Published in final edited form as:

Biometrics. 2019 March ; 75(1): 36–47. doi:10.1111/biom.12941.

Joint Skeleton Estimation of Multiple Directed Acyclic Graphs for Heterogeneous Population

Jianyu Liu¹, Wei Sun³, and Yufeng Liu^{1,2,*}

¹Department of Statistics and Operations Research, The University of North Carolina at Chapel Hill, Chapel Hill, North Carolina U.S.A.

²Department of Genetics, Department of Biostatistics, Carolina Center for Genome Sciences, Lineberger Comprehensive Cancer Center, The University of North Carolina at Chapel Hill, Chapel Hill, North Carolina U.S.A.

³Biostatistics Program, Public Health Sciences Division Fred Hutchinson Cancer Research Center, Seattle, Washington, U.S.A.

Summary:

The directed acyclic graph (DAG) is a powerful tool to model the interactions of high-dimensional variables. While estimating edge directions in a DAG often requires interventional data, one can estimate the skeleton of a DAG (i.e., an undirected graph formed by removing the direction of each edge in a DAG) using observational data. In real data analyses, the samples of the high-dimensional variables may be collected from a mixture of multiple populations. Each population has its own DAG while the DAGs across populations may have significant overlap. In this paper, we propose a two-step approach to jointly estimate the DAG skeletons of multiple populations while the population origin of each sample may or may not be labeled. In particular, our method allows a probabilistic soft label for each sample, which can be easily computed and often leads to more accurate skeleton estimation than hard labels. Compared with separate estimation of skeletons for each population, our method is more accurate and robust to labeling errors. We study the estimation consistency for our method, and demonstrate its performance using simulation studies in different settings. Finally, we apply our method to analyze gene expression data from breast cancer patients of multiple cancer subtypes.

Keywords

Bi-level selection; DAG; PC algorithm; Probabilistic labels; Soft classifier

1. Introduction

Our method development is motivated by the problem of analyzing genome-wide gene expression data. In a typical gene expression study at human population, the variables of

* yfliu@email.unc.edu.

⁸Supplementary Materials

Web Appendices, Tables, Figures referenced in Sections 2, 5, and 6, and the computer code are available with this article at the Biometrics website on Wiley Online Library.

interest are the expression of 10,000 – 50,000 genes, and the sample size is around a few hundreds. Co-expression of two genes implies a regulatory effect (one gene regulates the expression of the other gene) or that the two genes share some regulatory components. The co-expression of all the genes can be conveniently studied by a directed acyclic graph (DAG) in which a node represents a gene and directed edges specify regulatory effects. These graphs can be very useful for understanding the molecular basis of a disease or to prioritize drug target. For example, if disrupting a specific gene help treating a certain type of cancer but no drug is available to target this gene, one may use a directed graph model to identify the immediate parent nodes of this gene as potential drug targets.

A DAG is a directed graph of which the edges do not form any directed cycle. Statistically, a DAG is a model that describes conditional dependence of random variables. A directed edge $I \rightarrow j$ indicates that j is dependent on I given any subset of the remaining nodes. On the other hand, the absence of an edge indicates that the two nodes are marginally independent, or conditionally independent given some subset of the remaining nodes. By removing the direction of each edge in a DAG, the resulting undirected graph is the *skeleton* of the DAG. Estimation of a DAG is often infeasible due to lack of interventional data. Instead, the DAG skeleton can be estimated using observational data. Given the DAG skeleton, one can orient as many edges as possible to form a completed partially directed graph (CPDAG) using a set of deterministic rules (Pearl, 2009). The focus of this paper is on skeleton estimation.

There are various approaches to estimate a DAG or its skeleton. A typical search-and-score approach searches for a skeleton that maximizes the regularized likelihood or the posterior probability (Heckerman et al., 1995; Friedman and Koller, 2003). These methods can quickly become computationally infeasible for problems with thousands of variables. An alternative solution for skeleton estimation is constraint-based approaches, which are often computationally more efficient. Among these constraint-based methods, the PC algorithm (Spirtes et al., 2000; Kalisch and Bühlmann, 2007; Colombo and Maathuis, 2014), which is based on conditional independence tests, has been shown to be consistent in high dimensions. There are also hybrid methods that combine the search-and-score and constraint-based approaches (Tsamardinos et al., 2006; Schmidt et al., 2007; Han et al., 2016). Nandy et al. (2015) showed that some hybrid methods with greedy search over the constrained DAG space are consistent under high dimensional settings. In a recent paper, Ha et al. (2016) proposed a two-stage skeleton estimation method called PenPC, which first estimates a conditional independence graph by neighborhood selection, and then estimates the skeleton by a modified PC algorithm.

Despite the effectiveness of these aforementioned methods, they were originally developed for skeleton estimation of a homogeneous population. In practice, the samples may come from a heterogeneous population. For example, when we study gene expression data of the patients with a certain type of cancer, the patients may belong to different subtypes. The co-expression pattern of genes, hence the DAG model, may vary across subtypes. Though we can estimate a DAG for each subtype separately, joint estimation can be more efficient by exploiting DAG similarities across subtypes. There have been extensive studies on the joint estimation of multiple Gaussian graphical models (Guo et al., 2011; Danaher et al., 2014). In contrast, not much work has been done for joint DAG or skeleton estimation (Oates et al.,

2016). Furthermore, in practice, clustering and classification methods are commonly used to label samples into different classes with potential errors. However, these labels are often used as if they were true for multiple graph estimation. To the best of our knowledge, no previous work has considered clustering or classification errors in graph estimation.

In this paper, we propose a new method, MPenPC, to jointly estimate multiple skeletons for high dimensional variables measured in a set of heterogeneous samples. The MPenPC is a two-step method. It first jointly estimates the conditional independence graphs for multiple classes, and then applies the PC-stable algorithm to construct the skeletons. The MPenPC can accommodate both *hard labels* (i.e., discrete class assignment) or *soft labels* (i.e., posterior probability) for class assignments. Specifically, to use soft labels, we first estimate probabilistic class labels by clustering or classification, and then use them as weights in both steps of MPenPC. This approach benefits skeleton estimation by mitigating the impact of mistaken hard labels, as we will demonstrate in this paper.

The rest of this paper is organized as follows. In Section 2, we review some important properties of the DAG skeleton and existing estimation methods, then propose our method MPenPC. In Section 3, we give some implementation details of MPenPC. In Section 4, we discuss some theoretical properties of MPenPC. In Sections 5 and 6, we evaluate the performance of MPenPC by simulations and real data analysis, respectively. We conclude the paper with some discussions on possible generalizations of MPenPC in Section 7.

2. Methodology

We first give an overview of DAG skeleton estimation methods under Gaussian settings in Section 2.1. Then in Sections 2.2 and 2.3, we introduce our MPenPC methods using hard and soft labels, respectively.

