entropy

MDPI

Article

Nonparametric Causal Structure Learning in High Dimensions

Shubhadeep Chakraborty and Ali Shojaie *

Department of Biostatistics, University of Washington, Seattle, WA 98195, USA; deep20@uw.edu
* Correspondence: ashojaie@uw.edu

Abstract: The PC and FCI algorithms are popular constraint-based methods for learning the structure of directed acyclic graphs (DAGs) in the absence and presence of latent and selection variables, respectively. These algorithms (and their order-independent variants, PC-stable and FCI-stable) have been shown to be consistent for learning sparse high-dimensional DAGs based on partial correlations. However, inferring conditional independences from partial correlations is valid if the data are jointly Gaussian or generated from a linear structural equation model—an assumption that may be violated in many applications. To broaden the scope of high-dimensional causal structure learning, we propose nonparametric variants of the PC-stable and FCI-stable algorithms that employ the conditional distance covariance (CdCov) to test for conditional independence relationships. As the key theoretical contribution, we prove that the high-dimensional consistency of the PC-stable and FCI-stable algorithms carry over to general distributions over DAGs when we implement CdCov-based nonparametric tests for conditional independence. Numerical studies demonstrate that our proposed algorithms perform nearly as good as the PC-stable and FCI-stable for Gaussian distributions, and offer advantages in non-Gaussian graphical models.

Keywords: causal structure learning; consistency; FCI algorithm; high dimensionality; nonparametric testing; PC algorithm

Citation: Chakraborty, S.; Shojaie, A. Nonparametric Causal Structure Learning in High Dimensions. *Entropy* **2022**, *24*, 351. https://doi.org/10.3390/e24030351

Academic Editors: S. Ejaz Ahmed and Farouk Nathoo

Received: 20 January 2022
Accepted: 25 February 2022
Published: 28 February 2022

1. Introduction

Directed acyclic graphs (DAGs) are commonly used to represent causal relationships among random variables [1–3]. The PC algorithm [3] is the most popular constraint-based method for learning DAGs from observational data under the assumption of causal sufficiency, i.e., when there are no unmeasured common causes and no selection variables. It first estimates the skeleton of a DAG by recursively performing a sequence of conditional independence tests, and then uses the information from the conditional independence relations to partially orient the edges, resulting in a completed partially directed acyclic graph (CPDAG). In Section 2, we provide a review of these and other notions commonly used in the graphical modeling literature that are relevant to our work. In addition, we refer to estimating the CPDAG as structure learning of the underlying DAG throughout the rest of the paper.

Observational studies often involve latent and selection variables, which complicate the causal structure learning problem. Ignoring such unmeasured variables can make the causal inference based on the PC algorithm erroneous; see, e.g., Section 1.2 in [4] for some illustrations. The Fast Causal Inference (FCI) algorithm and its variants [3–6] utilize similar strategies as the PC algorithm to learn the DAG structure in the presence of latent and selection variables.

Both PC and FCI algorithms adopt a hierarchical search strategy—they recursively perform conditional independence tests given subsets of increasingly larger cardinalities in some appropriate search pool. The PC algorithm is usually order-dependent, in the sense that its output depends on the order in which pairs of adjacent vertices and subsets of their adjacency sets are considered. The FCI algorithm suffers from a similar limitation. To overcome this limitation, Ref. [7] proposed two variants of the PC and FCI algorithms, namely the PC-stable and FCI-stable algorithms that resolve the order dependence at different stages of the algorithms.

In general, testing for conditional independence is a problem of central importance in the causal structure learning. The literature on the PC and FCI algorithms predominantly uses partial correlations to infer conditional independence relations. It is well-known that the characterization of conditional independence by partial correlations, or, in other words, equivalence between conditional independence and zero partial correlations only holds for multivariate normal random variables. Therefore, the high-dimensional consistency results for the PC and FCI algorithms [4,8] are limited to Gaussian graphical models, where the nodes correspond to random variables with a joint Gaussian distribution. Although the Gaussian graphical model is the standard parametric model for continuous data, it may not hold in many real data applications. Although this limitation can be somewhat relaxed by considering linear structural equation models (SEMs) with general noise distributions [9], linear SEMs and joint Gaussianity are essentially equivalent [10]. Moreover, neither approach is appropriate when the observations are categorical, discrete, or are supported on a subset of the real line. In Section 4.3, for example, we present a real application where all the observed variables are categorical, and therefore far from being Gaussian. As an improvement, ref. [11] used rank-based partial correlations to test for conditional independence relations, showing that the high-dimensional consistency of the PC algorithm holds for a broader class of Gaussian copula models. Some nonparametric versions of the PC algorithm have been also proposed in the literature via kernel-based tests for conditional independence [12,13]; however, they lack theoretical justifications of the correctness of the algorithms, and are not studied in high dimensions.

This work aims to broaden the applicability of the PC-stable and FCI-stable algorithms to general distributions by employing a nonparametric test for conditional independence relationships. To this end, we utilize recent developments on dependence metrics that quantify nonlinear and non-monotone dependence between multivariate random variables. More specifically, our work builds on the idea of distance covariance (dCov) proposed by [14] and its extension to conditional distance covariance (CdCov) by [15] as a nonparametric measure of nonlinear and non-monotone conditional independence between two random vectors of arbitrary dimensions given a third. Utilizing this flexibility, we use the conditional distance covariance (CdCov) to test for conditional independence relationships in the sample versions of the PC-stable and FCI-stable algorithms. The resulting algorithms—which, for distinction, are termed *nonPC* and *nonFCI*—facilitate causal structure learning from general distributions over DAGs and are shown to be consistent in sparse high-dimensional settings. We establish the consistency of the proposed algorithms using some moment and tail conditions on the variables, without requiring strict distributional assumptions. To our knowledge, the proposed generalizations of PC/PC-stable or the FCI/FCI-stable algorithms provide the first general nonparametric framework for causal structure learning with theoretical guarantees in high dimensions.

The rest of the paper is organized as follows: In Section 2, we review the relevant background, including preliminaries on graphical modeling (Section 2.1), an outline of the PC-stable and FCI-stable algorithms (Section 2.2) and a brief overview of dCov and CdCov (Section 2.3). The nonparametric version of the PC-stable algorithm is presented in Section 3.1. As a key contribution of the paper, we establish that the algorithm consistently estimates the skeleton and the equivalence class of the underlying sparse high-dimensional DAG in a general nonparametric framework. We then present the nonparametric version of the FCI-stable algorithm in Section 3.2 and establish its consistency in sparse high-dimensional settings. As the FCI involves the adjacency search of the PC algorithm, any improvement on the PC/PC-stable directly carries over to the FCI/FCI-stable as well. In Section 4, we compare the performances of our algorithms with the PC-stable and FCI-stable using both simulated datasets (involving both Gaussian and non-Gaussian examples), as well as a real dataset. These numerical studies clearly demonstrate that nonPC and nonFCI algorithms are comparable with PC-stable and FCI-stable for Gaussian data and offer improvements for non-Gaussian data.

2. Background

2.1. Preliminaries on Graphical Modeling

We start with introducing some necessary terminologies and background information. Our notations and terminologies follow standard conventions in graphical modeling (see, e.g., [3]). A graph $\mathcal{G} = (V, E)$ consists of a vertex set $V = \{1, \ldots, p\}$ and an edge set $E \subseteq V \times V$. In a graphical model, the vertices or nodes are associated with random variables X_a for $1 \leq a \leq p$. Throughout, we index the nodes by the corresponding random variables. We also allow the edge set E of the graph $\mathcal{G}$ to contain (a subset of) the following six types of edges: $\rightarrow$ (*directed*), $\leftrightarrow$ (*bidirected*), $-$ (*undirected*), $\circ\!\!-\!\!\circ$ (*nondirected*), $\circ\!\!-$ (*partially undirected*) and $\circ\!\!\rightarrow$ (*partially directed*). The endpoints of an edge are called marks, which can be tails, arrowheads or circles. A "$\circ$" at the end of an edge indicates it is not known whether an arrowhead should occur at that place. We use the symbol '$\star$' to denote an arbitrary edge mark; for example, the symbol $\star\!\!\rightarrow$ represents an edge of the type $\rightarrow$, $\leftrightarrow$ or $\circ\!\!\rightarrow$ in the graph. A *mixed graph* is a graph containing directed, bidirected and undirected edges. A graph containing only directed edges $(\rightarrow)$ is called a *directed graph*, one containing only undirected edges $(-)$ is called an *undirected graph*, and one containing directed and undirected edges is called a *partially directed graph*.

The *adjacency set* of a vertex X_a in the graph $\mathcal{G} = (V, E)$, denoted $\text{adj}(\mathcal{G}, X_a)$, is the set of all vertices in V that are adjacent to X_a, or, in other words, are connected to X_a by an edge. The *degree* of a vertex X_a, $|\text{adj}(\mathcal{G}, X_a)|$, is defined as the number of vertices adjacent to it. A graph is *complete* if all pairs of vertices in the graph are adjacent. A vertex $X_b \in \text{adj}(\mathcal{G}, X_a)$ is called a *parent* of X_a if $X_b \rightarrow X_a$, a *child* of X_a if $X_a \rightarrow X_b$ and a *neighbor* of X_a if $X_a - X_b$. The *skeleton* of the graph $\mathcal{G}$ is the undirected graph obtained by replacing all the edges of $\mathcal{G}$ by undirected edges, in other words, ignoring all the edge orientations. Three vertices $\langle X_a, X_b, X_c \rangle$ are called an *unshielded triple* if X_a and X_b are adjacent, X_b and X_c are adjacent, but X_a and X_c are not adjacent. A *path* is a sequence of distinct adjacent vertices. A node X_a is an *ancestor* of its *descendent* X_b, if $\mathcal{G}$ contains a directed path $X_a \rightarrow \cdots \rightarrow X_b$. A non-endpoint vertex X_a on a path is called a collider on the path if both the edges preceding and succeeding it have an arrowhead at X_a, or, in other words, the path contains $\star\!\!\rightarrow X_a \leftarrow\!\!\star$. An unshielded triple $\langle X_a, X_b, X_c \rangle$ is called a *v-structure* if X_b is a collider on the path $\langle X_a, X_b, X_c \rangle$.

A *cycle* occurs in a graph when there is a path from X_a to X_b, and X_a and X_b are adjacent. A directed path from X_a to X_b forms a *directed cycle* together with the edge $X_b \rightarrow X_a$, and it forms an *almost directed cycle* together with the edge $X_b \leftrightarrow X_a$. Three vertices that form a cycle are called a *triangle*. A *directed acyclic graph* (DAG) is a directed graph that does not contain any cycle. A DAG entails conditional independence relationships via a graphical criterion called *d-separation* (Section 1.2.3 in [16]). Two vertices X_a and X_b that are not adjacent in a DAG $\mathcal{G}$ are d-separated in $\mathcal{G}$ by a subset $X_S \subseteq V \backslash \{X_a, X_b\}$. A probability distribution P on $\mathbb{R}^p$ is said to be *faithful* with respect to the DAG $\mathcal{G}$ if the conditional independence relationships in P can be inferred from $\mathcal{G}$ using d-separation and vice versa; in other words, $X_a \perp\!\!\!\perp X_b | X_S$ if and only if X_a and X_b are d-separated in $\mathcal{G}$ by X_S.

A graph that is both (partially) directed and acyclic is called a *partially directed acyclic graph (PDAG)*. DAGs that encode the same set of conditional independence relations form a Markov equivalence class [17]. Two DAGs belong to the same Markov equivalence class if and only if they have the same skeleton and the same v-structures. A Markov equivalence class of DAGs can be uniquely represented by a *completed partially directed acyclic graph (CPDAG)*, which is a PDAG that satisfies the following: (i) $X_a \rightarrow X_b$ in the CPDAG if $X_a \rightarrow X_b$ in every DAG in the Markov equivalence class, and (ii) $X_a - X_b$ in the CPDAG if the Markov equivalence class contains a DAG in which $X_a \rightarrow X_b$ as well as a DAG in which $X_a \leftarrow X_b$.

