

High dimensional sparse covariance estimation via directed acyclic graphs

Philipp Rütimann and Peter Bühlmann

Seminar für Statistik

ETH Zentrum

CH-8092 Zürich, Switzerland

e-mail: rutimann@stat.math.ethz.ch; buehlmann@stat.math.ethz.ch

Abstract: We present a graph-based technique for estimating sparse covariance matrices and their inverses from high-dimensional data. The method is based on learning a directed acyclic graph (DAG) and estimating parameters of a multivariate Gaussian distribution based on a DAG. For inferring the underlying DAG we use the PC-algorithm [27] and for estimating the DAG-based covariance matrix and its inverse, we use a Cholesky decomposition approach which provides a positive (semi-)definite sparse estimate. We present a consistency result in the high-dimensional framework and we compare our method with the Glasso [12, 8, 2] for simulated and real data.

AMS 2000 subject classifications: Primary 62H12; secondary 62F12.

Keywords and phrases: Concentration matrix, covariance matrix, directed acyclic graphs, graphical Lasso, high-dimensional data, PC-algorithm.

Received November 2009.

Contents

1	Introduction	1134
2	Graph terminology and graphical models	1135
	2.1 Graphs	1135
	2.2 Graphical models and Markov properties	1136
3	Covariance estimation based on DAGs	1137
	3.1 Estimating the covariance matrix from a DAG	1137
	3.2 Inferring a directed acyclic graph	1139
	3.2.1 Estimating the CPDAG	1139
	3.2.2 The PC-DAG covariance estimator	1141
4	Consistency	1142
5	Simulation and real data analysis	1143
	5.1 Simulation study	1144
	5.1.1 DAG models	1144
	5.1.2 Non DAG models	1145
	5.2 Real data	1147
	5.2.1 Isoprenoid gene pathways in <i>Arabidopsis thaliana</i>	1147
	5.2.2 Breast Cancer data	1149
6	A robust PC-DAG covariance estimator	1149
	6.1 Simulation study for non-Gaussian data	1151

7 Summary and discussion 1151
 Appendix 1153
 A Proof of Lemma 4.1 1153
 B Modifications of the PC-DAG covariance estimator 1158
 References 1158

1. Introduction

Estimation of covariance matrices is an important part of multivariate analysis. There are many problems with high-dimensional data where an estimation of the covariance matrix is of interest, for example in principal component analysis or classification by discriminant analysis. Application areas where such problems arise include gene micro-arrays, imaging and image classification or text retrieval. In many of these applications, the primary goal is the estimation of the inverse of a covariance matrix Σ^{-1} , also known as the precision or concentration matrix, rather than the covariance Σ itself. In low-dimensional settings with $p < n$, where p denotes the row- or column-dimension of Σ and n the sample size, we can obtain an estimate of Σ^{-1} by estimation and inversion of the Gaussian maximum likelihood estimator $\hat{\Sigma}_{MLE}$. But when p is large, inversion of this estimate is problematic and its accuracy is very poor.

Recently, two classes for high-dimensional covariance estimation have emerged: those that rely on a natural ordering among variables and typically assuming that variables far apart in the ordering are only weakly correlated, and those which are invariant to variable permutation. Regularized estimation by banding or tapering [3, 13, 5] or using sparse Cholesky factors of the inverse covariance matrix relying on the natural ordering of the variables [31, 14, 19] are members of the first class of covariance estimators. When having no natural ordering among the variables, estimators should be permutation invariant with respect to indexing the variables. A popular approach to obtain a sparse permutation-invariant estimate is to add a Lasso penalty on the entries of the concentration matrix to the negative Gaussian log-likelihood [12, 8, 2, 26]. This amounts to shrinking some of the elements of the inverse covariance matrix exactly to zero. Alternatively, the Lasso can be used for inferring an undirected conditional independence graph using node-wise regressions [24] and a covariance estimate can then be obtained using the structure of the graph. Other approaches include a simple hard-thresholding of the elements of the unpenalized maximum likelihood estimator [4], with the disadvantage that the resulting estimate is not necessarily positive (semi-) definite.

The method which we present here is also invariant under permutation of the variables. The type of regularization which we pursue is based on exploiting a sparse graphical model structure first and then estimating the covariance matrix and its inverse using non-regularized estimation. Because of the sparsity of the graphical model structure, the second step does not need any regularization anymore. More precisely, we use a sparsely structured Cholesky decomposition of the concentration matrix for estimation of the covariance and concentration

matrix. To obtain the structure of such a Cholesky factor, we estimate a DAG (in fact, an equivalence class of DAGs). Thus, this approach enforces a completely different sparsity structure on the Cholesky factor than proposals for ordered data as in e.g. [3, 13, 5, 9].

For a given DAG, our approach equals the iterative conditional fitting (ICF) method presented in [10, 6] which reduces here to the standard technique of fitting Gaussian DAG models. Our contribution is to use an estimated DAG, i.e. an estimated equivalence class of DAGs from the PC-algorithm [27], and to analyze the method in the high-dimensional case taking the uncertainty of structure estimation of the equivalence class of DAGs into account. We argue in this paper that within the class of methods which are invariant under variable permutation, a graph-structured approach can be worthwhile for a range of scenarios, sometimes resulting in performance gains up to 30-50% over shrinkage methods.

In Section 2 we give a brief overview over graph terminology and graphical models. Section 3 introduces our methodology and we show asymptotic consistency of the method in the high-dimensional framework in Section 4. Simulations and real data examples are presented in Section 5 and we propose a robustified version of our procedure in Section 6.

2. Graph terminology and graphical models

2.1. Graphs

Let $G = (V, E)$ be a graph with a set of vertices V and a set of edges $E \subseteq V \times V$. In our context, we use $V = \{1, \dots, p\}$ corresponding to some random variables X_1, \dots, X_p .

A graph can be *directed*, *undirected* or *partially directed*. An edge between two vertices, for example i and j , is called *directed* if the edge has an arrowhead: $i \leftarrow j$ or $i \rightarrow j$. An edge without arrowhead is an *undirected* edge: $i - j$. A graph in which all edges are directed is called a *directed graph*; and vice-versa, a graph in which no edge is directed is called an *undirected graph*. A graph which may contain both directed and undirected edges is called a *partially directed graph*. The underlying undirected graph of a (partially) directed graph G which we derive by removing all the arrowheads is called the *skeleton* of G .

Two vertices i and j are *adjacent* if there is any kind of edge between them. The *adjacency set* of a vertex i , denoted by $adj(i, G)$, is the set of all vertices that are adjacent to i in G . A *path* is a sequence of vertices $\{1, \dots, k\}$ such that i is adjacent to $i + 1$ for each $i = 1, \dots, k - 1$. A *directed path* is a path with directed edges that follows the direction of the arrows. When the first and last vertices coincide, the directed path is called a *directed cycle*. An *acyclic* graph is a graph that contains no directed cycles. A directed graph with no directed cycles is called *directed acyclic graph* (DAG).

If $i \rightarrow j$, then i is called a *parent* of j and j is called a *child* of i . The set of parents of i in G is denoted as $pa(i)$ and the set of children as $ch(i)$. If there

is a directed path from i to j , then i is called an *ancestor* of j and j is called an *descendant* of i . The set of ancestors of i is denoted as $an(i)$, the set of descendants as $de(i)$ and the set of non-descendants as $nde(i)$. A *v-structure* in a graph G is an ordered triple of vertices (i, j, r) , such that $i \rightarrow j$ and $j \leftarrow r$, and i and r are not adjacent in G . The vertex j is then called a *collider*.

A path Q from i to j in a directed acyclic graph G is said to be *blocked* by S , if it contains a vertex $v \in Q$ such that either (i) $v \in S$ and v is no collider; or (ii) $v \notin S$ nor has v any descendants in S , and v is a collider. A path that is not blocked by S is said to be *active*. Two subsets A and B are said to be *d-separated* by S if all paths from A to B are blocked by S . In other words, there is no active path from A to B .

2.2. Graphical models and Markov properties

Graphical models form a probabilistic tool to analyze and visualize conditional dependence between random variables, using some encoding with edges in the graph. Fundamental to the idea of a graphical model is, based on graph theoretical concepts and algorithms, the notion of modularity where a complex system is built by combining simpler parts. One can distinguish between three main graphical models. Here we focus on DAG models, where all the edges of the graph are directed. According to [18], a DAG model may exhibit several directed Markov properties. In the following, we present only two of them.

We use the following notation. Let P denote the distribution of (X_1, \dots, X_p) . For $x \in \mathbf{R}^p$, we denote by $x_A = \{x_j; j \in A\}$ for $A \subseteq V = \{1, \dots, p\}$ and analogously for the random vector X_A . Furthermore, for disjoint subsets A, B and S , we denote by $X_A \perp\!\!\!\perp X_B | X_S$ conditional independence between X_A, X_B given X_S .

Definition 2.1. [Directed global Markov property]

Let A, B and S be disjoint subsets of V and G a DAG on V . If

$$X_A \perp\!\!\!\perp X_B | X_S$$

whenever A and B are *d-separated* by S in the graph G , we say P obeys the directed global Markov property relative to the DAG G .

Definition 2.2. [Recursive factorization property]

We say that P admits a recursive factorization according to a DAG G whenever there exist non-negative functions $f_i(\cdot|\cdot)$ ($i = 1, \dots, p$), such that

$$\int f_i(x_i|x_{pa(i)})\nu(dx_i) = 1$$

and P has a density f with respect to the measure ν , where

$$f(X_1, \dots, X_p) = \prod_{i=1}^p f_i(X_i | X_{pa(i)}).$$

If the density f of P is strictly positive, as for example in the case of a multivariate Gaussian distribution, both Markov properties in Definitions 2.1 and 2.2 are equivalent. For more details see [18, pp. 46–52].

A DAG model encodes conditional independence relationships via the notion of d -separation. Several DAGs could encode the same set of conditional independence relationships. These DAGs form an equivalence class, consisting of DAGs with the same skeleton and v -structures. A *complete partially directed acyclic graph* (CPDAG) uniquely describes such an equivalence class. In fact, directed edges in the CPDAG are common to all DAGs in the equivalence class. Undirected edges in the CPDAG correspond to edges that are directed one way in some DAGs and another way in other DAGs of the equivalence class. The absence of edges in the CPDAG means that all DAG members in the equivalence class have no corresponding edge. If all the conditional independence relationships of a distribution P and no additional conditional independence relations, can be inferred from the graph, we say that the distribution P is *faithful* to the DAG G . More precisely, if P is faithful to the DAG G : for any triple of disjoint sets A, B and S in V ,

$$X_A \perp\!\!\!\perp X_B | X_S \Leftrightarrow A \text{ and } B \text{ are } d\text{-separated by } S \text{ in } G.$$

Note that the directed global Markov property in Definition 2.1 implies the implication from the right- to the left-hand side; the other direction is due to the faithfulness assumption.

3. Covariance estimation based on DAGs

Our methodology is based on two steps. We first infer the CPDAG, i.e. the equivalence class of DAGs, and we then estimate the covariance (concentration) matrix based on the CPDAG structure.