2.1 Review of DAG estimation

We first review some key concepts and properties of the DAG that will be used in this paper.

A node l is a *parent* of node j if there is an edge $l \rightarrow j$, and j is called a *child* of l . If two unconnected nodes are parents of a common child ($i \rightarrow j \leftarrow l$), then they form a *v-structure*. The *skeleton* of a DAG is the undirected graph formed by removing directions of all the edges in the DAG. For any DAG $\mathcal{D}$, we denote its skeleton by $\mathcal{D}^u$. There are often more than one DAGs that can describe the conditional dependence embedded in a probability distribution. These DAGs are probabilistically equivalent, and they form a *Markov equivalence class*. It can be shown that two DAGs belong to a Markov equivalence class if and only if they share the skeleton and *v-structures* (Chickering, 2002).

We consider a DAG model, denoted by $\mathcal{D}$, for p random variables $X_1, \dots, X_p$ under Gaussian settings. Given a sample $\{\mathbf{x}_i; i = 1, \dots, n\}$, we assume $\mathbf{x}_i \stackrel{iid}{\sim} N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, and denote the precision matrix by $\boldsymbol{\Omega} = \boldsymbol{\Sigma}^{-1}$. Given the DAG, the data generating process can be specified by a set of structure equations:

$$X_j = \nu_j + \sum_{l \in \mathbf{pa}_j} b_{j,l} X_l + Z_j, \quad (1)$$

for $j = 1, \dots, p$, where $\mathbf{pa}_j$ represents the set of parent nodes of j , and $Z_j \sim N(0, \lambda_j^2)$ independently for all j . Let $\mathbf{X} = (X_1, \dots, X_p)^T$, $\mathbf{v} = (\nu_1, \dots, \nu_p)^T$, and $\mathbf{Z} = (Z_1, \dots, Z_p)^T$. The p structure equations can be rewritten in a concise form: $\mathbf{X} = \mathbf{v} + \mathbf{B}\mathbf{X} + \mathbf{Z}$, where $B_{j,l} = b_{j,l}$ for $l \in \mathbf{pa}_j$ and 0 otherwise. The direct results of the model equivalence are $\boldsymbol{\mu} = (\mathbf{I} - \mathbf{B})\mathbf{v}$, $\boldsymbol{\Sigma} = (\mathbf{I} - \mathbf{B})^{-1} \boldsymbol{\Lambda} (\mathbf{I} - \mathbf{B})^{-T}$, and

$$\boldsymbol{\Omega} = (\mathbf{I} - \mathbf{B})^T \boldsymbol{\Lambda}^{-1} (\mathbf{I} - \mathbf{B}), \quad (2)$$

where $\boldsymbol{\Lambda} = \text{diag}(\lambda_1^2, \dots, \lambda_p^2)$. Let $\mathcal{G}$ be the conditional independence graph (CIG) for $X_1, \dots, X_p$, then nodes j and l are not connected if and only if $\Omega_{j,l} = \Omega_{l,j} = 0$. Owing to equation (2), $\mathcal{G}$ is also a sparse graph because of the sparsity of $\mathbf{B}$. In fact, we can establish a relationship between the DAG $\mathcal{D}$ and $\mathcal{G}$ as follows.

LEMMA 1: For any two nodes $j, l \in \{1, \dots, p\}$,

1. (1) there is an edge $l \rightarrow j$ in $\mathcal{D}$ only when there is an edge $l - j$ in $\mathcal{G}$;
2. (2) if there is an edge $j - l$ in $\mathcal{G}$ then j and l are either connected in $\mathcal{D}$, or they are parents of a v-structure in $\mathcal{D}$.

We refer readers to Chapter 3.7 of Spirtes et al. (2000) for a proof of Lemma 1. Ha et al. (2016) took advantage of this relationship and proposed PenPC for skeleton estimation. It consists of two steps: i) estimate $\mathcal{G}$ by neighborhood selection, denoted by $\widehat{\mathcal{G}}$, and ii) estimate the skeleton on the basis of $\widehat{\mathcal{G}}$. The first step is a Gaussian graphical model estimation problem, which is well studied in the literature (Meinshausen and Bühlmann, 2006; Yuan and Lin, 2007). Given a sparse $\widehat{\mathcal{G}}$ estimated by neighborhood selection, the skeleton can be efficiently estimated by a modified PC-stable algorithm. In high dimensional cases, the PenPC method has shown significant advantages over the PC-stable algorithm, both in terms of accuracy and computational efficiency.

2.2 Joint Estimation of Multiple Skeletons with Hard Labels

Denote the observed data by $\{(\mathbf{x}_i, g_i); i = 1, \dots, n\}$, where $\mathbf{x}_i \in \mathbb{R}^p$ and $g_i \in \{1, \dots, K\}$ are respectively the feature vector and the population label of the sample i . We assume the feature vector $\mathbf{X}$ is conditional Gaussian, i.e., $\mathbf{X}|(G = k) \sim N(\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)})$ for $k = 1, \dots, K$. For the k -th population, we denote the DAG and its skeleton by $\mathcal{D}^{(k)}$ and $\mathcal{D}_u^{(k)}$, the precision matrix by $\boldsymbol{\Omega}^{(k)}$, and the CIG by $\mathcal{G}^{(k)}$.

One can estimate a DAG skeleton for each population separately. However, a joint estimation approach can be more efficient when there is certain similarity across the DAGs.

We propose a two-step method, the multi-PenPC (MPenPC), for multiple-skeleton estimation by extending the PenPC method. At the first step, we use joint neighborhood selection to estimate the CIGs (Meinshausen and Bühlmann, 2006). For any node j , the node-wise regression models of K populations are

$$X_j^{(k)} = \gamma_{j,0}^{(k)} + \sum_{l \neq j} \gamma_{j,l}^{(k)} X_l^{(k)} + \epsilon_j^{(k)}, \quad k = 1, \dots, K, \quad (3)$$

where $X_j^{(k)}$ is the feature j of population k , and $\epsilon_j^{(k)}$ is an error term. For each class k , we define the coefficient vector $\gamma_j^{(k)} = (\gamma_{j,1}^{(k)}, \dots, \gamma_{j,j-1}^{(k)}, \gamma_{j,j+1}^{(k)}, \dots, \gamma_{j,p}^{(k)})^T \in \mathbb{R}^{p-1}$. Denote $\gamma_j^{*(k)} = -\Omega_{-j,j}^{(k)} \Omega_{j,j}^{(k)}$ the true coefficient vector, then $\gamma_{j,l}^{*(k)} \neq 0$ if and only if the edge $j-l$ exists in $\mathcal{G}^{(k)}$. Thus we can recover the graph $\mathcal{G}^{(k)}$ by estimating $\{\gamma_j^{*(k)}; j = 1, \dots, p\}$.