2.2. The PC-Stable and FCI-Stable Algorithms

In this section, we provide an outline of the PC/PC-stable and FCI/FCI-stable algorithms. Estimation of the CPDAG by the PC algorithm involves two steps: (1) estimation of the skeleton and separating sets (also called the adjacency search step); and (2) partial orientation of edges; see Algorithms 1 and 2 in [8] for details.

Intuitively, the PC algorithm works as follows. In the first step (the adjacency search step), the algorithm starts with a complete undirected graph Then, for conditioning sets of increasing cardinality, $k = 0, 1, \ldots$, the algorithm removed an edge $X_a - X_b$ if X_a and X_b are conditionally independent given a subset S of size k chosen among the current neighbors of nodes a and b. This process continues up to the order $q - 1$, where q is the maximum degree of the underlying DAG. By searching over the neighboring nodes, the algorithm is adaptive and can efficiently infer sparse high-dimensional DAGs, where the sparsity is characterized by the maximum node degree, q.

In the presence of latent and selection variables, one needs a generalization of an DAG, called a *maximal ancestral graph* (MAG). A mixed graph is called an *ancestral graph* if it contains no directed or almost directed cycles and no subgraph of the type $X_a - X_b \leftarrow \star X_c$. DAGs form a subset of ancestral graphs. A MAG is an ancestral graph in which every missing edge corresponds to a conditional independence relationship via the m-separation criterion [18], a generalization of the notion of d-separation. Multiple MAGs may represent the same set of conditional independence relations. Such MAGs form a Markov equivalence class which can be represented by a *partial ancestral graph* (PAG) [19]; see [18] for additional details.

Under the faithfulness assumption, the Markov equivalence class of a DAG with latent and selection variables can be learned using the FCI algorithm (e.g., Algorithm 3.1 in [4]), which is a modification of the PC algorithm. The FCI algorithm first employs the adjacency search of the PC algorithm, and then performs additional conditional independence queries because of the presence of latent variables followed by partial orientation of the edges, resulting in an estimated PAG. The FCI algorithm adopts the same hierarchical search strategy as the PC algorithm: It starts with a complete undirected graph and recursively removes edges via conditional independence queries given subsets of increasingly larger cardinalities in some appropriate search pool.

The PC algorithm is usually order-dependent, in the sense that its output depends on the order in which pairs of adjacent vertices and subsets of their adjacency sets are considered. The FCI algorithm suffers from a similar limitation, as it shares the adjacency search step of the PC algorithm as its first step. To overcome this limitation, ref. [7] proposed variants of the PC and FCI algorithms, namely the PC-stable and FCI-stable algorithms that resolve the order dependence at different stages of the algorithms. The basic difference between the PC algorithm and the PC-stable algorithm is that, in the adjacency search step, the latter computes and stores the adjacency sets of all the variables after each new cardinality, $k = 0, 1, \ldots$, of the conditioning sets. These stored adjacency sets are then used to search for conditioning sets of this given size k. As a consequence, the removal of an edge no longer affects which conditional independence relations need to be checked for other pairs of variables at this given size of the conditioning sets.

We would refer the reader to Appendix A, where we provide in full detail the pseudocodes of the *oracle* versions of the PC-stable and FCI-stable algorithms. In the *oracle* versions of the algorithms, it is assumed that perfect knowledge is available about all the necessary conditional independence relations. As such, conditional independence relations are not estimated from data. Of course, this perfect knowledge is not available in practice. *Sample* versions of the PC-stable and FCI-stable algorithms can be obtained by replacing the conditional independence queries by a suitable test for conditional independence at some pre-specified level. For example, if the variables are jointly Gaussian, one can test for zero partial correlations (see, e.g., [8]). The next subsection is devoted to discussions on nonparametric tests for independence and conditional independence.

2.3. Distance Covariance and Conditional Distance Covariance

We start by describing the notation used throughout the paper. We denote by $\|\cdot\|_p$ the Euclidean norm of $\mathbb{R}^p$ and use $\|\cdot\|$ when the dimension is clear from the context. We use $X \perp\!\!\!\perp Y$ to denote the independence of X and Y and use $\mathbb{E}_U$ to denote expectation with respect to the probability distribution of the random variable U. For any set S, we denote its cardinality by $|S|$.

We use the usual asymptotic notation, 'O' and 'o', as well as their probabilistic counterparts, O_p and o_p, which denote stochastic boundedness and convergence in probability,

respectively. For two sequences of real numbers $\{a_n\}_{n=1}^{\infty}$ and $\{b_n\}_{n=1}^{\infty}$, $a_n \asymp b_n$ if and only if $a_n/b_n = O(1)$ and $b_n/a_n = O(1)$ as $n \to \infty$. We use the symbol "$a \lesssim b$" to indicate that $a \leq C\, b$ for some constant $C > 0$. For a matrix $A = (a_{kl})_{k,l=1}^{n} \in \mathbb{R}^{n \times n}$, we denote its determinant by $|A|$ and define its $\mathcal{U}$-centered version $\tilde{A} = (\tilde{a}_{kl})_{k,l=1}^{n}$ as

$$\tilde{a}_{kl} = \begin{cases} a_{kl} - \dfrac{1}{n-2}\sum_{j=1}^{n} a_{kj} - \dfrac{1}{n-2}\sum_{i=1}^{n} a_{il} + \dfrac{1}{(n-1)(n-2)}\sum_{i,j=1}^{n} a_{ij}, & k \neq l, \\ 0, & k = l, \end{cases} \tag{1}$$

for $k, l = 1, \ldots, n$. We denote the indicator function of any set A by $\mathbf{1}(A)$. Finally, we denote the integer part of $a \in \mathbb{R}$ by $\lfloor a \rfloor$.

Ref. [14], in their seminal paper, introduced the notion of distance covariance (dCov, henceforth) to quantify nonlinear and non-monotone dependence between two random vectors of arbitrary dimensions. Consider two random vectors $X \in \mathbb{R}^p$ and $Y \in \mathbb{R}^q$ with $\mathbb{E}\|X\|_p < \infty$ and $\mathbb{E}\|Y\|_q < \infty$. The distance covariance between X and Y is defined as the positive square root of

$$\mathrm{dCov}^2(X, Y) = \frac{1}{c_p c_q} \int_{\mathbb{R}^{p+q}} \frac{|f_{X,Y}(t,s) - f_X(t) f_Y(s)|^2}{\|t\|_p^{1+p}\, \|s\|_q^{1+q}} dt ds$$

where f_X, f_Y and $f_{X,Y}$ are the individual and joint characteristic functions of X and Y, respectively, and $c_p = \pi^{(1+p)/2} / \Gamma((1+p)/2)$ is a constant with $\Gamma(\cdot)$ being the complete gamma function.

The key feature of dCov is that it completely characterizes the independence between two random vectors, or in other words $\mathrm{dCov}(X, Y) = 0$ if and only if $X \perp\!\!\!\perp Y$. According to Remark 3 in [14], dCov can be equivalently expressed as

$$\begin{aligned} \mathrm{dCov}^2(X, Y) \;&= \mathbb{E}\,\|X - X'\|_p \|Y - Y'\|_q \;+\; \mathbb{E}\,\|X - X'\|_p\, \mathbb{E}\,\|Y - Y'\|_q \\ &\quad - 2\,\mathbb{E}\,\|X - X'\|_p \|Y - Y''\|_q\,. \end{aligned}$$

This alternate expression comes handy in constructing V or U-statistic type estimators for the quantity. For an observed random sample $(X_i, Y_i)_{i=1}^{n}$ from the joint distribution of X and Y, define the distance matrices $d^X = \left(d_{ij}^X\right)_{i,j=1}^{n}$ and $d^Y = \left(d_{ij}^Y\right)_{i,j=1}^{n} \in \mathbb{R}^{n \times n}$, where $d_{ij}^X := \|X_i - X_j\|_p$ and $d_{ij}^Y := \|Y_i - Y_j\|_q$. Following the $\mathcal{U}$-centering idea in [20], an unbiased U-statistic type estimator of $\mathrm{dCov}^2(X, Y)$ can be expressed as

$$\mathrm{dCov}_n^2(X, Y) \;:=\; (\tilde{d}^X \cdot \tilde{d}^Y) \;:=\; \frac{1}{n(n-3)} \sum_{i \neq j} \tilde{d}_{ij}^X \tilde{d}_{ij}^Y\,, \tag{2}$$

where $\tilde{d}^X = (\tilde{d}_{ij}^X)_{i,j=1}^{n}$ and $\tilde{d}^Y = (\tilde{d}_{ij}^Y)_{i,j=1}^{n}$ are the $\mathcal{U}$-centered versions of the matrices d^X and d^Y, respectively, as defined in (1).

Ref. [15] generalized the notion of dCov and introduced the conditional distance covariance (CdCov, henceforth) as a measure of conditional dependence between two random vectors of arbitrary dimensions given a third. CdCov essentially replaces the characteristic functions used in the definition of dCov by conditional characteristic functions. Consider a third random vector $Z \in \mathbb{R}^r$ with $\mathbb{E}(\|X\|_p + \|Y\|_q \mid Z) < \infty$. Denote by $f_{X,Y|Z}$ the conditional joint characteristic function of X and Y given Z, and by $f_{X|Z}$ and $f_{Y|Z}$ the conditional marginal characteristic functions of X and Y given Z, respectively. Then, CdCov between X and Y given Z is defined as the positive square root of

$$\mathrm{CdCov}^2(X, Y|Z) = \frac{1}{c_p c_q} \int_{\mathbb{R}^{p+q}} \frac{|f_{X,Y|Z}(t,s) - f_{X|Z}(t) f_{Y|Z}(s)|^2}{\|t\|_p^{1+p}\, \|s\|_q^{1+q}} dt ds.$$

The key feature of CdCov is that $\mathrm{CdCov}\,(X, Y|Z) = 0$ almost surely if and only if $X \perp\!\!\!\perp Y|Z$, which is quite straightforward to see from the definition.

Similar to dCov, an equivalent alternative expression can be established for CdCov that avoids complicated integrations involving conditional characteristic functions. Let $\{W_i = (X_i, Y_i, Z_i)\}_{i=1}^n$ be an i.i.d. sample from the joint distribution of $W := (X, Y, Z)$. Define $d_{ijkl} := \left(d_{ij}^X + d_{kl}^X - d_{ik}^X - d_{jl}^X\right)\left(d_{ij}^Y + d_{kl}^Y - d_{ik}^Y - d_{jl}^Y\right)$, which is not symmetric with respect to $\{i, j, k, l\}$, and therefore necessitates defining the following symmetric form: $d_{ijkl}^S := d_{ijkl} + d_{ijlk} + d_{ilkj}$. Lemma 1 in [15] establishes an equivalent representation of $\mathrm{CdCov}^2(X, Y|Z = z)$ as

$$\mathrm{CdCov}^2(X, Y|Z = z) \;=\; \frac{1}{12}\,\mathbb{E}\left[d_{1234}^S \,|\, Z_1 = z, Z_2 = z, Z_3 = z, Z_4 = z\right]. \tag{3}$$

Remark 1. *In a recent work, [21] explore the connection between conditional independence measures induced by distances on a metric space and reproducing kernels associated with a reproducing kernel Hilbert space (RKHS). They generalize CdCov to arbitrary metric spaces of negative type—termed generalized CdCov (gCdCov)—and develop a kernel-based measure of conditional independence, namely the Hilbert–Schmidt conditional independence criterion (HSCIC). Theorem 1 in their paper establishes an equivalence between gCdCov and HSCIC, or, in other words, between distance and kernel-based measures of conditional independence.*

For $w \in \mathbb{R}^r$, let $K_H(w) := |H|^{-1} K(H^{-1}w)$ be a kernel function, where H is the diagonal matrix $\mathrm{diag}(h, \ldots, h)$ determined by a bandwidth parameter h. K_H is typically considered to be the Gaussian kernel $K_H(w) = (2\pi)^{-\frac{r}{2}} |H|^{-1} \exp\left(-\frac{1}{2} w^T H^{-2} w\right)$, where $w \in \mathbb{R}^r$.