We assume throughout the paper that the data are

$$\begin{aligned} X^{(r)} &= (X_1^{(r)}, \dots, X_p^{(r)}), \quad r = 1, \dots, n \\ X^{(1)}, \dots, X^{(n)} &\text{ i.i.d. } \sim P \end{aligned} \tag{1}$$

with P being multivariate normal $\mathcal{N}_p(0, \Sigma)$, Markovian (as in Definition 2.1 or 2.2) and faithful to a DAG G .

The Gaussian assumption implies that $\mathbf{E}[X_i | X_{pa(i)}]$ is linear in $X_{pa(i)}$ which will be useful in the second estimation step for the concentration or covariance matrix. Moreover, it allows us to equate conditional independence with zero partial correlation which makes estimation for the CPDAG much easier.

3.1. Estimating the covariance matrix from a DAG

We first assume that the underlying DAG is given. Using the factorization property from Definition 2.2 in Section 2.2 we have:

$$f(X_1, \dots, X_p) = \prod_{i=1}^p f(X_i | X_{pa(i)}).$$

We use here and in the sequel the short-hand notation $f(\cdot|\cdot)$ instead of $f_i(\cdot|\cdot)$. For data as in (1), we can then write the likelihood function as

$$L = \prod_{r=1}^n f(X_1^{(r)}, \dots, X_p^{(r)}) = \prod_{r=1}^n \prod_{i=1}^p f(X_i^{(r)} | X_{pa(i)}^{(r)}) = \prod_{i=1}^p \prod_{r=1}^n f(X_i^{(r)} | X_{pa(i)}^{(r)}).$$

Using the Gaussian assumption this leads to the likelihood in terms of the unknown parameter Σ (or Σ^{-1} respectively)

$$L(\Sigma) = \prod_{i=1}^p L_i(\mu_{i|pa(i)}, \Sigma_{i|pa(i)})$$

where $\mu_{i|pa(i)}$ and $\Sigma_{i|pa(i)}$ are the conditional expectation and variance of X_i given the parents $X_{pa(i)}$. Note that the conditional covariance is a fixed quantity whereas the conditional mean depends on the variables $X_{pa(i)}$. For a single random variable X_i we have:

$$\begin{aligned} \mu_{i|pa(i)} &= \mathbf{E}[X_i | X_{pa(i)}] = \mu_i + \Sigma_{i,pa(i)}(\Sigma_{pa(i),pa(i)})^{-1}(X_{pa(i)} - \mu_{pa(i)}) \\ &= \Sigma_{i,pa(i)}(\Sigma_{pa(i),pa(i)})^{-1}X_{pa(i)}, \end{aligned} \tag{2}$$

with assumption $\mu_i = 0 \forall i$ from above, and:

$$\Sigma_{i|pa(i)} = \Sigma_{i,i} - \Sigma_{i,pa(i)}(\Sigma_{pa(i),pa(i)})^{-1}\Sigma_{pa(i),i}. \tag{3}$$

The expressions $\Sigma_{i,pa(i)}$ and $\Sigma_{pa(i),pa(i)}$ are sub-matrices formed by selecting the corresponding rows and columns from the full covariance matrix Σ . For example, $\Sigma_{i,pa(i)}$ is the sub-matrix (or vector) of Σ with row i and columns $j \in pa(i)$. The values $\mu_{i|pa(i)}$ and $\Sigma_{i|pa(i)}$, in the i th factor L_i of the likelihood, are connected to regression, as described next.

Consider for each node i a regression from X_i on $X_{pa(i)}$, where $X_i | X_{pa(i)} \sim \mathcal{N}(\mu_{i|pa(i)}, \Sigma_{i|pa(i)})$. We can represent these p regressions in matrix notation as follows:

$$A \begin{bmatrix} X_1 \\ \vdots \\ X_p \end{bmatrix} = \epsilon \tag{4}$$

where A is a $p \times p$ matrix corresponding to the regressions and ϵ is the vector of the error terms. That is:

$$A_{ij} = \begin{cases} -(\Sigma_{i,pa(i)}(\Sigma_{pa(i),pa(i)})^{-1})_j & \text{if } j \in pa(i) \\ 1 & \text{if } j = i \\ 0 & \text{otherwise} \end{cases}.$$

Now we can easily compute Σ or Σ^{-1} , because we can write (4) as

$$(X_1, \dots, X_p)^T = A^{-1}\epsilon.$$

Hence,

$$\Sigma = \text{Cov}((X_1, \dots, X_p)^T) = \text{Cov}(A^{-1}\epsilon) = A^{-1} \text{Cov}(\epsilon) (A^{-1})^T,$$

where

$$\text{Cov}(\epsilon) = \begin{bmatrix} \Sigma_{1|pa(1)} & & 0 \\ & \ddots & \\ 0 & & \Sigma_{p|pa(p)} \end{bmatrix}. \tag{5}$$

We then also easily obtain

$$\Sigma^{-1} = (A^{-1} \text{Cov}(\epsilon) (A^{-1})^T)^{-1} = A^T \text{Cov}(\epsilon)^{-1} A.$$

See [7, Chap. 3] and [21] for more details.

Since Σ and Σ^{-1} are based on the structure of the DAG G (via the matrix A) we write Σ_G and Σ_G^{-1} . An estimator is now constructed as follows. Consider the maximum likelihood estimator

$$\hat{\Sigma}^{MLE} = n^{-1} \sum_{j=1}^n (X^{(j)} - \bar{X})(X^{(j)} - \bar{X})^T \tag{6}$$

as an “initial” estimator $\hat{\Sigma}_{init}$ of Σ and use the following plug-in estimators:

$$\begin{aligned} \hat{\Sigma}_G &= \hat{A}^{-1} \widehat{\text{Cov}}(\epsilon) (\hat{A}^{-1})^T, \\ \hat{\Sigma}_G^{-1} &= \hat{A}^T \widehat{\text{Cov}}(\epsilon)^{-1} \hat{A}, \end{aligned} \tag{7}$$

where \hat{A} and $\widehat{\text{Cov}}(\epsilon)$ are as in (5) but with the plug-in estimates $\hat{\Sigma}_{i,pa(i)}^{MLE}$, $(\hat{\Sigma}_{pa(i),pa(i)}^{MLE})^{-1}$ (for \hat{A}) and $\hat{\Sigma}_{i|pa(i)}^{MLE}$ (for $\widehat{\text{Cov}}(\epsilon)$) using formula (3).

Note that the estimators in (7) are automatically positive semi-definite having eigenvalues ≥ 0 (and positive definite assuming $\hat{\Sigma}_{i|pa(i)}^{MLE} > 0$ for all j , which would fail only in very pathological cases). Furthermore, we could use another “initial” estimator than $\hat{\Sigma}^{MLE}$ for estimating $\Sigma_{i,pa(i)}$, $\Sigma_{pa(i),i}$ and $\Sigma_{pa(i),pa(i)}$. We are exploiting this possibility for a robustified version, as discussed in Section 6. Finally, the estimator in (7) is implemented in the R-package `ggm` [6].

3.2. Inferring a directed acyclic graph

The conditional dependencies between X_1, \dots, X_p and hence the DAG are usually not known. We use the PC-algorithm [27] with estimated conditional dependencies to infer the corresponding CPDAG G , i.e. the equivalence class of DAGs (inferring the true DAG itself is well-known to be impossible due to identifiability problems).

Estimation of the skeleton and partial orientation of edges are the two major parts of inferring a CPDAG. In the following we will describe these two steps.

3.2.1. Estimating the CPDAG

In a first step, we start from a complete undirected graph. When two variables X_i, X_j are found to be conditional independent given X_K for some set K , the

edge $i - j$ is deleted: details are given in Algorithm 1. In a second step, the edges are oriented using the conditioning sets K which made edges drop out in the first step: details are given in Algorithm 2.

In the first step of the PC-algorithm, we need to estimate the conditional independence relations between X_1, \dots, X_p . Under the Gaussian assumption conditional independencies can be inferred from partial correlations. Then, the conditional independence of X_i and X_j given $X_K = \{X_r; r \in K\}$, where $K \subseteq \{1, \dots, p\} \setminus \{i, j\}$, is equivalent to the following: the partial correlation of X_i and X_j given $\{X_r; r \in K\}$, denoted by $\rho_{i,j|K}$, is equal to zero. This is an elementary property of the multivariate normal distribution, see [18, Prop. 5.2]. Hence to obtain estimates of conditional independencies we can use estimated partial correlations $\hat{\rho}_{i,j|K}$. For testing whether an estimated partial correlation is zero or not, we apply Fisher's z-transform

$$Z(i, j | K) = \frac{1}{2} \log \left(\frac{1 + \hat{\rho}_{i,j|K}}{1 - \hat{\rho}_{i,j|K}} \right).$$

Since $Z(i, j | K)$ has a $\mathcal{N}(0, (n - |K| - 3)^{-1})$ distribution if $\rho_{i,j|K} = 0$ [1], we have evidence that $\rho_{i,j|K} \neq 0$ if

$$\sqrt{n - |K| - 3} |Z(i, j | K)| > \Phi^{-1}(1 - \frac{\alpha}{2}),$$

where Φ is the cumulative distribution function of the standard Normal distribution and the significance level $0 < \alpha < 1$ is a tuning (threshold) parameter of the PC-algorithm described in Algorithms 1 and 2.

Algorithm 1: The PC-algorithm for the skeleton

Input: z-transform of estimated partial correlations, tuning parameter α
Output: Skeleton of CPDAG G , separation sets S (used later for directing the skeleton)

- 1 Form the complete undirected graph \tilde{G} on the set $\{1, \dots, p\}$;
- 2 $l = -1$; $G = \tilde{G}$;
- 3 **repeat**
- 4 $l = l + 1$;
- 5 **repeat**
- 6 Select an ordered pair of adjacent variables i, j in G such that $|adj(i, G) \setminus \{j\}| \geq l$;
- 7 **repeat**
- 8 Choose $K \subseteq adj(i, G) \setminus \{j\}$ with $|K| = l$;
- 9 **if** $\sqrt{n - |K| - 3} |Z(i, j | K)| \leq \Phi^{-1}(1 - \alpha/2)$ **then**
- 10 Delete edge i, j ;
- 11 Denote this new graph by G ;
- 12 Save K in $S(i, j)$ and $S(j, i)$;
- 13 **until** edge i, j is deleted or all $K \subseteq adj(i, G) \setminus \{j\}$ with $|K| = l$ have been chosen ;
- 14 **until** all ordered pairs of adjacent variables i and j , such that $|adj(i, G) \setminus \{j\}| \geq l$ and $K \subseteq adj(i, G) \setminus \{j\}$ with $|K| = l$, have been tested for conditional independence ;
- 15 **until** for each ordered pair of adjacent nodes i, j : $|adj(i, G) \setminus \{j\}| < l$;

If $\rho_{i,j|K} = 0$ is plausible, the edge $i - j$ is deleted and K is saved in $S(i, j)$. We call $S = \{S(i, j); i, j \in \{1, \dots, p\}, i \neq j\}$ the separation sets. These sets are important for extending the estimated skeleton to a CPDAG as described below in Algorithm 2.

