HHS Public Access

Author manuscript

Neural Comput. Author manuscript; available in PMC 2019 November 23.

Published in final edited form as:

Neural Comput. 2019 June ; 31(6): 1183–1214. doi:10.1162/neco_a_01190.

Learning Moral Graphs in Construction of High-Dimensional Bayesian Networks for Mixed Data

Suwa Xu,

Department of Biostatistics, University of Florida, Gainesville, FL 32611, U.S.A.

Bochao Jia,

Lilly Corporate Center, Eli Lilly and Company, Indianapolis, IN 46285, U.S.A.

Faming Liang

Department of Statistics, Purdue University, West Lafayette, IN 47906, U.S.A.

Abstract

Bayesian networks have been widely used in many scientific fields for describing the conditional independence relationships for a large set of random variables. This letter proposes a novel algorithm, the so-called p-learning algorithm, for learning moral graphs for high-dimensional Bayesian networks. The moral graph is a Markov network representation of the Bayesian network and also the key to construction of the Bayesian network for constraint-based algorithms. The consistency of the p-learning algorithm is justified under the small-n, large-p scenario. The numerical results indicate that the p-learning algorithm significantly outperforms the existing ones, such as the PC, grow-shrink, incremental association, semi-interleaved hiton, hill-climbing, and max-min hill-climbing. Under the sparsity assumption, the p-learning algorithm has a computational complexity of $O(p^2)$ even in the worst case, while the existing algorithms have a computational complexity of $O(p^3)$ in the worst case.

1. Introduction

Graphical models have proven to be a useful tool for describing conditional independence relationships for a large set of random variables. Two types of graphical models are commonly used, Markov networks and Bayesian networks. The Markov network, also known as the Markov random field, is a model over an undirected graph. During the past decade, the gaussian graphical model (GGM), as a special case of Markov networks, has been used in many scientific fields, from computer vision to natural language processing to genomics. Due to the mathematical tractability of the gaussian distribution, some efficient algorithms have been developed for learning the structure of GGMs—for example, graphical Lasso (Yuan & Lin, 2007; Friedman, Hastie, & Tibshirani, 2008), nodewise regression (Meinshausen & Bühlmann, 2006), and ψ-learning (Liang, Song, & Qiu, 2015).

Correspondence should be addressed to F.L. fmliang@purdue.edu.

S.X. and B.J. are co-first authors and contributed equally to this letter.

The Bayesian network is a model over a directed acyclic graph. Regarding the relationship between Markov networks and Bayesian networks, Pearl (1988) stated that the major weakness of Markov networks is their inability to represent induced and nontransitive dependencies; two independent variables will be directly connected by an edge merely because some other variable depends on both. As a result, many useful independencies go unrepresented in the network. Bayesian networks overcome this deficiency by using the richer language of directed graphs, where the directions of the arrows permit us to distinguish genuine dependencies from spurious dependencies induced by hypothetical observations. To illustrate this point, let us consider a set of random variables, where there can be four combinations of independence statements for any two variables. Table 1 gives an example for each of the three cases that are representable by Markov networks. The fourth case, that X and Y are marginally independent but dependent conditioned on variable Z, is not representable by a Markov network. As a compromise, the Markov network uses a cycle Ⓧ –Ⓩ –Ⓨ – Ⓧto represent the mutual dependence of the three variables. However, the fourth case can be easily represented by a Bayesian network using a υ-structure (defined in section 2) Ⓧ → Ⓩ ← Ⓨ, which includes two convergent directions on the edges Ⓧ – Ⓩ and Ⓨ – Ⓩ. In Bayesian formula, this situation can be described by

$$\pi(X, Y|Z) = \frac{\pi(Z|X, Y)\pi(X)\pi(Y)}{\pi(Z)} \neq \pi(X|Z)\pi(Y|Z),$$

which quite often holds for real problems. In Bayesian networks, the direction of edges represents the “parent of” relationship. For this reason, Bayesian networks have often been used in causal inference (see e.g., Spirtes, 2010).

Although the Bayesian network is statistically attractive, learning its structure can be difficult, especially under the small-n, large-p scenario, where n denotes the sample size and p denotes the number of random variables involved in the network. None of the existing algorithms developed for high-dimensional GGMs (e.g., graphical Lasso, nodewise regression, and ψ-learning) can be trivially extended to Bayesian networks due to the fundamental difference in their structures. In particular, the υ-structure needs care, especially when extending a Markov network learning algorithm to Bayesian networks.

The existing Bayesian network learning algorithms can be traced to three categories: constraint based, score based, and hybrid. The constraint-based algorithms, stemming from the inductive causation (IC) algorithm (Verma & Pearl, 1991), are to learn Bayesian networks by conducting a series of conditional independence tests. The grow-shrink (GS; Margaritis, 2003), incremental association (Tsamardinos, Aliferis, & Statnikov, 2003; Yaramakala & Margaritis, 2005), and PC (Spirtes, Glymour, & Scheines, 2000) algorithms belong to this category. These algorithms basically consist of three stages: they first learn the moral graph of the Bayesian network, then identify the υ-structures contained in the moral graph, and finally identify the derived directions for nonconvergent edges according to logic rules. The moral graph, which is formally defined in section 2, can be viewed as a Markov network representation of the Bayesian network. The difficulty with these algorithms is that they are not well scaled for high-dimensional problems (Aliferis, Statnikov, Tsamardinos, Mani, & Koutsoukos, 2010). They often involve some conditional tests with the conditioning

set size close to p, which cannot be carried out or is very unreliable when p is greater than n. It is remarkable that under the sparsity assumption, which bounds the neighborhood size of each node, the PC algorithm has been shown by Kalisch and Bühlmann (2007) to be consistent and can execute in a polynomial time of p. Therefore, the PC algorithm has been considered in the literature as the state-of-the-art algorithm for learning high-dimensional Bayesian networks. Recent applications and extensions of the algorithm can be found in Colomboi, Maathuis, Kalisch, and Richardson (2012), Verdugo et al. (2013), Harris and Drton (2013), McGeachie, Chang, and Weiss (2014), Cui, Groot, and Heskes (2016), Ha, Sun, and Xie (2016), among others.

The score-based algorithms are to find a network that optimizes a selected scoring function (e.g., entropy; Herskovits & Cooper, 1990), minimum description length (Lam & Bacchus, 1994), and Bayesian scores (Cooper & Herskovits, 1992; Heckerman, Geiger, & Chickering, 1995), which measures the fitness of each feasible network to the data. Under appropriate conditions, the score-based algorithms can also be shown to be consistent (see Chickering, 2002, and Nandy, Hauser, & Maathuis, 2016, for the low- and high-dimensional cases, respectively). Unfortunately, the task of finding a network structure that optimizes the scoring function is NP-hard (Chickering, 1996), and the search process often stops at a local optimal structure. The hybrid algorithms are to combine constraint-based and score-based algorithms to offset their respective weakness. Both the sparse candidate algorithm (Friedman, Pe'er, & Nachman, 1999) and the max-min hill-climbing (MMHC) algorithm (Tsamardinos, Brown, & Aliferis, 2006) belong to this category. They first restrict the parent set of each node to a smaller set and then search for the network that maximizes a scoring function subject to the constraints imposed by the restricted parent sets.

In this letter, we propose a new algorithm for learning moral graphs for high-dimensional mixed types of data. With the moral graph, the structure of the Bayesian network can be easily determined by completing the remaining stages of the constrained-based algorithms: υ-structure identification and derived direction identification. For example, the υ-structure can be identified using the collider set algorithm (Pellet & Elisseeff, 2008) or local neighborhood algorithm (Margaritis & Thrun, 2000). Upon completion of the υ-structure identification stage, the skeleton and colliders of the Bayesian network can be identified. Given the skeleton and colliders, a maximally directed Bayesian network can be obtained following the four necessary and sufficient rules (see, e.g., Verma & Pearl, 1992, and Kjaerulff & Madsen, 2010), which ensure that no directed cycles and additional colliders are created in the graph. The consistency of the proposed algorithm is justified under the small-n, large-p scenario. The numerical results indicate the superiority of the proposed algorithm over the existing ones. Under the sparsity assumption, the proposed algorithm has a computational complexity bounded by $O(p^2)$, while the computational complexity of the existing algorithms is $O(p^{2+a})$ for some $a > 0$.

In this letter, the mixed data are restricted to those consisting of gaussian and multinomial or binomial variables only. In this scenario, the joint distribution of the mixed variables is well defined for the moral graph (see Lee & Hastie, 2015), for which the conditional distribution of each continuous variable given the rest is still gaussian and the conditional distribution of each discrete variable given the rest is still multinomial. Therefore, all conditional

independence tests involved in the proposed algorithm can be conducted under the framework of generalized linear models (GLMs). Extension of the proposed algorithm to other types of mixed data is discussed in section 6.

The remainder of this letter is organized as follows. Section 2 gives a brief review of the theory of Bayesian networks. Section 3 describes the proposed algorithm, with the theoretical justification for its consistency deferred to the appendix. Section 4 illustrates the proposed algorithm using simulated examples along with comparisons with some existing algorithms. Section 5 reports the results for two real data examples. Section 6 concludes with a brief discussion of possible extensions of the proposed algorithm to other types of mixed data.

2. A Brief Review of Bayesian Network Theory

This section gives a brief review for the Bayesian network theory required by this letter. For a full account of the theory, we refer to Jensen and Nielsen (2007) and Scutari and Denis (2015).

A Bayesian network can be represented by a directed acyclic graph (DAG) $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$, where $\boldsymbol{V}$, with a slight abuse of notation, denotes a set of p nodes corresponding to the p variables $X_1, \ldots, X_p$, and $\boldsymbol{E} = (e_{ij})$ denotes the adjacency matrix or arc sets. The joint distribution of $X_1, \ldots, X_p$ is given by

$$P(\boldsymbol{X}) = \prod_i q\left(X_i | Pa\left(X_i\right)\right), \tag{2.1}$$

where $Pa(X_i)$ denotes the parent nodes or variables of X_i in the network, and $q(\cdot|\cdot)$ specifies the conditional distribution of X_i given its parent nodes. In Bayesian networks, each node X_i is conditionally independent of its non-descendants (the nodes for which there is no path to reach from X_i) given its parents. This is the so-called local Markov property of Bayesian networks. The local Markov property implies that the parents are not completely independent from their children in the Bayesian network. With Bayes's theorem, it is easy to show how information on a child can change the distribution of the parent. A convergent connection $X_i \rightarrow X_k \leftarrow X_j$ is called a υ-structure if there is no arc connecting X_i and X_j. In addition, X_k is often called a collider node, and the convergent connection is then called an unshielded collider. The υ-structure enables Bayesian networks to represent a type of relationship that Markov networks cannot, that is, X_i and X_j are marginally independent but also dependent conditional on X_k.

The Markov blanket of a node X_i is the set consisting of the parents of X_i, the children of X_i, and the spouse nodes that share a child with X_i. The Markov blanket of a node $X_i \in \boldsymbol{V}$ is the minimal subset of $\boldsymbol{V}$ such that X_i is independent of all other nodes conditioned on it. The Markov blanket is symmetric; if node X_i is in the Markov blanket of X_j, then X_j is also in the Markov blanket of X_i.

The moral graph, illustrated by Figure 1, is an undirected graph that is constructed by (1) connecting the nonadjacent nodes in each υ-structure with an undirected arc and (2)

ignoring the directions of other arcs. This transformation, called moralization, provides a simple way to transform a Bayesian network into the corresponding Markov network. In the Markov network, all dependencies are explicitly represented, even those that would be implicitly implied by υ-structures in a Bayesian network. In the moral graph, the neighboring set of each node forms its Markov blanket.

Finally, we give the definition for the faithfulness of graphical models. Let $\boldsymbol{M}$ denote the dependence structure of the probability distribution of $\boldsymbol{X}$—the set of conditional dependence relationships between any triplet $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ of subsets of $\boldsymbol{X}$. The graph $\boldsymbol{G}$ is said to be faithful or isomorphic to $\boldsymbol{M}$ if for all disjoint subsets $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ of $\boldsymbol{X}$, we have

$$\boldsymbol{A} \perp_p \boldsymbol{B}|\boldsymbol{C} \Leftrightarrow \boldsymbol{A} \perp_G \boldsymbol{B}|\boldsymbol{C}, \tag{2.2}$$

where the left denotes the conditional independence in probability, and the right denotes the separation in graph (i.e., $\boldsymbol{C}$ is a separator of $\boldsymbol{A}$ and $\boldsymbol{B}$). For a Markov network, $\boldsymbol{C}$ is said to be a separator of $\boldsymbol{A}$ and $\boldsymbol{B}$ if for every $a \in \boldsymbol{A}$ and $b \in \boldsymbol{B}$, all paths from a to b have at least one node in $\boldsymbol{C}$. For Bayesian networks, $\boldsymbol{C}$ is said to be a separator of $\boldsymbol{A}$ and $\boldsymbol{B}$ if, along every path between a node in $\boldsymbol{A}$ and a node in $\boldsymbol{B}$, there is a node υ satisfying one of the following two conditions: (1) υ has converging arcs and neither υ nor any of its descendants is in $\boldsymbol{C}$, and (2) υ is in $\boldsymbol{C}$ and does not have converging arcs. The faithfulness provides a theoretical basis for establishing consistency for constraint-based algorithms.

3. Learning High-Dimensional Moral Graphs

3.1 The *p*-Learning Algorithm.

Under the assumption of faithfulness, the moral graph can be learned via conditional independence tests $X_i \perp_P X_j|\boldsymbol{S}_{ij} \setminus \{X_i, X_j\}$ for all ordered pairs of (i, j), where $\boldsymbol{S}_{ij}$ denotes the Markov blanket of X_i or X_j. If the conditional independence is true, then there is no arc between X_i and X_j. Otherwise, X_i and X_j are in each other's Markov blanket.