Denote $\mathbf{X}^{(k)}$ the feature matrix of the sample from population k , $\mathbf{X}_j^{(k)}$ the j -th column of $\mathbf{X}^{(k)}$, and $\mathbf{X}_{-j}^{(k)}$ the feature matrix without the j -th column. Then a joint neighborhood selection method can be formulated as

$$\underset{\gamma_j}{\operatorname{argmin}} \frac{1}{2n} \sum_{k=1}^K \left\| \mathbf{X}_j^{(k)} - \gamma_{j,0}^{(k)} - \mathbf{X}_{-j}^{(k)} \gamma_j^{(k)} \right\|_2^2 + P(\gamma_j), \quad (4)$$

where $\gamma_j = (\gamma_{j,0}^{(1)}, \gamma_j^{(1)T}, \dots, \gamma_{j,0}^{(K)}, \gamma_j^{(K)T})^T$. If the penalty function is defined as a summation of penalties across the K groups, i.e., $P(\gamma_j) = \sum_{k=1}^K P^{(k)}(\gamma_j^{(k)})$, then (4) is equivalent to estimating the neighborhood of node j separately for K populations. Thus $P(\gamma_j)$ should possess a group selection effect to induce similar sparsity patterns for $\gamma_j^{(k)}$'s.

There are various group penalties that encourage certain parameters to be zero or nonzero simultaneously. Some of them, including the group lasso (Yuan and Lin, 2006), are rigid in the sense that parameters in one group usually are either all zero or all non-zero. Whereas some others, such as group bridge and group exponential penalties (Huang et al., 2009; Breheny, 2015), can achieve a bi-level selection effect, i.e., the parameters in one group can contain both zero and non-zero values. In our case, the graphs are often only partially overlapped, so an edge may or may not be shared by all DAGs, and thus a group penalty with the bi-level selection effect is more appropriate. In this paper, we employ the group exponential (GEL) penalty (Breheny, 2015) and the regularization in (4) becomes

$$P(\gamma_j) = P_{\lambda, \tau}(\gamma_j) = \lambda^2 \tau^{-1} \sum_{l \neq j} \left\{ 1 - \exp\left(-\lambda^{-1} \tau \|\gamma_{j,l}\|_1\right) \right\},$$

where $\gamma_{j,l} = (\gamma_{j,l}^{(1)}, \dots, \gamma_{j,l}^{(K)})^T$. The tuning parameters λ and τ can be chosen by the extended BIC (Chen and Chen, 2008). When there is only one class, the regularization becomes $R(\gamma_j) = \lambda^2 \tau^{-1} \prod_{j,l} \{1 - \exp(-\lambda^{-1} \tau \gamma_{j,l})\}$, which is non-convex and the resulting estimator has oracle properties (Fan and Li, 2001).

For the second step of our proposed MPenPC method, we apply the PC-stable algorithm (Colombo and Maathuis, 2014) for skeleton estimation while using $\{\widehat{\mathcal{G}}^{(k)}; k = 1, \dots, K\}$ as initial graphs. For completeness, we describe the implementation of the algorithm as follows. Assume we are estimating a DAG skeleton of p nodes using $\mathcal{G}$ as the initial graph, and a p-value cutoff α . Denote $\mathcal{N}(j)$ the neighborhood of node j in $\mathcal{G}$. Given an ordering of the p nodes, denoted by $\text{ORDER}(p)$, the search of ordered node pairs will be based on $\text{ORDER}(p)$. Starting with $s = 0$, we search over all ordered node pairs (j, l) such that $l \in \mathcal{N}(j)$ and $|\mathcal{N}(j) \setminus l| \geq s$. For each pair (j, l) , we search over all size- s subsets of $\mathcal{N}(j) \setminus l$ for a d -separation set $\mathcal{S}$, i.e. $(X_j \perp X_l) | X_{\mathcal{S}}$, by partial correlation tests. If we can find such a d -separation set $\mathcal{S}(j, l)$, we record the separation set and stop searching. After searching over all the node pairs, we update $\mathcal{G}$ by deleting all edges between node pairs with d -separation sets. Then we increase s by 1 and continue the procedure until each pair of adjacent nodes (j, l) satisfies $|\mathcal{N}(j) \setminus l| \leq s$. The resulting $\mathcal{G}$ is our estimated skeleton. With the d -separation sets $\{\mathcal{S}(j, l)\}$, we can direct a subset of the edges by a set of deterministic rules.

2.3 Joint Estimation of Multiple Skeletons with Soft Labels

In many real data analysis settings, the samples are labeled by experts or statistical methods (e.g., clustering or classification) with non-ignorable error rates. For example, breast cancer patients are typically classified into four major subtypes based on gene expression data, and some patients cannot be confidently classified into any subtype (Dai et al., 2015). To address this challenge, we propose to use *soft labeling*: instead of assigning a hard label to each observation, a probabilistic label vector $(w^{(1)}, \dots, w^{(K)})^T$ is computed such that $\approx \Pr(G = k | X)$. We refer to the method based on probabilistic labeling as *Soft MPenPC*.

Soft labels can be produced by either classification or clustering. When prior information or labels are available, we can compute probabilistic labels by soft classifiers, such as naive Bayes or quadratic discriminant analysis (QDA). Otherwise, we can estimate soft labels using probabilistic clustering methods, or apply soft classifiers on the clustered sample. The following toy example demonstrates that soft labels can provide more accurate estimates of class labels than hard labels. Let $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ represents n samples collected from a mixture of three Gaussian populations with equal weights. For any $k \in \{1, 2, 3\}$, the feature vector in class k has a Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}^{(k)}, \mathbf{I})$. Specifically, $\boldsymbol{\mu}^{(1)} = (2, 0)^T$, $\boldsymbol{\mu}^{(2)} = (0, 2)^T$, and $\boldsymbol{\mu}^{(3)} = (\sqrt{3} + 1, \sqrt{3} + 1)^T$. We create hard labels by k-means clustering with $k = 3$. Then we construct probabilistic labels using a naive Bayes classifier based on the clustering labels. We measure the accuracy of the estimated class labels by their average Manhattan distance from the true labels: $\sum_{i=1}^n \sum_{k=1}^K |\hat{w}_i^{(k)} - \mathbb{I}(g_i = k)| / n$, where $\hat{w}_i^{(k)} = \mathbb{I}(\hat{g}_i = k)$ for hard labels. The results show that soft labels often provide more accurate estimates of class labels (see Figure 1 in the Supplementary Materials).