Algorithm 2: The PC-algorithm: extending the skeleton to a CPDAG

Input: Skeleton G of CPDAG, separation sets S
Output: CPDAG

```

1 forall pairs of nonadjacent variables  $i, j$  with common neighbor  $k$  do
2   if  $k \notin S(i, j)$  then
3     Replace  $i - k - j$  in Skeleton of  $G$  by  $i \rightarrow k \leftarrow j$ ;
4 repeat
5   R1 Orient  $j - k$  into  $j \rightarrow k$  whenever there is an arrow  $i \rightarrow j$  such that  $i$  and  $k$  are nonadjacent;
6   R2 Orient  $i - j$  into  $i \rightarrow j$  whenever there is a chain  $i \rightarrow k \rightarrow j$ ;
7   R3 Orient  $i - j$  into  $i \rightarrow j$  whenever there are two chains  $i \rightarrow k \rightarrow j$  and  $i \rightarrow l \rightarrow j$  such that  $k$  and  $l$  are nonadjacent;
8 until no more orienting of undirected edges is possible by the rules R1 to R3 ;

```

[23] showed that the rules in Algorithm 2 are sufficient to orient all arrows in the CPDAG, see also [25, pp. 50]. The PC-algorithm, described in Algorithms 1 and 2, yields an estimate $\hat{G}_{\text{CPDAG}}(\alpha)$ of the true underlying CPDAG which depends on the tuning parameter α .

3.2.2. The PC-DAG covariance estimator

Having an estimate $\hat{G}_{\text{CPDAG}}(\alpha)$ of the CPDAG, we pick any DAG $\hat{G}_{\text{DAG}}(\alpha)$ in the equivalence class of the CPDAG. This can be done by directing undirected edges in the CPDAG at random without creating additional v-structures or cycles. The estimate for the covariance and concentration matrix is then:

$$\hat{\Sigma}_{\hat{G}_{\text{DAG}}(\alpha)}, \hat{\Sigma}_{\hat{G}_{\text{DAG}}(\alpha)}^{-1} \text{ as in formula (7),} \tag{8}$$

and since the PC-algorithm for DAGs is involved, we call it the PC-DAG covariance estimator. Its only tuning parameter is α used in the PC-algorithm. As described in Section 3.2.1, it has the interpretation of a significance level for a single test whether a partial correlation is zero or not. The choice of this tuning parameter α can be done using cross-validation of the negative out-of-sample log-likelihood.

We remark that the zeros in $\hat{\Sigma}_{\hat{G}_{\text{DAG}}(\alpha)}^{-1}$ are the same for any choice of a DAG in the estimated CPDAG $\hat{G}_{\text{CPDAG}}(\alpha)$. However, the non-zero estimated elements of the estimated matrices will be slightly different. To avoid an unusual random realization when selecting a DAG from $\hat{G}_{\text{CPDAG}}(\alpha)$, we can sample many DAGs and average the corresponding estimates for Σ^{-1} or Σ .

In some cases, we need some small modifications of the PC-DAG covariance estimator which are described in Appendix B. Estimation of a CPDAG as de-

scribed in Algorithm 1 and 2 is efficiently implemented in the R-package `pccalg`, as described in its reference manual [17].

4. Consistency

We prove asymptotic consistency of the estimation method in high-dimensional settings where the number of variables p can be much larger than the sample size n . In such a framework, the model depends on n and this is reflected notationally by using the subscript n . We assume:

- (A) The data is as in (1) with distribution P_n of (X_1, \dots, X_{p_n}) being multivariate normal $\mathcal{N}(0, \Sigma_n)$, Markovian as in Definition 2.1 or 2.2 and faithful to a DAG G_n .
- (B) The variances satisfy: $\text{Var}(X_i) = \sigma_{n;i}^2 \leq \sigma^2 < \infty$ for all $i = 1, \dots, p_n$.
- (C) The dimension $p_n = O(n^a)$ for some $0 \leq a < \infty$.
- (D) The maximal cardinality $q_n = \max_{i=1, \dots, p_n} |\text{adj}(i, G_n)|$ of the adjacency sets in G_n satisfies $q_n = O(n^{\frac{1}{2}-b})$ for some $0 < b \leq 1/2$.
- (E) For any $i, j \in 1, \dots, p_n$, let $\rho_{n;i,j|S}$ denote the partial correlation between X_i and X_j given S , where $S \in \{1, \dots, p_n\} \setminus \{i, j\}$. These partial correlations are bounded above and below:

$$\sup_{n, i \neq j, S} |\rho_{n;i,j|S}| \leq M$$

for some $M < 1$, and

$$\inf_{i,j,S} \{|\rho_{n;i,j|S}|; \rho_{n;i,j|S} \neq 0\} \geq c_n$$

with $c_n^{-1} = O(n^d)$ for some $0 < d < 1/4 + b/2$, where b is as in (D).

- (F) For every DAG in the equivalence class of the true underlying CPDAG (induced by the distribution in assumption (A)), the conditional variances satisfy the following bound:

$$\inf_{1 \leq i \leq p_n, j \in \text{pa}(i)} \text{Var}(X_j | X_{\text{pa}(i) \setminus j}) \geq r > 0,$$

$$\inf_{1 \leq i \leq p_n} \text{Var}(X_i | X_{\text{pa}(i)}) \geq r > 0.$$

Assumption (C) allows the number of variables p_n to grow as an arbitrary polynomial in the sample size and reflects the high-dimensional setting. Assumption (D) is a sparseness assumption, requiring that the maximal number of neighbors per node grows at a slower rate than $O(n^{\frac{1}{2}})$. Assumption (F) is a regularity condition on the conditional variances. Assumption (E), in particular the second part, is a restriction which corresponds to the detectability of non-zero partial correlations: obviously, we cannot consistently detect non-zero partial correlations of smaller order than $\frac{1}{\sqrt{n}}$. For sparse graphs with b close to $1/2$ in (D), the value d close to $1/2$ is allowed. i.e. close to the $1/\sqrt{n}$ detection

limit. Under assumptions (A)-(E), the PC-algorithm was shown to be consistent for inferring the true underlying CPDAG [15, Th.2]. More precisely, we denote by $\hat{G}_{\text{CPDAG};n}(\alpha)$ the estimate for the underlying CPDAG, using the PC-algorithm with tuning parameter α (Algorithms 1 and 2), and by $G_{\text{CPDAG};n}$ the true underlying CPDAG. Then, assuming (A)-(E) and for $\alpha_n = 2(1 - \Phi(n^{1/2}c_n/2))$:

$$P[\hat{G}_{\text{CPDAG};n}(\alpha_n) = G_{\text{CPDAG};n}] \rightarrow 1 \quad (n \rightarrow \infty). \tag{9}$$

Concerning the consistency of DAG based estimation of the concentration matrix, we have the following new result.

Lemma 4.1. *Under assumptions (A)-(D) and (F) the following holds. For any DAG G in the equivalence class of the true underlying CPDAG and using the estimator $\hat{\Sigma}_G^{-1}$ in (7):*

$$\sup_{i,j} \left| \hat{\Sigma}_{G,n;i,j}^{-1} - \Sigma_{n;i,j}^{-1} \right| \xrightarrow{P} 0 \quad (n \rightarrow \infty).$$

A proof is given in the Appendix. We then obtain the main theoretical result.

Theorem 4.1. *Under assumptions (A)-(F) and using the tuning parameter $\alpha_n = 2(1 - \Phi(n^{1/2}c_n/2))$ in the PC-algorithm, the following holds for the estimator in (8):*

$$\sup_{i,j} \left| \hat{\Sigma}_{\hat{G}_{\text{DAG}}(\alpha),n;i,j}^{-1} - \Sigma_{n;i,j}^{-1} \right| \xrightarrow{P} 0 \quad (n \rightarrow \infty).$$

Proof. The estimate $\hat{G}_{\text{DAG};n}(\alpha)$ is a DAG element of the estimated equivalence class encoded by the estimated CPDAG $\hat{G}_{\text{CPDAG};n}(\alpha)$. Denote this DAG by G_* . Consider the event

$$A_n = \{\hat{G}_{\text{CPDAG};n}(\alpha_n) = G_{\text{CPDAG};n}\},$$

whose probability $P[A_n] \rightarrow 1$ ($n \rightarrow \infty$), see (9). On A_n , G_* must be a DAG element of the true equivalence class $G_{\text{CPDAG};n}$ and hence on A_n , Lemma 4.1 yields consistency:

$$\sup_{i,j} \left| \hat{\Sigma}_{\hat{G},n;i,j}^{-1} - \Sigma_{n;i,j}^{-1} \right| = \sup_{i,j} \left| \hat{\Sigma}_{G_*,n;i,j}^{-1} - \Sigma_{n;i,j}^{-1} \right| \xrightarrow{P} 0 \quad (n \rightarrow \infty).$$

Since $P[A_n] \rightarrow 1$, the proof is complete. □

5. Simulation and real data analysis

We examine the behavior of our PC-DAG estimator using simulated and real data and compare it to the Glasso method [12, 2]. The Glasso is defined as:

$$\hat{\Sigma}_{\text{Glasso}}^{-1} = \underset{\Sigma^{-1} \text{ non-neg. def.}}{\text{arg min}} \quad (-\log \det \Sigma^{-1} + \text{tr}(\hat{\Sigma}^{MLE} \Sigma^{-1}) + \lambda \|\Sigma^{-1}\|_1) \tag{10}$$

where $\hat{\Sigma}^{MLE}$ is the empirical covariance matrix in (6), $\|\Sigma^{-1}\|_1 = \sum_{i < j} |\Sigma_{ij}^{-1}|$ and the minimization is over non-negative definite matrices.

All computations are done with the R-packages `pcalg` [17] and `glasso`.

5.1. Simulation study

We consider a DAG and a non-DAG model for generating the data.

5.1.1. DAG models

We focus on the following class of DAG models. We generate recursively

$$X_1 = \epsilon_1 \sim \mathcal{N}(0, 1),$$

$$X_i = \sum_{r=1}^{i-1} B_{ir} X_r + \epsilon_i \quad (i = 2, \dots, p),$$

where $\epsilon_1, \dots, \epsilon_p$ i.i.d. $\sim \mathcal{N}(0, 1)$ and B is an adjacency matrix generated as follows. We first fill the matrix B with zeros and replace every matrix entry in the lower triangle by independent realizations of Bernoulli(s) random variables with success probability s where $0 < s < 1$. Afterwards, we replace each entry having a 1 in the matrix B by independent realizations of a Uniform([0,1,1]) random variable. If $i < j$ and $B_{ji} \neq 0$ the corresponding DAG has a directed edge from node i to node j . The variables X_1, \dots, X_p have a multivariate Gaussian distribution with mean zero and covariance Σ which can be computed from B . We consider this model for different settings of n , s and p :

- D1: $n = 30, s = 0.01, p = 40, 50, 60, 70, 80, 90, 100, 110, 120$
- D2: $n = 50, s = 0.01, p = 40, 50, 60, 70, 80, 90, 100, 110, 120$
- D3: $n = 30, s = 0.05, p = 40, 50, 60, 70, 80, 90, 100, 110, 120$
- D4: $n = 50, s = 0.05, p = 40, 50, 60, 70, 80, 90, 100, 110, 120$

The settings D1 to D4 mainly differ in the sparsity s of the generated data, which is related to the expected neighborhood size $\mathbf{E}[adj(i, G)] = s(p - 1)$ for all i . For each of these settings we estimate the covariance and the concentration matrix with both methods, our PC-DAG and the Glasso estimator.