In the literature, quite a few algorithms have been proposed for learning Markov blankets, including the grow-shrink Markov blanket (Margaritis, 2003) and incremental association (Tsamardinos, Aliferis, & Statnikov, 2003; Yaramakala & Margaritis, 2005) algorithms. The grow-shrink Markov blanket algorithm works like a forward selection procedure, which first continues to add new variables to the conditioning set (starting with an empty set) until the conditional independence holds or there are no more variables to add, and then shrinks the conditioning set by removing the variables outside the blanket. The incremental association algorithm is an enhancement of the grow-shrink Markov blanket algorithm, which reduces the number of conditional tests by arranging the order of the variables to add to the conditioning set. A fundamental problem with these algorithms is that they often need to perform some conditional tests with the size of the conditioning set close to p. When p is greater than n, such tests cannot be carried out or are very unreliable. Their computational complexity is $O(p^{2+a})$ for some $0 < a \leq 1$, where the factor p^a accounts for the number of conditional independence tests performed for each of p^2 pairs of nodes. In the worst case that the graph is fully connected, a is equal to 1 for all the algorithms.

In what follows, we present a new algorithm for learning moral graphs, which can work under the scenario $n \ll p$ and has a computational complexity of $O(p^2)$ even in the worst case. Instead of identifying the exact Markov blanket for each node, we propose to identify a super-Markov blanket $\tilde{\boldsymbol{S}}_i$ for each node X_i such that $\boldsymbol{S}_i \subseteq \tilde{\boldsymbol{S}}_i$ holds, where $\boldsymbol{S}_i$ denotes the Markov blanket of the node X_i. Let ϕ_{ij} denote the output of the conditional independence test $X_i \perp_P X_j | \boldsymbol{S}_i \setminus \{X_i, X_j\}$, that is, $\phi_{ij} = 1$ if the conditional independence holds and 0 otherwise. Let $\tilde{\phi}_{ij}$ denote the output of the conditional independence test $X_i \perp_P X_j | \tilde{\boldsymbol{S}}_i \setminus \{X_i, X_j\}$. Theorem 1 shows that under the faithfulness assumption, ϕ_{ij} and $\tilde{\phi}_{ij}$ are equivalent in learning moral graphs.

Theorem 1. *Assume the faithfulness holds. Let $\boldsymbol{S}_i$ denote the Markov blanket of X_i, and let $\tilde{\boldsymbol{S}}_i$ denote a superset of $\boldsymbol{S}_i$. Then ϕ_{ij} and $\tilde{\phi}_{ij}$ are equivalent in learning moral graphs in the sense that*

$$\phi_{ij} = 1 \Leftrightarrow \tilde{\phi}_{ij} = 1.$$

Proof. If $\phi_{ij} = 1$, then $\boldsymbol{S}_i \setminus \{X_i, X_j\}$ forms a separator of X_i and X_j. Since $\boldsymbol{S}_i \subset \tilde{\boldsymbol{S}}_i$, $\tilde{\boldsymbol{S}}_i \setminus \{X_i, X_j\}$ is also a separator of X_i and X_j. By faithfulness, we have $\tilde{\phi}_{ij} = 1$. If $\tilde{\phi}_{ij} = 1$, then X_i and X_j are conditionally independent and $\tilde{\boldsymbol{S}}_i \setminus \{X_i, X_j\}$ forms a separator of X_i and X_j. Since $\tilde{\boldsymbol{S}}_i \subset \boldsymbol{V}$, $\boldsymbol{V} \setminus \{X_i, X_j\}$ is also a separator of X_i and X_j and the conditional independence $X_i \perp_P X_j | \boldsymbol{V} \setminus \{X_i, X_j\}$ holds. By the total conditioning property (property 7 in Pellet & Elisseeff, 2008), which shows that $X_j \in \boldsymbol{S}_i \Leftrightarrow X_i \not\perp_p X_j | \boldsymbol{V} \setminus \{X_i, X_j\}$, we have $X_j \notin \boldsymbol{S}_i$. Therefore, $\phi ij = 1$ holds. □

By the symmetry of X_i and X_j, theorem 1 also holds if $\boldsymbol{S}_i$ is replaced by $\boldsymbol{S}_j$ and $\tilde{\boldsymbol{S}}_i$ is replaced by $\tilde{\boldsymbol{S}}_j$. Although ϕ_{ij} and $\tilde{\phi}_{ij}$ are equivalent in learning moral graphs, the size of the super-Markov blanket $\tilde{\boldsymbol{S}}_i$ should be as small as possible considering the power of the conditional independence tests. A large $\tilde{\boldsymbol{S}}_i$ often reduces the power of the conditional independence test.

Based on theorem 1, we propose the so-called *p*-learning algorithm (see algorithm 1) for learning moral graphs, which provides an efficient way to learn the Markov blanket for each node simultaneously.

Algorithm 1: *p*-Learning Algorithm

a. Screening for parents and children nodes: Find a superset of parentsand children for each node X_i:

- **i.** For each ordered pair of nodes (X_i, X_j), $i, j = 1, 2, \ldots, p$, conduct the marginal independence test $X_i \perp_p X_j$ and obtain the *p*-value.
- **ii.** Conduct a multiple hypothesis test at level α_1 to identify the pairs of nodes that are dependent. Denote the superset by $\boldsymbol{A}_i$ for $i = 1, \ldots, p$. If the size of $\boldsymbol{A}_i$ is greater than $n/(c_{n1} \log(n))$ for a prespecified constant

c_{n1}, reduce it to $n/(c_{n1}\log(n))$ by removing the variables having larger p-values in the marginal independence tests.

b. Spouse nodes amendment. For each node X_i, find the spouse nodes that are not included in $\boldsymbol{A}_i$, that is, find the set $\boldsymbol{B}_i = \{X_j : X_j \notin \boldsymbol{A}_i, \exists X_k \in \boldsymbol{A}_i \cap \boldsymbol{A}_j\}$ for $i = 1, \ldots, p$, where X_j is a node not connected but sharing a common neighbor with X_i. If the size of $\boldsymbol{B}_i$ is greater than $n/(c_{n2}\log(n))$ for a prespecified constant c_{n2}, reduce it to $n/(c_{n2}\log(n))$ by removing the variables having larger p-values in the spouse test $X_i \perp_p X_j | X_k$.

c. Screening for the moral graph. Construct the moral graph based on conditional independence tests:

i. For each ordered pair of nodes (X_i, X_j), $i, j = 1, 2, \ldots, p$, conduct the conditional independence test $X_i \perp_P X_j | \tilde{\boldsymbol{S}}_{ij} \backslash \{i, j\}$, where $\tilde{\boldsymbol{S}}_{ij} = \boldsymbol{A}_i \cup \boldsymbol{B}_i$ if $|\boldsymbol{A}_i \cup \boldsymbol{B}_i \setminus \{i, j\}| \leq |\boldsymbol{A}_j \cup \boldsymbol{B}_j \setminus \{i, j\}|$ and $\tilde{\boldsymbol{S}}_{ij} = \boldsymbol{A}_j \cup \boldsymbol{B}_j$ otherwise.

ii. Conduct a multiple hypothesis test at level α_2 to identify the pairs of nodes for which they are conditionally dependent, and set the adjacency matrix $\hat{E}_{mb}$ accordingly, where $\hat{E}_{mb}$ denotes the adjacency matrix of the moral graph.

As annotated in algorithm 1, step a is to find a superset of parents and children for each node. As pointed out in the appendix, A_i also contains the spouse nodes that are marginally dependent with X_i. Step b is to find the spouse nodes that are not included in the superset A_i, that is, the nodes that are marginally independent of X_i but dependent on X_i conditioned on their common child. Then for each node X_i, we have $\boldsymbol{S}_i \subset \boldsymbol{A}_i \cup \boldsymbol{B}_i$. Hence, we can set $\tilde{\boldsymbol{S}}_i = \boldsymbol{A}_i \cup \boldsymbol{B}_i$. It follows from theorem 1 that this algorithm is valid for learning moral graphs.

The p-values of the individual tests for the marginal independence and conditional independence were obtained via the likelihood ratio tests (LRT) under the GLM setting. The multiple hypothesis tests were done using an empirical Bayes method developed by Liang and Zhang (2008). The advantage of this method is that it allows for the general dependence between test statistics. Other multiple hypothesis tests, which account for the dependence between test statistics (e.g., Benjamini, Krieger, & Yekutieli, 2006) can also be applied here. The performance of multiple hypothesis tests depends on their significance levels. Following from theorem 1, a slightly large value of α_1 should be used to reduce the risk of $\boldsymbol{S}_i \nsubseteq \boldsymbol{A}_i \cup \boldsymbol{B}_i$. On the other hand, the power of the conditional independence tests in step c is adversely affected by the size of the superset $\tilde{\boldsymbol{S}}_i$ and thus by the value of α_1. However, we also find that such an effect is not very sensitive to the size of $\tilde{\boldsymbol{S}}_i$; including a few extra variables in $\tilde{\boldsymbol{S}}_i$ will not hurt the power of the moral graph screening tests much. To balance the two ends, we suggest setting $\alpha_1 = 0.1$ or 0.2. Throughout examples in this letter, we set $\alpha_1 = 0.1$ and $\alpha_2 = 0.05$ unless otherwise stated.

In the algorithm, we have restricted the sizes of $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ based on the sparsity assumption, given by condition C of section 3.2, for the high-dimensional Bayesian network. By

assuming that each conditional distribution $q(\cdot)$ in equation 2.1 can be represented by the probability distribution function of a normal linear regression or multiclass logistic regression, we are able to bound the size of each set $\boldsymbol{A}_i$ by $O(n/\log(n))$ based on the theory of sure independence screening (Fan & Lv, 2008; Fan & Song, 2010). (Refer to the appendix for details on theoretical development). Further, under the sparsity assumption, we are also able to bound the size of each set $\boldsymbol{B}_i$ by $O(n/\log(n))$. Therefore, the size of each superset $\widetilde{\boldsymbol{S}}_i = \boldsymbol{A}_i \cup \boldsymbol{B}_i$ can be bounded by $O(n/\log(n))$. With appropriate choices of c_{n1} and c_{n2}, we can always have $|\tilde{\boldsymbol{S}}_i| < n$ holding for all $i = 1, 2, \ldots, p$ when n is reasonably large. In this letter, we set $c_{n1} = c_{n2} = 1$ for all examples. In practice, when the sample size n is small, even the size of $\boldsymbol{B}_i$ is smaller than the prespecified threshold, we might still conduct spouse tests to reduce its size further. Since the size of $\tilde{\boldsymbol{S}}_i$ adversely affects the power of the moral graph screening test, a smaller $\boldsymbol{B}_i$ is always preferred.

Since both the marginal tests in step a and the conditional independence tests in step c need to be performed only once for each ordered pair of nodes, and the multiple hypothesis tests can be done in a linear time of the total number of p-values, the computational complexity of the p-learning algorithm is $O(p^2)$, which is independent of the underlying structure of the Bayesian network. In the worst case, the computational complexity of the existing algorithms is $O(p^3)$.

3.2 Consistency of the *p*-Learning Algorithm.

This section establishes the consistency of the proposed p-learning algorithm. To achieve this goal, we assume that the joint distribution of the underlying true moral graph can be reexpressed as

$$p(\boldsymbol{x}, \boldsymbol{y}|\Theta) \propto \exp\left\{ -\frac{1}{2}\sum_{s=1}^{p_c}\sum_{t=1}^{p_c}\theta_{st}x_s x_t + \sum_{s=1}^{p_c}\vartheta_s x_s + \sum_{s=1}^{p_c}\sum_{j=1}^{p_d}\rho_{sj}(y_j)x_s + \sum_{j=1}^{p_d}\sum_{r=1}^{p_d}\psi_{rj}(y_r, y_j)\right\}, \tag{3.1}$$

where x_s denotes the sth of p_c continuous variables and y_j denotes the jth of p_d discrete variables. The joint model is parameterized by $\Theta = [\{\theta_{st}\}, \{\vartheta_s\}, \{\rho_{sj}\}, \{\psi_{rj}\}]$. (See Yang et al., 2014, for more general developments for the joint distribution of mixed graphical models.)

As Lee and Hastie (2015) showed, the conditional distributions of equation 3.1 are given by gaussian linear regression and multiclass logistic regressions. Therefore, all the conditional independence tests conducted in the moral graph learning and υ-structure identification stages are well defined, which are equivalent to test whether the corresponding regression coefficients equal to zero. To be specific, the test in step a of algorithm 1 is equivalent to testing the coefficient of X_j in the GLM,

$$X_i \sim 1 + X_j; \tag{3.2}$$

the test in step c of algorithm 1 is equivalent to testing the coefficient of X_j in the GLM,

$$X_i \sim 1 + X_j + \sum_{k \in \hat{S}_{ij} \setminus \{i,j\}} X_k; \tag{3.3}$$

and the test in the υ-structure identification stage is equivalent to testing the coefficient of X_j in the GLM

$$X_i \sim 1 + X_j + \sum_{k \in \boldsymbol{D}_{ij}} X_k, \tag{3.4}$$

where $\boldsymbol{D}_{ij}$ denotes a subset of $Bd(X_i) \setminus \{X_j\}$ and $Bd(X_i)$ denotes the neighboring set of X_i in the moral graph.

Under the GLM assumption, the consistency of algorithm 1 can be proved based on the theory of sure independence screening established in Fan and Song (2010), the theory of the ψ-learning algorithm established in Liang et al. (2015), and the theory established in Kalisch and Bühlmann (2007) for the PC algorithm. Parallel to the conditions assumed by the PC algorithm for the gaussian case, we assume the following conditions:

A. *Faithfulness:* The moral graph is faithful, for which the joint distribution can be expressed in a gaussian-multinomial distribution, equation 3.1.

B. *High dimensionality:* The dimension $p_n = O(\exp(n^{\delta}))$, where $0 \le \delta < (1 - 2\kappa)\alpha/(\alpha + 2)$ for some positive constants $\kappa < 1/2$ and $\alpha > 0$, and the subscript n of p_n indicates the dependence of the dimension p on the sample size n.