Let $K_{iu} := K_H(Z_i - Z_u) = |H|^{-1} K(H^{-1}(Z_i - Z_u))$ and $K_i(Z) := K_H(Z - Z_i)$ for $1 \le i, u \le n$. Then, by virtue of the equivalent representation of CdCov in (3), a V-statistic type estimator of $\mathrm{CdCov}^2(X, Y|Z)$ can be constructed as

$$\mathrm{CdCov}_n^2(X, Y|Z) \;:=\; \sum_{i,j,k,l} \frac{K_i(Z)\, K_j(Z)\, K_k(Z)\, K_l(Z)}{12\left(\sum_{i=1}^n K_i(Z)\right)^4}\, d_{ijkl}^S\,. \tag{4}$$

Under certain regularity conditions, Theorem 4 in [15] shows that, conditioned on Z, $\mathrm{CdCov}_n^2(X, Y|Z) \xrightarrow{P} \mathrm{CdCov}^2(X, Y|Z)$ as $n \to \infty$.

3. Methodology and Theory

3.1. The Nonparametric PC Algorithm in High Dimensions

To obtain a measure of conditional independence between X and Y given Z that is free of Z, we define

$$\rho_0^*\,(X, Y|Z) \;:=\; \mathbb{E}\left[\mathrm{CdCov}_n^2(X, Y|Z)\right]. \tag{5}$$

Clearly, $\rho_0^*\,(X, Y|Z) = 0$ if and only if $X \perp\!\!\!\perp Y \,|\, Z$. Consider a plug-in estimate of $\rho_0^*\,(X, Y|Z)$ as

$$\begin{aligned} \widehat{\rho}^{\,*}(X, Y|Z) \;&:=\; \frac{1}{n}\sum_{u=1}^n \mathrm{CdCov}_n^2(X, Y|Z_u) \;=\; \frac{1}{n}\sum_{u=1}^n \Delta_{i,j,k,l;u} \\ \text{where} \qquad \Delta_{i,j,k,l;u} \;&:=\; \sum_{i,j,k,l} \frac{K_{iu}\, K_{ju}\, K_{ku}\, K_{lu}}{12\left(\sum_{i=1}^n K_{iu}\right)^4}\, d_{ijkl}^S\,. \end{aligned} \tag{6}$$

We reject $H_0 : X \perp\!\!\!\perp Y|Z$ vs $H_A : X \not\perp\!\!\!\perp Y|Z$ at level $\alpha \in (0, 1)$ if $\widehat{\rho}^{\,*}(X, Y|Z) > \xi_\alpha$, for a suitably chosen threshold ξ_α. In Appendix A, we present a local bootstrap procedure for choosing ξ_α in practice, which is also used in our numerical studies. Henceforth, we will often denote $\rho_0^*\,(X, Y|Z)$ and $\widehat{\rho}^{\,*}(X, Y|Z)$ simply by ρ_0^* and $\widehat{\rho}^{\,*}$ respectively for notational simplicity, whenever there is no confusion.

In view of the complete characterization of conditional independence by ρ_0^*, we propose testing for conditional independence relations nonparametrically in the sample version of the PC-stable algorithm based on ρ_0^*, rather than partial correlations. We coin the resulting algorithm the 'nonPC' algorithm, to emphasize that it is a nonparametric generalization of parametric PC-stable algorithms.

The *oracle version* of the first step of nonPC, or the skeleton estimation step, is exactly the same as that of the PC-stable algorithm (Algorithm A1 in Appendix A). The second step, which extends the skeleton estimated in the first step to a CPDAG (Algorithm A2 in Appendix A), is comprised of some purely deterministic rules for edge orientations, and is exactly the same for both the nonPC and PC-stable as well. The only difference lies in the implementation of the tests for conditional independence relationships in the *sample versions* of the first step. Specifically, we replace all the conditional independence queries in the first step by tests based on $\rho_0^*(X, Y|Z)$. At some pre-specified significance level α, we infer that $X_a \perp\!\!\!\perp X_b \,|\, X_S$ when $\widehat{\rho}^{\,*}(X_a, X_b|X_S) \leq \xi_{n,\alpha}$, where $a, b \in V$ and $S \subseteq V$, $|S| \neq \phi$. When $|S| = \phi$, $\widehat{\rho}^{\,*}(X_a, X_b|X_S) = \mathrm{dCov}_n^2(X_a, X_b)$ and $\rho_0^*(X, Y|Z) = \mathrm{dCov}^2(X, Y)$. The critical value $\xi_{n,\alpha}$ in this case is obtained by a bootstrap procedure (see, e.g., Section 4 in [22] with $d = 2$).

Given that the equivalence between conditional independence and zero partial correlations only holds for multivariate normal random variables, our generalization broadens the scope of applicability of causal structure learning by the PC/PC-stable algorithm to general distributions over DAGs. This nonparametric approach is thus a natural extension of Gaussian and Gaussian copula models. It enables capturing nonlinear and non-monotone conditional dependence relationships among the variables, which partial correlations fail to detect.

Next, we establish theoretical guarantees on the correctness of the nonPC algorithm in learning the true underlying causal structure in sparse high-dimensional settings. Our consistency results only require mild moment and tail conditions on the set of variables, without making any strict distributional assumptions. Denote by m_p the maximum cardinality of the conditioning sets considered in the adjacency search step of the PC-stable algorithm. Clearly, $m_p \leq q$, where $q := \max_{1 \leq a \leq p} |\mathrm{adj}(\mathcal{G}, a)|$ is the maximum degree of the DAG $\mathcal{G}$. For a fixed pair of nodes $a, b \in V$, the conditioning sets considered in the adjacency search step are elements of $J_{a,b}^{m_p} := \{S \subseteq V \backslash \{a, b\} : |S| \leq m_p\}$.

We first establish a concentration inequality that gives the rate at which the absolute difference of $\rho_0^*(X_a, X_b|X_S)$ and its plug-in estimate $\widehat{\rho}^{\,*}(X_a, X_b|X_S)$ decays to zero, for any fixed pair of nodes a and $b \in V$ and a fixed conditioning set S. Towards that, we impose the following regularity conditions.

(A1) There exists $s_0 > 0$ such that, for $0 \leq s < s_0$, $\sup_p \max_{1 \leq a \leq p} \mathbb{E} \exp(sX_a^2) < \infty$.

(A2) The kernel function $K(\cdot)$ is non-negative and uniformly bounded over its support.

Condition (A1) imposes a sub-exponential tail bound on the squares of the random variables. This is a quite commonly used condition, for example, in the high-dimensional feature screening literature (see, for example, [23]). Condition (A2) is a mild condition on the kernel function $K(\cdot)$ that is guaranteed by many commonly used kernels, including the Gaussian kernel. Under conditions (A1) and (A2), the next result shows that the plug-in estimate $\widehat{\rho}^{\,*}(X_a, X_b|X_S)$ converges in probability to its population counterpart $\rho_0^*(X_a, X_b|X_S)$ exponentially fast.

Theorem 1. *Under conditions (A1) and (A2), for any $\epsilon > 0$, there exist positive constants A, B and $\gamma \in (0, 1/4)$ such that*

$$\mathbb{P}(|\,\widehat{\rho}^{\,*}(X_a, X_b|X_S) - \rho_0^*(X_a, X_b|X_S)\,| > \epsilon) \;\leq\; O\Big(2\exp\Big(-A\,n^{1-2\gamma}\,\epsilon^2\Big) + n^4 \exp\big(-B\,n^{\gamma}\big)\Big).$$

The proof of Theorem 1 is long and somewhat technical; it is thus relegated to Appendix B. Theorem 1 serves as the main building block towards establishing the consistency of the nonPC algorithm in sparse high-dimensional settings.

For notational convenience, henceforth, we denote $\rho_0^*(X_a, X_b|X_S)$ and $\widehat{\rho}^*(X_a, X_b|X_S)$ by $\rho^*_{0;\,a\,b|S}$ and $\widehat{\rho}^*_{ab|S}$, respectively. In Theorem 2 below, we establish a uniform bound for the errors in inferring conditional independence relationships using the ρ_0^*-based test in the skeleton estimation step of the sample version of the nonPC algorithm.

Theorem 2. *Under conditions (A1) and (A2), for any $\epsilon > 0$, there exist positive constants A, B and $\gamma \in (0, 1/4)$ such that*

$$\begin{aligned}\sup_{\substack{a,b\in V\\ S\in J_{a,b}^{m_p}}} \mathbb{P}\big(|\,\widehat{\rho}^*_{ab|S} - \rho^*_{0;ab|S}\,| > \epsilon\big) \;&\le\; \mathbb{P}\Big(\sup_{\substack{a,b\in V\\ S\in J_{a,b}^{m_p}}} |\,\widehat{\rho}^*_{ab|S} - \rho^*_{0;ab|S}\,| > \epsilon\Big)\\ &\le\; O\Big(p^{m_p+2}\,\big[\,2\,\exp\big(-A\,n^{1-2\gamma}\,\epsilon^2\big) \,+\, n^4\,\exp\big(-B\,n^{\gamma}\big)\big]\Big)\,.\end{aligned} \tag{7}$$

Next, we turn to proving the consistency of the nonPC algorithm in the high-dimensional setting where the dimension p can be much larger than the sample size n, but the DAG is considered to be sparse. We impose the following regularity conditions, which are similar to the assumptions imposed in Section 3.1 of [8] in order to prove the consistency of the PC algorithm for Gaussian graphical models. We let the number of variables p grow with the sample size n and consider $p = p_n$, and also the DAG $\mathcal{G} = \mathcal{G}_n := (V_n, E_n)$ and the distribution $P = P_n$.

(A3) The dimension p_n grows at a rate such that the right-hand side of (7) tends to zero as $n \to \infty$. In particular, this is satisfied when $p_n = O(n^r)$ for any $0 \le r < \infty$.

(A4) The maximum degree of the DAG $\mathcal{G}_n$, denoted by $q_n := \max_{1\le a\le p_n} |adj(\mathcal{G}_n, a)|$, grows at the rate of $O(n^{1-b})$, where $0 < b \le 1$.

(A5) The distribution P_n is faithful to the DAG $\mathcal{G}_n$ for all n. In other words, for any $a, b \in V_n$ and $S \in J_{a,b}^{m_{p_n}}$,

$$X_a \text{ and } X_b \text{ are d-separated by } X_S \iff X_a \perp\!\!\!\perp X_b \,|\, X_S \iff \rho^*_{0;\,a\,b|S} = 0\,.$$

Moreover, $\rho^*_{0;\,a\,b|S}$ values are uniformly bounded both from above and below. Formally,

$$\begin{aligned} C_{min} \;&:=\; \inf_{\substack{a,b\in V_n\\ S\in J_{a,b}^{m_{p_n}}\\ \rho^*_{0;ab|S}\neq 0}} \rho^*_{0;ab|S} \;\ge\; \lambda_{min}\;\lambda_{min}^{-1} = O(n^v)\\ \text{and}\qquad C_{max} \;&:=\; \sup_{\substack{a,b\in V_n\\ S\in J_{a,b}^{m_{p_n}}}} \rho^*_{0;ab|S} \;\le\; \lambda_{max}\end{aligned}$$

where λ_{max} is a positive constant and $0 < v < 1/4$.

Condition (A3) allows the dimension to grow at any arbitrary polynomial rate of the sample size. Condition (A4) is a sparsity assumption on the underlying true DAG, allowing the maximum degree of the DAG to also grow, but at a slower rate than n. Since $m_p \le q_n$, we also have $m_p = O(n^{1-b})$. Finally, Condition (A5) is the strong faithfulness assumption (Definition 1.3 in [24]) on P_n and is similar to condition (A4) in [8]. This essentially requires $\rho^*_{0;ab|S}$ to be bounded away from zero when the vertices X_a and X_b are not d-separated by X_S. It is worth noting that the faithfulness assumption alone is not enough to prove the consistency of the PC/PC-stable/nonPC algorithms in high-dimensional settings, and the more stringent strong faithfulness condition is required.