Soft labels can be naturally incorporated into both steps of the MPenPC. Denote the full feature matrix $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)^T$, $\mathbf{n}^{(k)} = \sum_{i=1}^n w_i^{(k)}$, $\mathbf{w}^{(k)} = (w_1^{(k)}, \dots, w_n^{(k)})^T$, $\mathbf{W}^{(k)} = \text{diag}(\mathbf{w}^{(k)})$, and $\bar{\mathbf{x}}^{(k)} = \mathbf{X}^T \mathbf{w}^{(k)} / n^{(k)}$. In the neighborhood selection step, we use soft labels for weighted nodewise regressions. That is, for node j we estimate its neighborhood by

$$\underset{\gamma_j}{\operatorname{argmin}} \frac{1}{2n} \sum_{k=1}^K \left\| \mathbf{X}_j - \gamma_{j,0}^{(k)} - \mathbf{X}_{-j} \boldsymbol{\gamma}_j^{(k)} \right\|_{\mathbf{W}^{(k)}}^2 + P_{\lambda, \tau}(\boldsymbol{\gamma}_j), \quad (5)$$

where $\mathbf{X}_j$ denotes the j -th column of $\mathbf{X}$, and $\|\mathbf{a}\|_{\mathbf{W}} = (\mathbf{a}^T \mathbf{W} \mathbf{a})^{1/2}$. In the PC step, we can compute a weighted covariance matrix with soft labels:

$\hat{\Sigma}^{(k)} = (\mathbf{X} - \mathbf{1} \bar{\mathbf{x}}^{(k)T})^T \mathbf{W}^{(k)} (\mathbf{X} - \mathbf{1} \bar{\mathbf{x}}^{(k)T}) / n^{(k)}$, for $k=1, \dots, K$. Based on the weighted covariance matrices, we can perform partial correlation tests and apply the PC-stable algorithm. In the partial correlation tests for population k , we use $n^{(k)}$ as the sample size.

3. Computation of Soft Labels and Tuning Parameter Selection

In this section, we discuss some implementation details of the MPenPC methods. The neighborhood selection step is implemented based on the `grpreg` R package. The PC-stable algorithm in the skeleton estimation step is implemented using the `ParallelPC` R package.

3.1 Computation of Soft Labels

Given hard labels, which are often estimated by clustering, we can compute soft labels by probabilistic classification methods, for example, QDA and naive Bayes. Since we are interested in high dimensional problems, dimension reduction has to be done prior to applying methods like QDA. When hard labels are not available, we can either construct them by clustering, or estimate the soft labels directly by probabilistic clustering methods, such as Gaussian mixture models. From our experience, it is often better to perform dimension reduction before clustering, for example, by principal component analysis.

3.2 Parameter Tuning

In the neighborhood selection step of MPenPC, we use extended BIC (EBIC) (Chen and Chen, 2008) for parameter tuning of λ and τ . For standard regression, EBIC is defined as

$$\text{EBIC}_{\gamma} = -2 \log L_n(\hat{\boldsymbol{\beta}}) + s \log n + 2\psi \log \binom{p}{s}, 0 \leq \psi \leq 1,$$

where L_n denotes the likelihood, and $s = \|\hat{\boldsymbol{\beta}}\|_0$. As suggested by Chen and Chen (2008), ψ can be taken as $1 - (2 \log p / \log n)^{-1}$. For the nodewise regression in our proposed MPenPC, we consider the joint regression as a model with $(p-1)K$ parameters. Following the notations of (4), EBIC for the nodewise regression can be defined as

$$\text{EBIC}_{j,\gamma} = n \log \hat{\sigma}^2 + s \log n + 2\psi \log \binom{(p-1)K}{s},$$

where $s = s = \sum^{(k)} \|\hat{\gamma}_j^{(k)}\|_0$, and $\hat{\sigma}^2 = n^{-1} \sum_{k=1}^K \|\mathbf{X}_j^{(k)} - \hat{\gamma}_{j,0}^{(k)} - \mathbf{X}_j^{(k)} \hat{\gamma}_j^{(k)}\|_2^2$. For the soft MPenPC, we define $\hat{\sigma}^2 = n^{-1} \sum_{k=1}^K \|\mathbf{X}_j - \hat{\gamma}_{j,0}^{(k)} - \mathbf{X}_j \hat{\gamma}_j^{(k)}\|_{\mathbf{W}^{(k)}}^2$.

4. Theoretical Properties

We first define some notations. For the k -th population, $\mathbf{\Omega}^{(k)}$ denotes the precision matrix, from which the CIG $\mathcal{G}^{(k)}$ can be deduced. We denote the neighborhood of node j in $\mathcal{G}^{(k)}$ by $A_j^{(k)} = \{l: \Omega_{j,l}^{(k)} \neq 0 \text{ and } l \neq j\}$, and its complement by $\bar{A}_j^{(k)} = \{l: \Omega_{j,l}^{(k)} = 0\}$. Moreover, denote $A_j = \cup_{k=1}^K A_j^{(k)}$ and $\bar{A}_j = \cap_{k=1}^K \bar{A}_j^{(k)}$. Then each node in A_j is connected with node j in at least one $\mathcal{G}^{(k)}$, and nodes in $\bar{A}_j$ are *not* connected with the node j in any $\mathcal{G}^{(k)}$.

We assume some regularity conditions (C1)–(C8) on the underlying model and tuning parameters. More details are provided in the Supplementary Materials. We have the following theorem regarding the neighborhood selection (Stage I) of the MPenPC with hard labels.

THEOREM 1 (Stage I Consistency): (i) Under (C1)–(C5), with a probability of $1 - O(1/p)$, for all $j \in \{1; \dots; p\}$ there is a local minimizer $\hat{\gamma}_j$ to problem (4) such that

$$j-l \in \widehat{\mathcal{G}}^{(k)} \text{ if } l \in A_j^{(k)}, \text{ and } j-l \notin \widehat{\mathcal{G}}^{(k)} \text{ if } l \in \bar{A}_j^{(k)};$$

(ii) Under (C1)–(C6), with a probability of $1 - O(1/p)$, for all $j \in \{1; \dots; p\}$ there is a local minimizer $\hat{\gamma}_j$ to problem (4) such that $j-l \in \widehat{\mathcal{G}}^{(k)}$ if $l \in A_j^{(k)}$, and $j-l \notin \widehat{\mathcal{G}}^{(k)}$ if $l \in \bar{A}_j^{(k)}$.

Theorem 1 considers two different types of consistency for the neighborhood selection. In particular, with (C1)–(C5), the estimation is guaranteed to recover all edges that appear in at least one conditional independence graph $\mathcal{G}^{(k)}$. We call this group-union selection consistency. In this case, we can recover the *union* of the undirected graphs asymptotically. With condition (C6), we can obtain stronger consistency that correctly identifies all edges for each graph. Similar results for the soft MPenPC, with a condition on the soft labels, are included in the Supplementary Materials.