We use two different performance measures to compare the two estimation techniques. First, the Frobenius norm of the difference between the estimated and the true matrix $\|\hat{\Sigma} - \Sigma\|_F$ and $\|\hat{\Sigma}^{-1} - \Sigma^{-1}\|_F$. And second, the Kullback-Leibler Loss $\Delta_{KL}(\hat{\Sigma}^{-1}, \Sigma^{-1}) = tr(\Sigma \hat{\Sigma}^{-1}) - \log |\Sigma \hat{\Sigma}^{-1}| - p$.

We sample data $X^{(1)}, \dots, X^{(n)}$ i.i.d. from the DAG model described above for each value of p in settings D1-D4. Then we derive, on a separate validation data-set $X^{(1)*}, \dots, X^{(n)*}$, the optimal value of the tuning parameters α (PC-DAG) or λ (Glasso), with respect to the negative Gaussian log-likelihood. The two different performance measures are evaluated for the estimates based on the training data $X^{(1)}, \dots, X^{(n)}$ with optimal tuning parameter choice based on the validation data. All results are based on 50 independent simulation runs.

Figures 1 and 2 show that in the sparse settings D1 and D2, the PC-DAG estimator clearly outperforms Glasso. Concerning the more dense settings D3 and D4, the PC-DAG method degrades only for the covariance matrix, whereas for the inverse covariance matrix Σ^{-1} , the figures still show an improvement of the PC-DAG estimator compared to the Glasso. If we match Figure 1 (a)

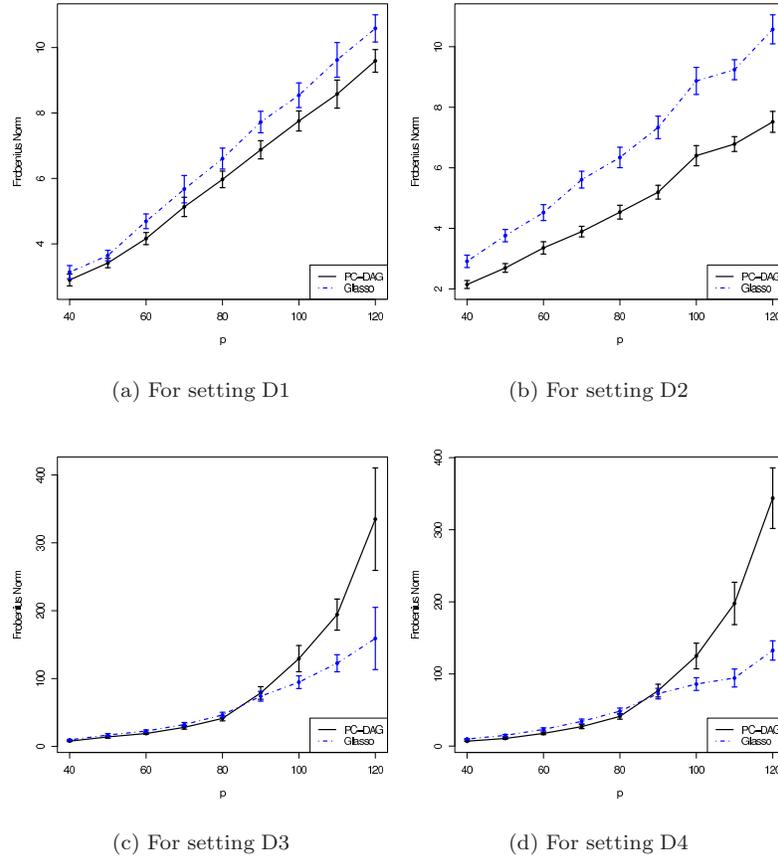


FIG 1. Plots of $\|\hat{\Sigma} - \Sigma\|_F$ for DAG models. Vertical bars indicate (pointwise) 95% confidence intervals.

with Figure 1 (b) and Figure 2 (a) with Figure 2 (b), we see that for a small increase of the sample size the Glasso improves substantially less compared to the PC-DAG estimator. The results in terms of the Kullback-Leibler loss are summarized in Table 1.

5.1.2. Non DAG models

Next we generate data from a non-DAG model proposed by [26]. The concentration matrix equals

$$\Sigma^{-1} = B + \delta I,$$

where each off-diagonal entry in B is generated independently and equals 0.5 with probability π or 0 with probability $1 - \pi$, all diagonal entries of B are zero, and δ is chosen such that the condition number of Σ^{-1} is p . The concentration

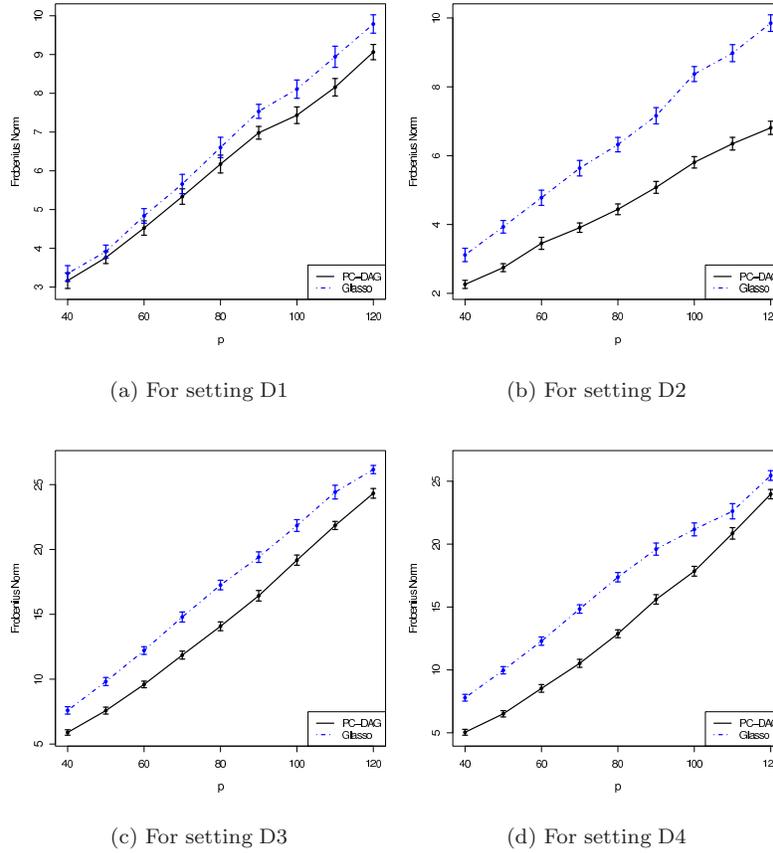


FIG 2. Plots of $\|\hat{\Sigma}^{-1} - \Sigma^{-1}\|_F$ for DAG models. Vertical bars indicate (pointwise) 95% confidence intervals.

matrices, which we generate from this model vary in their level of sparsity: for $\Sigma_{(1)}^{-1}$ we take $\pi = 0.1$ and for $\Sigma_{(2)}^{-1}$ we choose $\pi = 0.5$, i.e. $\Sigma_{(1)}^{-1}$ is sparser than $\Sigma_{(2)}^{-1}$. Note that the expected numbers of non-zero entries in $\Sigma_{(1)}^{-1}$ and $\Sigma_{(2)}^{-1}$ are proportional to p^2 .

We generate Gaussian data $X^{(1)}, \dots, X^{(n)}$ i.i.d. $\sim \mathcal{N}_p(0, \Sigma)$ with Σ^{-1} constructed as above, according to the following settings:

- nD1: $n = 30, \pi = 0.1, p = 40, 50, 60, 70, 80, 90, 100, 110, 120$
- nD2: $n = 50, \pi = 0.1, p = 40, 50, 60, 70, 80, 90, 100, 110, 120$
- nD3: $n = 30, \pi = 0.5, p = 40, 50, 60, 70, 80, 90, 100, 110, 120$
- nD4: $n = 50, \pi = 0.5, p = 40, 50, 60, 70, 80, 90, 100, 110, 120$

We tune and compare the estimation methods as described in Section 5.1.1.

In Figures 3 and 4 we see that in case of the dense model with $\pi = 0.5$, the two methods do not differ much (some of the differences are so small that they

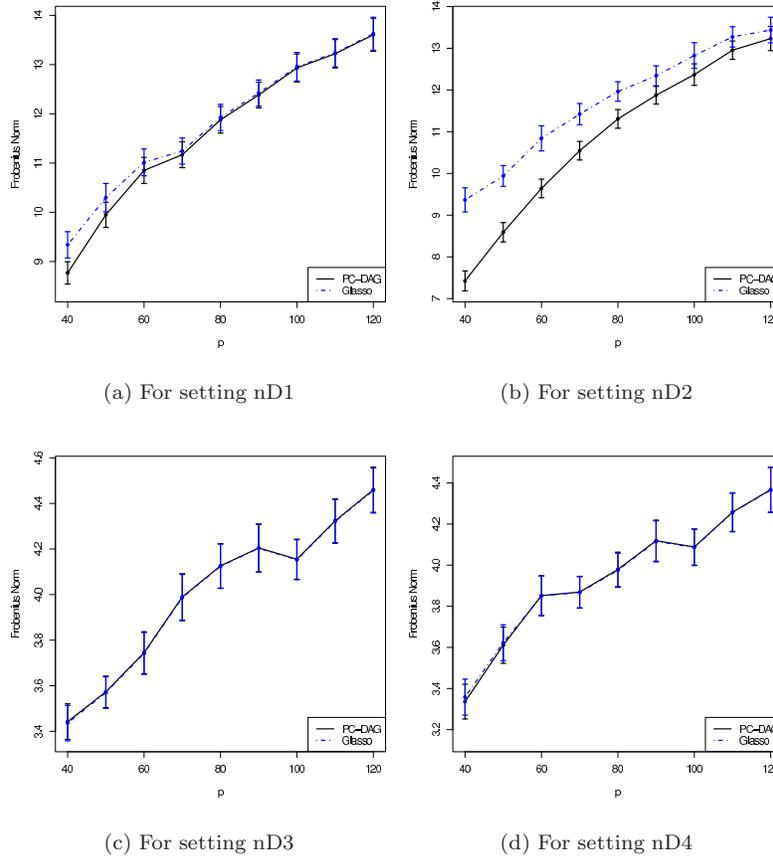


FIG 3. Plots of $\|\hat{\Sigma} - \Sigma\|_F$ for non DAG models. Vertical bars indicate (pointwise) 95% confidence intervals.

are invisible on the scales shown in the plots). But for the sparse model with $\pi = 0.1$ we observe that our PC-DAG estimator is better than the Glasso, in particular for the setting nD2. The results in terms of the Kullback-Leibler loss are summarized in Table 1.

