LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

INSTITUT FÜR STATISTIK

Vincent Guillemot, Andreas Bender, Anne-Laure Boulesteix

iPACOSE: an iterative algorithm for the estimation of gene regulation networks

Technical Report Number 133, 2012
Department of Statistics
University of Munich

http://www.stat.uni-muenchen.de

iPACOSE: a new algorithm for the estimation of gene regulation networks

Vincent Guillemot[(1)] Andreas Bender[(1)]
Anne-Laure Boulesteix[(1)] *

[1] Department of Medical Informatics, Biometry and Epidemiology, University of Munich, Marchioninistr. 15, 81377 Munich, Germany

Abstract

In the context of Gaussian Graphical Models (GGMs) with high-dimensional small sample data, we present a simple procedure to estimate partial correlations under the constraint that some of them are strictly zero. This method can also be extended to covariance selection. If the goal is to estimate a GGM, our new procedure can be applied to re-estimate the partial correlations after a first graph has been estimated in the hope to improve the estimation of non-zero coefficients. In a simulation study, we compare our new covariance selection procedure to existing methods and show that the re-estimated partial correlation coefficients may be closer to the real values in important cases.

1 Introduction

The robust estimation of the inverse covariance matrix is crucial in many multivariate statistical methods such as discriminant analysis or linear regression [22]. Many variants of these multivariate methods aim at somehow "regularizing" the estimation of the covariance matrix to make it invertible or better conditioned, for example ridge regression (RR), diagonal discriminant analysis or regularized discriminant analysis [6]. A large body of literature is devoted to the estimation of the inverse covariance matrix in high-dimensional small sample settings, i.e. when the number of observations n is much smaller than the number of variables p. A well-known

*Corresponding author. Email: boulesteix@ibe.med.uni-muenchen.de.

example is the shrinkage estimator by [12] which is defined as a weighted sum of the sample covariance matrix and a fixed (invertible) target matrix. This method can be considered as "agnostic" in the sense that it estimates the covariance matrix in a completely data-driven way, i.e. without prior knowledge.

In this article, we first propose a method that directly estimates the partial correlation matrix while taking into account prior information of the dependencies between variables materialized by a given undirected graph. In a nutshell, our new method takes such a graph – called "independence graph" – as input and estimates the non-zero coefficients of the partial correlation matrix by ridge regression using the regression-based definition of partial correlations. The inverse covariance matrix can then be simply obtained from the partial correlation matrix by incorporating estimates of the variances. In this sense, our method can be seen as a *covariance selection* algorithm [3]. Although many covariance selection methods have been proposed in the literature (see Section 2 for details), none of these methods is designed to estimate the inverse covariance matrix in high-dimensional settings while incorporating a *non-decomposable* independence graph.

Furthermore, we suggest a new iterative algorithm called "PACOSE" - standing for PArtial COrrelation SElection - that estimates an independence graph from a data set using our new partial correlation estimate in a recursive way. Briefly, PACOSE takes as an input a data set and a significance level for the partial correlation and gives as an output an estimated independence graph. We show on simulated datasets that the fact that the partial correlation coefficients are re-estimated recursively yields graphs closer to the real one than a simple thresholding of an estimated partial correlation matrix.

The rest of the paper is structured as follows. Section 2 presents our iterative method and the associated covariance selection and also briefly reviews existing covariance selection methods. In Section 3, we compare our new method to existing GRNs estimation algorithms on simulated data. Finally, we present in section 4 some results obtained on real datasets.

2 Methods

2.1 Introduction

The estimation of Gene Regulation Networks (GRNs) is a burning issue in bioinformatics. Gaussian Graphical Models (GGMs) have been widely used for this purpose in the last few years GRNs can indeed provide the biologists with new candidates for interactions between a set of given genes through the analysis of now widely available gene expression microarray data. Unfortunately, the very property of the microarray data that makes them attractive for this task, which is the high number of genes present on such supports, is what makes the task of estimating GRNs statistically challenging: the number of individuals (n) is always very low compared to the number of variables (p). It is then mandatory to adopt regularization strategies to cope with this $n \ll p$ situation when estimating GGMs.

The method we propose achieves the estimation of a graph through the use of a regularized partial correlation matrix estimation. We call this new method "(PACOSE)", standing for PArtial COrrelation SElection. It is able to estimate a partial correlation matrix under the constraint that some given coefficients are equal to 0. The term "Selection" refers to covariance selection algorithms. They are themselves defined as methods to:

- Estimate the covariance and/or the precision matrix matrix knowing that some given coefficients are null in the precision matrix [3];
- Estimate the covariance matrix knowing that there is a certain amount of zeros in the precision matrix [2, 11].

To avoid any confusion with these sensibly different definitions, we chose an acronym closely related to the parameters that we want to estimate: the partial correlations.

Briefly, PACOSE is an iterative algorithm based on the subsequent application of a threshold to an estimated partial correlation matrix knowing a pattern of zeros. It is synthesized in Figure 1 and further detailed in part 2.7.