C. *Sparsity:* The maximum size of the Markov blanket of each node, denoted by $\tilde{q}_n = \max_{1 \le j \le p_n} |\boldsymbol{S}_i|$, satisfies $\tilde{q}_n = O\left(n^b\right)$ for some constant $0 \le b < (1 - 2\kappa)\alpha/(\alpha + 2)$, where $\boldsymbol{S}_i$ denotes the Markov blanket of node i.

D. *Identifiability:* The regression coefficients satisfy

$$\inf\left\{|\beta_{ij|\boldsymbol{C}}|; \beta_{ij|\boldsymbol{C}} \neq 0, i, j = 1, 2, \ldots, p_n, \boldsymbol{C} \subseteq \left\{1, 2, \ldots, p_n\right\} \setminus \{i, j\}, |\boldsymbol{C}| \le O(n/\log(n))\right\} \ge c_0 n^{-\kappa},$$

for some constant $c_0 > 0$, where κ is as defined in condition B and $\beta_{ij|C}$ denotes the true regression coefficient of X_j in the GLM, equations 3.2, 3.3, or 3.4.

Since the p-learning algorithm works based on the theory of sure independence screening, we follow Fan and Song (2010) to give some conditions for GLMs (see the appendix for details) such that the resulting Bayesian network satisfies the sparsity condition C. Fan and Song (2010) showed that variable screening can be done in regression coefficients or in p-values of the conditional independence tests (χ^2 test with a degree of freedom of 1), which are equivalent to each other. For this reason, the identifiability condition, D, is given in terms of regression coefficients. Under these conditions, we show in the appendix that algorithm 1 is consistent, that is, $P\left(\widehat{\boldsymbol{E}}_{mb}^{(n)} = \boldsymbol{E}_{mb}^{(n)}\right) \to 1$ and $P\left(\widehat{\boldsymbol{E}}_{v}^{(n)} = \boldsymbol{E}_{v}^{(n)} | \widehat{\boldsymbol{E}}_{mb}^{(n)} = \boldsymbol{E}_{mb}^{(n)}\right) \to 1$ as $n \to \infty$, where $\boldsymbol{E}_{mb}^{(n)}$ denotes the adjacency matrix of the moral graph, $\boldsymbol{E}_{v}^{(n)}$ denotes the set of υ-structures,

and $\widehat{\boldsymbol{E}}_{mb}^{(n)}$ and $\widehat{\boldsymbol{E}}_{v}^{(n)}$ denote the estimators of $\boldsymbol{E}_{mb}^{(n)}$ and $\boldsymbol{E}_{v}^{(n)}$ obtained by the p-learning algorithm, respectively.

4. Simulation Studies

This section illustrates algorithm 1 for learning moral graphs using simulated examples, along with comparisons with a variety of existing algorithms.

4.1 Mixed Data with an AR(2) Structure.

Following Kalisch and Bühlmann (2007), we simulated the mixed data in the following procedure: (1) fix an order of variables, (2) randomly mark half of the variables as continuous and the rest as binary, (3) fill the adjacency matrix $\boldsymbol{E}$ with some given structures, and (4) generate the data according to the adjacency matrix in a sequential manner.

For this example, the variable X_1, which corresponds to the first node of the Bayesian network, was generated through a gaussian random variable $Y_1 \sim N(0, 1)$. We set1 $X_1 = Y_1$ if X_1 was set to be continuous, and $X_1 \sim Binomial\left(n, 1/\left(1+e^{-Y_1}\right)\right)$ otherwise. The other variables X_j's, $j = 2, 3, \ldots, p$, were then sequentially generated by setting

$$Y_j = \sum_{i=1}^{p} 0.5 E_{ij} X_i, \tag{4.1}$$

$$X_j = \begin{cases} Y_j + \epsilon_j, & \text{if } X_j \text{ is continuous,} \\ \text{Binomial}\left(n, \dfrac{\exp(Y_j)}{1+\exp(Y_j)}\right), & \text{if } X_j \text{ is binary,} \end{cases} \tag{4.2}$$

where $\epsilon_1, \ldots, \epsilon_p$ are independent and identically distributed standard gaussian random variables, and E_{ij} denotes the (i, j)th entry of $\boldsymbol{E}$. In this example, we first set $\boldsymbol{E}$ to be of an AR(2) structure given by

$$E_{i,j} = \begin{cases} 1, & \text{if } j - i = 1, i = 1, \ldots, (j-1), \\ 1, & \text{if } j - i = 2, i = 1, \ldots, (j-2), \\ 0, & \text{otherwise .} \end{cases} \tag{4.3}$$

Let p_c and p_d denote the numbers of continuous and discrete variables, respectively. In our simulations, we fixed $p_c = p_d = 100$, while varying the sample size n at four values: $n = 100$, 200, 500, and 1000. For each value of n, 10 data sets were generated independently. The p-learning algorithm was first applied to this example with the default settings $\alpha_1 = 0.1$ and $\alpha_2 = 0.05$ and then compared with several popular algorithms that were originally designed for learning Bayesian networks. The popular algorithms include the constraint-based algorithms such as PC, grow-shrink (GS), incremental association Markov blanket (IAMB), and semi-interleaved HITON-PC (hiton); the score-based algorithm such as hill-climbing (HC); and the hybridbased algorithm such as max-min hill-climbing (MMHC). All of these algorithms

were first employed to learn the Bayesian networks using the R package *bnlearn* under their default settings, and then the moral graphs were generated from the learned Bayesian networks via the function *moral* in the R package. The Markov blanket discovery algorithms by Gao and Ji (2017a, 2017b) can also be applied to produce a moral graph, but their codes are not available to the public and thus are not included for comparison.

To evaluate the performance of each algorithm, the receiver operating characteristic (ROC) curve was drawn. The ROC curve is a plot of false-positive rate (FPR) versus true-positive rate (TPR), defined by

$$\text{FPR} = \frac{FP}{FP + TN}, \quad \text{TPR} = \frac{TP}{TP + FN},$$

where *TP*, *FP*, and *FN* denote true positives, false positives and false negatives, respectively. Figure 2 shows the average ROC curves over 10 independent data sets for each algorithm. For the algorithm 1, in order to plot the ROC curve, we vary the value of a_2. For all other algorithms, the package *bnlearn* provides a bootstrap method to calculate the arc presence probabilities, and the ROC curve can be plotted by varying the cutoff value of the probability. Table 2 reports the averaged area under the ROC curve and the associated standard deviation for all the algorithms. The comparison indicates the superiority of the proposed algorithm over the existing ones.

4.2 Sensitivity Analysis.

The *p*-learning algorithm consists of two parameters: a_1 and a_2. The a_1 controls the size of the super-Markov blanket for each node, while a_2 controls the false discovery rate (FDR) and thus sparsity of the resulting moral graph. In general, we suggest that a_1 be set to a reasonably large value in order to reduce the risk of $S_i \not\subseteq \tilde{S}_i$, where S_i and $\tilde{S}_i$ denote the Markov blanket and super-Markov blanket of node *i*, respectively. The a_2 is a user-specified parameter, which should be set by the user according to his or her own purpose. For $a1$, we conducted a sensitivity analysis with the results reported in Table 3, where the data were generated as in section 4.1 with an AR(2) structure and $p_c = p_d = p/2$, and the ROC curve was plotted by fixing the value of a_1 and varying the value of a_2 from 0 to 1. The results show that the performance of the *p*-learning algorithm is quite robust to the choice of a_1; the AUC (area under the ROC curve) values are not much changed as a_1 varies from 0.05 to 0.25.

4.3 Mixed Data with General Dependence Structures.

For a thorough comparison, we also considered several other moral graph structures such as alarm, barley, ecoli, and magic, which are four popular networks obtained at the Bayesian Network Repository (http://www.bnlearn.com/bnrepository/). Since the data for these networks are not of mixed type, we simulated the mixed type of data with their known network structures as follows. We first randomly marked half of the variables as continuous and the rest as binary, filled the adjacency matrix $\boldsymbol{E}$ with the given DAG structure, and then simulated the observations in the following steps:

1. Define an ancestor set of variables $\boldsymbol{A}$, which refer to the nodes with no parents.

2. For each node $i \in \boldsymbol{A}$, generate a gaussian random variable $Y_i \sim N(0, 1)$. Set $X_i = Y_i$ if X_i is continuous, and set $X_i \sim Binomial\,(n, 1/(1 + e^{-Y_i}))$ otherwise.

3. Define the offspring set $\boldsymbol{O}$, which refers to the nodes with parents in the ancestor set $\boldsymbol{A}$ but $\boldsymbol{O} \cap \boldsymbol{A} = \emptyset$.

4. For each node $j \in \boldsymbol{O}$, generate X_j by setting

$$Y_j = \sum_{i \in A} 0.5 E_{ij} X_i, \tag{4.4}$$

$$X_j = \begin{cases} Y_j + \epsilon_j, & \text{if } X_j \text{ is continuous,} \\ \text{Binomial}\left(n, \dfrac{\exp(Y_j)}{1 + \exp(Y_j)}\right), & \text{if } X_j \text{ is binary,} \end{cases} \tag{4.5}$$

where ϵ_j is a standard gaussian random variable and E_{ij} denotes the (i, j) entry of $\boldsymbol{E}$.

5. Update the ancestor set by $\boldsymbol{A} = \boldsymbol{A} \cup \boldsymbol{O}$.

6. Iterate between steps 3 and 5 until all nodes are included in the ancestor set $\boldsymbol{A}$.

For each structure, 10 independent data sets were simulated. Figure 3 and Table 4 report the averaged ROC curves and areas under the curves (AUCs) produced by different algorithms for these data sets. The comparison shows that algorithm 1 outperforms other algorithms `for all types of Bayesian network structures.

4.4 Binary Data with an AR(2) Structure.

For a thorough test for the performance of the proposed algorithm, we have also considered the case with binary variables only. We simulated 10 independent data sets as in section 4.1, except that all $p = 200$ variables were set to binary. Figure 4 shows the average ROC curves over the 10 data sets for different algorithms, and Table 5 reports the averaged area under the ROC curve and the associated standard deviation. The comparison indicates the superiority of the proposed algorithm over the existing ones.

4.5 Time Complexity.

This study compares the time complexity of the proposed algorithm with the existing ones. In this study, we let the dimension p increase with n in the polynomial $p = 0.01n^2$. Such a polynomial setting facilitates the measurement of the time complexity of each algorithm in the form of $O(p^\nu)$. Different settings of (n, p) were considered, including (100, 100), (141, 200), (200, 400), (264, 700), and (300, 900). For each setting of (n, p), an independent data set was simulated as in section 4.1 with $p_c = p_d = p/2$, different algorithms were applied to learn the moral graph from the data set, and the CPU time (in minutes) was recorded on a Xeon Gold 6126 CPU@2.60 GHz machine. (See Table 6 for details.) For each algorithm, the recorded CPU time was fitted by a linear regression

$$\log(T) = \beta_0 + \nu \log(p) + \epsilon, \quad \epsilon \sim N\left(0, \sigma^2\right),$$

where σ^2 denotes the variance of the random error and T denotes the recorded CPU time. The fitting results, including R^2 and the OLS estimate of ν and its standard deviation, are reported in Table 6. In addition, we report in Table 6 the p-values of the tests for the hypotheses,

$$H_0 : \nu_m \leq \nu_p \quad \text{versus} \quad H_1 : \nu_m > \nu_p, m \in \{\text{GS, IAMB, hiton, PC, hc, mmhc}\},$$

where $\nu_p = 1.861$ denotes the value of ν for the p-learning algorithm and ν_m denotes the value of ν for the algorithm m. The tests show that the time complexity of the p-learning algorithm is significantly lower than the existing algorithms (at a significance level of 0.05). More important, algorithm 1 outperforms the existing algorithms in recovering the underlying moral graph.

5. Real Example

This study aims to learn an interactive genomics network for Breast Cancer (BRCA), which incorporates gene expressions (mRNA-array data), mutations, and DNA methylations. The data set was downloaded from the Cancer Genome Atlas (TCGA) at https://tcga-data.nci.nih.gov/tcga/. For mRNA gene expressions, we used the microarray data collected from the Agilent custom 244,000 array (Agilent) platform, which includes 17,814 normalized mRNA expressions. The mutations were defined by a binary variable, where 1 stands for all the nonsilent mutations and 0 for silent mutations or not being mutated, resulting in 16,806 genes with mutations. DNA methylations were measured at the probe level, where each probe represents a CpG site. This data set consists of 27,578 CpG sites. Based on the suggestions from Zhang, Burdette, and Wang (2014), we classified methylation levels using the k-means clustering algorithm into either hyper or hypo states, which are represented as 0 and 1, respectively. In summary, the data set consists of mRNA gene expressions, mutations, and DNA methylations, which are either gaussian or binary distributed. We present our analysis for the genes that overlap with the BRCA pathways available in the *Kyoto Encyclopedia of Genes and Genomes* (KEGG). For mutations, we use only those included in the BRCA pathway. For methylations, we include only the CpG islands where the genes in the BRCA pathway locate. As a result, we have a data set with 129 mRNA gene expressions, 11 mutations, and 315 DNA methylations, with 287 observations.

In the genomics network, there exist some parents-children associations from mutations or DNA methylation to gene expressions, which imply that the mutations and DNA methylations can regulate gene expressions. However, according to biological knowledge, there should not exist the parents-children association among themselves; that is, any mutation-mutation, methylation-methylation, or mutation-methylation edges should not exist in the skeleton of Bayesian networks. Therefore, to generate biologically meaningful network structures, edge restriction rules should be considered when constructing mixed

graphical models. Under the framework of algorithm 1, these restriction rules can be easily incorporated into network construction.