To the extent of our knowledge, this is the first theoretical result regarding the selection consistency of GEL-penalized estimation. It can be extended to other bi-level group penalties. According to the irrepresentability condition (C6), estimating $\gamma_{j,l}^{(k)}$ correctly as 0 becomes more difficult as the edge $j-l$ being shared by more graphs. By replacing the GEL with another bi-level sparsity-inducing penalty composed of concave penalties for both levels (Breheny and Huang, 2009; Chen and Sun, 2017), condition (C6) can be greatly relaxed. We do not pursue that approach in this work, however, due to the additional

computational burden for tuning parameter selection. Even if (C6) is not satisfied, the first part of Theorem 1 still guarantees consistent estimate of the union of edges from $\mathcal{G}^{(k)}$'s, and the next theorem guarantees that the PC step of the MPenPC can produce consistent skeleton estimates given the graph union.

THEOREM 2 (Stage II Consistency): *Assume we have perfect estimation of all the CIGs or their union from the Stage I, under (C1)–(C2) and (C8), there exists a p -value cutoff $\alpha \rightarrow 0$ such that the skeletons are recovered perfectly for all subpopulations with probability*

$$1 - O\left(\exp\left(-Cn^{1-2d_2}\right)\right) \rightarrow 1, \text{ for some } d_2 > 0.$$

Therefore, by combining the results of Theorems 1 and 2, the MPenPC method produces a consistent skeleton estimation with probability going to 1.

5. Simulation Studies

In this section, we use simulated examples to study the performance of our MPenPC methods. In particular, we justify the joint estimation and the usage of soft labeling by comparing the MPenPC with the original PenPC method. When applying the PenPC method, we can either estimate a single skeleton with all the samples, or estimate one skeleton for each class. We call the first approach as PenPC without grouping (PenPC - No Grouping) and the second as group-specific PenPC (Hard PenPC). For a comprehensive comparison, we also consider a group-specific PenPC with soft labels (Soft PenPC), which performs weighted regression at Stage I and uses weighted covariance for the PC-stable algorithm at Stage II. In this simulation, we implement the Hard and Soft MPenPC. The exponential penalty, i.e. $P(\boldsymbol{\theta}) = \lambda^2 \tau^{-1} \prod_j \{1 - \exp(-\lambda^{-1} \tau |\theta_j|)\}$, is used in both PenPC approaches. We first introduce the simulation settings and the implementation of methods in Section 5.1. Then the methods are compared by stage in Sections 5.2 and 5.3.

5.1 Simulation Settings

In the simulation, we consider Gaussian mixture settings under which the corresponding DAGs share some common edges. To this end, we first generate K DAGs with certain similarity, based on which we specify the Gaussian distributions. In particular, the k -th Gaussian component is specified through the structure equation model,

$$X = \boldsymbol{\nu}^{(k)} + \mathbf{B}^{(k)}X + Z, \quad (6)$$

where $Z \sim N(\mathbf{0}, \mathbf{I})$. For convenience, the variables are ordered such that $b_{j,l}^{(k)} = 0, \forall l \geq j$, where $b_{j,l}^{(k)}$ is the (j, l) -th entry of $\mathbf{B}^{(k)}$. The bias vector $\boldsymbol{\nu}^{(k)}$ is set as $\nu_j^{(k)} = \delta$ if $4(k-1) < j < 4k$ and 0 otherwise, so we can adjust the group difference via δ .

The DAGs are generated with two different models, the Erdos-Renyi (ER) and the Barabasi-Albert (BA), respectively. Denote p the dimension of $\mathbf{X}$, then $\{\mathbf{B}^{(k)}; k=1, \dots, K\}$ are $p \times p$ lower triangular matrices. With the two models, we generate $\mathbf{B}^{(k)}$'s as follows.

- (ER model) We generate $K + 1$ random $p \times p$ matrices, denoted by $\mathbf{A}^{(0)}, \dots, \mathbf{A}^{(K)}$, independently. Initialize each $\mathbf{A}^{(k)}$ with all 0's. Randomly select $\lceil \pi_E p(p-1)/2 \rceil$ entries in the lower triangular matrix (excluding the diagonal), and fill them with random values from $\text{Uniform}([-1, -0.5] \cup [0.5, 1])$. Using $\mathbf{A}^{(0)} = (a_{j,l}^{(0)})_{p \times p}$ as the basis matrix, we construct $\mathbf{B}^{(k)}$ by taking $b_{j,l}^{(k)} = a_{j,l}^{(0)}$ for $\lceil \pi_0 p^2 \rceil$ random entries and $b_{j,l}^{(k)} = a_{j,l}^{(k)}$ for the rest, where $\pi_0 \in [0, 1]$, and $a_{j,l}^{(k)}$ is the (j, l) -the entry of $\mathbf{A}^{(k)}$.
- (BA Model) The procedure is basically the same as above, except $\mathbf{A}^{(k)}$'s. For each $\mathbf{A}^{(k)}$, we generate a random DAG with the BA model as follows. Starting from an empty DAG with the node 1, we add the node 2 and the edge $1 \rightarrow 2$ to the DAG. Then at each step, we add the node j and e random edges to the graph such that the probability of edge $l \rightarrow j$ is proportional to the neighborhood size of node l ($l < j$). Then we construct $\mathbf{A}^{(k)}$ by filling $a_{j,l}^{(k)}$ with random values from $\text{Uniform}([-1, -0.5] \cup [0.5, 1])$ if the edge $l \rightarrow j$ exists in the DAG.

Note that the graph sparsity is determined by π_E and e respectively in the two models, and π_0 tunes the similarity among the K DAGs. In particular, any two of the K DAGs have about $\pi_0^2 \times 100\%$ overlapping in expectation. Examples of ER and BA models with the same sparsity are shown in Figure 1. One may observe that the BA model has more variation on the degree of connections per node, with both hubs (i.e. heavily connected nodes) as well as nodes with few connections. This is due to the *scale free* property of the BA model. The different sparsity patterns of ER and BA models can result in different challenges in skeleton estimation.