2.2 Partial correlation and Gaussian Graphical Models

This section briefly reviews the basics of GGM theory used in this paper. Let X denote a p-variate random vector $X = (X_1, \ldots, X_p)^\top$ such that the

variables $X_1, \ldots, X_p$ all have a mean and a variance. $\mathcal{G}$ denotes the graph describing the conditional independencies between the p variables: $\mathcal{G}$ is thus an undirected graph with p nodes. The covariance matrix of X, denoted by Σ, is supposed to be invertible. Its inverse $\Omega = \Sigma^{-1}$ is from now on referred to as the *precision matrix*.

The partial correlation coefficient ρ_{ij} of X_i and X_j given all the other variables $\{X_1, \ldots, X_p\} \setminus \{X_i, X_j\}$ can be estimated as

$$\widehat{\rho}_{ij} = \frac{\widehat{cov}\left(X_i - \widehat{X}_i, X_j - \widehat{X}_j\right)}{\sqrt{\widehat{var}\left(X_i - \widehat{X}_i\right)\widehat{var}\left(X_j - \widehat{X}_j\right)}}, \tag{1}$$

where $\widehat{cov}$ and $\widehat{var}$ denote the empirical covariance and variance, respectively, and $\widehat{X}_i$ stands for the fitted value of X_i in a linear regression model including all variables except X_i and X_j as covariates. In a word $\widehat{\rho}_{ij}$ is the correlation of the residuals of the linear models regressing X_i against all variables except X_j and vice-versa.

Another method to compute $\widehat{\rho}_{ij}$ based on linear regressions results from the following property [20]:

$$\widehat{\rho}_{ij} = sign(\widehat{\beta}_{ij})\sqrt{\widehat{\beta}_{ij}\widehat{\beta}_{ji}}, \tag{2}$$

where $\widehat{\beta}_{ij}$ is the estimated coefficient of variable X_j in the linear model regressing X_i against all the other variables. Note that both formulations (1) and (2) implicitly assume that the considered linear regression models can be estimated, which is for instance not the case in high-dimensional data with $n < p$. This issue will be discussed later. Moreover, it can also be shown [20] that the partial correlation coefficient ρ_{ij} is related to the precision matrix $\Omega = [\omega_{ij}] = \Sigma^{-1}$ as follows:

$$\rho_{ij} \;=\; -\frac{\omega_{ij}}{\sqrt{\omega_{ii}}\sqrt{\omega_{jj}}}, \text{ for } i \neq j. \tag{3}$$

If $X_1, \ldots, X_p$ are Gaussian, the following important property can be shown for $i, j, k \in \{1, \ldots, p\}$ $(k \neq i, j)$, see for instance [21]:

$$X_i \perp\!\!\!\perp X_j | X_k \Leftrightarrow \rho_{ij} = 0, \tag{4}$$

which means that two variables are conditionally independent if and only if their partial correlation equals zero.

The formulation (2) is exploited by numerous methods to estimate gene regulatory networks from high-dimensional microarray gene expression data [7, 10, 15]. Note, however, that these data often have much more variables (genes) than observations (arrays), hence the term "$n \ll p$ data". A regularized regression technique has then to be used to estimate β_{ij} and β_{ji}, since least squares regression cannot be performed with $n < p$ data. Another popular approach [12] to estimate GGMs from high-dimensional data consists in applying Eq. (3) using a regularized (invertible) estimator of Σ .

All these methods yield an estimate of the partial correlation matrix. Some methods are essentially sparse, i.e. yield a matrix with many zeros [7]. In this case, the graph is simply derived from the partial correlation matrix by connecting pairs of variables with non-zero partial correlations. For other methods [12, 10], however, a threshold has to be applied to decide which variables have to be connected. We further discuss this aspect in section 2.6.

2.3 Correlation and covariance selection

The concepts briefly reviewed in the above section are important for understanding our novel method, whose main idea is to combine formulation (2) along with the information given in an a priori independence graph $\mathcal{G}$ between the variables. In a nutshell, this is done by setting β_{ij} and β_{ji} to 0 if X_i and X_j are not connected in the graph. It immediately results from Eq. (2) that $\widehat{\rho}_{ij} = 0$. Other partial correlation coefficients ρ_{ik} or ρ_{jk} involving X_i or X_j are also affected by the constraint $\beta_{ij} = \beta_{ij} = 0$, which essentially removes one covariate in the considered linear regression models.

More precisely, our graph-constrained estimator of the partial correlation between X_i and X_j is given as

$$\widehat{\rho}^{\mathcal{G}}_{ij} = sign(\widehat{\beta}^{\mathcal{G}}_{ij})\sqrt{\widehat{\beta}^{\mathcal{G}}_{ij}\widehat{\beta}^{\mathcal{G}}_{ji}}, \tag{5}$$

where

* $\widehat{\beta}^{\mathcal{G}}_{ij} = 0$ if X_i and X_j are not connected in $\mathcal{G}$,
* $\widehat{\beta}^{\mathcal{G}}_{ij}$ is the estimated regression coefficient of X_j in the regression of X_i

against its connected variables otherwise, i.e. the coefficient $\widehat{\beta}_{ij}^{\mathcal{G}}$ in the model

$$X_i \;=\; \beta_{i0}^{\mathcal{G}} \;+\; \sum_{k:\; k\sim i} \beta_{ik}^{\mathcal{G}} X_k \;+\; \epsilon_i. \tag{6}$$

This definition implicitly assumes that the estimates of the regression coefficients exist, which may not be the case in high-dimensional settings. This problem is addressed in the next section.