Suppose that edges in the skeleton network can exist only among a subset of variables or they are exempt among certain types of variables. In this scenario, we define a set $\tilde{\boldsymbol{E}}$, which contains all pairs of variables with possible edges. For example, in the genomic network, edges are exempted from the pairs between discrete variables; hence, $\tilde{\boldsymbol{E}}$ contains all (i, j) pairs, $i, j = 1, 2, \ldots, p$ excluding the pairs between discrete variables. Then a restricted p-learning algorithm can be proposed by modifying steps a.i and c.i in algorithm 1 as follows:

a.i′. For each ordered pair of nodes (X_i, X_j), where $(i, j) \in \tilde{\boldsymbol{E}}$, conduct the marginal independence test $X_i \perp_p X_j$ and obtain the p-value.

c.i′. For each ordered pair of nodes (X_i, X_j), where $(i, j) \in \tilde{\boldsymbol{E}}$, conduct the conditional independence test $X_i \perp_p X_j | \tilde{\boldsymbol{S}}_{ij} \backslash \{i, j\}$, where $\tilde{\boldsymbol{S}}_{ij} = \boldsymbol{A}_i \cup \boldsymbol{B}_i$ if $|\boldsymbol{A}_i \cup \boldsymbol{B}_i \backslash \{i, j\}| \leq |\boldsymbol{A}_j \cup \boldsymbol{B}_j \backslash \{i, j\}|$ and $\tilde{\boldsymbol{S}}_{ij} = \boldsymbol{A}_j \cup \boldsymbol{B}_j$ otherwise.

Under the edge restriction case, the independence and conditional independence screening are conducted only for the potential pairs of nodes in the set $\tilde{\boldsymbol{E}}$, and therefore the edges in the resulting network can be chosen only from the set $\tilde{\boldsymbol{E}}$. However, based on the definition of moral graphs, which should link the two nodes together if they have at least one common child, we should finally add edges between the pairs $(i, j) \notin \tilde{\boldsymbol{E}}$ if they share at least a common child. In applying the p-learning algorithm to this example, we set the parameters $\alpha_1 = 0.05$ and $\alpha_2 = 0.02$. The resulting moral network is shown in Figure 5.

From Figure 5, some hub genes, mutations, or methylations can be identified, which might play an important role in the development of breast cancer. A hub gene refers to a gene with with strong connectivity to other genes, mutations, or methylations. The hub mutation or hub methylation can be defined similarly, which might regulate the expression of quite a few genes. Table 7 lists the top five hub genes, mutations, and methylations identified by algorithm 1 for this data set, some of which have been verified in the existing literature. For example, PIK3R1 is the first hub gene, for which Cizkova et al. (2013) stated that PIK3R1 underexpression is an independent prognostic marker in breast cancer. NOTCH2 is another hub gene, for which Wang et al. (2016) claimed that NOTCH2 is downregulated and plays suppressive roles in breast cancer and the high NOTCH2 expression is shown to predict good survival for breast cancer patients. We also identified the mutation TP53, which is known to be the most frequent genetic alterations in breast cancer (Bertheau et al., 2013; Silwal-Pandit et al., 2014). Moreover, Silwal-Pandit et al. (2014) reported that TP53 mutation status is a strong marker of prognosis and has distinct prognostic relevance across different breast cancer subtypes. As for hub methylations, we identified CpG sites cg01230931, which is located in the promoter region of the gene APC. Virmani et al. (1998) mentioned that the aberrant of APC methylation had been reported in breast cancer and the frequency of APC methylation is significantly higher in breast cancer cases groups than healthy controls and also increases with tumor stage and size.

For comparison, we have applied other algorithms to this data set. Figure 6 showed the networks produced by the incremental association Markov blanket (IAMB), hill-climbing

(hc), max-min hill-climbing (mmhc), and semi-interleaved HITON-PC (hiton) algorithms under their default settings in the *bnlearn* package. The networks produced by the IAMB and hiton algorithms are too sparse and lose too much information about genomics associations. The network produced by the mmhc algorithm reveals some hub genes such as PIK3CD and FGF1, but it fails to identify the TP53 mutation and many important methylations. The hc algorithm produced an overly dense network, which has too many highly connected genes. Both the grow-shrink (GS) and PC algorithms produced an empty network. The comparison indicates that the proposed algorithm outperforms all others for this example.

6. Discussion

We have proposed a novel algorithm, the so-called p-learning algorithm, for learning moral graphs for high-dimensional Bayesian networks, and justified the consistency of the p-learning algorithm under the small-n, large-p scenario. The numerical results indicate that algorithm 1 significantly outperforms the existing ones, such as the PC, grow-shrink, IAMB, hiton-pc, hill-climbing, and max-min hill-climbing algorithms.

In this letter, we consider only the binary data and the mixed data of gaussian and binary variables. Extension of algorithm 1 to some other types of mixed data is straightforward. For example, for nongaussian continuous random variables, the nonparanormal transformation proposed by Liu, Lafferty, and Wasserman (2009) can be applied to gaussianize the data prior to applying the proposed algorithm. For Poisson random variables, the random-effect model-based transformation proposed by Jia, Xu, Xiao, Lamba, and Liang (2017) can be first applied to continuize the data, and the nonparanormal transformation can then be applied to gaussianize the data. The negative binomial data can be treated in the same way. For some other types of discrete data, we might regroup and treat them as multinomial data.

Finally, we note that the moral graph is a Markov network representation of a Bayesian network, and learning the Markov network for mixed data is of great interest in the current literature. For example, Cheng, Li, Levina, and Zhu (2013) proposed a conditional gaussian distribution-based algorithm and Fan, Liu, and Ning (2017) proposed a semiparametric latent variable algorithm to tackle the problem. The conditional gaussian distribution used in Cheng et al. (2013) is similar to equation 3.1 but includes more interaction terms. They used the nodewise regression method to estimate the Markov network structure. The semiparametric latent variable algorithm works by introducing a latent gaussian variable for each of the discrete variables and then estimating the Markov network using a regularization method. However, as Fan et al. (2017) stated, the conditional independence between the latent variables does not imply the conditional independence between the observed discrete variables. The copula PC algorithm (Cui et al., 2016) might suffer from the same problem.

Acknowledgments

F.L.'s research was partially supported by grants DMS-1612924, R01GM117597, and R01-GM126089. We thank the editor, associate editor, and two referees for their constructive comments, which led to significant improvement of this letter.

Appendix:: Consistency of Moral Graph Learning

To indicate that p can grow as a function of n, we rewrite p as p_n, rewrite the distribution function P in (2.1) as $P^{(n)}$, and rewrite the true Bayesian network $\boldsymbol{G}$ as $\boldsymbol{G}^{(n)} = (\boldsymbol{V}^{(n)}, \boldsymbol{E}^{(n)})$. Let $\mathcal{G}^{(n)} = \left(\mathcal{V}^{(n)}, \mathcal{E}^{(n)}\right)$ denote the marginal association network, where $\mathcal{V}^{(n)} = V^{(n)}$ and the association are measured by the coefficients of the marginal regression,

$$X_i \sim 1 + X_j, \quad i, j = 1, 2, \ldots, p_n, \tag{A.1}$$

which can be normal linear regression or multiclass logistic regression depending on the type of X_i. Let γ_{ij} denote the coefficient of X_j in equation A.1, which is called the marginal regression coefficient (MRC) in this letter. Then we have

$$\mathcal{E}^{(n)} = \left\{(i, j): \gamma_{ij} \neq 0, i, j = 1, \ldots, p_n\right\}.$$

Let v_n denote a threshold value of the MRC, let $\widehat{\mathcal{E}}_{v_n}$ denote the edge set of the network obtained through MRC thresholding at v_n, and let $\widehat{\mathcal{E}}_{v_n}$ denote the neighborhood of node i in $\widehat{\mathcal{E}}_{v_n}$. That is, we define

$$\widehat{\mathcal{E}}_{v_n} = \left\{(i, j): |\hat{\gamma}_{ij}| > v_n\right\}, \quad \text{and} \quad \widehat{\mathcal{E}}_{v,i} = \left\{j: j \neq i, |\hat{\gamma}_{ij}| > v_n\right\}. \tag{A.2}$$

For convenience, we call the network with the edge set $\widehat{\mathcal{E}}_{v_n}$ the thresholding MRC network.

Similarly, we let β_{ij} denote the regression coefficient of X_j in the nodewise GLM:

$$X_i \sim 1 + X_j + \sum_{k \in \boldsymbol{V}^{(n)} \setminus \{i, j\}} X_k. \tag{A.3}$$

Following from the total conditioning property of Bayesian networks (Pellet & Elisseeff, 2008), which shows that $X_j \in \boldsymbol{S}_i \Leftrightarrow X_i \not\perp\!\!\!\perp_p X_j | \boldsymbol{V} \setminus \left\{X_i, X_j\right\}$, we have $\beta_{ij} \neq 0 \Leftrightarrow X_j \notin \boldsymbol{S}_i$. Let $\boldsymbol{E}_{mb}^{(n)} = \left\{(i, j): \beta_{ij} \neq 0, i, j = 1, \ldots, p_n\right\}$ denote the edge set of the moral graph. We partition $\boldsymbol{E}_{mb}^{(n)}$ into two subsets: $\boldsymbol{E}_{\tilde{p}}^{(n)} = \left\{(i, j): \beta_{ij} \neq 0, \gamma_{ij} \neq 0\right\}$ and $\boldsymbol{E}_{\tilde{s}}^{(n)} = \left\{(i, j): \beta_{ij} \neq 0, \gamma_{ij} = 0\right\}$ The former set contains the parent-child links as well as the spouse links for which the two spouse variables are marginally dependent. The latter set contains the spouse links for which the two spouse variables are marginally independent but dependent conditioned on their common child.

Let $\boldsymbol{Z}_i = \left(1, X_{i,1}, \ldots, X_{i,q_n}\right)'$, where $\left\{X_{i,1}, \ldots, X_{i,q_n}\right\} \subset \left\{X_1, X_2, \ldots, X_{p_n}\right\} \setminus \left\{X_i\right\}$, and q_n is bounded by $O(n/\log(n))$. In this letter, q_n is allowed to increase with n at an appropriate rate. The regression model $X_i \sim \boldsymbol{Z}_i$ is assumed with quasi-likelihood function $-l\left(\boldsymbol{Z}_i^{\top}\boldsymbol{\xi}_i, X_i\right)$, where $\boldsymbol{\xi}_i$ denotes the vector of regression coefficients. Let

$$\boldsymbol{\xi}_i^* = \arg\min_{\boldsymbol{\xi}_i} El(\mathbf{Z}_i\boldsymbol{\xi}_i, X_i), \tag{A.4}$$

be the population parameter and

$$\hat{\boldsymbol{\xi}}_i^* = \arg\min_{\boldsymbol{\xi}_i} P_n l(\mathbf{Z}_i\boldsymbol{\xi}_i, X_i), \tag{A.5}$$

be the maximum likelihood estimator (MLE), where $P_n f(X, Y) = n^{-1}\sum_{i=1}^{n} f(X_i, Y_i)$ is the empirical measure and

$$l(X;\theta) = -[\theta X - b(\theta) - \log c(X)],$$

denotes the log-density function (in the canonical form) of the exponential family, where $b(\cdot)$ and $c(\cdot)$ denote some known functions. Assume that ξ_i^* is an interior point of a sufficiently large, compact, and convex set $\boldsymbol{F} \in \boldsymbol{R}^{q_n+1}$. For any pair $(\mathbf{Z}_i, X_i)$, the following conditions are assumed:

E1: The Fisher information,

$$I(\boldsymbol{\xi}_i) = E\left\{\left[\frac{\partial}{\partial\boldsymbol{\xi}_i} l(\mathbf{Z}_i^T\boldsymbol{\xi}_i, X_i)\right]\left[\frac{\partial}{\partial\boldsymbol{\xi}_i} l(\mathbf{Z}_i^T\boldsymbol{\xi}_i, X_i)\right]^T\right\},$$

is finite and positive at $\boldsymbol{\xi}_i = \boldsymbol{\xi}_i^*$. Moreover, $\|I(\boldsymbol{\xi}_i)\|_F = \sup_{\boldsymbol{\xi}:\, \boldsymbol{\xi}\in\boldsymbol{F}, \|z\|=1} \|I(\boldsymbol{\xi}_j)^{1/2} z\|$ exists, where $\|\cdot\|$ is the Euclidean norm.

E2: The function $l(z_i^T\boldsymbol{\xi}_i, x_i)$ satisfies the Lipschitz property with positive constant k_n,

$$|l(z_i^T\boldsymbol{\xi}_i, x_i) - l(z_i^T\boldsymbol{\xi}_i', x_i)| I_n(z_i, x_i) \le k_n |z_i^T\boldsymbol{\xi}_i - z_i^T\boldsymbol{\xi}_i'| I_n(z_i, x_i),$$

for $\boldsymbol{\xi}_i, \boldsymbol{\xi}_i' \in \boldsymbol{F}$, where $I_n(\mathbf{z}_i, x_i) = I((\mathbf{z}_i, x_i) \in \Omega_n)$ with $\Omega_n = \{(\mathbf{z}, x) : |(\mathbf{z}, x)|_\infty \le K_n\}$ for some sufficiently large, positive constants K_n, and $\|\cdot\|_\infty$ is the supremum norm. In addition, there exists a sufficiently large constant C such that with $b_n = Ck_nV_n^{-1}(q/n)^{1/2}$ and

$$\sup_{\boldsymbol{\xi}_i\in\boldsymbol{F}, \|\boldsymbol{\xi}_i - \boldsymbol{\xi}_i^*\| \le b_n} |E\left[l(\mathbf{Z}_i^T\boldsymbol{\xi}, X_i) - l(\mathbf{Z}_i^T\boldsymbol{\xi}_i^*, X_i)\right](1 - I_n(\mathbf{Z}_i, X_i))| \le o(q/n),$$

where V_n is the constant given in condition E3.

E3: The function $l(\mathbf{X}_i^T\boldsymbol{\xi}_i, X_i)$ is convex in $\boldsymbol{\xi}_i$, satisfying

$$E\left(l\left(\mathbf{Z}_i^T\boldsymbol{\xi}_i, X_i\right) - l\left(\mathbf{Z}_i^T\boldsymbol{\xi}_i^*, X_i\right)\right) \geq V_n\|\boldsymbol{\xi}_i - \boldsymbol{\xi}_i^*\|^2$$

for all $\|\boldsymbol{\xi}_i - \boldsymbol{\xi}_i^*\| \leq b_n$ and some positive constants V_n.