In the simulation, we assign equal weights to K classes, and generate n samples from the Gaussian mixture distribution, denoted by $\{x_i, 1 \leq i \leq n\}$. We perform principal component analysis (PCA) on the whole data. Based on the first 20 principal components, the hard labels are constructed by the k -means clustering, denoted by $\{g_i, 1 \leq i \leq n\}$. Soft labels are then computed by QDA based on the hard labels, using the same 20 principal components. Let $K = 4$, $p = 500$, and $n = 400$. For the ER model, we set $\pi_E = 1/500$, and for the BA model, we set $e = 1$. For both models, we consider a *high overlapping* ($\pi_0 = 0.7$) setting as well as a *low-overlapping* one ($\pi_0 = 0.3$). In the latter case, any two graphs have only about 10% edges in common. For each simulation setting, we run 100 repetitions and evaluate the performance of all the methods by stage. Evaluation criteria include true positive rate (TPR) and false positive rate (FPR) averaged over classes for both stages. In particular, denote $\mathcal{G}$ and $\widehat{\mathcal{G}}$ as the true CIG or DAG skeleton and its estimate. Then

$$\text{TPR} = \frac{|\widehat{\mathcal{G}} \cap \mathcal{G}|}{|\mathcal{G}|}, \quad \text{FPR} = \frac{|\widehat{\mathcal{G}} \setminus \mathcal{G}|}{(p^2 - p)/2 - |\mathcal{G}|},$$

which are respectively 1 and 0 if $\widehat{\mathcal{G}}$ recovers $\mathcal{G}$ perfectly. For Stage II, we also include the estimation accuracy results of the CPDAGs in terms of the average Structural Hamming

Distance (SHD), which is computed as the number of flipping, addition, and deletion operations to turn each estimated CPDAG to the true CPDAG (the smaller the better).

Due to limited space, we only display the results for the BA example in the paper. The results for the ER example and additional scenarios are included in the Supplementary Materials.

5.2 Stage I: Neighborhood Selection

In this subsection, we compare the CIG estimation (Stage I) of all methods under different settings. We select the tuning parameters of all methods (λ for PenPC, τ and λ for MPenPC) by EBIC. As we can see from Table 1 (upper panel), all four methods, except the PenPC without grouping, recover most of the edges effectively. Both the group-specific PenPC and the MPenPC benefit from using soft labels (Soft PenPC vs. Hard PenPC, Soft MPenPC vs. Hard MPenPC) – with slightly higher FPR, they have substantially higher TPR. We also notice that while the MPenPC methods can take advantage of the graph overlaps, they appear to be less effective than the group-specific PenPC under the low overlapping setting at this stage. However, the sparsity level of estimated graphs vary dramatically across methods, which poses challenges to our analysis. In particular, the estimates of the PenPC without grouping have far fewer edges than those of other methods, thus we do not include it in later comparisons.

For a more thorough comparison of the methods, we evaluate the estimation of all the methods at different sparsity levels by changing the tuning parameter λ in nodewise regressions. Figure 2 displays the results for BA model. In the high overlapping setting, the MPenPC methods always have higher TPRs than the PenPC methods at the same FPR level. Even in the low-overlapping setting, the Soft MPenPC has a better accuracy than the group-specific PenPC methods. Moreover, both the group-specific PenPC and the MPenPC with soft labels outperform their counterparts with hard labels, which justifies the usage of soft labeling.

5.3 Stage II: Skeleton Estimation

At Stage II, we apply the PC-stable algorithm to obtain skeleton estimation based on the CIGs estimated in Stage I (selected by EBIC). For each of the K classes, the input of the PC-stable algorithm includes an initial graph and a correlation matrix. The procedure is similar for all methods, except that the soft PenPC and the soft MPenPC use weighted correlation, and the PenPC without grouping uses the overall correlation of all samples. Table 1 (lower panel) summarizes the performance evaluation of all the methods at Stage II for the BA model with the p -value cutoff $\alpha = 0.02$.

From Table 1, we observe that all methods except the PenPC without grouping produce much sparser skeletons compared to the undirected graphs at Stage I. The sparsity of group-specific PenPC and MPenPC estimates becomes comparable. In the high overlapping setting, the MPenPC with soft labels has a clear edge over the group-specific PenPC with soft labels, with higher TPR and lower FPR. The comparison is the same for their counterparts with hard labels. Under the low overlapping setting, the differences between the group-specific PenPC and the MPenPC become smaller compared with the results at Stage I.

The results of CPDAG estimation also confirm the advantages of the proposed method in terms of lower SHD.

With a fixed initial graph, the sparsity of skeleton estimation can be tuned by α for each of the methods. For a complete comparison among the methods, we also present the results for a range of significance levels, namely 0.005, 0.01, ..., 0.3, in Figure 3. As α increases, the skeleton estimation becomes less sparse and both FPR and TPR increase. With respect to the BA example, the relative performance of different methods is relatively stable at each significance level. The comparisons mostly conform to our earlier observations. In the high overlapping setting, both MPenPC variants have significant advantages over their group-specific PenPC counterpart. The PenPC and the MPenPC with soft labels are more accurate than those with hard labels. In the low overlapping setting, the performance of MPenPC methods and their group-specific PenPC counterpart are similar. The group-specific PenPC estimates have slightly higher TPRs than the MPenPC, but also with higher FPRs.

The simulation results for the ER model show similar patterns of relative performance of all the methods (Supplementary Table 1, Figures 2–3). We have also conducted simulations with non-overlapping DAGs. In this scenario, the group-specific PenPC methods have slightly better performance than the MPenPC methods, but soft-label-based methods still have better performance than hard-label-based methods (Supplementary Figures 4–7). In summary, our simulation study shows that our joint estimation methods, including both Hard and Soft MPenPC, often produce more accurate skeleton estimates than separate estimation, when the multiple DAGs have reasonable similarities. Moreover, for either the PenPC or the MPenPC method, the soft labels always benefit the estimation.

6. Cancer Genomic Applications

Breast cancer is the most commonly diagnosed cancer type in females and second leading cancer death in females (Siegel et al., 2016). Gene expression data collected from cancer samples are very informative to study the molecular characteristics of breast cancer. For example, based on the gene expression pattern, breast cancer can be divided into five subtypes: basal, HER2 over-expression (HER2), luminal A (lumA), luminal B (lumB), and normal-like (The Cancer Genome Atlas Network, 2012). We seek to use DAG skeleton to study gene co-expression patterns in breast cancer. To account for the similarity and differences across subtypes, we apply our MPenPC method to jointly estimate DAG skeletons of multiple subtypes.

We obtain level 3 gene expression data (UNC IlluminaHiSeq_RNASeqV2 pipeline) of TCGA breast cancer patients (The Cancer Genome Atlas Network, 2012) from TCGA data portal. The gene expression dataset is in the format of raw counts of sequencing reads for more than 20,000 genes. We limit our analysis on 405 Caucasian patients and filter out genes with low expression in the majority of the patients. Specifically, we require the raw counts to be 20 for at least 25% of the individuals and 15,816 genes pass this filtering. The distribution of these 405 patients in the 5 breast cancer subtypes are: basal ($n = 64$), HER2 ($n = 21$), lumA ($n = 222$), lumB ($n = 91$), and normal-like ($n = 8$). In the following analysis, we remove the 8 individuals of normal-like subtype due to its small sample size.