2.4 High dimensional settings

When the number of variables connected to i is greater than the number of observations, the estimation of the coefficients of the linear regression model (6) cannot be performed by ordinary least squares. Unfortunately, it is likely to sometimes occur in practical analyses with high-dimensional data. That is why we suggest to replace least squares regression by one of its its regularized versions: ridge regression [9], PLS regression [23, 24], Lasso [16] or adaptive Lasso [25]. Here we focus on Ridge Regression which can be seen as the standard regularized regression approach. The regularization parameters are estimated by k-fold cross-validation (CV).

The estimated partial correlation coefficients are then given as:

$$\widehat{\rho}_{ij}^{\mathcal{G}} = sign(\widehat{\beta}_{ij}^{\mathcal{G},\mathcal{RR}})\sqrt{\widehat{\beta}_{ij}^{\mathcal{G},\mathcal{RR}}\widehat{\beta}_{ji}^{\mathcal{G},\mathcal{RR}}}, \tag{7}$$

where $\widehat{\beta}_{ij}^{\mathcal{G},\mathcal{RR}}$ is the regression coefficient of variable X_j in the ridge regression model regressing X_i against all the other variables, where the penalty is optimized by CV.

Once the partial correlation coefficients are estimated, an estimator of the precision matrix $\Omega = \Sigma^{-1}$ is obtained via the following procedure:

1. compute for each variable its partial variance with respect to the other variables,

2. use the relation between the precision matrix, the partial covariance matrix and the partial variances, build the precision matrix estimate as

$$\rho_{ij} = \frac{-\omega_{ij}}{\sqrt{\omega_{ii}}\sqrt{\omega_{jj}}}$$

According to [20], the partial correlation coefficient are linked to the coefficients of the precision matrix $\omega_{ij} = [\Omega]_{ij}$ through Eq. (3). All the methods from the literature allow to compute directly the precision matrix, we then use equivalence (2) to compare them to our method based on the ridge regression. With this algorithm it is then possible to estimate a partial correlation matrix and a precision matrix.

2.5 Competing approaches

To our knowledge, there is no method in the literature allowing to compute directly the partial correlation matrix with the knowledge of an undirected graph. But there are numerous methods dedicated to the estimation of the inverse covariance matrix knowing a given graph. The literature refers to these methods as covariance selection algorithms. These algorithms are usually used to estimate the covariance matrix, but they can also be used to estimate the precision matrix.

A first method could be to estimate the precision matrix with a shrinkage algorithm [12] and then to set to 0 all the coefficients corresponding to the non connected variables in the given graph:

$$\widehat{\Omega} = (S + \lambda I)^{-1} .$$

To take into account the given graph, all the coefficients $\widehat{\Omega}_{ij}$ such that $i \nsim j$ are set to 0.

When the graph is decomposable:

- if it is completely connected, the maximum likelihood estimate for the precision matrix is simply $\widehat{\Omega} = S^{-1}$.
- otherwise, the maximum likelihood estimate of the precision matrix is a linear combination of block matrices:

$$\widehat{\Omega}(\mathcal{G}) = \sum_{\mathcal{C}_i} n \left[S_{\mathcal{C}_i}^{-1} \right]^0 - \sum_{\mathcal{S}_j} n \left[S_{\mathcal{S}_j}^{-1} \right]^0 ,$$

However, the previous estimator's variance is outperformed by the variance of the estimators presented by Wiesel [21]: given a decomposition of the graph $\mathcal{G}$ into cliques $\mathcal{C}_i$ and separators $\mathcal{S}_j$, then the proposed biased

estimator is

$$\widehat{\Omega}_{Wi}(\mathcal{G}) = \sum_{\mathcal{C}_i} (n - c_i - 1 - d) \left[S_{\mathcal{C}_i}^{-1}\right]^0 - \sum_{\mathcal{S}_j} (n - s_j - 1 - d) \left[S_{\mathcal{S}_j}^{-1}\right]^0,$$

with $d = -2tr(\nabla D)/D$ and $D = \sum_{\mathcal{C}_i} S_{\mathcal{C}_i}^{-1} - \sum_{\mathcal{S}_j} S_{\mathcal{S}_j}^{-1}$.

Nevertheless, the methods presented in [21] are not able to cope with a non decomposable graph. This is a major drawback because most of the graphs we deal with in bioinformatics are not decomposable. Hence, one needs to turn to iterative methods ([19] or [20]) or methods based on the optimization of a criterion independent from the nature of the graph, such as the method `glasso` [7].