5.2. Real data

In this section we compare the two estimation methods for real data.

5.2.1. Isoprenoid gene pathways in Arabidopsis thaliana

We analyze the gene expression data from the isoprenoid biosynthesis pathway in *Arabidopsis thaliana* given in [30]. Isoprenoids comprehend the most diverse class of natural products and have been identified in many different organisms.

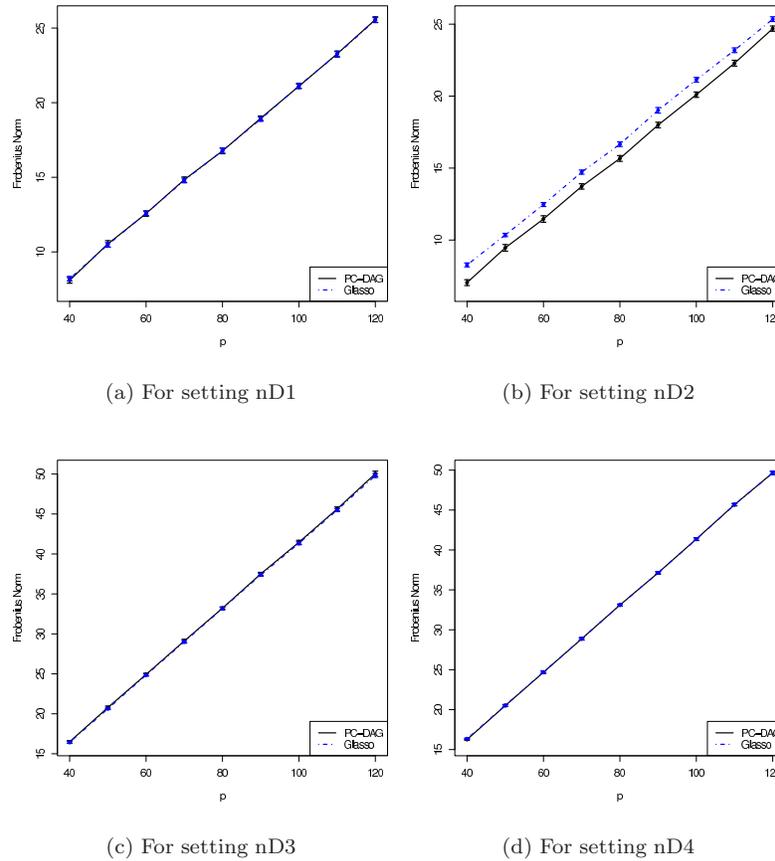


FIG 4. Plots of $\|\hat{\Sigma}^{-1} - \Sigma^{-1}\|_F$ for non DAG models. Vertical bars indicate (pointwise) 95% confidence intervals.

In plants isoprenoids play important roles in a variety of processes such as photosynthesis, respiration, regulation of growth and development.

This data set consists of $p = 39$ isoprenoid genes for which we have $n = 118$ gene expression patterns under various experimental conditions. As performance measure we use the 10-fold cross-validated negative Gaussian log-likelihood for centered data.

The results are described in Figure 5. We find that none of the two methods performs substantially better than the other and the slight superiority of Glasso is in the order of 1% only. The marginal difference in the negative log-likelihood between the two estimation techniques may be due to the high noise in the data.

TABLE 1
Kullback-Leibler Loss (standard error in parentheses)

Kullback-Leibler Loss				
DAG				
$n = 30$				
p	$s = 0.01$ (D1)		$s = 0.05$ (D3)	
	Glasso	PC-DAG	Glasso	PC-DAG
40	3.78(0.17)	3.38(0.16)	13.64(0.41)	9.27(0.29)
80	12.75(0.34)	11.36(0.29)	54.63(0.9)	41.69(0.67)
120	25.5(0.41)	22.93(0.42)	79.34(1.35)	104.43(1.47)
DAG				
$n = 50$				
p	$s = 0.01$ (D2)		$s = 0.05$ (D4)	
	Glasso	PC-DAG	Glasso	PC-DAG
40	3.12(0.15)	1.88(0.08)	13.3(0.31)	6.26(0.18)
80	11.07(0.26)	6.32(0.17)	53.08(1.22)	31.83(0.53)
120	24.35(0.47)	13.76(0.27)	66.21(2.78)	87.11(0.93)
non DAG				
$n = 30$				
p	Model $\Sigma_{(1)}^{-1}$ (nD1)		Model $\Sigma_{(2)}^{-1}$ (nD3)	
	Glasso	PC-DAG	Glasso	PC-DAG
40	15.61(0.21)	14.91(0.22)	13.53(0.16)	13.71(0.16)
80	35.63(0.45)	35.9(0.49)	29.36(0.33)	29.49(0.33)
120	56.44(0.67)	56.88(0.7)	45.34(0.45)	45.76(0.46)
non DAG				
$n = 50$				
p	Model $\Sigma_{(1)}^{-1}$ (nD2)		Model $\Sigma_{(2)}^{-1}$ (nD4)	
	Glasso	PC-DAG	Glasso	PC-DAG
40	15.38(0.24)	10.58(0.18)	12.76(0.16)	12.91(0.17)
80	34.28(0.4)	32.13(0.3)	27.49(0.28)	27.68(0.34)
120	53.69(0.7)	53.16(0.67)	42.85(0.5)	43.08(0.5)

5.2.2. Breast Cancer data

Next, we explore the performance on a gene expression data set from breast tumor samples. The tumor samples were selected from the Duke Breast Cancer SPORE tissue bank on the basis of several criteria. For more details on the data set see [29]. The data matrix monitors $p = 7129$ genes in $n = 49$ breast tumor samples. We only use the 100 variables having the largest sample variance.

As before we first center the data and then compute the negative log-likelihood via 10-fold cross-validation. Figure 6 shows the result.

As for the Isoprenoid gene pathways data-set, we cannot nominate a winner here. In fact, the performances are even more indistinct than before.

6. A robust PC-DAG covariance estimator

In this section we propose a robust version of the PC-DAG estimator. According to Section 3, we need an initial covariance matrix estimation $\hat{\Sigma}_{init}$ in order to run the PC-DAG technique. In Section 3, we used the sample covariance

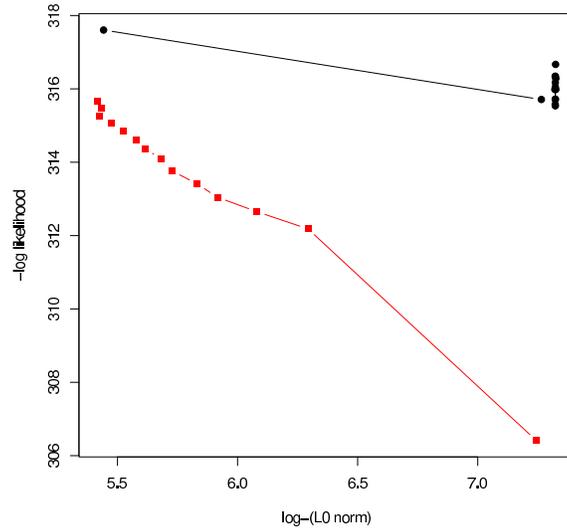


FIG 5. 10-fold CV of negative log-likelihood against the logarithm of the average number of non-zero entries of the estimated concentration matrix $\hat{\Sigma}^{-1}$. The squares stand for the Glasso and the circles for the PC-DAG estimator.

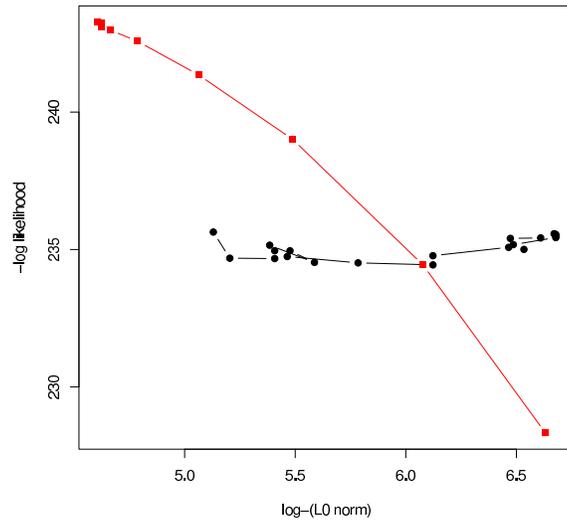


FIG 6. 10-fold CV of negative log-likelihood against the logarithm of the average number of non-zero entries of the estimated concentration matrix. The squares stand for the Glasso and the circles for the PC-DAG estimator.

$\hat{\Sigma}_{init} = \hat{\Sigma}_{MLE}$ from (6). It is well known that the standard sample covariance estimator is not robust against outliers or non-Gaussian distributions.

In order to get a robust version of the PC-DAG method we start with a robust estimate of Σ . We propose to use the orthogonalized Gnanadesikan-

Kettenring (OGK) estimator presented by [22]. Employing the OGK estimator in the PC-algorithm, i.e. estimating partial correlations from the OGK covariance estimate, we obtain a robustified estimate of the CPDAG, see also [16], and finally a robust PC-DAG covariance estimate as in (7) and (8) by using again the OGK covariance estimator instead of $\hat{\Sigma}_{MLE}$.

An “ad-hoc” robustification of the Glasso method can be achieved by using in (10) the robust OGK covariance estimate instead of the sample covariance $\hat{\Sigma}_{MLE}$.

6.1. Simulation study for non-Gaussian data

In order to analyze the behavior of the robust PC-DAG method we use a simulation model as in Section 5.1.1 but with different distributions for the errors ϵ . Regarding the latter, we consider the following distributions: $\mathcal{N}(0, 1)$, $0.9\mathcal{N}(0, 1) + 0.1t_3(0, 1)$ or $0.9\mathcal{N}(0, 1) + 0.1\text{Cauchy}(0, 1)$.

We compare the standard PC-DAG, robust PC-DAG, standard Glasso and the robust Glasso estimators for Gaussian, 10% t_3 contaminated Gaussian and 10% Cauchy contaminated Gaussian data for one specific parameter setting:

$$R : n = 50, p = 80, s = 0.01$$

In order to compare the four methods we use the Kullback-Leibler loss defined in Section 5.1.1. For the four estimation methods we plot the Kullback-Leibler loss against the logarithm of the average number of non-zero entries of the estimated concentration matrix $\hat{\Sigma}^{-1}$. The dotted vertical line represents the average number of non-zero entries of the true underlying concentration matrices. All the results are again based on 50 independent simulation runs.

Figures 7 (a) and 7 (b) show that without or with moderate outliers, the standard and robust PC-DAG estimators perform about as well as the standard and robust Glasso: the claim is based on the observation that the minimum Kullback-Leibler loss of each of the four methods is about the same, although the corresponding sparsity of the fitted concentration matrix may be very different. In the presence of more severe outliers, the robust PC-DAG technique is best as can be seen from Figure 7 (c). In summary, the robust PC-DAG estimator is a useful addition to gain robustness for estimating a high-dimensional concentration matrix.