To focus on genes that are more relevant to cancer biology, we select 3,466 genes that belong to at least one of 17 cancer-relevant gene sets (the Molecular Signatures Database C6 oncogenic signatures gene sets (Subramanian et al., 2005)). See the Supplementary Materials for a complete list of specific gene sets. For each of these 3,466 genes, we regress it against all other genes by penalized regression using the *log*-penalty (Sun et al., 2010) and select tuning parameters by EBIC. Two genes are connected if each of them is selected in the regression model for the other gene. Then we remove the genes that are not connected with any other gene of the same gene set, and use the remaining 1,528 genes in the following analysis.

As in the simulation study, we use different methods to estimate the skeletons for four subtypes. We use a PathwayCommons dataset (Cerami et al., 2010), which is based on multiple databases, as the benchmark. In this case, we have a single benchmark graph for all groups/subtypes. It is created by connecting any two genes that *interact with* or *are in complex with* each other. Since the subtypes have already been specified, we only need to create soft labels by classification. Naive Bayes is used to compute the soft labels. At the second stage of skeleton estimation (removing edges by conditional dependence testing), we use a significant level of $\alpha = 0.02$ for all methods.

We compare the performance of different methods at two stages. During Stage I, by varying tuning parameters, each method produces undirected graphs of varying sparsity level for each subtype. The MPenPC method with soft labels generally recover more edges defined in PathwayCommons than other methods, including the MPenPC with hard labels, at the same sparsity level (Figure 4). Since the EBIC-selected graph estimates of the four methods have very different sparsity levels, we compare the performances of these methods across sparsity levels of Stage I, after applying the PC-stable algorithm with a fixed significance level $\alpha = 0.02$ during Stage II (Figure 5). We can see that the soft MPenPC again produces the best skeleton estimates among all methods. Therefore, joint estimation and soft labels can benefit the skeleton estimation in this application.

To further explore the DAG skeleton within each gene set. We rerun our analysis using MPenPC for each gene set separately for three subtypes: basal, lumA or lumB. The HER2 subtype is not included due to its relatively small sample size. Since subtypes lumA and lumB are more similar, we expect more similarity between their skeleton estimates and less similarity between the skeleton estimates for lumA/lumB and basal. This is indeed the case, as shown in Supplementary Figure 8 in the Supplementary Materials.

We further illustrate the skeleton estimates for these three cancer subtypes for a gene set related with TP53. TP53 is a tumor suppressor gene that induces cell cycle arrest and apoptosis in abnormal cells. TP53 pathway activity is among the major differences between basal and lumA/lumB cancers. TP53 mutation rate is 12%, 32%, and 84% for lumA, lumB, and basal, respectively. Integrative analysis of multiple types of -omic data from TCGA samples suggest TP53 pathway remains largely functional in luminal A samples, is often inactivated in a subset of luminal B samples, and is inactivated in most basal samples (The Cancer Genome Atlas Network, 2012). The gene set we analyzed related with TP53 is the union of two gene sets of MSigDB: P53_DN.V1_UP and P53_DN.V1_DN, which

corresponds to genes that are up-regulated or down-regulated in NCI-60 panel of cell lines with mutated TP53. The DAG skeleton estimated by either soft MPenPC or soft PenPC show that genes involved in negative regulation of apoptosis are enriched among the genes with 4 or more connections in basal subtype, but less so for lumA/lumB, suggesting both methods identify biologically interesting subtype-specific features. In addition, soft MPenPC identify more edges shared between basal and lumA/lumB subtypes than soft PenPC (112 edges by MPenPC vs. 66 edges by PenPC, Chi-squared test p -value 1.4×10^{-6}), suggesting the advantage of joint analysis of multiple subtypes.

7. Conclusion

In this paper, we propose the MPenPC method to estimate the DAG skeletons with heterogeneous samples. By taking advantage of the similarity among the DAGs, the MPenPC method can produce more accurate estimates than separate estimation. In particular, we take into account possible labeling errors in this scenario, and propose a remedy with soft labels. The effectiveness of our method is demonstrated with numerical examples.

The ideas of joint estimation and soft labels can also be combined with other DAG estimation methods. For example, Nandy et al. (2015) has demonstrated how to appropriately combine the CIG estimation and the greedy equivalence search for single-DAG estimation. Similar hybrid approaches may be used for heterogeneous populations.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

ACKNOWLEDGEMENTS

The authors thank the editor, the associate editor, and the reviewer for their helpful suggestions that led to significant improvement of the article. The research was supported in part by NSF grant IIS1632951, and NIH grants R01GM126550 and P01CA142538.

REFERENCES

- Breheny P (2015). The group exponential lasso for bi-level variable selection. *Biometrics* 71, 731–740. [PubMed: 25773593]
- Breheny P and Huang J (2009). Penalized methods for bi-level variable selection. *Statistics and its interface* 2, 369. [PubMed: 20640242]
- Cerami EG, Gross BE, Demir E, Rodchenkov I, Babur Ö, Anwar N, Schultz N, Bader GD, and Sander C (2010). Pathway commons, a web resource for biological pathway data. *Nucleic acids research* 39, D685–D690. [PubMed: 21071392]
- Chen J and Chen Z (2008). Extended bayesian information criteria for model selection with large model spaces. *Biometrika* 95, 759–771.
- Chen T-H and Sun W (2017). Prediction of cancer drug sensitivity using high-dimensional omic features. *Biostatistics* 18, 1–14. [PubMed: 27324412]
- Chickering DM (2002). Learning equivalence classes of bayesian-network structures. *Journal of Machine Learning Research* 2, 445–498.
- Colombo D and Maathuis MH (2014). Order-independent constraint-based causal structure learning. *Journal of Machine Learning Research* 15, 3741–3782.