The iterative algorithm presented in [19] is based on the fact that if one coefficient only has to be null in the precision matrix, then there is an equivalence between the empirical covariance matrix, its inverse and the other coefficients. The algorithm consists in iterating on the coefficients supposed to be zero until they all reach an acceptable level defined by the user. This estimator will be denoted $\widehat{\Omega}_{We}(\mathcal{G})$.

The iterative algorithm presented in [20] (p. 182) is based on an iterative proportional fitting algorithm [13]. It consists, for each clique a of the graph, to replace at iteration $n+1$ $K_{a,a}^{n+1}$ by

$$S_{a,a}^{-1} + B_{b|a}^n K_{a,a}^n (B_{b|a}^n)^\top,$$

with $B_{b|a}^n = cov^n(X_b, X_a) var^n(X_a)^{-1}$. The resulting estimator of the precision matrix is then the limit of the iterative process : $\widehat{\Omega}_{Wh}(\mathcal{G}) = lim_n K^n$.

Finally, the method `glasso` [7] allows to compute an estimate of the precision matrix $\widehat{\Omega}_{Gl}(\mathcal{G})$. It has to be noted that the version of the method used to take into account depends on a regularization parameter. Since we do not have a suitable method to determine it in a proper way, we decided to set it to 0.

2.6 Thresholding the partial correlation matrix

Regardless the method used to estimate a partial correlation matrix, whether a graph is also used in the process or not, the user may want to know which coefficients are significant and which are not. The goal could be to transform a partial correlation matrix, e.g. estimated thanks to the method of Sch?fer

et al. [12], into a graph. Since for such an application a certain value of partial correlation is determined above which the partial correlation coefficients are considered significant, we will refer to this value as the "threshold".

Many methods are available to determine a suitable threshold. The first step is to apply to the partial correlation coefficients an inverse hyperbolic tangent transform, also known as Fisher's transformation [5], and compute a p-value by considering the obtained values as normally distributed. Then, the p-values are corrected for multiple tests with

1. the local-fdr strategy, this is the strategy adopted for example in [10],
2. the FNDR controlling strategy [14].

Finally, the threshold is the minimum value for which the p-values are considered significant. We choose the last of the previous strategies to determine a suitable threshold and use the functions implemented in the library `fdrtool` [14].

2.7 PACOSE

In a nutshell, our algorithm PACOSE operates as follows:

1. Estimate a first version of the partial correlation matrix with an already existing method (for example `pcor.shrink` or `ridge.net` or `pls.net` or `adalasso.net` or `lasso.net`).
2. Define on this matrix a threshold that allows transforming the partial correlation matrix into a graph that can be used in PACOSE; this threshold is then fixed once and for all.
3. Apply a partial correlation selection to the dataset with the previously defined graph in order to estimate a new partial correlation matrix.
4. Apply the threshold defined in 2 to the partial correlation matrix estimated with partial correlation selection to define a new version of the graph.
5. Iterate steps 3 and 4 until the graph in 4 is the same as the graph in 3.

We experimentally verified that it is not satisfactory to set a new threshold for each new iteration: it leads to far too sparse graphs.

3 Results on simulated data

We address three different issues in this section:

(a) the estimation of the partial correlation matrix and the precision matrix knowing a certain undirected graph,

(b) the evaluation of PACOSE's performance.

When one want to simulate data knowing a given graph of independence, there is the possibility to use the characteristic given by the theory of Gaussian Graphical Models. To this constraint is added the fact that the randomly generated precision matrix has to be positive definite. One could see this problem as a so-called "positive definite completion matrix" issue [8]. But the work on this specific issue is once again mainly focused on decomposable graph. We decided to adopt a more empirical method, which in practice gives a very satisfying range of partial correlation coefficients, and at the end of the algorithm, the respect of constraint (4).

We used simulated data to compare our method to the methods presented in the literature. Erdős-Rényi's graphs [4] were used to model the interactions between genes, which allows loops, hubs, and multiple connected components. We used the following algorithm:

(i) Compute a first random Erdős-Rényi [4] or Barabasi [1] graph $\mathcal{G}^{(0)}$, with adjacency matrix ,

(ii) Get the "upper triangular" adjacency matrix $A^{(0)}$ of this graph and replace any non null coefficient by a random realization of a uniform variable (e.g. $\mathcal{U}(]-1,-0.8] \cup [0.8,1[)$, but any interval is possible), which then allows to define an upper triangular weight matrix $W^{(0)}$,

(iii) compute then the following matrix $M = (W^{(0)}+I)^\top(W^{(0)}+I)$, where I is the identity matrix, defining a new graph $\mathcal{G}$ slightly different from the initial graph, but above all defining a sparse positive definite matrix M,

(iv) Normalize this matrix to get a partial correlation matrix $\Pi = M^\star$,

(v) generate a multivariate Gaussian random variables $X \sim \mathcal{N}(\mathbf{0}, \Sigma = \Omega^{-1})$.