7. Summary and discussion

We have introduced the PC-DAG estimator, a graphical model based technique for estimating sparse covariance matrices and their inverses from high-dimensional data. The method is based on very different methodological concepts than shrinkage estimators. Our PC-DAG procedure is invariant to variable permutation, yields a positive definite estimate of the covariance and concentration matrix, and we have proven asymptotic consistency for sparse high-dimensional settings. An implementation of the estimator is based on the R-

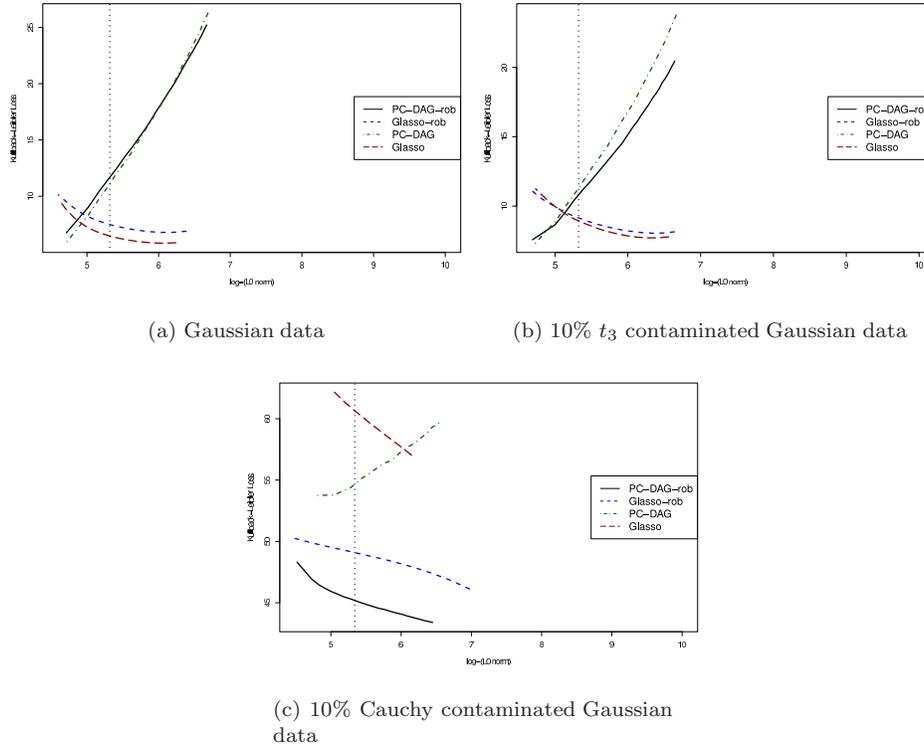


FIG 7. Kullback-Leibler loss against the logarithm of the average number of non-zero elements of Σ^{-1} for Gaussian data (a), 10% t_3 contaminated Gaussian data (b) and 10% Cauchy contaminated Gaussian data (c).

package `pcalg` [17]. We remark that alternatively, one could construct a high-dimensional covariance estimate based on a sparse undirected conditional independence graph which itself can be inferred from data using e.g. the node-wise Lasso procedure from [24].

We have compared our PC-DAG estimator with the Glasso [12, 2] in two simulation models. For the concentration matrix, our PC-DAG approach clearly outperforms the Glasso technique for some parameter settings, with performance gains up to 30-50%, while it keeps up with Glasso for the rest of the considered scenarios. For estimation of covariances, the conclusions are similar but slightly less pronounced than for inferring concentration matrices. Furthermore, we have compared the two methods in two real data-sets and found only marginal differences in performance. If the data generating mechanism is well approximated by a DAG-model, the PC-DAG estimator is undoubtedly better than the shrinkage-based Glasso. However, it is very hard to know a-priori how well a DAG-model describes the underlying true distribution. Finally, we have presented a robustification of our PC-DAG estimator for cases where the Gaussian data is contaminated by outliers.

Appendix

Appendix A: Proof of Lemma 4.1

A key element of the proofs is the analysis of low-order regression problems described in Section 3.1. For a DAG-structure with sets of parents, we consider regressions of the form

$$X_i = \sum_{j \in pa(i)} \beta_j^{(i)} X_j + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma_{i|pa(i)}^2),$$

and ε_i independent of $X_{pa(i)}$. The corresponding OLS estimates based on n i.i.d. samples $X^{(1)}, \dots, X^{(n)}$ as in (1) are denoted by

$$\hat{\beta}_j^{(i)}, \quad \hat{\sigma}_{i|pa(i)}^2 = (n - |pa(i)|)^{-1} \sum_{r=1}^n \left(X_i^{(r)} - \sum_{j \in pa(i)} \hat{\beta}_j^{(i)} X_j^{(r)} \right)^2.$$

Lemma A.1. *Suppose that the Gaussian assumption in (A), assumptions (B) and (F) hold. Then, for every $\epsilon > 0$,*

$$\begin{aligned} & \mathbb{P} \left[\sup_{i=1, \dots, p_n, j \in pa(i)} \left| \hat{\beta}_j^{(i)} - \beta_j^{(i)} \right| > \frac{\epsilon}{q_n} \right] \\ & \leq \frac{C_1}{\epsilon} q_n p_n \exp \left(-C_2 \frac{\epsilon^2}{q_n^2} (n - q_n - 1) \right) + 2 \exp \left(-C_3 \left(\frac{n}{2} - q_n - 1 \right) \right), \end{aligned} \tag{11}$$

$n \geq 2(q_n + C_4)$ where $C_1, C_2 > 0$ are constants depending on σ^2 and r (see Assumptions (B) and (F)), and $C_3, C_4 > 0$ are absolute constants.

Proof. The proof is analogous to Lemma 7.1 in [20]. For completeness, we give a detailed derivation. The union bound yields

$$\mathbb{P} \left[\sup_{i=1, \dots, p_n, j \in pa(i)} \left| \hat{\beta}_j^{(i)} - \beta_j^{(i)} \right| > \frac{\epsilon}{q_n} \right] \leq p_n q_n \sup_{i,j} \mathbb{P} \left[\left| \hat{\beta}_j^{(i)} - \beta_j^{(i)} \right| > \frac{\epsilon}{q_n} \right]. \tag{12}$$

Next we analyze $\sup_{i,j} \mathbb{P} \left[\left| \hat{\beta}_j^{(i)} - \beta_j^{(i)} \right| > \tilde{\epsilon} \right]$ for a general $\tilde{\epsilon} > 0$.

Let $i \in \{1, \dots, p_n\}$ and denote by $s(i, j) = pa(i) \setminus j$. We consider first the conditional distribution of $\hat{\beta}_j^{(i)} | X_{pa(i)}$. The variance of $X_i | X_{pa(i)}$ is $\sigma_{i|pa(i)}^2$ and we denote the variance of $X_j | X_{s(i,j)}$ by $\sigma_{j|s(i,j)}^2$. Further, we denote the sample variance of X_j by $\hat{\sigma}_j^2$, the sample variance of $X_j | X_{s(i,j)}$ by $\hat{\sigma}_{j|s(i,j)}^2$ and the sample multivariate correlation coefficient between X_j and $X_{s(i,j)}$ by $R_{j|s(i,j)}^2$. Then, when conditioning on $\mathbf{X}_{pa(i)} = \{X_{r,j}; r = 1, \dots, n, j \in pa(i)\}$,

$$\text{Var} \left(\hat{\beta}_j^{(i)} \mid \mathbf{X}_{pa(i)} \right) = \frac{1}{1 - R_{j|s(i,j)}^2} \frac{\sigma_{i|pa(i)}^2}{(n - 1)\hat{\sigma}_j^2} = \frac{\sigma_{i|pa(i)}^2}{(n - |s(i, j)| - 1)\hat{\sigma}_{j|s(i,j)}^2}, \tag{13}$$

where the first equality follows from e.g. [11, p.120] and the second equality follows from $1 - R_{j|s(i,j)}^2 = \frac{(n-|s(i,j)|-1)\hat{\sigma}_{j|s(i,j)}^2}{(n-1)\hat{\sigma}_j^2}$. With (13), $\mathbf{E} \left[\hat{\beta}_j^{(i)} \mid \mathbf{X}_{pa(i)} \right] = \beta_j^{(i)}$ and the Gaussian assumption in (A), we get

$$\begin{aligned} & \mathbf{P} \left[|\hat{\beta}_j^{(i)} - \beta_j^{(i)}| > \tilde{\epsilon} \mid \mathbf{X}_{pa(i)} \right] = \\ & \mathbf{P} \left[|Z| > \frac{\tilde{\epsilon} \sqrt{n - |s(i,j)| - 1} \hat{\sigma}_{j|s(i,j)}}{\sigma_{i|pa(i)}} \mid \mathbf{X}_{pa(i)} \right], \end{aligned} \tag{14}$$

where Z is a standard normal random variable.

We first analyze (14) on the set $B_{js(i,j)} = \{\hat{\sigma}_{j|s(i,j)}^2 > \frac{1}{2}\sigma_{j|s(i,j)}^2\}$. From assumption (F) and $\text{Var}(X_i \mid X_{pa(i)}) \leq \sigma^2$ it follows that

$$\inf_{i=1, \dots, p_n, j \in pa(i)} \frac{\text{Var}(X_j \mid X_{pa(i) \setminus j})}{\text{Var}(X_i \mid X_{pa(i)})} \geq \frac{r}{\sigma^2} = v^2, \tag{15}$$

where $v > 0$. Using this bound from (15) we obtain

$$\begin{aligned} & \mathbf{P} \left[|Z| > \frac{\tilde{\epsilon} \sqrt{n - |s(i,j)| - 1} \hat{\sigma}_{j|s(i,j)}}{\sigma_{i|pa(i)}} \mid \mathbf{X}_{pa(i)} \right] \mathbb{I}_{B_{js(i,j)}} \\ & \leq \mathbf{P} \left[|Z| > \tilde{\epsilon} v \frac{\sqrt{n - |s(i,j)| - 1}}{\sqrt{2}} \right] \\ & \leq \mathbf{P} \left[|Z| > C \tilde{\epsilon} \sqrt{n - |q_n| - 1} \right], \end{aligned} \tag{16}$$

where C depends on v in (15). We then bound the tail probability of the standard normal distribution by $\mathbf{P}[|Z| > a] \leq \frac{2}{\sqrt{2\pi}a} \exp(-\frac{a^2}{2})$ for $a > 0$. Hence, (16) can be further bounded by

$$\frac{C_1}{\tilde{\epsilon}} \exp(-C_2 \tilde{\epsilon}^2 (n - q_n - 1)) \tag{17}$$

for all n such that $q_n < n - 2$, where $C_1, C_2 > 0$ are constants depending on v in (15), i.e. they depend on σ^2 and r in assumptions (B) and (F).

