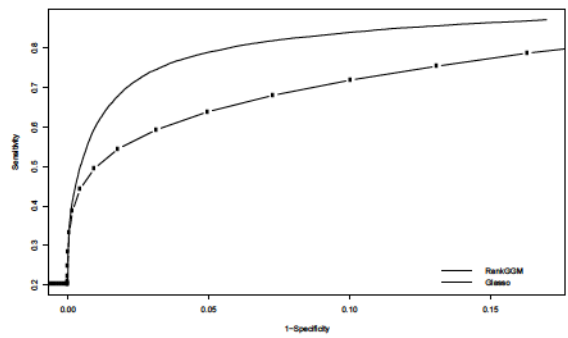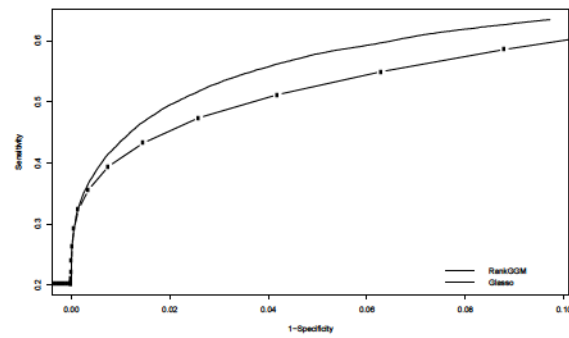
(b) $p = 60$

(c) $p = 90$

(d) $p = 120$

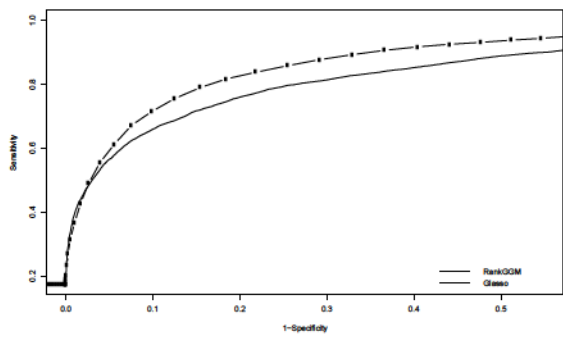Figure 11: ROC curves for AR(2) Model with $n = 100$.

(a) $p = 30$

(b) $p = 60$

(c) $p = 90$

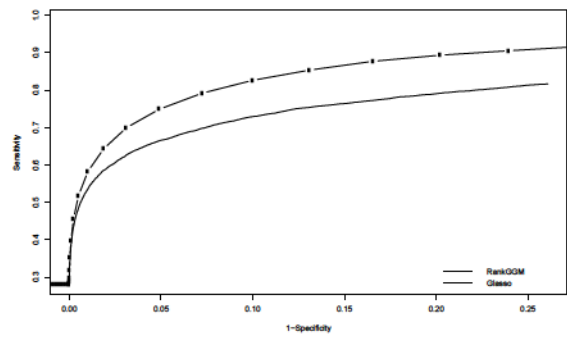(d) $p = 120$

Figure 12: ROC curves for AR(3) Model with $n = 100$.
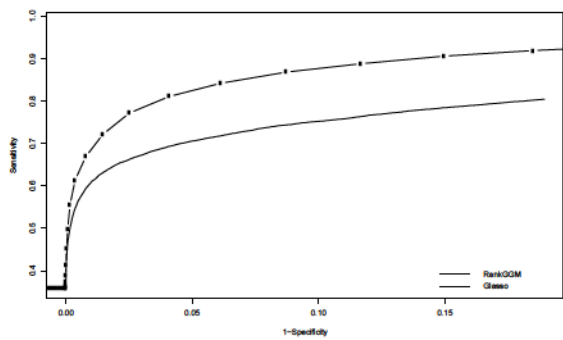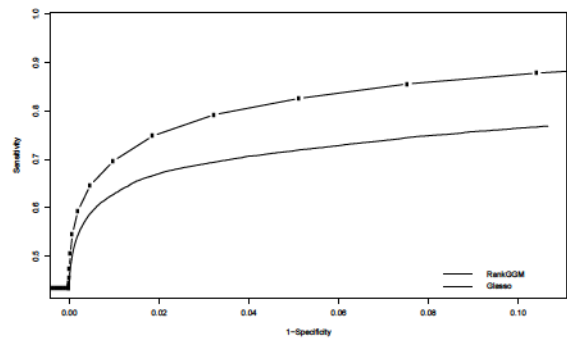
(a) $p = 30$

(b) $p = 60$

(c) $p = 90$

(d) $p = 120$

Figure 13: ROC curves for Hub Model with $n = 100$.

References

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

Cai, T., Liu, W., and Luo, X. (2011). A constrained 1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association*, 106(494):594–607.

Dempster, A. P. (1972). Covariance selection. *Biometrics*, pages 157–175.

Edwards, D. (2000). *Introduction to graphical modelling*. Springer Science & Business Media.

Efron, B., Hastie, T., Johnstone, I., Tibshirani, R., et al. (2004). Least angle regression. *The Annals of statistics*, 32(2):407–499.

Fan, J., Feng, Y., and Wu, Y. (2009). Network exploration via the adaptive lasso and scad penalties. *The annals of applied statistics*, 3(2):521.

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

Karoui, N. E. (2008). Operator norm consistent estimation of large-dimensional sparse covariance matrices. *The Annals of Statistics*, pages 2717–2756.

Lam, C. and Fan, J. (2009). Sparsistency and rates of convergence in large covariance matrix estimation. *Annals of statistics*, 37(6B):4254.

Lauritzen, S. L. (1996). *Graphical models*. Oxford University Press.

Ledoit, O. and Wolf, M. (2004). A well-conditioned estimator for large-dimensional covariance matrices. *Journal of multivariate analysis*, 88(2):365–411.

Liu, H. and Wang, L. (2012). Tiger: A tuning-insensitive approach for optimally estimating gaussian graphical models. *arXiv preprint arXiv:1209.2437*.

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

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

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

Sachs, K., Perez, O., Pe'er, D., Lauffenburger, D. A., and Nolan, G. P. (2005). Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721):523–529.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288.

Tomal, J. H., Welch, W. J., and Zamar, R. H. (2013). Ensembling classification models based on phalanxes of variables with applications in drug discovery. *arXiv preprint arXiv:1303.4805*.

Weisberg, S. (2005). *Applied linear regression*, volume 528. John Wiley & Sons.

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

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