We prefer this algorithm to for example the algorithm presented in [18] (the same as the one presented in [10]) because the latter produces partial correlation coefficients often very close to 0 when p is greater than a few dozens. The drawback of this method is that it alters the degree structure of the initial structure - in a drastic way for Erdős-Rényi graph, and in a very moderate way for Barabasi's graph.

The covariance selection algorithm presented in [19] were implemented in R and C by ourselves, so were the algorithms presented in [21]. Whittaker's method [20] is implemented in the R package `ggm`, and Friedman's et al method [7] in the package `glasso`.

3.1 Estimation of the partial correlation matrix

The presented estimator has interesting mean square errors compared to the other estimators, as figure 3.1 shows it on an example. The function :

$$MSE(\widehat{\Omega}) = \frac{1}{N}\sum_{ij}(\widehat{\Omega}_{ij} - \Omega_{ij})^2,$$

$$MSE(\widehat{\Pi}) = \frac{1}{N}\sum_{ij}(\widehat{\Pi}_{ij} - \Pi_{ij})^2,$$

with $N = p(p-1)/2$.

This is when the setting is favorable (much more variables than individuals). When the setting is less favorable, the results show a better performance of our estimator, both in stability and accuracy, see figure 3.1.

3.2 PACOSE results

PACOSE was applied to simulated datasets in order to recover the underlying partial independence graphs.

To compare the estimated graphs with the real graph, we used the positive predictive value (ppv) and the sensibility (sen):

$$ppv = \frac{TP}{TP+FP} \text{ and } sen = \frac{TP}{TP+FN},$$

where TP, FP and FN are defined in table 1

Biological networks are indeed often described as sparse, and indicators based on the number of edges are more suitable in this case [17].

	$i \sim j$	$i \nsim j$
$\widehat{\rho}_{ij} \neq 0$	TP	FP
$\widehat{\rho}_{ij} = 0$	FN	TN

Table 1: The definitions of true and false positives (resp. TP and FP), true and false negatives (resp. TN and FN) in the context of graph inference.

On Figures 3.2 and 3.2 are represented the sensibility and the ppv of the estimated graphs as a function of the threshold. PACOSE clearly outperforms the classical graph estimation strategies based on partial correlation.

4 Conclusion

We present in this paper a new straightforward way of integrating a network in the estimation of a partial correlation matrix. It constitutes the basis of our new iterative algorithm for the estimation of Gaussian Graphical Models.

References

[1] Barabasi and Albert. Emergence of scaling in random networks. Science, 286(5439):509–512, Oct 1999.

[2] Alexandre D'Aspremont, Onureena Banerjee, and Laurent El Ghaoui. First-order methods for sparse covariance selection. Siam Journal On Matrix Analysis and Applications, 30(1):56–66, 2008.

[3] A. P. Dempster. Covariance selection. Biometrics, 28(1):157–175, 1972.

[4] P. Erdős and A. Rényi. On random graphs. Publicationes Mathematicæ, 6:290–297, 1959.

[5] R. A. Fisher. Frequency distribution of the values of the correlation coefficient in samples from an indefinitely large population. Biometrika, 10(4):pp. 507–521, 1915.

[6] Jerome Friedman. Regularized discriminant analysis. Journal of the American Statististical Association, 84:165, 1998.

[7] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Sparse inverse covariance estimation with the graphical lasso. Biostatistics, 9(3):432–441, Jul 2008.

[8] Robert Grone, Charles R. Johnson, Eduardo M. S, and Henry Wolkowicz. Positive definite completions of partial hermitian matrices. Linear Algebra and its Applications, 58:109 – 124, 1984.

[9] A. E. Hoerl and R. W. Kennard. Ridge regression: biased estimation for nonorthogonal problems. Technometrics, 12:55–77, 1970.

[10] N. Krämer, J. Schäfer, and A.-L. Boulesteix. Regularized estimation of large scale gene association networks using gaussian graphical models. BMC Bioinformatics, 10:384, 2009.

[11] Vijay Krishnamurthy and Alexandre d'Aspremont. A pathwise algorithm for covariance selection. In OPT 2009: 2nd NIPS Workshop on Optimization for Machine Learning. MIT Press, 2009.

[12] J. Schäfer and K. Strimmer. A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. Stat Appl Genet Mol Biol, 4:Issue 1, Article 32, 2005.

[13] T. P. Speed and H. T. Kiiveri. Gaussian markov distributions over finite graphs. The Annals of Statistics, 14(1):138–150, March 1986.

[14] Korbinian Strimmer. fdrtool: a versatile r package for estimating local and tail area-based false discovery rates. Bioinformatics, 24(12):1461–1462, Jun 2008.

[15] A. Tenenhaus, V. Guillemot, X. Gidrol, and V. Frouin. Gene association networks from microarray data using a regularized estimation of partial correlation based on pls regression. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 7(2):251–262, April-June 2010.

[16] R. Tibshirani. Regression shrinkage and selection via the lasso. Journal of the Royal Statistics Society, Series B 58:267–288, 1996.