- Dai X, Li T, Bai Z, Yang Y, Liu X, Zhan J, and Shi B (2015). Breast cancer intrinsic subtype classification, clinical use and future trends. *American Journal of Cancer Research* 5, 2929–2943. [PubMed: 26693050]
- Danaher P, Wang P, and Witten DM (2014). The joint graphical lasso for inverse covariance estimation across multiple classes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 76, 373–397. [PubMed: 24817823]
- Fan J and Li R (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association* 96, 1348–1360.
- Friedman N and Koller D (2003). Being bayesian about network structure. a bayesian approach to structure discovery in bayesian networks. *Machine Learning* 50, 95–125.
- Guo J, Levina E, Michailidis G, Zhu J, et al. (2011). Joint estimation of multiple graphical models. *Biometrika* 98, 1–15. [PubMed: 23049124]
- Ha MJ, Sun W, and Xie J (2016). Penpc: A two-step approach to estimate the skeletons of high-dimensional directed acyclic graphs. *Biometrics* 72, 146–155. [PubMed: 26406114]
- Han SW, Chen G, Cheon M-S, and Zhong H (2016). Estimation of directed acyclic graphs through two-stage adaptive lasso for gene network inference. *Journal of the American Statistical Association* 111, 1004–1019. [PubMed: 28239216]
- Heckerman D, Geiger D, and Chickering DM (1995). Learning bayesian networks: The combination of knowledge and statistical data. *Machine Learning* 20, 197–243.
- Huang J, Ma S, Xie H, and Zhang C-H (2009). A group bridge approach for variable selection. *Biometrika* 96, 339–355. [PubMed: 20037673]
- Kalisch M and Bühlmann P (2007). Estimating high-dimensional directed acyclic graphs with the pc-algorithm. *Journal of Machine Learning Research* 8, 613–636.
- Meinshausen N and Bühlmann P (2006). High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics* 34, 1436–1462.
- Nandy P, Hauser A, and Maathuis MH (2015). High-dimensional consistency in score- based and hybrid structure learning. *arXiv preprint arXiv:1507.02608*.
- Oates CJ, Smith JQ, Mukherjee S, and Cussens J (2016). Exact estimation of multiple directed acyclic graphs. *Statistics and Computing* 26, 797–811.
- Pearl J (2009). *Causality*. Cambridge university press.
- Schmidt M, Niculescu-Mizil A, and Murphy K (2007). Learning graphical model structure using l1-regularization paths. In *Proceedings of the 22nd National Conference on Artificial Intelligence*, volume 2, pages 1278–1283. AAAI Press.
- Siegel RL, Miller KD, and Jemal A (2016). Cancer statistics, 2016. *CA: A Cancer Journal for Clinicians* 66, 7–30. [PubMed: 26742998]
- Spirtes P, Glymour CN, and Scheines R (2000). *Causation, Prediction, and Search Adaptive computation and machine learning*. MIT Press.
- Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gillette MA, Paulovich A, Pomeroy SL, Golub TR, Lander ES, and Mesirov JP (2005). Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences* 102, 15545–15550.
- Sun W, Ibrahim JG, and Zou F (2010). Genomewide multiple-loci mapping in experimental crosses by iterative adaptive penalized regression. *Genetics* 185, 349–359. [PubMed: 20157003]
- Tsamardinos I, Brown LE, and Aliferis CF (2006). The max-min hill-climbing bayesian network structure learning algorithm. *Machine Learning* 65, 31–78.
- Yuan M and Lin Y (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 68, 49–67.
- Yuan M and Lin Y (2007). Model selection and estimation in the gaussian graphical model. *Biometrika* 94, 19–35.

Figure 1. Examples of DAGs generated by the ER model (left) with $\pi_E = 0.02$ and the BA model (right) with $e = 2$. In both models, we set $K = 3$, $p = 50$, and $\pi_0 = 0.4$.

Figure 2.

Neighborhood selection performance of different methods at Stage I in BA scenario: $K = 4$, $p = 500$, $e = 1$, $\delta^2 = 0.05$, $\pi_0 = 0.7$ for (a) and 0.3 for (b). The x -axes and y -axes represent FPR and TPR respectively. The graph sparsity at the neighborhood selection stage varies for different tuning parameter λ . The tuning parameter γ for MPenPC methods is preselected by EBIC.

Figure 3. Skeleton estimation performance of different methods at Stage II in BA scenario. The high overlapping scenarios have $\pi_0 = 0.7$, and the low overlapping scenarios have $\pi_0 = 0.3$. The x -axes represent the significance level α of the PC-stable algorithm. The y -axes represent TPR (the left panel) and FPR (the right panel) respectively.

Figure 4.

Performance comparison of all methods at Stage I by cancer subtype. The x - axes represent the total number of edges in estimated graphs corresponding to different λ values; the y -axes represent the number of overlapping edges in estimated graphs.

Figure 5.

Performance comparison of all methods at Stage II by cancer subtype. The x - axes represent the significance level α of the PC-stable algorithm. The y -axes represent the number of overlapping edges (left panel) and total number of edges (right panel) in estimated skeletons.

Table 1

Performance of different methods at both stages for the BA-model examples ($K = 4$, $p = 500$, $e = 1$, $\pi_0 = 0.7$ (High Overlapping) or 0.3 (Low Overlapping), and $\delta^2 = 0.05$). $TPR = |\widehat{\mathcal{G}} \cap \mathcal{G}|/|\mathcal{G}|$ and $FPR = |\widehat{\mathcal{G}} \setminus \mathcal{G}| / \{(p^2 - p)/2 - |\mathcal{G}|\}$, where $\mathcal{G}$ and $\widehat{\mathcal{G}}$ denote the true and the estimated graphs respectively. The numbers outside and inside parentheses are averages and standard errors respectively based on 100 repetitions.

Stage I. Neighborhood Selection		
High Overlapping	TPR	FPR
PenPC (No Grouping)	0.6469(0.0022)	0.0017(0.0000)
Hard PenPC	0.8176(0.0022)	0.0353(0.0000)
Soft PenPC	0.8474(0.0022)	0.0390(0.0002)
Hard MPenPC	0.8451(0.0019)	0.0446(0.0002)
Soft MPenPC	0.8675(0.0019)	0.0473(0.0002)
Low Overlapping	TPR	FPR
PenPC (No Grouping)	0.4515(0.0020)	0.0024(0.0000)
Hard PenPC	0.7991(0.0044)	0.0359(0.0000)
Soft PenPC	0.8338(0.0029)	0.0391(0.0000)
Hard MPenPC	0.7884(0.0048)	0.0452(0.0000)
Soft MPenPC	0.8178(0.0039)	0.0472(0.0000)
Stage II. Skeleton Estimation		
High Overlapping	TPR	FPR
PenPC (No Grouping)	0.6348(0.0025)	0.0016(0.0000)
Hard PenPC	0.8375(0.0030)	0.0055(0.0000)
Soft PenPC	0.8553(0.0030)	0.0054(0.0000)
Hard MPenPC	0.8629(0.0030)	0.0052(0.0000)
Soft MPenPC	0.8785(0.0030)	0.0052(0.0000)
Low Overlapping	TPR	FPR
PenPC (No Grouping)	0.4548(0.0017)	0.0022(0.0000)
Hard PenPC	0.8369(0.0060)	0.0055(0.0000)
Soft PenPC	0.8615(0.0046)	0.0054(0.0000)
Hard MPenPC	0.8321(0.0066)	0.0053(0.0000)
Soft MPenPC	0.8574(0.0051)	0.0052(0.0000)
Stage II. CPDAG Estimation		
High Overlapping	SHD	
PenPC (No Grouping)	612.99(3.57)	
Hard PenPC	565.46(2.94)	
Soft PenPC	550.22(2.27)	
Hard MPenPC	498.87(2.54)	
Soft MPenPC	492.61(2.28)	
Low Overlapping	SHD	
PenPC (No Grouping)	784.79(2.94)	

Hard PenPC	580.61(2.22)
Soft PenPC	558.45(2.45)
Hard MPenPC	511.22(2.55)
Soft MPenPC	496.84(2.45)