E4: There exist some positive constants m_0, m_1, s_0, s_1, and α, such that for sufficiently large t,

$$P\left(|X_j| > t\right) \leq \left(m_1 - s_1\right)\exp\left\{-m_0 t^{\alpha}\right\}, \quad j = 1, \ldots, p_n,$$

where α is as defined in condition B of section 3.2, and

$$E\exp\left(b\left(\tilde{\mathbf{Z}}_i^T\tilde{\boldsymbol{\xi}}_i + s_0\right) - b\left(\tilde{\mathbf{Z}}_i^T\tilde{\boldsymbol{\xi}}_i\right)\right) + E\exp\left(b\left(\tilde{\mathbf{Z}}_i^T\tilde{\boldsymbol{\xi}}_i - s_0\right) - b\left(\tilde{\mathbf{Z}}_i^T\tilde{\boldsymbol{\xi}}_i\right)\right) \leq s_1,$$

where $\tilde{\boldsymbol{\xi}}_i = \left\{\beta_{ij}: \beta_{ij} \neq 0, j \in \tilde{P}(i)\right\}, \tilde{P}(i) = \left\{j: (i,j) \in \boldsymbol{E}_{\tilde{p}}^{(n)}\right\}$,, and $\tilde{Z}_i$ contains the corresponding predictors defined in model 2.1—$\tilde{\mathbf{Z}}_i^T\tilde{\boldsymbol{\xi}}_i = \beta_{i0} + \sum_{j \in \tilde{P}(i)} X_j\beta_{ij}$.

E5: The variance $\mathrm{Var}\left(\tilde{\mathbf{Z}}_i^T\tilde{\boldsymbol{\xi}}_i\right)$ is bounded from above and below for all $i = 1, \ldots, p_n$, where $\tilde{\mathbf{Z}}_i$ and $\tilde{\xi}_i$ are as specified in condition D in section 3.2.

E6: Either $b''(\cdot)$ is bounded or $\boldsymbol{X}_M = \left(X_1, \ldots, X_{p_n}\right)^T$ follows an elliptically contoured distribution, that is,

$$\boldsymbol{X}_M = \Sigma^{1/2}R\boldsymbol{U},$$

and $|Eb'\left(\tilde{\mathbf{Z}}_i^\top\tilde{\boldsymbol{\xi}}_i\right)\left(\tilde{\mathbf{Z}}_i^\top\tilde{\boldsymbol{\xi}}_i - \beta_{i0}\right)|$ is bounded, where $\boldsymbol{U}$ is uniformly distributed on the unit sphere in p-dimensional Euclidean space, independent of the nonnegative random variable R, $\Sigma = \mathrm{Var}(\boldsymbol{X}_M)$, and $\lambda_{\max}(\Sigma) = O(n^{\tau})$for some constant $0 \leq \tau < 1 - 2\kappa$, where κ is as defined in condition B of section 3.2.

Assumption *E*6 implies that the largest eigenvalue of Σ is allowed to grow with n, but the growth rate should be restricted. Otherwise, the resulting thresholding network can be dense.

To establish the consistency of the p-screening algorithm for moral graph learning, we first note that following from the definition of $\tilde{P}(i)$ and condition D of section 3.2, there exists a constant c_2 such that

$$\min_i \min_{j \in \tilde{P}(i)} |\gamma_{ij}| \geq c_2 n^{-\kappa}. \tag{A.6}$$

Lemma 1 concerns the sure screening property of the thresholded association network, and lemma 2 concerns the neighborhood size of each node of the thresholded association

network. Their proofs can be simply modified from that of theorems 4 and 5 of Fan and Song (2010), respectively.

Lemma 1. *Suppose that the conditions A, B, and E1 to E4 hold:*

i. *If* $K_n = o(n^{(1-2\kappa)/(\alpha+2)})$, *then for any* $c_3 > 0$, *there exists a positive constant* c_4 *such that*

$$P\left(\max_{1 \le i,j \le p_n} |\hat{\gamma}_{ij} - \gamma_{ij}| \ge c_3 n^{-\kappa}\right) \le O\left(p_n^2 \exp\left(-c_4 n^{(1-2\kappa)\alpha/(\alpha+2)}\right)\right) = o(1). \quad \text{(A.7)}$$

ii. *If, in addition, condition D holds, then by taking* $\nu_n = c_5 n^{-\kappa}$ *with* $0 < c_5 \le c_2/2$, *we have*

$$P\left(\tilde{P}(i) \subseteq \hat{\mathscr{E}}_{\nu_n, i}\right) \ge 1 - O\left(p_n \exp\left(-c_4 n^{(1-2\kappa)\alpha/(\alpha+2)}\right)\right) = 1 - o(1), \quad \text{(A.8)}$$

$$P\left(\boldsymbol{E}_{\tilde{p}}^{(n)} \subseteq \hat{\mathscr{E}}_{\nu_n}\right) \ge 1 - O\left(p_n^2 \exp\left(-c_4 n^{(1-2\kappa)\alpha/(\alpha+2)}\right)\right) = 1 - o(1). \quad \text{(A.9)}$$

Lemma 2. *Suppose that conditions A, B, and E1 to E6 hold. If* $K_n = o(n^{(1-2\kappa)/(\alpha+2)})$, *then for any* ν_n $c_5 n^{-\kappa}$, *we have*

$$P\left(|\hat{\mathscr{E}}_{\nu_n, i}| \le O\left\{n^{2\kappa+\tau}\right\}\right) \ge 1 - O\left(p_n \exp\left(-c_4 n^{(1-2\kappa)\alpha/(\alpha+2)}\right)\right) = 1 - o(1). \quad \text{(A.10)}$$

Since the exact value of $2\kappa + \tau$ is unknown, we may bound the size of the neighboring set $\hat{\mathscr{E}}_{\nu_n, i}$ by $O(n/\log(n))$ in practice. However, when n is large, $n/\log(n)$ can be too large. An excessively large size of the set will adversely affect the power of the moral graph screening tests. To address this issue, we propose a multiple hypothesis test-based procedure: step a.ii for preidentification of the nonzero marginal association measure. To justify this procedure, we have the following lemmas.

Lemma 3. *Assume conditions A, B, D, and E1 to E4 hold. If* $\eta_n = \frac{1}{2} c_2 n^{-k}$, *where* c_2 *is defined in equation A.6, then*

$$P\left[\boldsymbol{E}_{\tilde{p}}^{(n)} \subset \hat{\mathscr{E}}_{\eta_n}\right] = 1 - o(1), \quad \text{as } n \to \infty.$$

Proof. Let A_{ij} denote that an error event occurs when testing the hypotheses $H_0 : \gamma_{ij} = 0$ versus $H : \gamma_{ij} \neq 0$ for variables X_i and X_j. Let A_{ij}^I and A_{ij}^{II} denote the false-positive and false-negative errors, respectively. Then $A_{ij} = A_{ij}^I \cup A_{ij}^{II}$, where

$$\begin{cases} \text{False-positive error } A_{ij}^{I}: |\hat{\gamma}_{ij}| > \frac{c_2}{2} n^{-\kappa} & \text{and } \gamma_{ij} = 0, \\ \text{False-negative error } A_{ij}^{II}: |\hat{\gamma}_{ij}| \leq \frac{c_2}{2} n^{-\kappa} & \text{and } \gamma_{ij} \neq 0 \end{cases}. \tag{A.11}$$

By equation A.6, $\min_{ij} |\gamma_{ij}| \geq c_2 n^{-\kappa}$ for the links in $\boldsymbol{E}_{\tilde{p}}^{(n)}$. Therefore, by lemma 1.i,

$$P[\text{Missing a link of } \boldsymbol{E}_{\tilde{p}}^{(n)} \text{ in } \widehat{\mathscr{E}}_{\eta_n}]$$

$$\leq P\left(\max_{1 \leq i,j \leq p_n} |\hat{\gamma}_{ij} - \gamma_{ij}| \geq c_2/2 n^{-\kappa} \right) \leq o(1). \tag{A.12}$$

□

Therefore, based on lemmas 1, 2, and 3, we propose to restrict the size of the set $\boldsymbol{A}_i$ (in algorithm 1) for each node to be

$$\min\left\{ |\widehat{\mathscr{E}}_{\eta_n, i}|, \frac{n}{c_{n1} \log(n)} \right\}, \tag{A.13}$$

where c_{n1} is a small constant—for example, $c_n = 1$, 2, or 3. The value of η_n can be determined through a simultaneous test for the hypotheses $H_0 : \gamma_{ij} = 0 \leftrightarrow H_1 : \gamma_{ij} \neq 0$, $1 \leq i \leq j \leq p_n$, at a significance level of α_1.

Lemma 4 concerns the convergence of the MLE of the regression coefficients for which all the true predictors have been included. The lemma is a restatement of theorem 1 of Fan and Song (2010):

Lemma 4. *Assume conditions A, B, and E1 to E3 hold. If* $K_n = o(n^{(1-2\kappa)/(\alpha+2)})$, *then for any constant* $c_7 > 0$, *there exists a constant* $c_8 > 0$ *such that*

$$P\left(\max_{1 \leq i \leq p_n} |\hat{\boldsymbol{\xi}}_i - \boldsymbol{\xi}_i^*| \geq c_7 n^{-\kappa} \right) \leq O\left(p_n \exp\left(-c_8 n^{(1-2\kappa)\alpha/(\alpha+2)} \right) \right) = o(1), \tag{A.14}$$

where $\boldsymbol{\xi}_i^*$ is defined in equation A.4 and $\hat{\boldsymbol{\xi}}_i$ is the MLE of $\boldsymbol{\xi}_i^*$.

Recall that if the Markov blanket $\boldsymbol{S}_i$ (of node X_i) is contained in $\boldsymbol{Z}_i$, then $\xi_{i,i_k}^* = \beta_{ij}$ for $j \in \boldsymbol{S}_i$ and $X_j = X_{i,i_k}$, and $\xi_{i,i_k}^* = 0$ otherwise.

Let $\hat{\beta}_{ij}$ denote the estimate of β_{ij} obtained in step c of algorithm 1. Let ζ_n denote the threshold value of $\hat{\beta}_{ij}$, and let $\widehat{\boldsymbol{E}}_{mb,\zeta_n}^{(n)}$ denote the network obtained through thresholding $\hat{\beta}_{ij}$. That is, we define

$$\hat{\boldsymbol{E}}^{(n)}_{mb,\zeta_n} = \left\{(i,j): |\hat{\beta}_{ij}| > \zeta_n\right\}.$$

To establish the consistency of $\hat{\boldsymbol{E}}_{mb,\zeta_n}$, we first note that as implied by condition D and the total conditioning property, there exists a constant c_6 such that the true regression coefficients $\{\beta_{ij}\}$defined in equation A.3 satisfy

$$\min_i \min_{j \in \boldsymbol{S}_i} |\beta_{ij}| \geq c_6 n^{-\kappa}, \tag{A.15}$$

where κ is as defined in condition (*B*). Let $\widehat{\mathscr{E}}_*$ denote the edge set of a marginal association network for which each node has a degree of $O(n/\log(n))$, adjacent to $O(n/\log(n))$ highest associated nodes. It follows from lemmas 1 and 2 that

$$P\left[\boldsymbol{E}^{(n)}_{\tilde{p}} \subseteq \widehat{\mathscr{E}}_*\right] \geq 1 - O\left(p_n^2 \exp\left(-c_4 n^{(1-2\kappa)\alpha/(\alpha+2)}\right)\right) = 1 - o(1). \tag{A.16}$$

Let $\tilde{\boldsymbol{B}} = \cup_{i=1}^{p_n} \boldsymbol{B}_i$, where $\boldsymbol{B}_i$ is defined in step b of the *p*-screening algorithm. We have $\boldsymbol{E}^{(n)}_{\tilde{s}} \subset \tilde{\boldsymbol{B}}$. Further, by lemma 3, we have $\boldsymbol{E}^{(n)}_{mb} = \boldsymbol{E}^{(n)}_{\tilde{p}} \cup \boldsymbol{E}^{(n)}_{\tilde{s}} \subset (\widehat{\mathscr{E}}_* \cap \widehat{\mathscr{E}}_{\eta_n}) \cup \tilde{\boldsymbol{B}}$.

Lemma 5 establishes the consistency of $\hat{\boldsymbol{E}}^{(n)}_{mb,\zeta_n}$ as an estimate of $\boldsymbol{E}^{(n)}_{mb}$ conditioned on $\boldsymbol{E}^{(n)}_{mb} \subseteq \left(\widehat{\mathscr{E}}_* \cap \widehat{\mathscr{E}}_{\eta_n}\right) \cup \tilde{\boldsymbol{B}}$. Its proof is based on equation A.15 and follows closely the proof of lemma 3, and is thus omitted here.

Lemma 5. *Assume that the conditions A, B, C, D, and E1 to E6 hold and that* $\boldsymbol{E}^{(n)}_{mb} \subseteq \left(\widehat{\mathscr{E}}_* \cap \widehat{\mathscr{E}}_{\eta_n}\right) \cup \tilde{\boldsymbol{B}}$ *is true. Let* $\zeta_n = \frac{1}{2} c_6 n^{-\kappa}$. *If* $K_n = o(n^{(1-2\kappa)/(\alpha+2)})$. *Then*

$$P\left[\hat{\boldsymbol{E}}^{(n)}_{mb,\zeta_n} = \boldsymbol{E}^{(n)}_{mb} \middle| \boldsymbol{E}^{(n)}_{mb} \subseteq \left(\widehat{\mathscr{E}}_* \cap \widehat{\mathscr{E}}_{\eta_n}\right) \cup \tilde{\boldsymbol{B}}\right] = 1 - o(1), \quad \text{as } n \to \infty.$$

As a summary for the above results, we have the following theorem, which establishes the consistency of $\hat{\boldsymbol{E}}^{(n)}_{mb,\zeta_n}$ as an estimate of the adjacency matrix of the moral graph $\boldsymbol{E}^{(n)}_{mb}$.