[17] Jean Philippe Vert. Reconstruction of biological networks by supervised machine learning approaches. Technical report, Mines ParisTech, Centre for Computational Biology, 2008.

[18] N. Verzelen and F. Villers. Tests for gaussian graphical models. Computational Statistics & Data Analysis, 53(5):1894–1905, March 2009.

[19] Nanny Wermuth and Eberhard Scheidt. Fitting a covariance selection model to a matrix. Applied Statistics Algorithms, pages 88 – 92, 1976.

[20] Joe Whittaker. Graphical models in applied multivariate statistics. Wiley, 1990.

[21] Ami Wiesel, Yonina C. Eldar, and Alfred O. Hero. Covariance estimation in decomposable gaussian graphical models. IEEE Transactions on Signal Processing, 58(3):1482–1492, 2010.

[22] Daniela M Witten and Robert Tibshirani. Covariance-regularized regression and classification for high-dimensional problems. J R Stat Soc Series B Stat Methodol, 71(3):615–636, Feb 2009.

[23] H. Wold. Path Models with Latent Variables: The NIPALS Approach, pages 307–357. 1975.

[24] S. Wold, A. Ruhe, H. Wold, W. J. Dunn, and III. The collinearity problem in linear regression. the partial least squares (pls) approach to generalized inverses. SIAM Journal on Scientific and Statistical Computing, 5(3):735–743, 1984.