Remark 2. *For notational convenience, treat X_a, X_b and X_S as X, Y and Z, respectively, for any $a, b \in V_n$ and $S \in J_{a,b}^{m_{p_n}}$. From Equation (3), we have*

$$\mathrm{CdCov}^2(\mathrm{X}, \mathrm{Y}|\mathrm{Z}) \;=\; \frac{1}{12}\,\mathbb{E}\,\big[\,\mathrm{d}^{\mathrm{S}}_{1234}\,|\,\mathrm{Z}_1 = \mathrm{Z}, \ldots, \mathrm{Z}_4 = \mathrm{Z}\big],$$

which implies

$$\rho_0^* \;=\; \mathbb{E}\left[\mathrm{CdCov}^2(\mathrm{X},\mathrm{Y}|\mathrm{Z})\right] \;=\; \frac{1}{12}\,\mathbb{E}\left[\,\mathrm{d}_{1234}^{\mathrm{S}}\right] \;=\; \frac{1}{12}\,\mathbb{E}\left[\,\mathrm{d}_{1234}+\mathrm{d}_{1243}+\mathrm{d}_{1432}\right].$$

Condition (A1) implies $\sup_p \max_{1\le a\le p} \mathbb{E}\, X_a^2 < \infty$. *With this and the definition of* d_{ijkl} *in Section 2.3, it follows from some simple algebra and the Cauchy–Schwarz inequality that* $\rho_0^* < \infty$. *This provides a justification for the second part of Assumption (A5) that* $\sup_{\substack{a,b\in V_n \\ S\in J_{a,b}^{mpn}}} \rho^*_{0;\,ab|S} \le \lambda_{max}$ *for some positive constant* λ_{max}.

The next theorem establishes that the nonPC algorithm consistently estimates the skeleton of a sparse high-dimensional DAG, thereby providing the necessary theoretical guarantees to our proposed methodology. It is worth noting that, in the sample version of the PC-stable and hence the nonPC algorithm, all the inference is done during the skeleton estimation step. The second step that involves appropriately orienting the edges of the estimated skeleton is purely deterministic (see Sections 4.2 and 4.3 in [7]). Therefore, to prove the consistency of the nonPC algorithm in estimating the equivalence class of the underlying true DAG, it is enough to prove the consistency of the estimated skeleton. We include the detailed proof of Theorem 3 in Appendix B.

Theorem 3. *Assume that Conditions (A1)–(A5) hold. Let* $\mathcal{G}_{skel,n}$ *be the true skeleton of the graph* $\mathcal{G}_n$, *and* $\hat{\mathcal{G}}_{skel,n}$ *be the skeleton estimated by the nonPC algorithm. Then, as* $n \to \infty$, $\mathbb{P}(\hat{\mathcal{G}}_{skel,n} = \mathcal{G}_{skel,n}) \to 1$.

Remark 3. *In the proof of Theorem 3, we consider the threshold* ξ_α *to be of constant order. However, the proof continues to work as long as* ξ_α *is of the same order as* C_{min} *as* $n \to \infty$.

3.2. The Nonparametric FCI Algorithm in High Dimensions

The FCI is a modification of the PC algorithm that accounts for latent and selection variables. Thus, generalizations of the PC algorithm naturally extend to the FCI as well. Similar to nonPC, we propose testing for conditional independence relations nonparametrically in the *sample version* of the FCI-stable algorithm (Algorithm A3 in Appendix A) based on ρ_0^*, instead of partial correlations. We coin the resulting algorithm the 'nonFCI' algorithm, to emphasize that it is a generalization of parametric FCI-stable algorithms. Again, the *oracle version* of the nonFCI is exactly the same as that of the FCI-stable algorithm. The difference is in the implementation of the tests for conditional independence relationships in their *sample versions*. This broadens the scope of the FCI algorithm in causal structural learning for observational data in the presence of latent and selection variables when Gaussianity is not a viable assumption. More specifically, it enables capturing nonlinear and non-monotone conditional dependence relationships among the variables that partial correlations would fail to detect.

Equipped with the theoretical guarantees we established for the nonPC in Section 3.1, we establish below in Theorem 4 the consistency of the nonFCI algorithm for general distributions in sparse high-dimensional settings. Let $\mathcal{H} = (V, E)$ be a DAG with the vertex set partitioned as $V = V_X \cup V_L \cup V_T$, where V_X indexes the set of p observed variables, V_L denotes the set of latent variables and V_T stands for the set of selection variables. Let $\mathcal{M}$ be the unique MAG over V_X. We let p grow with n and consider $p = p_n$, $\mathcal{H} = \mathcal{H}_n$ and $Q = Q_n$, where Q is the distribution of $(U_1, \ldots, U_p) := (X_1 \mid V_T, \ldots, X_p \mid V_T)$. We provide below the definition of possible-D-SEP sets (Definition 3.3 in [4]).

Definition 1. *Let* $\mathcal{C}$ *be a graph with any of the following edge types :* $\circ\!\!-\!\!\circ$, $\circ\!\!\rightarrow$ *and* $\leftrightarrow$. *A possible-D-SEP* (X_a, X_b) *in* $\mathcal{C}$, *denoted* $\mathrm{pds}(\mathcal{C}, X_a, X_b)$, *is defined as follows:* $X_c \in \mathrm{pds}(\mathcal{C}, X_a, X_b)$ *if and only if there is a path* π *between* X_a *and* X_c *in* $\mathcal{C}$ *such that, for every subpath* $\langle X_e, X_f, X_g\rangle$ *of* π, X_f *is a collider on the subpath in* $\mathcal{C}$ *or* $\langle X_e, X_f, X_g\rangle$ *is a triangle in* $\mathcal{C}$.

To prove the consistency of the nonFCI algorithm in sparse high-dimensional settings, we impose the following regularity conditions, which are similar to the assumptions imposed in Section 4 in [4].

(C3) The distribution Q_n is faithful to the underlying MAG $\mathcal{M}_n$ for all n.
(C4) The maximum size of the possible-D-SEP sets for finding the final skeleton in the FCI-stable algorithm (Algorithm A6 in Appendix A), q'_n, grows at the rate of $O(n^{1-b})$, where $0 < b \leq 1$.
(C5) For any $U_i, U_j \in \{U_1, \ldots, U_{p_n}\}$ and $U_S \subseteq \{U_1, \ldots, U_{p_n}\} \backslash \{U_i, U_j\}$ with $|U_S| \leq q'_n$, assume

$$\inf\left\{|\rho_0^*(U_i, U_j|U_S)| : \rho_0^*(U_i, U_j|U_S) \neq 0\right\} \geq \lambda'_{min}\ (\lambda'_{min})^{-1} = O(n^v)$$
$$\text{and} \qquad \sup |\rho_0^*(U_i, U_j|U_S)| \leq \lambda'_{max}$$

where λ'_{max} is a positive constant and $0 < v < 1/4$.

Theorem 4. *Suppose conditions (A1)–(A3) and (C3)–(C5) hold. Denote by $\mathcal{C}_n$ and $\mathcal{C}_n^*$ the true underlying FCI-PAG and the output of the nonFCI algorithm, respectively. Then, as $n \to \infty$, $\mathbb{P}(\mathcal{C}_n^* = \mathcal{C}_n) \to 1$.*

4. Numerical Studies

4.1. Performance of the NonPC Algorithm

In this subsection, we compare the performances of the nonPC and the PC-stable algorithms in finding the skeleton and the CPDAG for various simulated datasets. We simulate random DAGs in the following examples and sample from probability distributions faithful to them.

Example 1 (Linear SEM)**.** *We first fix a sparsity parameter $s \in (0, 1)$ and enumerate the vertices as $V = \{1, \ldots, p\}$. We then construct a $p \times p$ adjacency matrix Λ as follows. First, initialize Λ as a zero matrix. Next, fill every entry in the lower triangle (below the diagonal) of Λ by independent realizations of Bernoulli random variables with success probability s. Finally, replace each nonzero entry in Λ by independent realizations of a Uniform$(0.1, 1)$ random variable.*

In this scheme, each node has the same expected degree $\mathbb{E}(m) = (p-1)s$, where m is the degree of a node and follows a Binomial $(p-1, s)$ distribution. Using the adjacency matrix Λ, the data are then generated from the following linear structural equation model (SEM) :

$$X = \Lambda X + \epsilon$$

where $\epsilon = (\epsilon_1, \ldots, \epsilon_p)$ and $\epsilon_1, \ldots, \epsilon_p$ are jointly independent. To obtain samples $\{X_1^k, \ldots, X_p^k\}_{k=1}^n$ on $\{X_1, \ldots, X_p\}$, we first sample $\{\epsilon_1^k, \ldots, \epsilon_p^k\}_{k=1}^n$ from the three following data-generating schemes. For $1 \leq k \leq n$ and $1 \leq i \leq p$,

1. Normal: Generate ϵ_i^k's independently from a standard normal distribution.
2. Copula: Generate ϵ_i^k's as in (1) and then transform the marginals to a $F_{1,1}$ distribution.
3. Mixture: Generate ϵ_i^k's independently from a 50–50 mixture of a standard normal and a standard Cauchy distribution.

Example 2 (Nonlinear SEM)**.** *In this example, we first generate a $p \times p$ adjacency matrix Λ in the similar way as in Example 1 and then generate the data from the following nonlinear SEM (similar to [10]) : $X_i = \sum_{j:\Lambda_{ij} \neq 0} f_{ij}(X_j) + \epsilon_i$ with $\epsilon_i \overset{i.i.d.}{\sim} N(0, 1)$, where $1 \leq j < i \leq p$. If the functions f_{ij}'s are chosen to be nonlinear, then the data will typically not correspond to a well-known multivariate distribution. We consider $f_{ij}(x_j) = b_{ij1}x_j + b_{ij2}x_j^2$, where b_{ij1} and b_{ij1} are independently sampled from $N(0, 1)$ and $N(0, 0.5)$ distributions, respectively.*

With the exception of Example 1.1, the above examples are all non-Gaussian graphical models. We would thus expect the nonPC to perform better than the PC-stable in learning the unknown causal structure in these examples. For each of the four data generating

methods considered above, we compare the Structural Hamming Distance (SHD) [25] between the estimated and the true skeletons of the underlying DAGs using the nonPC and PC-stable algorithms. The SHD between two undirected graphs is the number of edge additions or deletions necessary to make the two graphs match. Therefore, larger SHD values between the estimated and the true skeleton correspond to worse estimates.

We consider 199 bootstrap replicates for the CdCov-based conditional independence tests in the implementation of our nonPC algorithm and the significance level $\alpha = 0.05$. Table 1 presents the average SHD for the different data generating schemes over 20 simulation runs, for different choices of n, p and $\mathbb{E}(m)$.

Table 1. Comparison of the average structural Hamming distances (SHD) of nonPC and PC-stable algorithms across simulation studies.

			Normal		Copula	
n	p	$\mathbb{E}(m)$	nonPC	PC-stable	nonPC	PC-stable
50	9	1.4	3.35	3.05	5.55	5.75
100	27	2.0	14.55	11.00	25.6	28.6
150	81	2.4	53.70	43.45	97.3	121.3
200	243	2.8	186.2	183.4	331.00	471.45
			Mixture		Nonlinear SEM	
n	p	$\mathbb{E}(m)$	nonPC	PC-stable	nonPC	PC-stable
50	9	1.4	3.8	3.5	2.9	3.7
100	27	2.0	17.75	18.00	15.05	20.05
150	81	2.4	69.05	77.75	62.583	95.083
200	243	2.8	250.3	336.1	213.70	375.45

The results in Table 1 demonstrate that the nonPC performs nearly as good as the PC-stable for the Gaussian data example, in terms of the average SHD. However, for each of the non-Gaussian data examples, the nonPC performs better than the PC-stable in estimating the true skeleton of the underlying DAGs. The improvement in SHD becomes more substantial as the dimension grows. The superior performance of the nonPC over PC-stable for the non-Gaussian graphical models is expected, as the characterization of conditional independence by partial correlations is only valid under the assumption of joint Gaussianity.