Theorem 2. *Consider a Bayesian network with distribution* $P^{(n)}$ *defined in equation 2.1 for mixed GLM variables. Assume the conditions A, B, C, D, and E1 to E6 hold. If* $K_n = o(n^{(1-2\kappa)/(\alpha+2)})$, *then*

$$P\left[\hat{\boldsymbol{E}}^{(n)}_{mb,\zeta_n} = \boldsymbol{E}^{(n)}_{mb}\right] \geq 1 - o(1), \quad \text{as } n \to \infty.$$

Proof. By invoking lemma 3, equation A.16, and lemma 5, we have

$$P\left[\hat{\boldsymbol{E}}^{(n)}_{mb,\zeta_n} = \boldsymbol{E}^{(n)}_{mb}\right] \geq P\left[\hat{\boldsymbol{E}}^{(n)}_{mb,\zeta_n} = \boldsymbol{E}^{(n)}_{mb} | \boldsymbol{E}^{(n)}_{mb} \subseteq \left(\widehat{\mathscr{E}}_* \cap \widehat{\mathscr{E}}_{\eta_n}\right) \cup \tilde{\boldsymbol{B}}\right] P\left[\boldsymbol{E}^{(n)}_{mb} \subseteq \left(\widehat{\mathscr{E}}_* \cap \widehat{\mathscr{E}}_{\eta_n}\right) \cup \tilde{\boldsymbol{B}}\right] \geq [1 - o(1)][1 - o(1) + 1 - o(1) - 1] = 1 - o(1).$$

□

References

Aliferis C, Statnikov A, Tsamardinos I, Mani S, & Koutsoukos XD (2010). Local causal and Markov blanket induction for causal discovery and feature selection for classification, part I: Algorithms and empirical evaluation. Journal of Machine Learning Research, 11, 171–234.

Benjamini Y, Krieger A, & Yekutieli D (2006). Adaptive linear step-up procedures that control the false discovery rate. Biometrika, 93, 491–507.

Bertheau P, Lehmann-Che J, Varna M, Dumay A, Poirot B, Porcher R, … de The H (2013). p53 in breast cancer subtypes and new insights into response to chemotherapy. Breast. (suppl. 2), S27–S29. [PubMed: 24074787]

Cheng J, Li T, Levina E, & Zhu J (2013). High-dimensional mixed graphical models. arXiv: 1304.2810v3.

Chickering D (1996). Learning Bayesian networks is NP-complete In Fisher D & Lenz H-J (Eds.), Learning from data: Artificial intelligence and statistics, 5 (pp. 121–130). New York: Springer-Verlag.

Chickering D (2002). Optimal structure identification with greedy search. Journal of Machine Learning Research, 3, 507–554.

Cizkova M, Vacher S, Meseure D, Trassard M, Susini A, Mlcuchova D, … Bieche I (2013). Pik3r1 underexpression is an independent prognostic marker in breast cancer. BMC Cancer, 13, 545. [PubMed: 24229379]

Colomboi D, Maathuis M, Kalisch M, & Richardson T (2012). Learning high-dimensional directed acyclic graphs with latent and selection variables. Annals of Statistics, 40, 294–321.

Cooper G, & Herskovits E (1992). A Bayesian method for the induction of probabilistic networks from data. Mach. Learn, 9, 309–347.

Cui R, Groot P, & Heskes T (2016). Copula PC algorithm for causal discovery from mixed data In Frasconi P, Landwehr N, Manco G, & Vreeken J (Eds.), Machine learning and knowledge discovery in databases (pp. 377–392). Cham: Springer.

Fan J, Liu H, & Ning Y (2017). High dimensional semiparametric latent graphical model for mixed data. Journal of the Royal Statistical Society, B, 79(2), 405–421.

Fan J, & Lv J (2008). Sure independence screening for ultrahigh dimensional feature space. Journal of the Royal Statistical Society, B, 70(5), 849–911.

Fan J, & Song R (2010). Sure independence screening in generalized linear models with NP-dimensionality. Annals of Statistics, 38(6), 3567–3604.

Friedman J, Hastie T, & Tibshirani R (2008). Sparse inverse covariance estimation with the graphical Lasso. Biostatistics, 9(3), 432–441. [PubMed: 18079126]

Friedman N, Pe'er D, & Nachman I (1999). Learning Bayesian network structure from massive datasets: The "sparse candidate" algorithm In Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence (pp. 206–221). San Mateo, CA: Morgan Kaufmann.

Gao T, & Ji Q (2017a). Efficient Markov blanket discovery and its application. IEEE Trans. on Cybernetics, 47, 1169–1179.

Gao T, & Ji Q (2017b). Efficient score-based Markov blanket discovery. International Journal of Approximate Reasoning, 80, 277–293.

Ha M, Sun W, & Xie J (2016). Penpc: Atwo-step approach to estimate the skeletons of high-dimensional directed acyclic graphs. Biometrics, 72, 146–155. [PubMed: 26406114]

Harris N, & Drton M (2013). PC algorithm for nonparanormal graphical models. Journal of Machine Learning Research, 14, 3365–3383.

Heckerman D, Geiger D, & Chickering D (1995). Learning Bayesian networks: The combination of knowledge and statistical data. Mach. Learn, 20, 197–243.

Herskovits E, & Cooper G (1990). Kutató: An entropy-driven system for the construction of probabilistic expert systems from datasets In Bonissone P (Ed.), Proceedings of the 6th Conference on Uncertainty in Artificial Intelligence (pp. 54–62). New York: Elsevier.

Jensen F, & Nielsen T (2007). Bayesian networks and decision graphs. New York: Springer.

Jia B, Xu S, Xiao G, Lamba V, & Liang F (2017). Learning gene regulatory networks from next generation sequencing data. Biometrics, 73(4), 1221–1230. [PubMed: 28294287]

Kalisch M, & Bühlmann P (2007). Estimating high-dimensional directed acyclic graphs with the PC-algorithm. Journal of Machine Learning Research, 8, 613–636.

Kjaerulff U, & Madsen A (2010). Bayesian networks and influence diagrams. New York: Springer.

Lam W, & Bacchus F (1994). Learning Bayesian belief networks: An approach based on the MDL principle. Comput. Intell, 10, 269–293.

Lee J, & Hastie T (2015). Learning the structure of mixed graphical models. Journal of Computational and Graphical Statistics, 24, 230–253. [PubMed: 26085782]

Liang F, Song Q, & Qiu P (2015). An equivalent measure of partial correlation coefficients for high dimensional gaussian graphical models. Journal of the American Statistical Association, 110, 1248–1265.

Liang F, & Zhang J (2008). Estimating the false discovery rate using the stochastic approximation algorithm. Biometrika, 95, 961–977.

Liu H, Lafferty J, & Wasserman L (2009). The nonparanormal: Semiparametric estimation of high dimensional undirected graphs. Journal of Machine Learning Research, 10, 2295–2328.

Margaritis D (2003). Learning Bayesian network structure models from data PhD diss, Carnegie-Mellon University.

Margaritis D, & Thrun S (2000). Bayesian network induction via local neighborhoods In Solla S, Leen T, & Müller K-R (Eds.), Advances in neural information processing systems, 12 (pp. 505–511). Cambridge, MA: MIT Press.

McGeachie M, Chang H-H, & Weiss S (2014). Ccbayesnets: Conditional gaussian Bayesian network learning and inference with mixed discrete and continuous data. PLoS Computational Biology, 10(6), e1003676. [PubMed: 24922310]

Meinshausen N, & Bühlmann P (2006). High-dimensional graphs and variable selection with the Lasso. Annals of Statistics, 34, 1436–1462.

Nandy P, Hauser A, & Maathuis M (2016). High-dimensional consistency in score-based and hybrid structure learning. https://arxiv.org/abs/1507.02608?context-math.

Pearl J (1988). Probabilistic reasoning in intelligent systems: Networks of Plausible Inference. San Mateo, CA: Morgan Kaufmann.

Pellet J-P, & Elisseeff A (2008). Using Markov blankets for causal structure learning. Journal of Machine Learning Research, 9, 1295–1342.

Scutari M, & Denis J-B (2015). Bayesian networks with examples in R. Boca Raton, FL: CRC Press.

Silwal-Pandit L, Vollan H, Chin S, Rueda O, McKinney S, Osako T, … Langerod A (2014). Tp53 mutation spectrum in breast cancer is subtype specific and has distinct prognostic relevance. Clin. Cancer Res, 20(13), 3569–3580. [PubMed: 24803582]

Spirtes P (2010). Introduction to causal inference. Journal of Machine Learning Research, 11, 1643–1662.

Spirtes P, Glymour C, & Scheines R (2000). Causation, prediction, and search. Cambridge, MA: MIT Press.

Tsamardinos I, Aliferis C, & Statnikov A (2003). Algorithms for large scale Markov blanket discovery In Proceedings of the 16th International Florida Artificial Intelligence Research Society Conference (pp. 376–381). Palo Alto, CA: AAAI Press.

Tsamardinos I, Brown L, & Aliferis C (2006). The max-min hill-climbing Bayesian network structure learning algorithm. Machine Learning, 65(1), 31–78.

Verdugo R, Zeller T, Rotival M, Wild P, Münzel T, Lackner K, … Tiret L (2013). Graphical modeling of gene expression in monocytes suggests molecular mechanisms explaining increased atherosclerosis in smokers. PLoS One, 8(1), e50888. [PubMed: 23372645]

Verma T, & Pearl J (1991). Equivalence and synthesis of causal models. Uncertainty in Artificial Intelligence, 6, 255–268.

Verma T, & Pearl J (1992). An algorithm for deciding if a set of observed independencies has a causal explanation In Proceedings of the 8th Conference on Uncertainty in Artificial Intelligence (pp. 323–330). Amsterdam: Elsevier.

Virmani A, Rathi A, Sathyanarayana U, Padar A, Huang C-X, Cunnigham H, … Gazdar A (1998). Aberrant methylation of the adenomatous polyposis coli (APC) gene promoter 1a in breast and lung carcinomas. Clin. Cancer Res, 7, 3569–3580.

Wang C, Li Q, Liu F, Chen X, Liu B, Nesa EU, … Cheng Y (2016). Notch2 as a promising prognostic biomarker for oesophageal squamous cell carcinoma. Sci. Rep, 6, 25722. [PubMed: 27158037]

Yang E, Ravikumar P, Allen GI, Baker Y, Wan Y-W, & Liu Z (2014). A general framework for mixed graphical models. arXiv:1411.0288v1.

Yaramakala S, & Margaritis D (2005). Speculative Markov blanket discovery for optimal feature selection In Proceedings of the 5th IEEE International Conference on Data Mining (pp. 809–812). Washington, DC: IEEE Computer Society.

Yuan M, & Lin Y (2007). Model selection and estimation in the gaussian graphical model. Biometrika, 95(1), 19–35.

Zhang Q, Burdette J, & Wang J (2014). Integrative network analysis of TCGA data for ovarian cancer. BMC Syst. Biol, 8, 1338. [PubMed: 25551281]