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

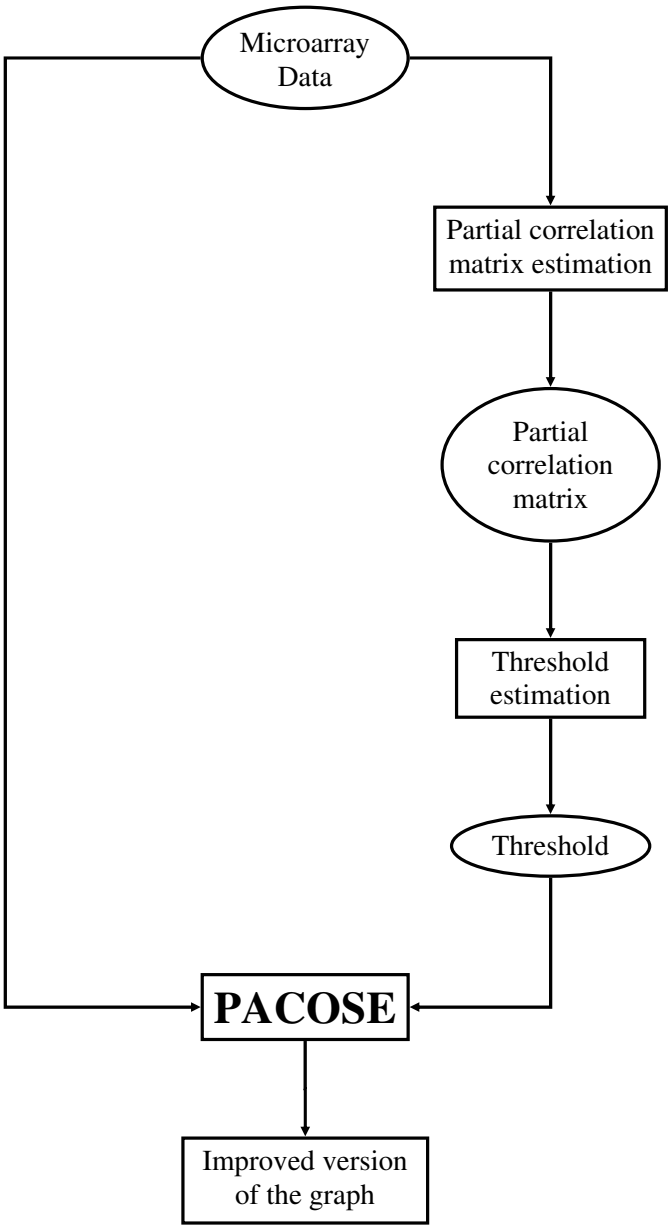

Figure 1: The PACOSE algorithm

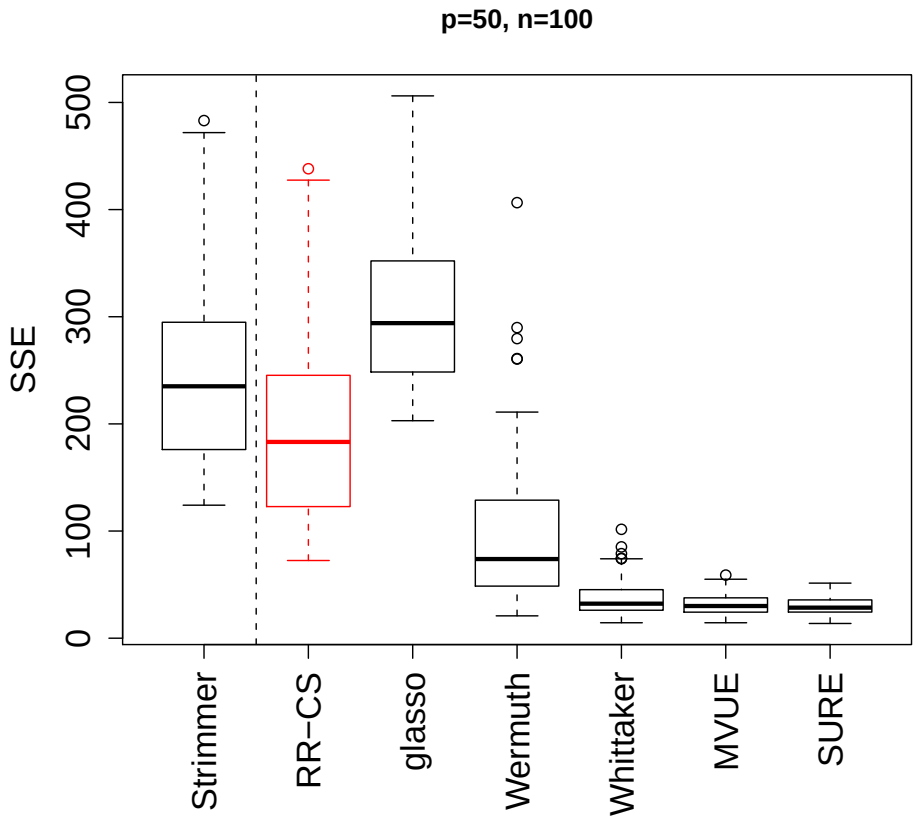

Figure 2: Sum of the square errors of the precision matrix estimates. $p = 50$ and $n = 100$, the graphs are decomposable.

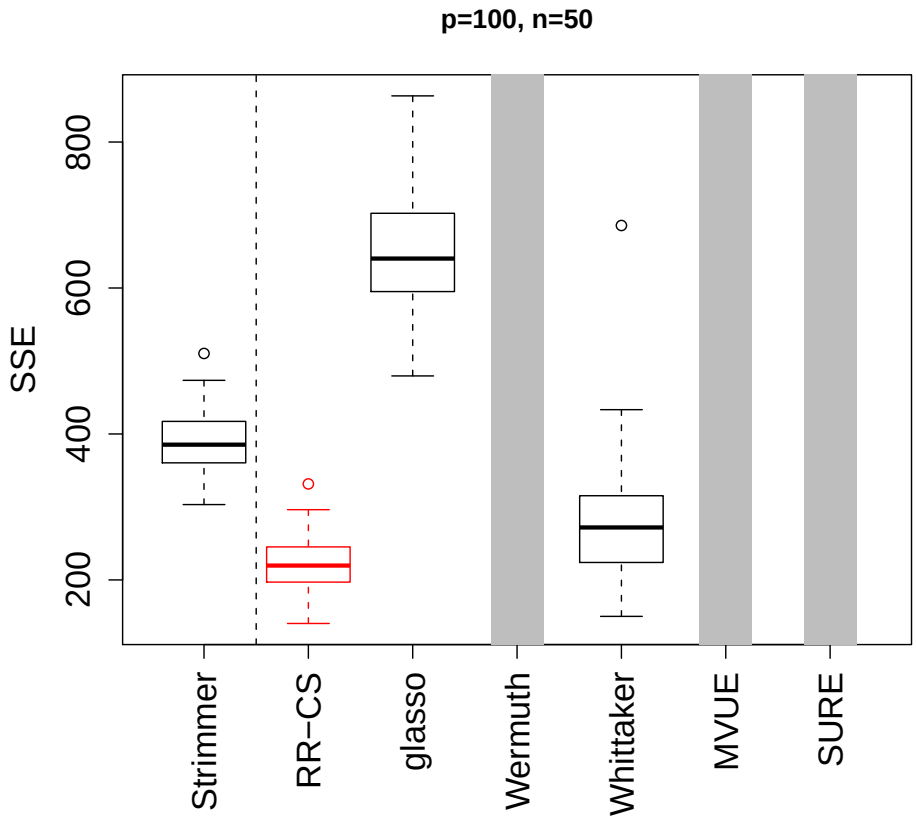

Figure 3: Sum of the square errors of the precision matrix estimates. $p = 100$ and $n = 50$, the graphs are not decomposable.