4.2. Performance of the NonFCI Algorithm

In this subsection, we compare the performances of the nonFCI and the FCI-stable algorithms over various simulated datasets. We first generate random DAGs as in Examples 1 and 2. To assess the impact of latent variables, we randomly define half of the variables with no parents and at least one child as latent. We do not consider selection variables. We run both the nonFCI and the FCI-stable algorithms on the above data examples with $n = 200$, $p = \{10, 20, 30, 100, 200\}$ and $\alpha = 0.01$, using 199 bootstrap replicates for the CdCov-based conditional independence tests. We consider 20 simulation runs for each of the data generating models. Table 2 reports the average SHD between the estimated and true PAG skeleton by the nonFCI and FCI-stable algorithms.

Table 2. Comparison of the average structural Hamming distances (SHD) of nonFCI and FCI-stable algorithms across simulation studies.

		Normal		**Copula**		**Mixture**		**Nonlinear SEM**	
p	$\mathbb{E}(m)$	**nonFCI**	**FCI-Stable**	**nonFCI**	**FCI-Stable**	**nonFCI**	**FCI-Stable**	**nonFCI**	**FCI-Stable**
10	2.0	7.15	7.60	1.3	1.8	5.65	6.80	7.15	8.20
20	2.0	14.55	17.60	4.55	6.85	13.65	18.55	19.0	20.8
30	2.0	27.65	33.95	5.25	10.15	19.3	27.8	33.40	37.85
100	3.0	109.30	150.35	26.95	60.05	62.25	111.10	115.2	149.0
200	3.0	287.75	371.40	76.733	157.267	136.05	255.10	289.6	354.1

The results in Table 2 demonstrate that, in both the Gaussian and non-Gaussian examples, the nonFCI algorithm outperforms the FCI-stable in estimating the true PAG skeleton.

4.3. Real Data Example

A major difficulty in assessing whether nonPC and nonFCI provide more reasonable estimates compared to the parametric versions of the algorithms in high-dimensional real data settings is that the true causal graph is not known in most of the cases. In absence of the truth, we may only be able to draw some conclusions about sensible causal mechanisms by examining known or logical relationships among pairs of variables. However, this becomes increasingly difficult for larger networks, where even visualization becomes challenging. This is why we first choose a relatively smaller dataset in Section 4.3.1, where we can draw upon background knowledge to glean insight into potential causal mechanisms in a setting where the data are clearly non-Gaussian. This example highlights the main focus of the paper that, with non-Gaussian data (categorical, as in this example), nonPC is expected to perform better than the PC-stable in learning the true causal structure of the underlying DAG. In Section 4.3.2, we consider a larger example and examine the performance of PC-stable and nonPC in learning the DAG from both seemingly Gaussian data as well as a categorized version of the same data. This example clearly illustrates the potential limitations of PC-stable: in contrast to nonPC, the output of PC-stable can be strikingly different when applied to a categorized version of the original data.

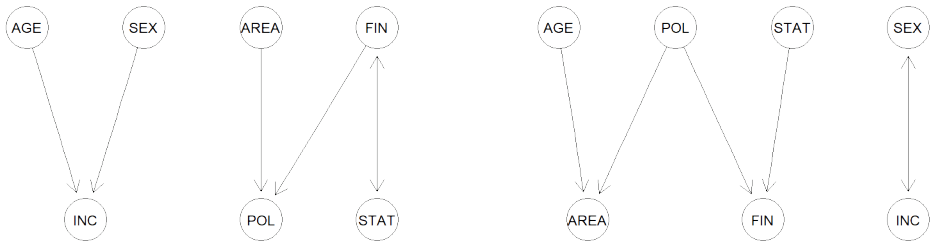
4.3.1. Montana Poll Dataset

To demonstrate the flexibility of our proposed framework, we first apply the nonPC algorithm to the Montana Economic Outlook Poll dataset. The poll was conducted in May 1992 where a random sample of 209 Montana residents were asked whether their personal financial status was worse, the same or better than a year ago, and whether they thought the state economic outlook was better than the year before. Accompanying demographic information on the respondents' age, income, political orientation, and area of residence in the state were also recorded. We obtained the dataset from the Data and Story Library (DASL), available at https://math.tntech.edu/e-stat/DASL/page4.html (accessed on 25 March 2021). The study is comprised of the following seven categorical variables: AGE = 1 for under 35, 2 for 35–54, 3 for 55 and over; SEX = 0 for male, 1 for female; INC = yearly income: 1 for under \$20 K, 2 for \$20–35 K, 3 for over \$35 K; POL = 1 for Democrat, 2 for Independent, 3 for Republican; AREA = 1 for Western, 2 for Northeastern, 3 for Southeastern Montana; FIN (=Financial status): 1 for worse, 2 for same, 3 for better than a year ago; and STAT (=State economic outlook): 1 for better, 0 for not better than a year ago.

After removing the cases with missing values, we are left with $n = 163$ samples. Since all the variables are categorical, the Gaussianity assumption is outrightly violated. Thus, we would expect the nonPC to perform better than the PC-stable in learning the true causal structure among the variables in this case. Figure 1 below presents the CPDAGs estimated by the nonPC and PC-stable algorithms at a significance level $\alpha = 0.1$. We consider 199 bootstrap replicates for the CdCov-based conditional independence tests in the implementation of the nonPC algorithm.

It is quite intuitive that age and sex are likely to affect the income; one's financial status and the area of residence might also influence their political inclination; and improvements

or downturns in the state economic outlook might impact an individual's financial status. The CPDAG estimated by the nonPC algorithm in Figure 1a affirms such common-sense understanding of these causal influences. However, in the CPDAG estimated by the PC-stable in Figure 1b, the edge between age and income is missing. In addition, the directed edges POL → AREA and POL → FIN seem to make little sense in this case.



(**a**) nonPC (**b**) PC-stable

Figure 1. CPDAGs estimated by the nonPC and PC-stable algorithms for the Montana poll dataset.

4.3.2. Protein Expression Data

We next consider a protein expression dataset of 410 patients with breast cancer from The Cancer Genome Atlas (TCGA). The dataset consists of $p = 118$ genes, and we randomly select a subset of $n = 100$ patients with PR-negative status. Since the true causal structure of the genes in the cancer cells may be different than that of normal cells [26], we apply both the nonPC and PC-stable algorithms to learn the causal structure. To put the performances of the nonPC and PC-stable under scrutiny as the data depart farther away from Gaussianity, we categorize the protein expression data for each of the p genes, denoted by $\{X_a^k\}_{k=1}^n$, $1 \le a \le p$, as follows. We compute the three quartiles $Q_{1;a}$, $Q_{2;a}$ and $Q_{3;a}$ of the protein expression values for every $1 \le a \le p$. Consequently, we obtain categorized protein expressions $\{X_{C;a}^k\}_{k=1}^n$ for $1 \le a \le p$, where

$$X_{C;a}^k := \begin{cases} 0 & \text{if } X_a^k \le Q_{1;a} \\ 1 & \text{if } Q_{1;a} < X_a^k \le Q_{2;a} \\ 2 & \text{if } Q_{2;a} < X_a^k \le Q_{3;a} \\ 3 & \text{if } X_a^k > Q_{3;a}\,. \end{cases}$$

We apply the nonPC and PC-stable algorithms to both the original and the categorized protein expression data at a significance level $\alpha = 0.01$. We consider 199 bootstrap replicates for the CdCov-based conditional independence tests in the implementation of the nonPC algorithm. Table 3 below shows the SHD between the skeletons estimated from the original and the categorized data by the nonPC and PC-stable algorithms. It can be seen that the SHD between the skeletons estimated from the original and categorized data by the PC-stable algorithm is much larger than that for nonPC. This example highlights the potential limitation of parametric implementations of the PC algorithm: when the data deviate farther away from Gaussianity (in this case, being categorical), the estimates produced by the PC-stable may deviate considerably more from the estimates from the original data. In contrast, the nonparametric test in nonPC delivers more stable estimates regardless of the data distribution.

Table 3. Comparison of the SHD between the skeletons estimated from the original and the categorized protein expression data by the nonPC and PC-stable algorithms.

nonPC	**PC-Stable**
22	79

5. Discussion

We proposed nonparametric variants of the widely popular PC-stable and FCI-stable algorithms, which employ conditional distance covariance (CdCov) to test for conditional independence relationships in their sample versions. Our proposed algorithms broaden the applicability of the PC/PC-stable and FCI/FCI-stable algorithms to general distributions over DAGs, and enable taking into account nonlinear and non-monotone conditional dependence among the random variables, which partial correlations fail to capture. We show that the high-dimensional consistency of the PC-stable and FCI-stable algorithms carry over to more general distributions over DAGs when we implement CdCov-based nonparametric tests for conditional independence. These results are obtained without imposing any strict distributional assumptions and only require moment and tail conditions on the variables.

There are several intriguing potential directions for future research. First, it is generally difficult to select the tuning parameter (i.e., the significance threshold for the CdCov test) in causal structure learning. One possible strategy is to use ideas based on *stability selection* [27,28]. By assessing the stability of the estimated graphs in multiple subsamples, this strategy allows us to choose the tuning parameter in order to control the false positive error. However, the repeated subsampling increases the computational burden. Second, the computational and sample complexities of the PC and FCI algorithms (and hence those of the nonPC and nonFCI) scale with the maximum degree of the DAG, which is assumed to be small relative to the sample size. However, in many applications, one encounters sparse graphs containing a small number of highly connected 'hub' nodes. In such cases, ref. [29] proposed a low-complexity variant of the PC algorithm, namely the *reduced PC* (rPC) algorithm that exploits the local separation property of large random networks [30]. The rPC is shown to consistently estimate the skeleton of a high-dimensional DAG by conditioning only on sets of small cardinality. More recently, ref. [31] have generalized this approach to account for unobserved confounders. In this light, it would be intriguing to develop computationally faster variants of the nonPC and nonFCI in the future by exploiting the idea of local separation.

Author Contributions: Conceptualization, S.C. and A.S.; methodology, S.C. and A.S.; formal analysis, S.C.; investigation, S.C.; writing–original draft preparation, S.C.; writing–review and editing, S.C. and A.S.; supervision, A.S.; funding acquisition, A.S. All authors have read and agreed to the published version of the manuscript.

Funding: The authors gratefully acknowledge the funding from grants R01GM114029 and R01GM133848 from the US National Institutes of Health and grant DMS-1915855 from the US National Science Foundation.

Data Availability Statement: The Montana Poll dataset has been accessed from the Data and Story Library (DASL) at https://math.tntech.edu/e-stat/DASL/page4.html (accessed on 25 March 2021).

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Preliminaries and Background

For the sake of completeness, we illustrate in this section the pseudocodes of the oracle versions of the PC-stable and FCI-stable algorithms. We also outline a local bootstrap procedure that can be used to approximate the threshold ξ_α mentioned in Section 3.1 and is used throughout the numerical studies in the paper.

Algorithm A1 presents the pseudocode of the oracle version of Step 1 of the PC-stable algorithm (Algorithm 4.1 of [7]), which estimates the skeleton of the underlying DAG. Algorithm A2 presents the pseudocode of Step 2 of the PC-stable algorithm (Algorithm 2 of [8]) that extends the skeleton estimated in Step 1 to the CPDAG. Algorithm A3 presents the pseudocode of the FCI-stable algorithm (Section 4.4 in [7]). It implements Algorithm A4 to obtain an initial skeleton of the underlying PAG, Algorithm A5 to orient the v-structures, and finally Algorithm A6 to obtain the final skeleton that the FCI-stable returns.