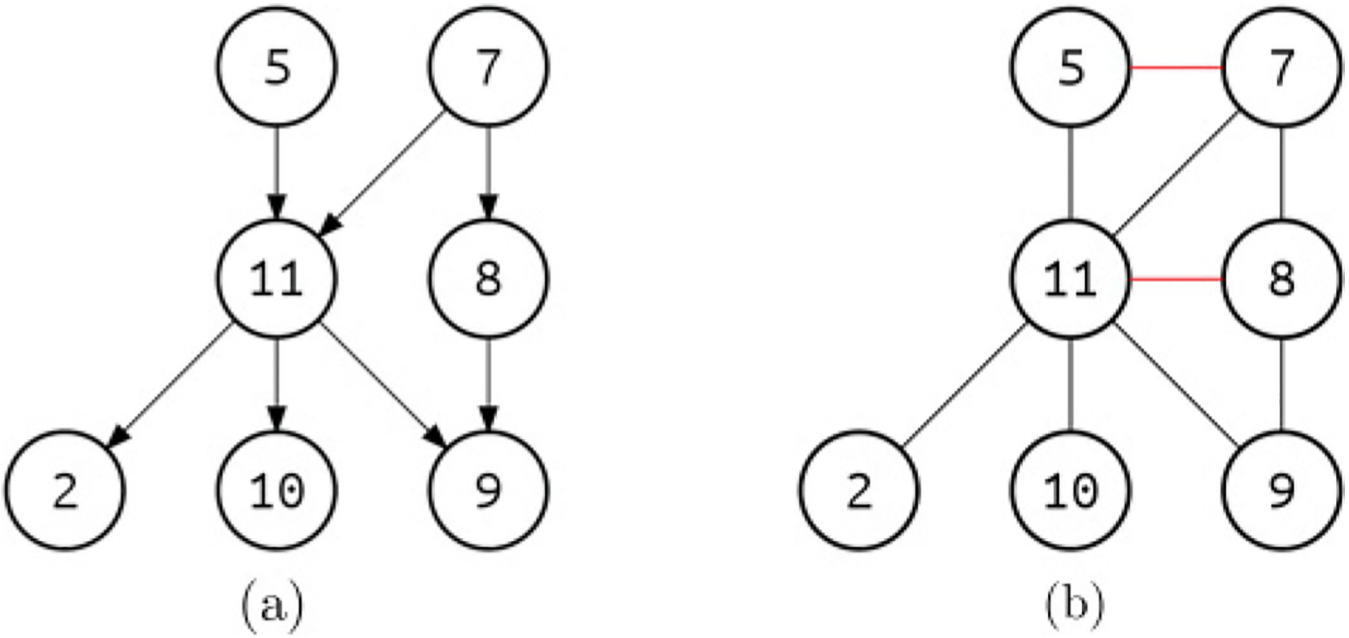
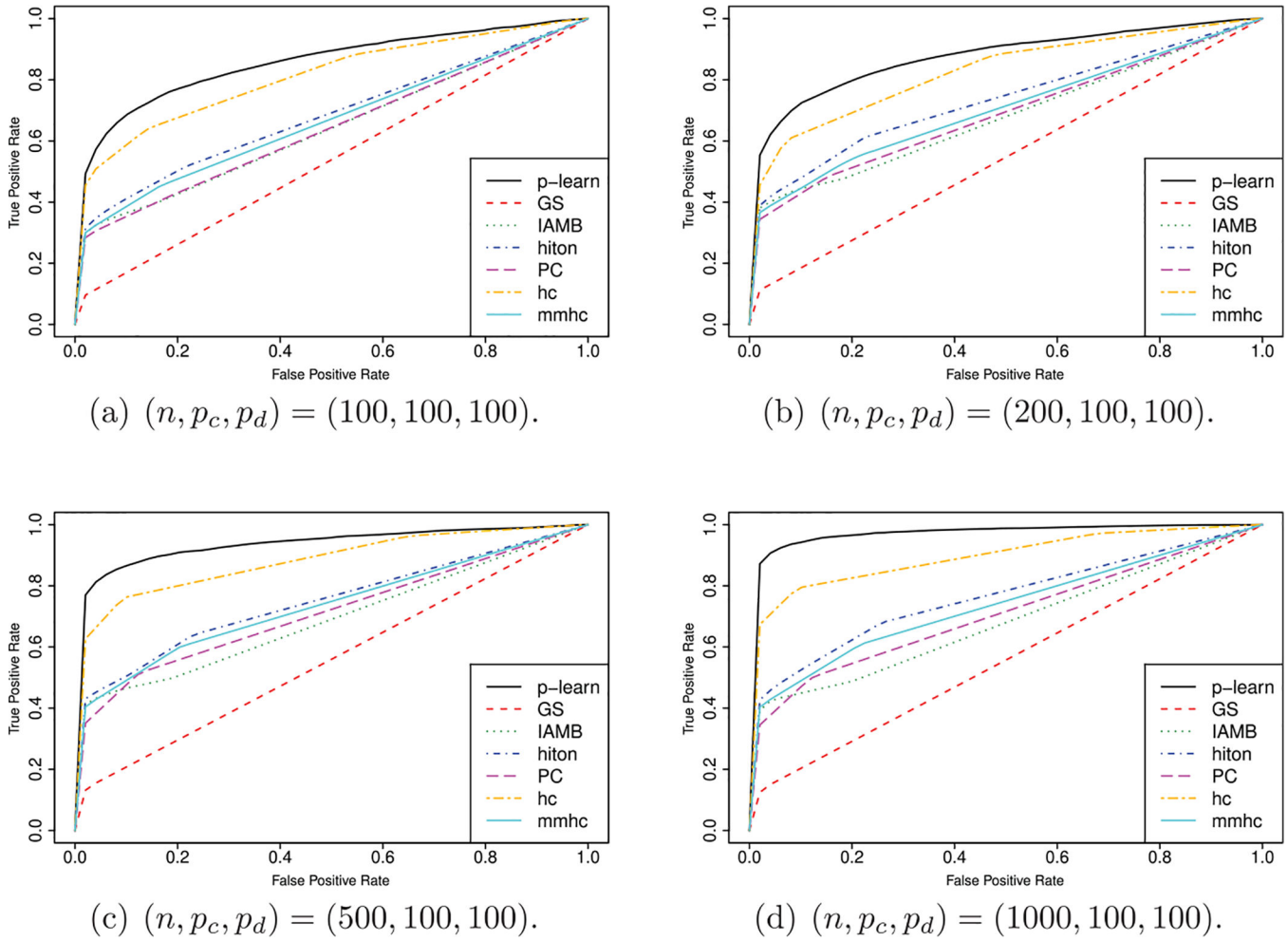


Figure 1:
(a) A Bayesian network. (b) The moral graph corresponding to panel a, where edges 5–7 and 8–11 are added.



(a) $(n, p_c, p_d) = (100, 100, 100)$. (b) $(n, p_c, p_d) = (200, 100, 100)$.

(c) $(n, p_c, p_d) = (500, 100, 100)$. (d) $(n, p_c, p_d) = (1000, 100, 100)$.

Figure 2:

Averaged ROC curves produced by different algorithms for learning the moral graph with an AR(2) structure: p-learn denotes the proposed algorithm; GS denotes the grow-shrink algorithm; IAMB denotes the incremental association Markov blanket algorithm; hiton denotes the semi-interleaved HITON-PC algorithm; PC denotes the PC algorithm; hc denotes the hill-climbing algorithm; and mmhc denotes the max-min hill-climbing algorithm.

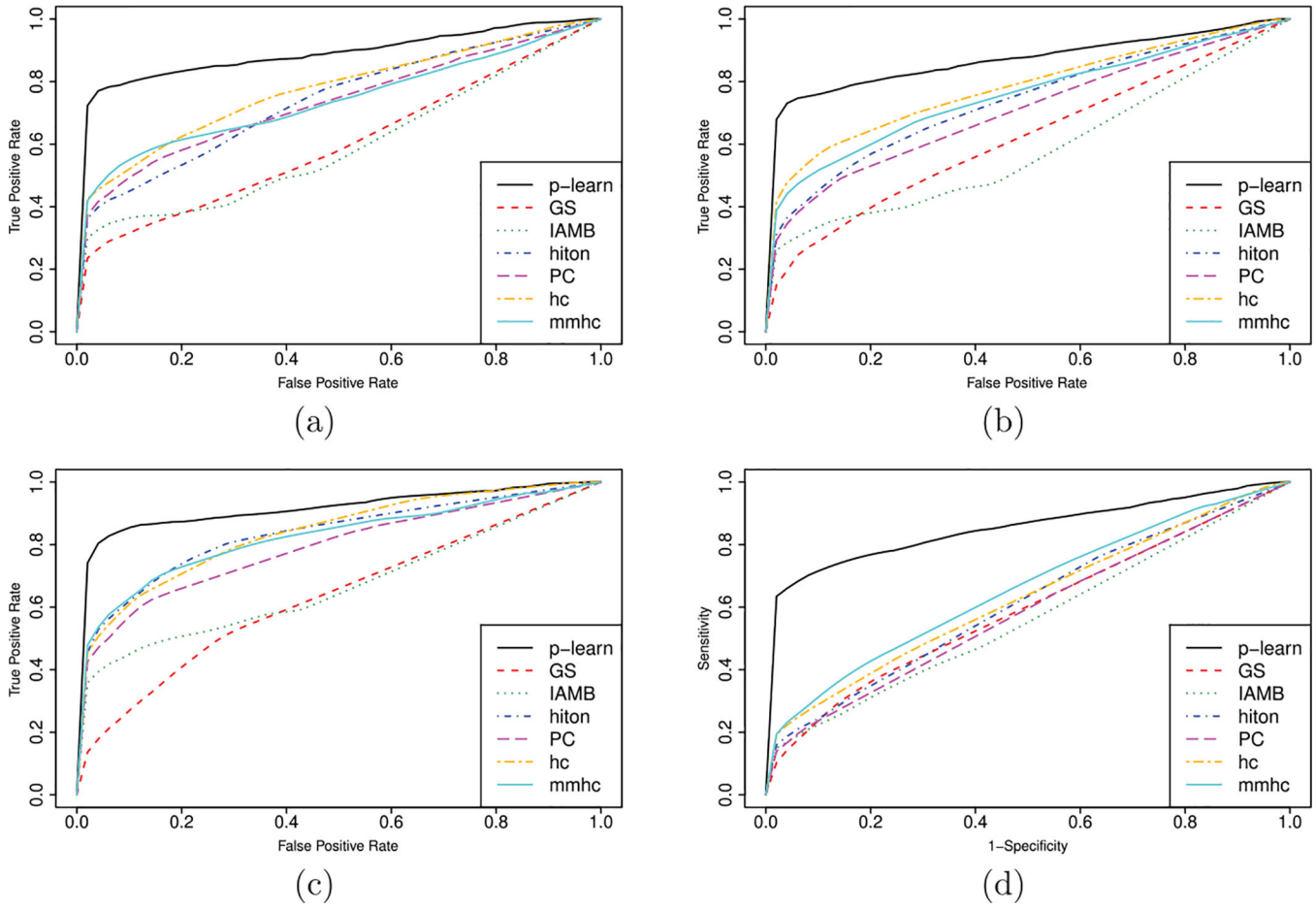


Figure 3:

Averaged ROC curves produced by different algorithms for learning moral graphs with different dependency structures. (a) Alarm structure with (n, p_c, p_d) = (3000, 19, 18). (b) Barley structure with (n, p_c, p_d) = (3000, 24, 24). (c) Ecoli structure with (n, p_c, p_d) = (3000, 23, 23). (d) Magic structure with (n, p_c, p_d) = (3000, 32, 32), where p-learn denotes the proposed algorithm; GS denotes the grow-shrink algorithm; IAMB denotes the incremental association Markov blanket algorithm; hiton denotes the semi-interleaved HITON-PC algorithm; PC denotes the PC algorithm; hc denotes the hill-climbing algorithm; and mmhc denotes the max-min hill-climbing algorithm.

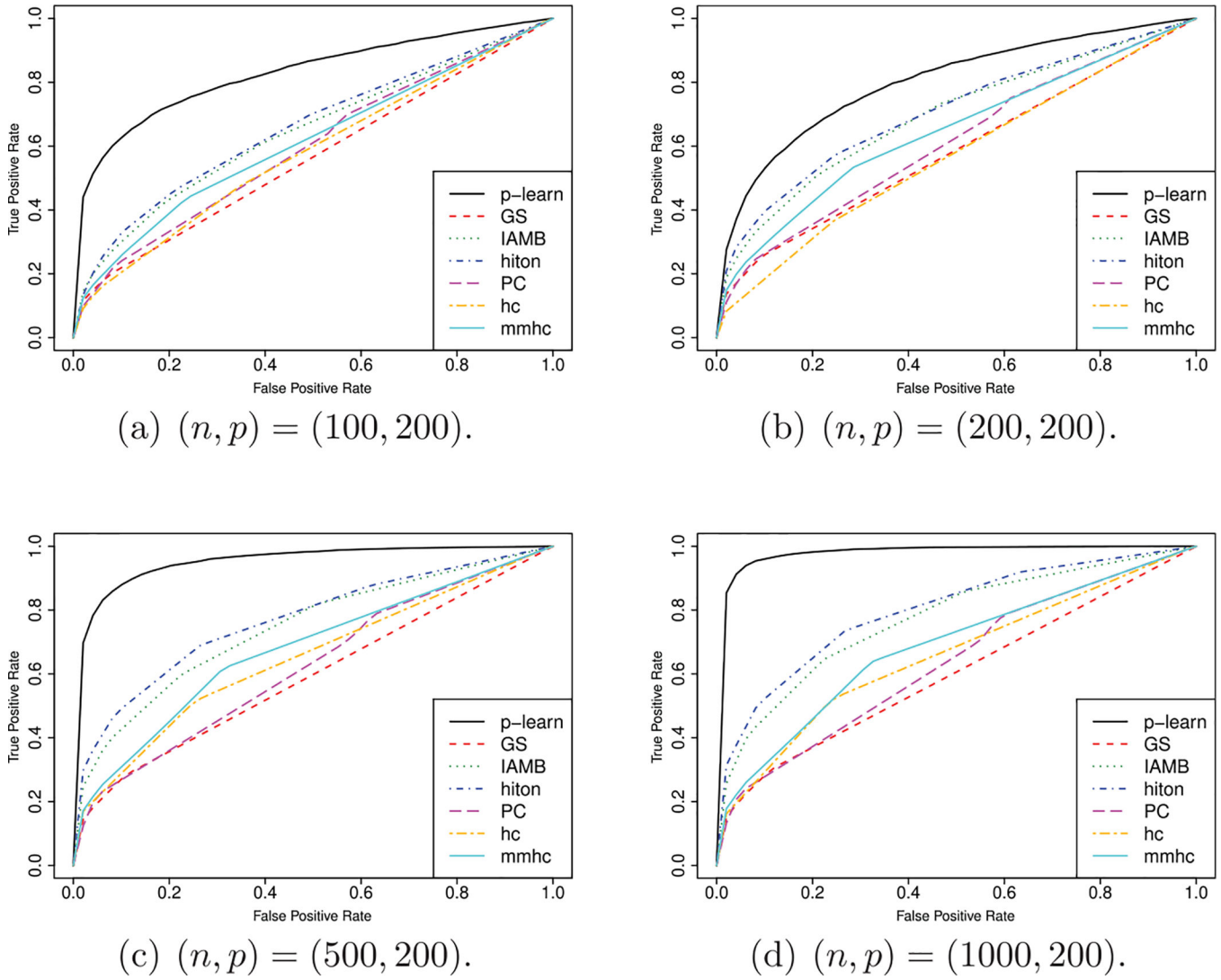


Figure 4:

Averaged ROC curves produced by different algorithms for learning the moral graph with an AR(2) structure and binary variables only. *p*-learn denotes the proposed algorithm; GS denotes the grow-shrink algorithm; IAMB denotes the incremental association Markov blanket algorithm; hiton denotes the semi-Interleaved HITON-PC algorithm; PC denotes the PC algorithm; hc denotes the hill-climbing algorithm; and mmhc denotes the max-min hill-climbing algorithm.