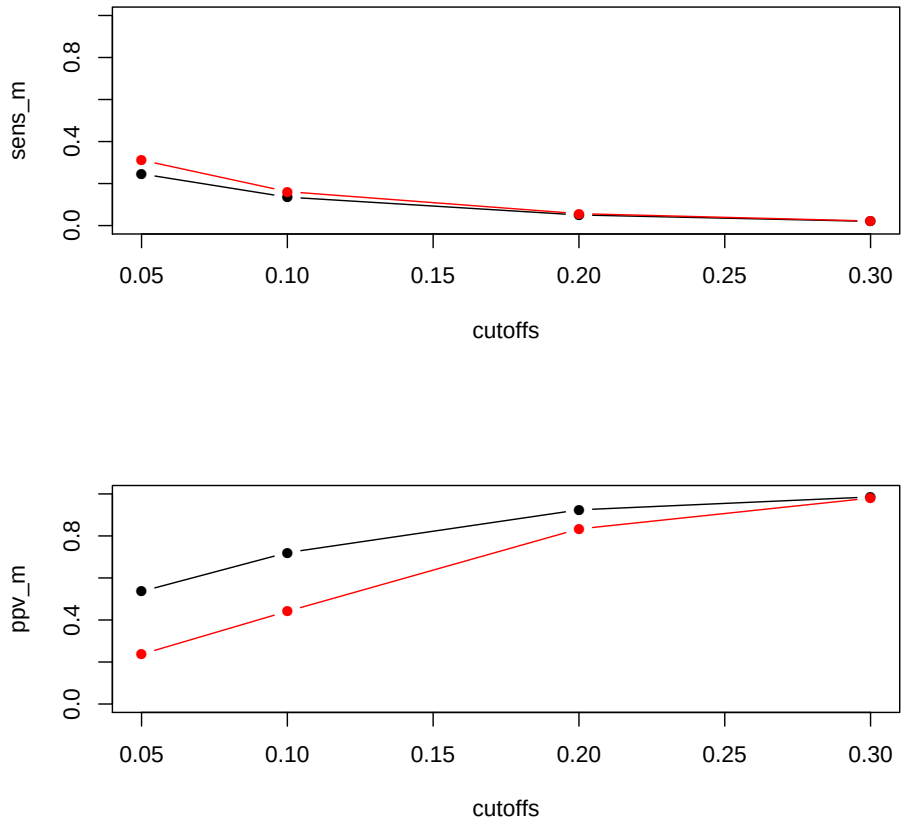

Figure 4: Sensibility and ppv. $p = 100$ and $n = 50$. Thresholds: 0.05, 0.1, 0.2 and 0.3. The results of PACOSE are represented by the black line and the results of the `pls.net` function with the red line. **UPPER FIGURE**: sensibility as a function of the threshold, **LOWER FIGURE**: ppv as a function of the threshold.

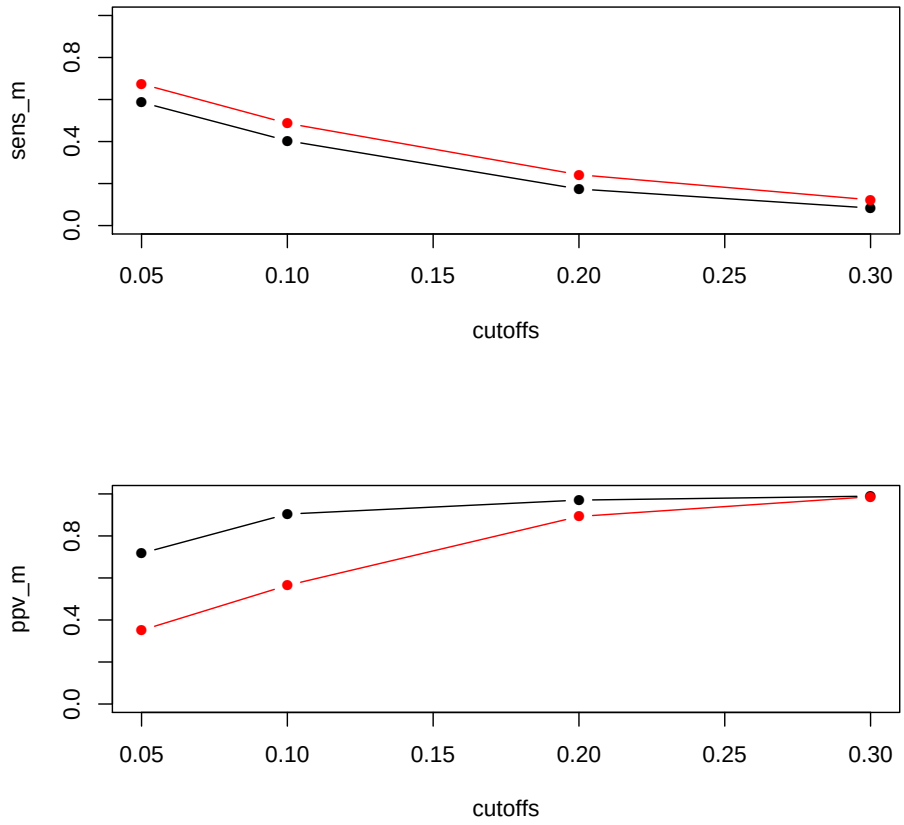

Figure 5: Sensibility and ppv. $p = 50$ and $n = 100$. Thresholds: 0.05, 0.1, 0.2 and 0.3. The results of PACOSE are represented by the black line and the results of the `pls.net` function with the red line. **UPPER FIGURE**: sensibility as a function of the threshold, **LOWER FIGURE**: ppv as a function of the threshold.