To approximate the threshold ξ_α to test for $H_0 : X \perp\!\!\!\perp Y|Z$ vs. $H_A : X \not\perp\!\!\!\perp Y|Z$ at level $\alpha \in (0,1)$ (see Section 3.1), we consider the following local bootstrap procedure in the light of Section 4.3 in [15]. Given the i.i.d. sample $\{W_i = (X_i, Y_i, Z_i)\}_{i=1}^n$ from the joint

distribution of $W = (X, Y, Z)$, draw a local bootstrap sample $\{W_i^\dagger = (X_i^\dagger, Y_i, Z_i)\}_{i=1}^n$ and compute the bootstrap statistic. The detailed steps are as follows :

Algorithm A1 Step 1 of the PC-stable algorithm (oracle version).

Require : Conditional independence information among all variables in V, and an ordering order(V) on the variables.
Form the complete undirected graph $\mathcal{C}$ on the vertex set V.
Let $l = -1$;
repeat
  $l = l + 1$;
  for all vertices X_a in $\mathcal{C}$ **do**
    let $u(X_a) = adj(\mathcal{C}, X_a)$
  end for
  repeat
    Select a (new) ordered pair of vertices (X_a, X_b) that are adjacent in $\mathcal{C}$ such that $|u(X_a) \setminus \{X_b\}| \geq l$, using order (V);
    repeat
      Choose a (new) set $S \subseteq u(X_a) \setminus \{X_b\}$ with $|S| = l$, using order(V);
      if $X_a \perp\!\!\!\perp X_b \,|\, S$ **then**
        Delete the edge $X_a - X_b$ from $\mathcal{C}$;
        Let *sepset* $(X_a, X_b) =$ *sepset* $(X_b, X_a) = S$;
      end if
    until X_a and X_b are no longer adjacent in $\mathcal{C}$ or all $S \subseteq u(X_a) \setminus \{X_b\}$ with $|S| = l$ have
    been considered
  until all ordered pairs of adjacent vertices (X_a, X_b) in $\mathcal{C}$ with $|u(X_a) \setminus \{X_b\}| \geq l$ have been
  considered
until all pairs of adjacent vertices (X_a, X_b) in $\mathcal{C}$ satisfy $|u(X_a) \setminus \{X_b\}| \leq l$
Output : The estimated skeleton $\mathcal{C}$, separation sets *sepset*.

Algorithm A2 Step 2 of the PC-stable algorithm.

Require : Skeleton $\mathcal{C}$, separation sets *sepset*.
for all all pair of nonadjacent vertices X_a, X_c with common neighbor X_b in $\mathcal{C}$ **do**
  if $X_b \notin sepset(X_a, X_c)$ **then**
    Replace $X_a - X_b - X_c$ in $\mathcal{C}$ by $X_a \to X_b \leftarrow X_c$;
  end if
end for
In the resulting PDAG, try to orient as many undirected edges as possible by repeated applications of the following rules :
(R1) Orient $X_b - X_c$ into $X_b \to X_c$ whenever there is an arrow $X_a \to X_b$ such that X_a and X_c are nonadjacent (otherwise, a new v-structure is created).
(R2) Orient $X_a - X_c$ into $X_a \to X_c$ whenever there is a chain $X_a \to X_b \to X_c$ (otherwise, a directed cycle is created).
(R3) Orient $X_a - X_c$ into $X_a \to X_c$ whenever there are two chains $X_a - X_b \to X_c$ and $X_a - X_d \to X_c$ such that X_b and X_d are nonadjacent (otherwise, a new v-structure or a directed cycle is created).

Algorithm A3 The FCI-stable algorithm (oracle version).

Require : Conditional independence information among all variables in V_X given V_T.
Use Algorithm A4 to find an initial skeleton ($\mathcal{C}$), separation sets (sepset) and unshielded triple list ($\mathcal{M}$);
Use Algorithm A5 to orient v-structures (update $\mathcal{C}$);
Use Algorithm A6 to find the final skeleton (update $\mathcal{C}$ and sepset);
Use Algorithm A5 to orient v-structures (update $\mathcal{C}$);
Use rules (R1)-(R10) of [6] to orient as many edge marks as possible (update $\mathcal{C}$);
Output : $\mathcal{C}$, sepset.

Algorithm A4 Obtaining an initial skeleton in the FCI-stable algorithm (Algorithm 4.1 in the supplement of [4]).

Require : Conditional independence information among all variables in V_X given V_T, and an ordering order(V_X) on the variables.
Form the complete undirected graph $\mathcal{C}$ on the vertex set V_X with edges $\circ\!\!-\!\!\circ$.
Let $l = -1$;
repeat
  $l = l + 1$;
  for all vertices X_a in $\mathcal{C}$ **do**
    let $u(X_a) = adj(\mathcal{C}, X_a)$
  end for
  repeat
    Select a (new) ordered pair of vertices (X_a, X_b) that are adjacent in $\mathcal{C}$ such that $|u(X_a) \setminus \{X_b\}| \geq l$, using order (V_X);
    repeat
      Choose a (new) set $Y \subseteq u(X_a) \setminus \{X_b\}$ with $|Y| = l$, using order(V_X);
      if $X_a \perp\!\!\!\perp X_b \,|\, Y \cup V_T$ **then**
        Delete the edge $X_a \circ\!\!-\!\!\circ X_b$ from $\mathcal{C}$;
        Let sepset$(X_a, X_b) =$ sepset$(X_b, X_a) = Y$;
      end if
    until X_a and X_b are no longer adjacent in $\mathcal{C}$ or all $Y \subseteq u(X_a) \setminus \{X_b\}$ with $|Y| = l$ have
    been considered
  until all ordered pairs of adjacent vertices (X_a, X_b) in $\mathcal{C}$ with $|u(X_a) \setminus \{X_b\}| \geq l$ have been
  considered
until all pairs of adjacent vertices (X_a, X_b) in $\mathcal{C}$ satisfy $|u(X_a) \setminus \{X_b\}| \leq l$
Form a list $\mathcal{M}$ of all unshielded triples $\langle X_c \cdot X_d \rangle$ (i.e., the middle vertex is left unspecified) in $\mathcal{C}$ with $c < d$.
Output : $\mathcal{C}$, sepset, $\mathcal{M}$.

Algorithm A5 Orienting v-structures in the FCI-stable algorithm (Algorithm 4.2 in the supplement of [4]).

Require : Initial skeleton ($\mathcal{C}$), separation sets (sepset) and unshielded triple list ($\mathcal{M}$).
for all elements $\langle X_a, X_b, X_c \rangle$ of $\mathcal{M}$ **do**
  if $X_b \notin$ sepset(X_a, X_c) **then** Orient $X_a \star\!\!-\!\!\circ X_b \circ\!\!-\!\!\star X_c$ as $X_a \star\!\!\rightarrow X_b \leftarrow\!\!\star X_c$
  end if
end for
Output : $\mathcal{C}$, sepset.

Algorithm A6 Obtaining the final skeleton in the FCI-stable algorithm (Algorithm 4.3 in the supplement of [4]).

Require: Partially oriented graph ($\mathcal{C}$) and separation sets (sepset).
for all vertices X_a in $\mathcal{C}$ **do**
  let $v(X_a) = \text{pds}(\mathcal{C}, X_a, \cdot)$;
  for all vertices $X_b \in adj(\mathcal{C}, X_a)$ **do**
    Let $l = -1$;
    repeat
      $l = l + 1$;
      repeat
        Choose a (new) set $Y \subseteq v(X_a) \setminus \{X_b\}$ with $|Y| = l$;
        if $X_a \perp\!\!\!\perp X_b \,|\, Y \cup V_T$ **then**
          Delete the edge $X_a \star\!\!-\!\!\star X_b$ from $\mathcal{C}$;
          Let sepset$(X_a, X_b) =$ sepset$(X_b, X_a) = Y$;
        end if
      until X_a and X_b are no longer adjacent in $\mathcal{C}$ or all $Y \subseteq v(X_a) \setminus \{X_b\}$ with $|Y| = l$ have
        been considered
    until X_a and X_b are no longer adjacent in $\mathcal{C}$ or $|v(X_a) \setminus \{X_b\}| < l$
  end for
end for
Reorient all edges in $\mathcal{C}$ as $\circ\!\!-\!\!\circ$.
Form a list $\mathcal{M}$ of all unshielded triples $\langle X_c \cdot X_d \rangle$ in $\mathcal{C}$ with $c < d$.
Output : $\mathcal{C}$, sepset, $\mathcal{M}$.

A. For $i = 1, \ldots, n$, draw $X_i^\dagger$ from

$$\widehat{F}_{X|Z=Z_i} \;=\; \frac{\sum_{j=1}^n K_{ij}\, \mathbf{1}(-\infty, X_j](x)}{\sum_{j=1}^n K_{ij}}.$$

Compute $\widehat{\rho}^{\,*\dagger}$ based on the local bootstrap sample $\{W_i^\dagger = (X_i^\dagger, Y_i, Z_i)\}_{i=1}^n$.

B. Repeat Step A B times to obtain $\{\widehat{\rho}_b^{\,*\dagger}\}_{b=1}^B$. Obtain $\xi_{n,\alpha}^{\,*}$ as the $100(1-\alpha)^{th}$ percentile of $\{nh^{r/2}\,\widehat{\rho}_b^{\,*\dagger}\}_{b=1}^B$. Then, $\frac{1}{nh^{r/2}}\,\xi_{n,\alpha}^{\,*}$ can be considered as an approximation for ξ_α.

Appendix B. Proofs of the Theoretical Results

In this section, we provide detailed technical proofs of the theoretical results presented in the paper. We first state a concentration inequality in Lemma A1. The result in Lemma A1 is not new and can be seen as a corollary of Theorem A in Section 5.6.1 of [32]; however, it is a key technical ingredient in the proof of Theorem 1, which is the main theoretical innovation of our paper. For completeness, we include a short proof for Lemma A1.

Lemma A1. *Consider a U-statistic* $U_n = U(X_1, \ldots, X_n) = \binom{n}{m}^{-1} \sum_{i_1 < \cdots < i_m} h(X_{i_1}, \ldots, X_{i_m})$ *with a symmetric kernel* h *such that* $\mathbb{E}\, U_n = \mathbb{E}\, h(X_1, \ldots, X_m) = \theta$. *Further suppose* $|h(X_1, \ldots, X_m)| \le M$ *for some* $M > 0$. *Then, for any* $\epsilon > 0$, *we have*

$$\mathbb{P}(|U_n - \theta| > \epsilon) \;\le\; 2\,\exp\left(-\frac{\epsilon^2\, k}{2M^2}\right)$$

where $k := \lfloor \frac{n}{m} \rfloor$.