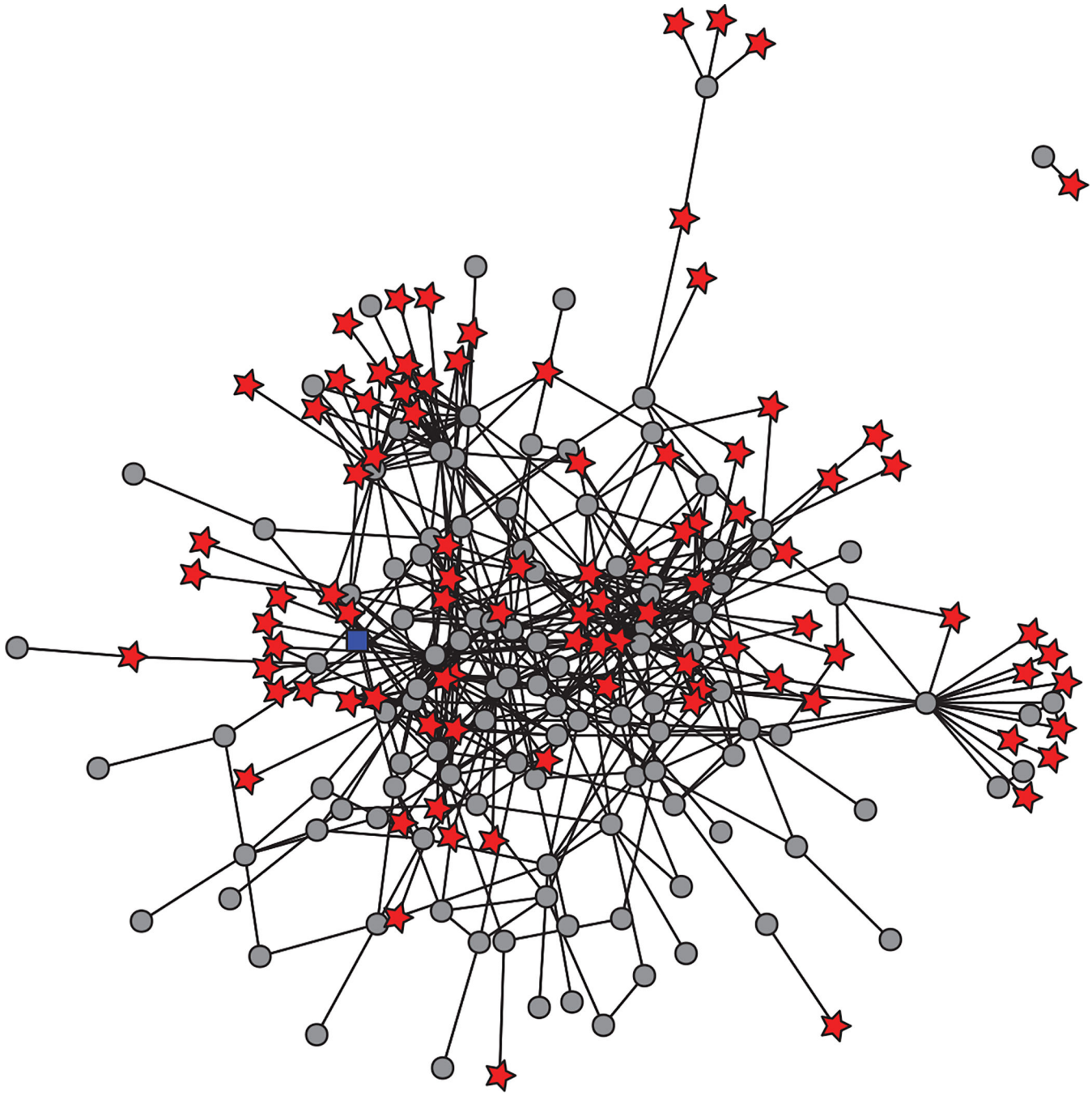


Figure 5:
The moral network produced by algorithm 1 for breast cancer, where the circle nodes denote genes, the square nodes denote mutations, and the star nodes denote DNA methylations.

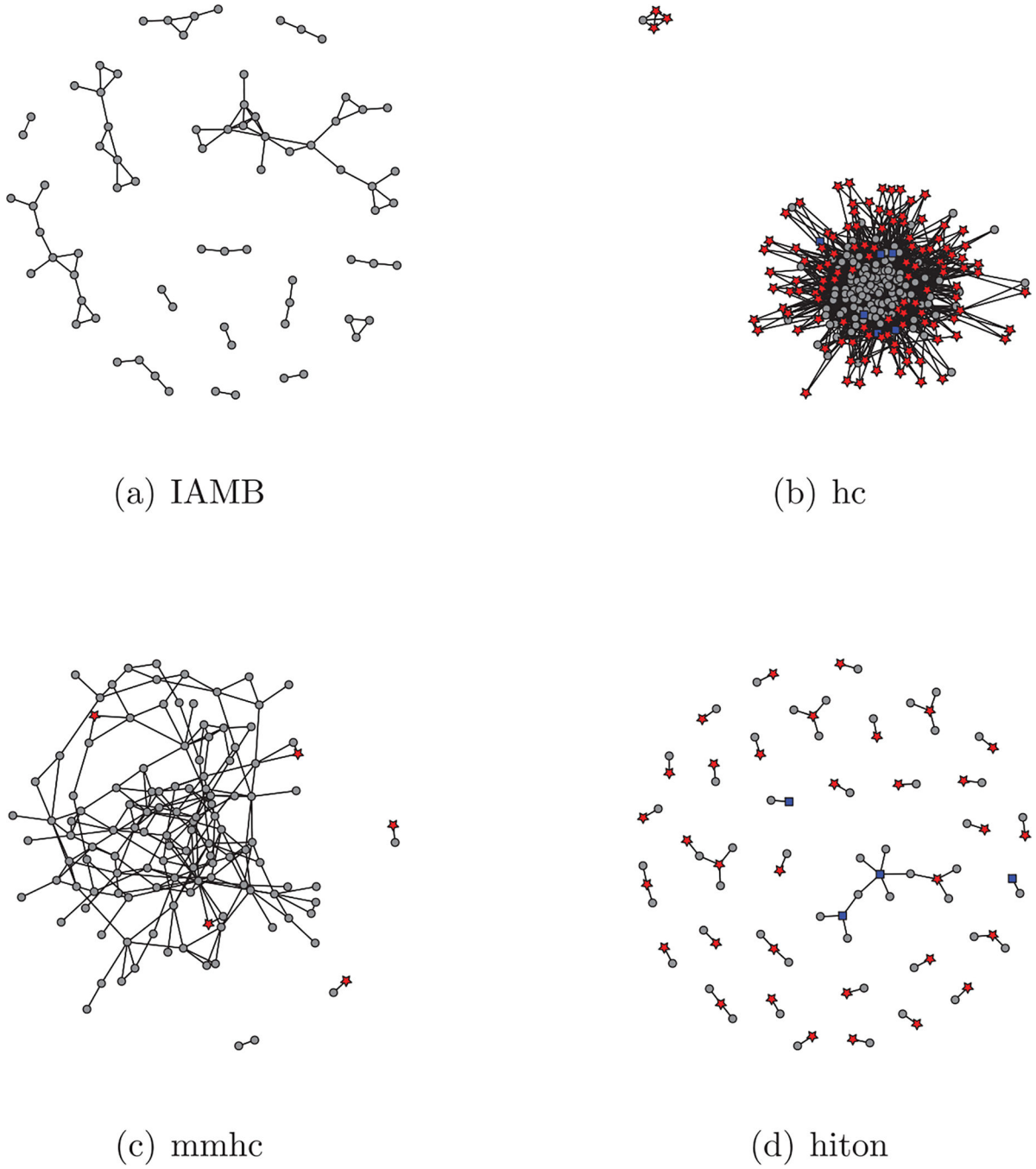


Figure 6:
The genomics networks produced by different algorithms for the breast cancer data set.

Table 1:

Conditional Independence Represented by Markov Networks.

Conditional Independence	**Marginal Independence**	
	$X \perp_p Y$	$X \not\perp_p Y$
$X \perp_p Y \mid Z$	Ⓧ Ⓨ − Ⓩ	Ⓧ − Ⓩ − Ⓨ
$X \not\perp_p Y \mid Z$	Nonrepresentable	Ⓧ − Ⓨ − Ⓩ

Table 2:

Averaged Areas under the ROC Curves Produced by Different Algorithms for Recovering the Moral Graph with an AR(2) Structure.

n	**GS**	**IAMB**	**hiton**	**PC**	**hc**	**mmhc**	***p*-learn**
100	0.538	0.642	0.683	0.642	0.813	0.667	**0.857**
	(0.001)	(0.001)	(0.002)	(0.003)	(0.005)	(0.003)	(0.005)
200	0.547	0.681	0.735	0.690	0.830	0.706	**0.877**
	(0.001)	(0.001)	(0.002)	(0.003)	(0.003)	(0.002)	(0.002)
500	0.559	0.693	0.751	0.715	0.885	0.737	**0.940**
	(0.002)	(0.001)	(0.003)	(0.004)	(0.003)	(0.004)	(0.002)
1000	0.557	0.683	0.765	0.709	0.899	0.737	**0.977**
	(0.002)	(0.001)	(0.002)	(0.003)	(0.002)	(0.004)	(0.001)

Notes: The numbers of continuous and binary variables (p_C, p_D) were fixed to (100, 100). The numbers in parentheses represents the standard deviation of the areas averaged over 10 data sets. The bold numbers indicate that the proposed algorithm 1 significantly outperforms the competitors.

Table 3:

Averaged Areas under the ROC Curves Produced by Algorithm 1 with Different Values of a_1.

$n \backslash a_1$	0.05	0.1	0.15	0.2	0.25
100	0.863	0.857	0.854	0.836	0.826
	(0.005)	(0.005)	(0.004)	(0.005)	(0.005)
200	0.873	0.877	0.872	0.869	0.868
	(0.003)	(0.002)	(0.003)	(0.003)	(0.002)
500	0.941	0.940	0.940	0.939	0.939
	(0.001)	(0.001)	(0.002)	(0.003)	(0.005)
1000	0.977	0.977	0.977	0.977	0.977
	(0.001)	(0.001)	(0.001)	(0.001)	(0.001)

Notes: The data were generated as in section 4.1 with an AR(2) structure and $p_C = p_d = 100$. The numbers in parentheses represent the standard deviation of the areas averaged over 10 data sets.

Table 4:

Averaged Areas under the ROC Curves Produced by Different Algorithms for Learning Moral Graphs with Different Dependency Structures.

Structure	GS	IAMB	hiton	PC	hc	mmhc	*p*-learn
Alarm	0.600	0.594	0.740	0.735	0.774	0.740	**0.894**
	(0.008)	(0.006)	(0.005)	(0.005)	(0.006)	(0.005)	(0.014)
Barley	0.619	0.581	0.738	0.707	0.782	0.755	**0.873**
	(0.005)	(0.003)	(0.003)	(0.007)	(0.003)	(0.005)	(0.012)
Ecoli	0.633	0.668	0.836	0.792	0.839	0.826	**0.919**
	(0.006)	(0.002)	(0.003)	(0.008)	(0.005)	(0.003)	(0.006)
magic	0.591	0.561	0.612	0.585	0.628	0.657	**0.857**
	(0.005)	(0.002)	(0.003)	(0.006)	(0.004)	(0.004)	(0.003)

Notes: The numbers in the parentheses represent the standard deviation of the areas averaged over 10 data sets. The bold numbers indicate that the proposed algorithm 1 significantly outperforms the competitors.

Table 5:

Averaged Areas under the ROC Curves Produced by Different Algorithms for Learning the Moral Graph with an AR(2) Structure and Binary Variables Only.

n	**GS**	**IAMB**	**hiton**	**PC**	**hc**	**mmhc**	***p*-learn**
100	0.564	0.646	0.659	0.596	0.580	0.613	**0.829**
	(0.002)	(0.005)	(0.008)	(0.006)	(0.005)	(0.006)	(0.006)
200	0.585	0.695	0.707	0.612	0.569	0.646	**0.820**
	(0.002)	(0.005)	(0.007)	(0.004)	(0.004)	(0.004)	(0.004)
500	0.593	0.743	0.769	0.623	0.650	0.676	**0.954**
	(0.003)	(0.004)	(0.004)	(0.004)	(0.006)	(0.005)	(0.002)
1000	0.601	0.772	0.799	0.636	0.657	0.684	**0.983**
	(0.002)	(0.003)	(0.003)	(0.005)	(0.004)	(0.005)	(0.001)

Notes: The numbers in the parentheses represent the standard deviation of the areas averaged over 10 data sets. The bold numbers indicate that the proposed algorithm 1 significantly outperforms the competitors.

Table 6:

Computational Time (in CPU Minutes Recorded on a Xeon Gold 6126 CPU@2.60 GHz Computer) of Different Algorithms for Recovering the Moral Graph with an AR(2) Structure.

	GS	**IAMB**	**hiton**	**PC**	**hc**	**mmhc**	***p*-learn**
(n, p)							
(100,100)	0.9	1.1	3.0	7.1	1.4	2.9	4.1
(141,200)	3.2	3.9	34.2	82.2	6.5	32.3	15.9
(200,400)	18.6	18.6	499.7	1304.1	49.7	520.5	54.4
(264,700)	50.0	46.8	5320	12239.3	253.0	4785.3	152.5
(300,900)	102.4	97.1	14560.8	33389.9	624.2	13545.3	254.8
Regression							
R^2	0.9962	0.9969	0.9987	0.9993	0.9954	0.9991	0.9997
$\hat{v}$	2.157	2.017	3.886	3.872	2.786	3.866	1.861
$\mathrm{sd}(\hat{v})$	0.076	0.065	0.080	0.059	0.110	0.068	0.019
Test							
p-value	7.9×10^{-5}	0.011	0	0	5.8×10^{-17}	0	—

Table 7:

Top Five Hub Genes, Mutations, and Methylations Identified by Algorithm 1.

Rank	Gene	Links	Mutation	Links	Methylation	Links
1	PIK3R1	13	TP53	4	cg01240931	15
2	NOTCH2	11	-	-	cg01830294	13
3	CTNNB1	11	-	-	cg04658354	12
4	FGF1	8	-	-	cg19088651	12
5	PIK3CD	8	-	-	cg15791248	10

Note: “Links” denotes the number of edges connected to the corresponding node in the network.