Next, we compute a bound for $\mathbf{P} \left[B_{js(i,j)}^C \right]$. Note that

$$\begin{aligned} \mathbf{P} \left[B_{js(i,j)}^C \mid X_{s(i,j)} \right] &= \mathbf{P} \left[\frac{(n - |s(i,j)| - 1) \hat{\sigma}_{j|s(i,j)}^2}{\sigma_{j|s(i,j)}^2} \leq \frac{(n - |s(i,j)| - 1)}{2} \mid X_{s(i,j)} \right] \\ &= \mathbf{P} \left[\chi_{n-|s(i,j)|-1}^2 \leq \frac{(n - |s(i,j)| - 1)}{2} \right] \\ &\leq \mathbf{P} \left[\chi_{n-q_n-1}^2 \leq \frac{n-1}{2} \right]. \end{aligned}$$

Now we apply Bernstein’s inequality [28, Lemma 2.2.11] by writing

$$\begin{aligned} \mathbb{P} \left[\chi_{n-q_n-1}^2 \leq \frac{n-1}{2} \right] &= \mathbb{P} \left[\chi_{n-q_n-1}^2 - (n-q_n-1) \leq \frac{-(n-1)}{2} + q_n \right] \\ &\leq \mathbb{P} \left[\left| \chi_{n-q_n-1}^2 - (n-q_n-1) \right| < \frac{(n-1)}{2} - q_n \right] \end{aligned}$$

and noting that $\chi_{n-q_n-1}^2 - (n-q_n-1)$ can be viewed as the sum of $n-q_n-1$ independent centered χ_1^2 random variables. Hence, the last term is bounded above by

$$2 \exp \left(- \frac{\left(\frac{n-1}{2} - q_n \right)^2}{C'_3 + C'_4 \left(\frac{n-1}{2} - q_n \right)} \right)$$

where $C'_3, C'_4 > 0$ are constants arising from moment conditions. This expression is in addition bounded above by

$$2 \exp \left(-C_3 \left(\frac{n}{2} - q_n - 1 \right) \right) \tag{18}$$

for all n such that $\frac{n-2}{2} - q_n > C'_3$, and $C_3 > 0$ is a constant arising from moment conditions. Because this bound in (18) holds for all $X_{s(i,j)}$ with $|s(i,j)| \leq q_n$, it also holds for the unconditional probability $\mathbb{P} \left[B_{js(i,j)}^C \right]$.

The upper bound for $\mathbb{P} \left[\left| \hat{\beta}_j^{(i)} - \beta_j^{(i)} \right| > \tilde{\epsilon} \right]$ now follows by combining (17) and (18):

$$\begin{aligned} &\mathbb{P} \left[\left| \hat{\beta}_j^{(i)} - \beta_j^{(i)} \right| > \tilde{\epsilon} \right] \\ &\leq \int_{B_{js(i,j)}} \mathbb{P} \left[\left| \hat{\beta}_j^{(i)} - \beta_j^{(i)} \right| > \tilde{\epsilon} \mid pa(i) \right] dF_{X_{j,s(i,j)}} + \mathbb{P} \left[B_{js(i,j)}^C \right] \\ &\leq \frac{C_1}{\tilde{\epsilon}} \exp \left(-C_2 \tilde{\epsilon}^2 (n - q_n - 1) \right) + 2 \exp \left(-C_3 \left(\frac{n}{2} - q_n - 1 \right) \right). \end{aligned}$$

Now by using $\tilde{\epsilon} = \frac{\epsilon}{q_n}$ we derive

$$\begin{aligned} &\sup_{i,j} \mathbb{P} \left[\left| \hat{\beta}_j^{(i)} - \beta_j^{(i)} \right| > \frac{\epsilon}{q_n} \right] \\ &\leq \frac{C_1 q_n}{\epsilon} \exp \left(-C_2 \frac{\epsilon^2}{q_n^2} (n - q_n - 1) \right) + 2 \exp \left(-C_3 \left(\frac{n}{2} - q_n - 1 \right) \right) \end{aligned} \tag{19}$$

which holds for all $n > 2(q_n + C'_3) + 2 = 2(q_n + C_4)$. Combining (19) with (12) we complete the proof of Lemma A.1. \square

Lemma A.2. *Suppose that the Gaussian distribution in assumption (A), assumptions (B) and (F) hold. Then, for every $\epsilon > 0$,*

$$\begin{aligned} &\mathbb{P} \left[\sup_{1 \leq i \leq p_n} \left| \frac{1}{\hat{\sigma}_{i|pa(i)}^2} - \frac{1}{\sigma_{i|pa(i)}^2} \right| > \frac{\epsilon}{q_n} \right] \\ &\leq p_n 2 \left(\exp \left(- \frac{\epsilon^2 (n - q_n)}{6C^2 q_n^2 \sigma^4 + 4C \epsilon q_n \sigma^2} \right) + \exp \left(- \frac{r^2 (n - q_n)}{24\sigma^4 + 8r\sigma^2} \right) \right) \end{aligned}$$

where $C > 0$ is an absolute constant and $r > 0$ as in assumption (F).

Proof. Using the union bound, for $\tilde{\epsilon} > 0$,

$$\mathbb{P} \left[\sup_{i=1, \dots, p_n} \left| \hat{\sigma}_{i|pa(i)}^2 - \sigma_{i|pa(i)}^2 \right| > \tilde{\epsilon} \right] \leq p_n \sup_{i=1, \dots, p_n} \mathbb{P} \left[\left| \hat{\sigma}_{i|pa(i)}^2 - \sigma_{i|pa(i)}^2 \right| > \tilde{\epsilon} \right].$$

For the conditional probability, when conditioning on $\mathbf{X}_{pa(i)} = \{X_{r,j}; r = 1, \dots, n, j \in pa(i)\}$, we have that $\mathbb{P} \left[\left| \hat{\sigma}_{i|pa(i)}^2 - \sigma_{i|pa(i)}^2 \right| > \tilde{\epsilon} \mid \mathbf{X}_{pa(i)} \right]$ is equal to

$$\begin{aligned} & \mathbb{P} \left[\left| \frac{\hat{\sigma}_{i|pa(i)}^2}{\sigma_{i|pa(i)}^2} - 1 \right| > \frac{\tilde{\epsilon}}{\sigma_{i|pa(i)}^2} \mid \mathbf{X}_{pa(i)} \right] = \\ & \mathbb{P} \left[\left| \frac{(n - |pa(i)|)\hat{\sigma}_{i|pa(i)}^2}{\sigma_{i|pa(i)}^2} - (n - |pa(i)|) \right| > \frac{\tilde{\epsilon}(n - |pa(i)|)}{\sigma_{i|pa(i)}^2} \mid \mathbf{X}_{pa(i)} \right]. \end{aligned}$$

Because $\frac{(n - |pa(i)|)\hat{\sigma}_{i|pa(i)}^2}{\sigma_{i|pa(i)}^2} - (n - |pa(i)|)$ is a sum of $(n - |pa(i)|)$ independent χ_1^2 -distributed centered random variables, we can use Bernstein's inequality [28, Lemma 2.2.11]. Hence, with $\sigma_{i|pa(i)}^2 \leq \sigma^2$ we get

$$\begin{aligned} & \mathbb{P} \left[\left| \frac{(n - |pa(i)|)\hat{\sigma}_{i|pa(i)}^2}{\sigma_{i|pa(i)}^2} - (n - |pa(i)|) \right| > \frac{\tilde{\epsilon}(n - |pa(i)|)}{\sigma_k^2} \mid \mathbf{X}_{pa(i)} \right] \\ & \leq 2 \exp \left(-\frac{\tilde{\epsilon}^2(n - |pa(i)|)}{6\sigma^4 + 4\tilde{\epsilon}\sigma^2} \right). \end{aligned}$$

Since this bound holds for all $\mathbf{X}_{pa(i)}$, the bound also applies to the unconditional probability:

$$\begin{aligned} & \mathbb{P} \left[\left| \hat{\sigma}_{i|pa(i)}^2 - \sigma_{i|pa(i)}^2 \right| > \tilde{\epsilon} \right] \\ & = \mathbb{P} \left[\left| \frac{(n - |pa(i)|)\hat{\sigma}_{i|pa(i)}^2}{\sigma_{i|pa(i)}^2} - (n - |pa(i)|) \right| > \frac{\tilde{\epsilon}(n - |pa(i)|)}{\sigma_k^2} \right] \\ & \leq 2 \exp \left(-\frac{\tilde{\epsilon}^2(n - |pa(i)|)}{6\sigma^4 + 4\tilde{\epsilon}\sigma^2} \right). \end{aligned} \tag{20}$$

We use now a Taylor expansion:

$$\frac{1}{\hat{\sigma}_{i|pa(i)}^2} = \frac{1}{\sigma_{i|pa(i)}^2} - \frac{1}{\tilde{\sigma}_{i|pa(i)}^4} (\hat{\sigma}_{i|pa(i)}^2 - \sigma_{i|pa(i)}^2),$$

where $\left| \hat{\sigma}_{i|pa(i)}^2 - \sigma_{i|pa(i)}^2 \right| \leq \left| \hat{\sigma}_{i|pa(i)}^2 - \sigma_{i|pa(i)}^2 \right|$.

Consider the set $B = \{\sup_{i=1, \dots, p_n} \left| \hat{\sigma}_{i|pa(i)}^2 - \sigma_{i|pa(i)}^2 \right| \leq r/2\}$ with $r > 0$ as in assumption (F). Then, on B , we have $\left| \frac{1}{\hat{\sigma}_i} \right| \leq \tilde{C} < \infty$ (and the bound does not

depend on the index i). Therefore,

$$\begin{aligned} & \mathbb{P} \left[\sup_{i=1, \dots, p_n} \left| \frac{1}{\hat{\sigma}_{i|pa(i)}^2} - \frac{1}{\sigma_{i|pa(i)}^2} \right| > \frac{\epsilon}{q_n} \right] \\ & \leq \mathbb{P} \left[\left\{ \tilde{C} \sup_{i=1, \dots, p_n} \left| \hat{\sigma}_{i|pa(i)}^2 - \sigma_{i|pa(i)}^2 \right| > \frac{\epsilon}{q_n} \right\} \cap B \right] + \mathbb{P} [B^C] \end{aligned}$$

The first term and second term on the right-hand side can be bounded using (20), leading to the bound in the statement of the lemma. This completes the proof of Lemma A.2. \square

Proof of Lemma 4.1. Let G be a DAG from the true underlying CPDAG, i.e. the true equivalence class. Using the union bound we have

$$\mathbb{P} \left[\sup_{i,j=1, \dots, p_n} \left| \hat{\Sigma}_{G,n;i,j}^{-1} - \Sigma_{n;i,j}^{-1} \right| > \gamma \right] \leq p_n^2 \sup_{i,j} \mathbb{P} \left[\left| \hat{\Sigma}_{G,n;i,j}^{-1} - \Sigma_{n;i,j}^{-1} \right| > \gamma \right]. \quad (21)$$