Proof of Lemma A1. Define

$$W(X_1, \ldots, X_n) \;:=\; \frac{1}{k}\left[h(X_1, \ldots, X_m) + h(X_{m+1}, \ldots, X_{2m}) + \cdots + h(X_{km-m+1}, \ldots, X_{km})\right].$$

Then, following Section 5.1.6 in [32], we can write

$$U_n \;=\; \frac{1}{n!}\sum_{\pi} W(X_{i_1},\ldots,X_{i_n}) \tag{A1}$$

where $\sum_{\pi}$ denotes summation over all $n!$ permutations $(i_1,\ldots,i_n)$ of $(1,2,\ldots,n)$. Thus, U_n can be expressed as an average of $n!$ terms, each of which is an average of k i.i.d. random variables. Using Markov's inequality, convexity of the exponential function and Jensen's inequality, we have, for any $t>0$,

$$\begin{aligned}
\mathbb{P}(U_n-\theta>\epsilon) \;&=\; \mathbb{P}\big(\exp\left(t\,(U_n-\theta)\right)>\exp(t\epsilon)\big)\\
&\le\; \exp(-t\epsilon)\exp(-t\theta)\,\mathbb{E}\left[\exp\left(t\,U_n\right)\right]\\
&=\; \exp(-t\epsilon)\exp(-t\theta)\,\mathbb{E}\left[\exp\left(t\,\frac{1}{n!}\sum_{p} W(X_{i_1},\ldots,X_{i_n})\right)\right]\\
&\le\; \exp(-t\epsilon)\exp(-t\theta)\,\frac{1}{n!}\sum_{\pi}\mathbb{E}\left[\exp\left(t\,W(X_{i_1},\ldots,X_{i_n})\right)\right]\\
&=\; \exp(-t\epsilon)\exp(-t\theta)\left[\mathbb{E}\left(\exp\left(\frac{t}{k}\,h\right)\right)\right]^k\\
&=\; \exp(-t\epsilon)\,\mathbb{E}^k\left[\exp\left(\frac{t}{k}(h-\theta)\right)\right]
\end{aligned} \tag{A2}$$

where, for notational simplicity, we use h to denote $h(X_1,\ldots,X_m)$. Using Hoeffding's Lemma, we have from (A2)

$$\mathbb{P}(U_n-\theta>\epsilon) \;\le\; \exp\left(-t\epsilon\;+\;k\,\frac{1}{8}\,\frac{t^2}{k^2}\,(2M)^2\right) \;=\; \exp\left(-t\epsilon\;+\;\frac{t^2M^2}{2k}\right).$$

Symmetrically, we obtain

$$\mathbb{P}(|U_n-\theta|>\epsilon) \;\le\; 2\,\exp\left(-t\epsilon\;+\;\frac{t^2M^2}{2k}\right). \tag{A3}$$

The right-hand side of (A3) is minimized at $t=\epsilon\,k/M^2$. Therefore, choosing $t=\epsilon\,k/M^2$, we obtain

$$\mathbb{P}(|U_n-\theta|>\epsilon) \;\le\; 2\,\exp\left(-\frac{\epsilon^2\,k}{2M^2}\right).$$

□

Proof of Theorem 1. When $|S|=0$, it can be shown in similar lines of Theorem 1 in Li et al. (2012) [33] that, for any $\epsilon>0$, there exist positive constants A, B and $\gamma\in(0,1/4)$ such that

$$\mathbb{P}(|\,\widehat{\rho}^{\,*}(X_a,X_b|X_S)-\rho_0^*\,(X_a,X_b|X_S)\,|>\epsilon) \;\le\; O\Big(2\,\exp\Big(-A\,n^{1-2\gamma}\,\epsilon^2\Big)\;+\;n\,\exp\left(-B\,n^{\gamma}\right)\Big).$$

Now, consider the case $0<|S|\le m_p$.

For notational convenience, we treat X_a, X_b and X_S as X, Y and Z, respectively.

Denote $\delta_Z := \mathrm{CdCov}^2(X,Y|Z)$. Then, $\rho_0^*=\mathbb{E}[\delta_Z]$. Recall that

$$\begin{aligned} \widehat{\rho}^{\,*}(X,Y|Z) \;&:=\; \frac{1}{n}\sum_{u=1}^{n} \mathrm{CdCov}_n^2(X,Y|Z_u) \;:=\; \frac{1}{n}\sum_{u=1}^{n}\Delta_{i,j,k,l;u} \\ \text{where} \qquad \Delta_{i,j,k,l;u} \;&:=\; \sum_{i,j,k,l} \frac{K_{iu}\,K_{ju}\,K_{ku}\,K_{lu}}{12\left(\sum_{i=1}^{n} K_{iu}\right)^4}\, d^S_{ijkl}\,. \end{aligned} \tag{A4}$$

From (A4), we have

$$\begin{aligned} &\mathbb{E}\left[\mathrm{CdCov}_n^2(X,Y|Z_u)|Z\right] \\ &= \frac{1}{12}\,\mathbb{E}\left[\,d^S_{1234}\,|\,Z_1=Z_u,\ldots,Z_4=Z_u\right] \sum_{i,j,k,l} K_{iu}\,K_{ju}\,K_{ku}\,K_{lu} \,/\, \left(\sum_{i=1}^{n} K_{iu}\right)^4 \\ &= \frac{1}{12}\,\mathbb{E}\left[\,d^S_{1234}\,|\,Z_1=Z_u,\ldots,Z_4=Z_u\right] \;=\; \delta_{Z_u} \end{aligned} \tag{A5}$$

where the last equality follows from Lemma 1 in [15]. Together, (A4) and (A5) imply $\mathbb{E}\left[\widehat{\rho}^{\,*}\right] \;=\; \rho_0^*$.

Now, consider the truncation

$$\begin{aligned} \rho_0^* \;&=\; \rho_{01}^* + \rho_{02}^* \\ &:=\; \mathbb{E}\left[\frac{1}{12}\,d^S_{i,j,k,l}\,\mathbf{1}\left(\left|\frac{1}{12}\,d^S_{i,j,k,l}\right| \le M\right)\right] \;+\; \mathbb{E}\left[\frac{1}{12}\,d^S_{i,j,k,l}\,\mathbf{1}\left(\left|\frac{1}{12}\,d^S_{i,j,k,l}\right| > M\right)\right] \end{aligned} \tag{A6}$$

where $M>0$ will be specified later. Then, using triangle inequality,

$$\begin{aligned} \mathbb{P}(|\widehat{\rho}^{\,*}-\rho_0^*|>\epsilon) \;&=\; \mathbb{P}\left(\left|\frac{1}{n}\sum_{u=1}^{n}\Big(\sum_{i,j,k,l}\Delta_{i,j,k,l;u}-\rho_0^*\Big)\right|>\epsilon\right) \\ &\le\; \mathbb{P}\left(\left|\frac{1}{n}\sum_{u=1}^{n}\left(\sum_{i,j,k,l}\Delta_{i,j,k,l;u}\,\mathbf{1}\left(\left|\frac{1}{12}\,d^S_{i,j,k,l}\right|\le M\right)-\rho_{01}^*\right)\right|>\epsilon/2\right) \\ &\quad+\; \mathbb{P}\left(\left|\frac{1}{n}\sum_{u=1}^{n}\left(\sum_{i,j,k,l}\Delta_{i,j,k,l;u}\,\mathbf{1}\left(\left|\frac{1}{12}\,d^S_{i,j,k,l}\right|> M\right)-\rho_{02}^*\right)\right|>\epsilon/2\right) \\ &=: \mathrm{I}+\mathrm{II}\,. \end{aligned} \tag{A7}$$

Clearly, from (A4), we have $|\Delta_{i,j,k,l;u}| \le M$ when $\left|\frac{1}{12}\,d^S_{i,j,k,l}\right| \le M$. With this observation, we have

$$\mathrm{I} \;\le\; 2\,\exp\left(-\frac{n\,\epsilon^2}{8\,M^2}\right) \tag{A8}$$

which follows from Lemma A1 by setting $m=1$, $k=\lfloor n \rfloor$ and $\epsilon=\epsilon/2$. Choosing $M=c\,n^{\gamma}$ for $\gamma\in(0,1/4)$ and some positive constant c, it follows from (A8) that

$$\mathrm{I} \;\le\; 2\,\exp\left(-C_1\,n^{1-2\gamma}\,\epsilon^2\right) \tag{A9}$$

for some $C_1>0$.

Now, to find a suitable upper bound for II, note that a simple application of triangle inequality yields

$$\begin{aligned} \frac{\epsilon}{2} &< \left| \frac{1}{n} \sum_{u=1}^{n} \sum_{i,j,k,l} \Delta_{i,j,k,l;u} \, \mathbf{1}\left(\left| \frac{1}{12} \, d^S_{i,j,k,l} \right| > M \right) - \rho^*_{02} \right| \\ &\leq \left| \frac{1}{n} \sum_{u=1}^{n} \sum_{i,j,k,l} \Delta_{i,j,k,l;u} \, \mathbf{1}\left(\left| \frac{1}{12} \, d^S_{i,j,k,l} \right| > M \right) \right| + \left| \rho^*_{02} \right| . \end{aligned} \tag{A10}$$

For the choice of $M = c\, n^{\gamma}$, we have

$$\rho^*_{02} \;=\; \mathbb{E}\left[\frac{1}{12} \, d^S_{i,j,k,l} \, \mathbf{1}\left(\left| \frac{1}{12} \, d^S_{i,j,k,l} \right| > M \right) \right] \;<\; \frac{\epsilon}{4} \tag{A11}$$

for sufficiently large n (see, for example, Exercise 6 in Chapter 5, [34]). Combining (A10) and (A11), we obtain

$$\begin{aligned} &\left\{ \left| \frac{1}{n} \sum_{u=1}^{n} \sum_{i,j,k,l} \Delta_{i,j,k,l;u} \, \mathbf{1}\left(\left| \frac{1}{12} \, d^S_{i,j,k,l} \right| > M \right) - \rho^*_{02} \right| > \epsilon/2 \right\} \\ &\subseteq \left\{ \left| \frac{1}{n} \sum_{u=1}^{n} \sum_{i,j,k,l} \Delta_{i,j,k,l;u} \, \mathbf{1}\left(\left| \frac{1}{12} \, d^S_{i,j,k,l} \right| > M \right) \right| > \epsilon/4 \right\} \\ &\subseteq \left\{ \left[\left| \frac{1}{12} \, d^S_{i,j,k,l} \right| > M \right] \text{ for some } 1 \leq i,j,k,l \leq n \right\}, \end{aligned}$$

which implies

$$\begin{aligned} &\mathbb{P}\left(\left| \frac{1}{n} \sum_{u=1}^{n} \sum_{i,j,k,l} \Delta_{i,j,k,l;u} \, \mathbf{1}\left(\left| \frac{1}{12} \, d^S_{i,j,k,l} \right| > M \right) - \rho^*_{02} \right| > \epsilon/2 \right) \\ &\leq \mathbb{P}\left(\left| \frac{1}{n} \sum_{u=1}^{n} \sum_{i,j,k,l} \Delta_{i,j,k,l;u} \, \mathbf{1}\left(\left| \frac{1}{12} \, d^S_{i,j,k,l} \right| > M \right) \right| > \epsilon/4 \right) \\ &\leq n^4 \, \mathbb{P}\left(\left| \frac{1}{12} \, d^S_{i,j,k,l} \right| > M \right). \end{aligned} \tag{A12}$$

This is because, if $\left| \frac{1}{12} \, d^S_{i,j,k,l} \right| \leq M$ for all $1 \leq i,j,k,l \leq n$, then

$$n^{-1} \sum_{u=1}^{n} \sum_{i,j,k,l} \Delta_{i,j,k,l;u} \, \mathbf{1}\left(\left| \frac{1}{12} \, d^S_{i,j,k,l} \right| > M \right) = 0.$$

Under Condition (A1), Lemma 2 in the supplementary materials of [35] proves that there exists $s > 0$ for which $\mathbb{E}\left[\exp\left(s \left| d^S_{1234} \right| \right) \right]$ is finite. Using Markov's inequality, we have

$$\begin{aligned} \mathbb{P}\left(\left| \frac{1}{12} \, d^S_{i,j,k,l} \right| > M \right) &\leq \mathbb{P}\left(\exp\left(s \left| \frac{1}{12} \, d^S_{i,j,k,l} \right| \right) > \exp(sM) \right) \\ &\leq \exp(-sM) \, \mathbb{E}\left[\exp\left(s \left| \frac{1}{12} \, d^S_{i,j,k,l} \right| \right) \right] \\ &\leq C_2 \exp(-sM) \;\leq\; C_2 \exp(-s_1 \, n^{\gamma}) \end{aligned} \tag{A13}$$

for some positive constants C_2 and s_1, where the last line uses the fact that $M = c\, n^{\gamma}$. Combining (A12) and (A13), we have

$$\text{II} \;\leq\; C_2 \, n^4 \exp(-s_1 \, n^{\gamma}) . \tag{A14}$$

Finally, combining (A7), (A9) and (A14), we obtain

$$\mathbb{P}(|\widehat{\rho}^{\,*} - \rho_0^*| > \epsilon/2) \;\le\; 2\,\exp\Big(-C_1 n^{1-2\gamma}\epsilon^2\Big) \;+\; C_2\; n^4\,\exp(-s_1 n^{\gamma})$$

for some positive constants γ, C_1, C_2 and s_1. This completes the proof of the theorem. □

Proof of Theorem 2. The first inequality in Theorem 2 simply follows by observing the fact that, for any generic random sequence $\{X_n\}_{n=1}^{\infty}$ and any $\epsilon > 0$,

$$P(|X_n| > \epsilon) \;\le\; P\big(\sup_n |X_n| > \epsilon\big)$$

for all $n \ge 1$, which, in turn, implies

$$\sup_n P(|X_n| > \epsilon) \;\le\; P\big(\sup_n |X_n| > \epsilon\big)\,.$$

The second inequality follows from union bound and Theorem 1. □

Proof of Theorem 3. Denote by $E_{ab|S}$ the event that "an error occurs while testing for $X_a \perp\!\!\!\perp X_b \,|\, X_S$" for $a, b \in V$ and $S \in J_{a,b}^{m_{p_n}}$. Then,

$$\mathbb{P}(\text{ an error occurs in the nonPC algorithm }) \;\le\; \mathbb{P}\Bigg(\bigcup_{\substack{a,b\in V\\ S\in J_{a,b}^{m_{p_n}}}} E_{ab|S}\Bigg) \;\lesssim\; p_n^{m_{p_n}+2}\,\mathbb{P}(E_{ab|S}) \tag{A15}$$

which is essentially due to the union bound. Now, we can write $E_{ab|S} = E^{\mathrm{I}}_{ab|S} \cup E^{\mathrm{II}}_{ab|S}$, where

$$\begin{aligned} &\text{(Type I error)} \qquad E^{\mathrm{I}}_{ab|S} \,:\, |\hat{\rho}^*_{ab|S}| > \xi_\alpha \qquad \text{when } \rho^*_{0;ab|S} = 0 \\ \text{and} \quad &\text{(Type II error)} \qquad E^{\mathrm{II}}_{ab|S} \,:\, |\hat{\rho}^*_{ab|S}| \le \xi_\alpha \qquad \text{when } \rho^*_{0;ab|S} > 0\,. \end{aligned}$$

Then, by using triangle inequality,

$$\begin{aligned} \mathbb{P}(E^{\mathrm{I}}_{ab|S}) \;&=\; \mathbb{P}(|\,\hat{\rho}^*_{ab|S}\,| > \xi_\alpha) \;=\; \mathbb{P}(|\,\hat{\rho}^*_{ab|S} - \rho^*_{0;ab|S} + \rho^*_{0;ab|S}\,| > \xi_\alpha) \\ &\le\; \mathbb{P}(|\,\hat{\rho}^*_{ab|S} - \rho^*_{0;ab|S}\,| > \xi_\alpha - C_{max}) \\ &\lesssim\; 2\,\exp\big(-A\,n^{1-2\gamma}(\xi_\alpha - C_{max})^2\big) \;+\; n^4\,\exp\big(-Bn^{\gamma}\big) \end{aligned} \tag{A16}$$

for positive constants A, B and $\gamma \in (0, 1/4)$, where the last inequality follows from Theorem 2. Similarly, using the definition of C_{min} and the identity $|a| - |b| \le |a - b|$ for $a, b \in \mathbb{R}$, we have

$$\begin{aligned} \mathbb{P}\Big(E^{\mathrm{II}}_{ab|S}\Big) \;&=\; \mathbb{P}(|\,\hat{\rho}^*_{ab|S}\,| \le \xi_\alpha) \;=\; \mathbb{P}(-|\,\hat{\rho}^*_{ab|S}\,| \ge -\xi_\alpha) \\ &=\; \mathbb{P}(|\rho^*_{0;ab|S}| - |\,\hat{\rho}^*_{ab|S}| \ge |\rho^*_{0;ab|S}| - \xi_\alpha) \\ &\le\; \mathbb{P}(|\rho^*_{0;ab|S} - \hat{\rho}^*_{ab|S}| \ge C_{min} - \xi_\alpha) \\ &\lesssim\; 2\,\exp\big(-A\,n^{1-2\gamma}(\xi_\alpha - C_{min})^2\big) \;+\; n^4\,\exp\big(-Bn^{\gamma}\big)\,. \end{aligned} \tag{A17}$$

Again, the last inequality follows from Theorem 2. Combining Equations (A15)–(A17), we have

$$\begin{aligned}
&\mathbb{P}\,(\text{ an error occurs in the nonPC algorithm }) \\
&= O\Big(p_n^{m_{p_n}+2} \big[2\,\exp\big(-A\,n^{1-2\gamma}(\xi_\alpha - C_{max})^2\big) + 2\,\exp\big(-A\,n^{1-2\gamma}(\xi_\alpha - C_{min})^2\big] \\
&\qquad + n^4 \exp\big(-B\,n^\gamma\big)\big]\Big) \\
&= o(1)
\end{aligned}$$

where the last step follows from the fact that $\gamma \in (0, 1/4)$ and Assumption (A5). This implies that, as $n \to \infty$,

$$\begin{aligned}
\mathbb{P}\big(\hat{G}_{\text{skel},n} = G_{\text{skel},n}\big) &= 1 - \mathbb{P}\,(\text{ an error occurs in the nonPC algorithm }) \\
&\to 1\,.
\end{aligned}$$

□

Proof of Theorem 4. The proof follows similar lines of the proof of Theorem 4.2 in [4], replacing Lemma 1.4 in their supplement by Theorem 2 in our paper.
□

References

1. Lauritzen, S.L. *Graphical Models*; Oxford University Press: Oxford, UK, 1996.
2. Maathuis, M.; Drton, M.; Lauritzen, S.; Wainwright, M. *Handbook of Graphical Models*; CRC Press: Boca Raton, FL, USA, 2019.
3. Spirtes, P.; Glymour, C.; Scheines, R. *Causation, Prediction, and Search*, 2nd ed; The MIT Press: Cambridge, MA, USA, 2000.
4. Colombo, D.; Maathuis, M.H.; Kalisch, M.; Richardson, T.S. Learning high-dimensional directed acyclic graphs with latent and selection variables. *Ann. Stat.* **2012**, *40*, 294–321. [CrossRef]
5. Spirtes, P. An anytime algorithm for causal inference. In Proceedings of the 8th International Workshop on Artificial Intelligence and Statistics, Key West, FL, USA, 3–6 January 2001; pp. 213–221.
6. Zhang, J. On the completeness of orientation rules for causal discovery in the presence of latent confounders and selection bias. *Artif. Intell.* **2008**, *172*, 1873–1896. [CrossRef]
7. Colombo, D.; Maathuis, M.H. Order-independent constraint-based causal structure learning. *J. Mach. Learn. Res.* **2014**, *15*, 3921–3962.
8. Kalisch, M.; Bühlmann, P. Estimating High-Dimensional Directed Acyclic Graphs with the PC-Algorithm. *J. Mach. Learn. Res.* **2007**, *8*, 613–636.
9. Loh, P-L.; Bühlmann, P. High-Dimensional Learning of Linear Causal Networks via Inverse Covariance Estimation. *J. Mach. Learn. Res.* **2014**, *15*, 3065–3105.
10. Voorman, A.; Shojaie, A.; Witten, D. Graph estimation with joint additive models. *Biometrika* **2014**, *99*, 1–25. [CrossRef]
11. Harris, N.; Drton, M. PC Algorithm for Nonparanormal Graphical Models. *J. Mach. Learn. Res.* **2013**, *14*, 3365–3383.
12. Sun, X.; Janzing, D.; Schölkopf, B.; Fukumizu, K. A kernel-based causal learning algorithm. In Proceedings of the 24th International Conference on Machine Learning, Corvalis, OR, USA, 20–24 June 2007; pp. 855–862.
13. Zhang, K.; Peters, J.; Janzing, D.; Schölkopf, B. Kernel-based conditional independence test and application in causal discovery. *arXiv* **2012**, arXiv:1202.3775.
14. Székely, G.J.; Rizzo, M.L.; Bakirov, N.K. Measuring and testing independence by correlation of distances. *Ann. Stat.* **2007**, *35*, 2769–2794. [CrossRef]
15. Wang, X.; Wenliang, P.; Hu, W.; Tian, Y.; Zhang, H. Conditional distance correlation. *J. Am. Stat. Assoc.* **2015**, *110*, 1726–1734. [CrossRef]
16. Pearl, J. *Causality*; Cambridge University Press: Cambridge, UK, 2000.
17. Verma, T.; Pearl, J. Equivalence and synthesis of causal models. In Proceedings of the Sixth Annual Conference on Uncertainty in Artificial Intelligence, Cambridge, MA, USA, 27–29 July 1990; pp. 255–270.
18. Richardson, T.S.; Spirtes, P. Ancestral graph markov models. *Ann. Stat.* **2002**, *30*, 962–1030. [CrossRef]
19. Ali, R.A.; Richardson, T.S.; Spirtes, P. Markov equivalence for ancestral graphs. *Ann. Stat.* **2009**, *37*, 2808–2837. [CrossRef]
20. Székely, G.J.; Rizzo, M.L. Partial distance correlation with methods for dissimilarities. *Ann. Stat.* **2014**, *42*, 2382–2412. [CrossRef]
21. Sheng, T.; Sriperumbudur, B.K. On distance and kernel measures of conditional independence. *arXiv* **2019**, arXiv:1912.01103.
22. Chakraborty, S.; Zhang, X. Distance Metrics for Measuring Joint Dependence with Application to Causal Inference. *J. Am. Stat. Assoc.* **2019**, *114*, 1638–1650. [CrossRef]
23. Liu, J.; Li, R.; Wu, R. Feature selection for varying coefficient models with ultrahigh-dimensional covariates. *J. Am. Stat. Assoc.* **2014**, *109*, 266–274. [CrossRef]

24. Uhler, C.; Raskutti, G.; Bühlmann, P.; Yu, B. Geometry of the faithfulness assumption in causal inference. *Ann. Stat.* **2013**, *41*, 436–463. [CrossRef]
25. Tsamardinos, I.; Brown, L.E.; Aliferis, C.F. The max-min hill-climbing Bayesian network structure learning algorithm. *Mach. Learn.* **2006**, *65*, 31–78. [CrossRef]
26. Shojaie, A. Differential network analysis: A statistical perspective. In *Wiley Interdisciplinary Reviews: Computational Statistics*; Wiley: New York, NY, USA, 2021; p. e1508.
27. Meinshausen, N.; Bühlmann, P. Stability selection. *J. R. Stat. Soc.* **2010**, *72*, 417–473. [CrossRef]
28. Shah, R.D.; Samworth, R.J. Variable selection with error control: Another look at stability selection. *J. R. Stat. Soc.* **2013**, *75*, 55–80. [CrossRef]
29. Sondhi, A.; Shojaie, A. The Reduced PC-Algorithm: Improved Causal Structure Learning in Large Random Networks. *J. Mach. Learn. Res.* **2019**, *20*, 1–31.
30. Anandkumar, A.; Tan, V.Y.F.; Huang, F.; Willsky, A.S. High-Dimensional Gaussian Graphical Model Selection: Walk Summability and Local Separation Criterion. *J. Mach. Learn. Res.* **2012**, *13*, 2293–2337.
31. Chen, W.; Drton, M.; Shojaie, A. Causal structural learning via local graphs. *arXiv* **2021**, arXiv:2107.03597.
32. Serfling, R. J. *Approximation Theorems of Mathematical Statistics*; Wiley: New York, NY, USA, 1980.
33. Li, R.; Zhong, W.; Zhu, L. Feature selection via distance correlation learning. *J. Am. Stat. Assoc.* **2012**, *107*, 1129–1139. [CrossRef] [PubMed]
34. Resnick, S. I. *A Probability Path*; Springer: Berlin/Heidelberg, Germany, 1999.
35. Wen, C.; Wenliang, P.; Huang, M.; Wang, X. Sure Independence Screening Adjusted for Confounding Covariates with Ultrahigh Dimensional Data. *Stat. Sin.* **2018**, *28*, 293–317.