Since $\hat{\Sigma}^{-1} = \hat{A}^T \widehat{\text{Cov}}(\epsilon)^{-1} \hat{A}$ we have $\hat{\Sigma}_{G,n;i,j}^{-1} = \sum_{k=1}^{p_n} \hat{\lambda}_k \hat{A}_{kj} \hat{A}_{ki}$ with $\hat{\lambda}_k = \frac{1}{\hat{\sigma}_k^2}$ and \hat{A} as in (7). Thus,

$$\begin{aligned} \left| \hat{\Sigma}_{G,n;i,j}^{-1} - \Sigma_{n;i,j}^{-1} \right| &= \left| \sum_{k=1}^{p_n} \left(\hat{\lambda}_k \hat{A}_{kj} \hat{A}_{ki} - \lambda_k A_{kj} A_{ki} \right) \right| \\ &\leq \sum_{k=1}^{p_n} \left| \hat{\lambda}_k \hat{A}_{kj} \hat{A}_{ki} - \lambda_k A_{kj} A_{ki} \right| \\ &= \sum_{k=1}^{p_n} \left| \hat{\lambda}_k \hat{A}_{kj} \hat{A}_{ki} - \hat{\lambda}_k A_{kj} A_{ki} + \hat{\lambda}_k A_{kj} A_{ki} - \lambda_k A_{kj} A_{ki} \right| \\ &= \sum_{k=1}^{p_n} \left| \hat{\lambda}_k \left(\hat{A}_{kj} \hat{A}_{ki} - A_{kj} A_{ki} \right) + A_{kj} A_{ki} \left(\hat{\lambda}_k - \lambda_k \right) \right| \\ &\leq \sum_{k=1}^{p_n} \left(\left| \hat{\lambda}_k \right| \left| \hat{A}_{kj} \hat{A}_{ki} - A_{kj} A_{ki} \right| + \left| A_{kj} A_{ki} \right| \left| \hat{\lambda}_k - \lambda_k \right| \right) \end{aligned}$$

Consider the terms $\left| \hat{A}_{kj} \hat{A}_{ki} - A_{kj} A_{ki} \right|$ and $\left| \hat{\lambda}_k \right|$:

$$\begin{aligned} \left| \hat{A}_{kj} \hat{A}_{ki} - A_{kj} A_{ki} \right| &= \left| \hat{A}_{kj} \hat{A}_{ki} - \hat{A}_{kj} A_{ki} + \hat{A}_{kj} A_{ki} - A_{kj} A_{ki} \right| \\ &= \left| \hat{A}_{kj} \left(\hat{A}_{ki} - A_{ki} \right) + A_{ki} \left(\hat{A}_{kj} - A_{kj} \right) \right| \\ &\leq \left| \hat{A}_{kj} \right| \left| \hat{A}_{ki} - A_{ki} \right| + \left| A_{ki} \right| \left| \hat{A}_{kj} - A_{kj} \right| \\ \left| \hat{\lambda}_k \right| &= \left| \hat{\lambda}_k - \lambda_k + \lambda_k \right| \leq \left| \hat{\lambda}_k - \lambda_k \right| + \left| \lambda_k \right| \end{aligned}$$

By plugging these bounds into the formula above and using that the summations are over at most q_n terms only (due to sparsity of \hat{A}_{ki} and A_{ki}), we obtain

$$\left| \hat{\Sigma}_{G,n;i,j}^{-1} - \Sigma_{n;i,j}^{-1} \right| \leq Cq_n\delta$$

where $C > 0$ is an absolute constant and δ the maximal absolute difference of \hat{A} 's and $\hat{\lambda}$'s:

$$\delta = \max\{\max_{i,k} |\hat{A}_{ki} - A_{ki}|, \max_k |\hat{\lambda}_k - \lambda_k|\}.$$

Hence

$$\mathbb{P} \left[\left| \hat{\Sigma}_{G,n;i,j}^{-1} - \Sigma_{n;i,j}^{-1} \right| > \gamma \right] \leq \mathbb{P} [Cq_n |\delta| > \gamma] = \mathbb{P} \left[|\delta| > \frac{\gamma}{Cq_n} \right] = \mathbb{P} \left[|\delta| > \frac{\epsilon}{q_n} \right]$$

with $\frac{\gamma}{C} = \epsilon$. Because the convergence of the term $\mathbb{P} \left[|\delta| > \frac{\epsilon}{q_n} \right]$ is covered either by Lemma A.1 or Lemma A.2, since $q_n^2 = O(n^{1-2b})$ ($0 < b \leq 1/2$) from assumption (D), and using (21), we complete the proof of Lemma 4.1. \square

Appendix B: Modifications of the PC-DAG covariance estimator

With finite sample size, the PC-algorithm may make some errors. One of them can produce conflicting v-structures when orienting the graph: if so, we deal with it by keeping one and discarding other v-structures. In our implementation, the result then depends on the order of the performed independence tests. Furthermore, it may happen that the output of the PC-algorithm is an invalid CPDAG which does not describe an equivalence class of DAGs. In such a case we use the *retry* type orientation procedure implemented in the `pcAlgo`-function of the `pcalg`-package, see the reference manual of the `pcalg`-package [17] for more information.

References

- [1] ANDERSON, T. W. (1984). *An Introduction to Multivariate Statistical Analysis*. Wiley, NY. [MR0771294](#)
- [2] BANERJEE, O., GHAOUI, L. E. and D'ASPREMONT, A. (2008). Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data. *Journal of Machine Learning Research* **9** 485-516. [MR2417243](#)
- [3] BICKEL, P. J. and LEVINA, E. (2004). Some theory for Fisher's linear discriminant function, "naive Bayes", and some alternatives when there are many more variables than observations. *Bernoulli* **10** 989-1010. [MR2108040](#)
- [4] BICKEL, P. J. and LEVINA, E. (2008). Covariance regularization by thresholding. *The Annals of Statistics* **36** 2577-2604. [MR2485008](#)

- [5] BICKEL, P. J. and LEVINA, E. (2008). Regularized estimation of large covariance matrices. *The Annals of Statistics* **36** 199–227. [MR2387969](#)
- [6] CHAUDHURI, S., DRTON, M. and RICHARDSON, T. S. (2007). Estimation of a covariance matrix with zeros. *Biometrika* **94** 1–18. [MR2307904](#)
- [7] COX, D. R. and WERMUTH, N. (1996). *Multivariate Dependencies*, First ed. *Monographs on Statistics and Applied Probability*. Chapman and Hall. [MR1456990](#)
- [8] D’ASPREMONT, A., BANERJEE, O. and GHAOUI, L. E. (2008). First-order methods for sparse covariance selection. *SIAM Journal on Matrix Analysis and Applications* **30** 56–66. [MR2399568](#)
- [9] DENG, X. and YUAN, M. (2009). Large Gaussian covariance matrix estimation with Markov structures. *Journal of Computational and Graphical Statistics* **18** 640–657.
- [10] DRTON, M. and RICHARDSON, T. S. (2004). Iterative conditional fitting for Gaussian ancestral graph models. In *AUAI '04: Proceedings of the 20th conference on Uncertainty in artificial intelligence* 130–137. AUAI Press, Arlington, Virginia, United States.
- [11] FOX, J. (1997). *Applied Regression Analysis, Linear Models, and Related Methods*. Sage Publications.
- [12] FRIEDMAN, J., HASTIE, T. and TIBSHIRANI, R. (2007). Sparse inverse covariance estimation with the graphical Lasso. *Biostatistics* **9** 432–441.
- [13] FURRER, R. and BENGTTSSON, T. (2007). Estimation of high-dimensional prior and posterior covariance matrices in Kalman filter variants. *Journal of Multivariate Analysis* **98** 227–255. [MR2301751](#)
- [14] HUANG, J. Z., LIU, N., POURAHMADI, M. and LIU, L. (2006). Covariance matrix selection and estimation via penalised normal likelihood. *Biometrika* **93** 85–98. [MR2277742](#)
- [15] KALISCH, M. and BÜHLMANN, P. (2007). Estimating high-dimensional directed acyclic graphs with the PC-algorithm. *Journal of Machine Learning Research* **8** 613–636.
- [16] KALISCH, M. and BÜHLMANN, P. (2008). Robustification of the PC-algorithm for directed acyclic graphs. *Journal of Computational and Graphical Statistics* **17** 773–789.
- [17] KALISCH, M. and MÄCHLER, M. Estimating the skeleton and equivalence class of a DAG. Manual to the R-package pcalg.
- [18] LAURITZEN, S. L. (1996). *Graphical Models. Oxford Statistical Science Series, 17*. Oxford Clarendon Press. [MR1419991](#)
- [19] LEVINA, E., ROTHMAN, A. and ZHU, J. (2008). Sparse estimation of large covariance matrices via a nested Lasso penalty. *The Annals of Applied Statistics* **2** 245–263. [MR2415602](#)
- [20] MAATHUIS, M. H., KALISCH, M. and BÜHLMANN, P. (2009). Estimating high-dimensional intervention effects from observational data. *The Annals of Statistics* **37** 3133–3164.
- [21] MARCHETTI, G. M. (2006). Independencies induced from a graphical Markov model after marginalization and conditioning: The R-package ggm. *Journal of Statistical Software* **15** 1–15.

- [22] MARONNA, R. A. and ZAMAR, R. H. (2002). Robust estimates of location and dispersion for high-dimensional datasets. *Technometrics* **44** 307–317. [MR1939680](#)
- [23] MEEK, C. (1995). Causal Inference and Causal Explanation with Background Knowledge. In *Proceedings of the 11th Annual Conference on Uncertainty in Artificial Intelligence (UAI-95)* 403–441. Morgan Kaufmann, San Francisco, CA. [MR1615025](#)
- [24] MEINSHAUSEN, N. and BÜHLMANN, P. (2006). High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics* **34** 1436–1462. [MR2278363](#)
- [25] PEARL, J. (2008). *Causality - Models, Reasoning, and Inference*. Cambridge University Press, NY. [MR1744773](#)
- [26] ROTHMAN, A. J., BICKEL, P. J., LEVINA, E. and ZHU, J. (2008). Sparse permutation invariant covariance estimation. *Electronic Journal of Statistics* **2** 494–515. [MR2417391](#)
- [27] SPIRITES, P., GLYMOUR, C. and SCHEINES, R. (2000). *Causation, Prediction, and Search*, 2nd ed. The MIT Press, Cambridge, Massachusetts, London, England. [MR1815675](#)
- [28] VAN DER VAART, A. W. and WELLNER, J. A. (1996). *Weak Convergence and Empirical Processes; With Applications to Statistics*. Springer Series in Statistics. Springer-Verlag, New York. [MR1385671](#)
- [29] WEST, M., BLANCHETTE, C., DRESSMAN, H., HUANG, E., ISHIDA, S., SPANG, R., ZUZAN, H., JR., J. O., MARKS, J. and NEVINS, J. (2001). Predicting the clinical status of human breast cancer by using gene expression profiles. *PNAS* **98** 11462–11467.
- [30] WILLE, A., ZIMMERMANN, P., VRANOVA, E., FÜRHOHL, A., LAULE, O., BLEULER, S., HENNIG, L., PRELIC, A., VON ROHR, P., THIELE, L., ZITZLER, E., GRUISSEM, W. and BÜHLMANN, P. (2004). Sparse graphical Gaussian modeling of the isoprenoid gene network in Arabidopsis thaliana. *Genome Biology* **5** R92.
- [31] WU, W. B. and POURAHMADI, M. (2003). Nonparametric estimation of large covariance matrices of longitudinal data. *Biometrika* **90** 831–844. [MR2024760](#)