Proceedings of Machine Learning Research vol 140:1–18, 2022 1st Conference on Causal Learning and Reasoning

Causal Structure Discovery between Clusters of Nodes Induced by Latent Factors

Author names withheld

Editor: Under Review for CLeaR 2022

Abstract

We consider the problem of learning the structure of a causal directed acyclic graph (DAG) model in the presence of latent variables. We define *latent factor causal models* (LFCMs) as a restriction on causal DAG models with latent variables, which are composed of clusters of observed variables that share the same latent parent and connections between these clusters given by edges pointing from the observed variables to latent variables. LFCMs are motivated by gene regulatory networks, where regulatory edges, corresponding to transcription factors, connect spatially clustered genes. We show identifiability results on this model and design a consistent three-stage algorithm that discovers clusters of observed nodes, a partial ordering over clusters, and finally, the entire structure over both observed and latent nodes. We evaluate our method in a synthetic setting, demonstrating its ability to almost perfectly recover the ground truth clustering even at relatively low sample sizes, as well as the ability to recover a significant number of the edges from observed variables to latent factors. Finally, we apply our method in a semi-synthetic setting to protein mass spectrometry data with a known ground truth network, and achieve almost perfect recovery of the ground truth variable clusters.

Keywords: Causal discovery, causal structure learning, causal identifiability, latent factor model

1. Introduction

Structural causal models are valuable tools for reasoning about decision-making, and as a result, have been widely adopted across fields such as genomics (Friedman et al., 2000), econometrics (Blalock, 2017), and epidemiology (Robins et al., 2000). To use causal models when the causal structure is not known *a priori*, it is necessary to learn the model from observed data, a task known as *causal structure learning* (Heinze-Deml et al., 2018). As a field, causal structure learning has recently experienced major developments and remains an active and widespread area of research. Recent works aim to address a number of challenges inherent to the problem of learning causal structure, such as the presence of unobserved confounders (Cai et al., 2019; Frot et al., 2019; Bernstein et al., 2020), the large search space over causal models (Chickering, 2002; Solus et al., 2021), identifiability of the underlying causal model (Shimizu et al., 2006; Peters and Bühlmann, 2014), and statistical issues stemming from high-dimensional datasets (Nandy et al., 2018). We focus on a setting which exhibits all of these challenges, and our proposed method addresses each of these challenges in a cohesive way. We devote particular attention to the issue of unobserved confounders,

A number of methods have been proposed to address the challenge of learning causal models in the presence of unobserved confounders. These methods fall into two general categories. First, some methods account for unobserved confounders by learning a graphical model over only the observed variables, albeit from a *different* class of graphical models (Richardson and Spirtes, 2002; Bernstein et al., 2020). However, in some cases, such as the one explored in this paper, it is possible to learn a graph over *both* the observed and latent variables. Existing methods (Silva et al., 2006;

© 2022 .

Kummerfeld and Ramsey, 2016; Xie et al., 2020; Agrawal et al., 2021) that seek to recover these structures often assume that the latent variables are *exogenous*, i.e., are not caused by any of the observed variables. However, this assumption is often violated in many applications. For example, in genomics, gene regulatory networks are often modeled using *transcription factories* (Stadhouders et al., 2019) as underlying latent variables with gene expression as the observable variables. These gene expressions then can have downstream impacts on other transcription factories, requiring a model that allows non-exogenous latent variables.

Contributions. In Section 2, we introduce the class of *latent factor causal models* (LFCMs), which allow for non-exogenous latent variables. Similar to prior work, this class of models prohibits direct edges between observed variables, i.e., the effect of one observed variable on another must be mediated by some latent variable. We likewise prohibit direct edges between latent variables, inducing a *bipartite* structure over the graph. Furthermore, we require that the latent variables *cluster* the observed variables, i.e., each observed variable has only a single latent parent, and that each latent variable has at least three observed children. These constraints on the model are motivated by how the DNA is organized in the cell nucleus to facilitate cell-type specific gene expression. The spatial clustering of genes in the cell nucleus facilitates their co-regulation by transcription factors (Belyaeva et al., 2017; Uhler and Shivashankar, 2017). The expression of each gene represents the observed variables, the spatial clustering of genes is unobserved, and the latent factors represent the presence of transcription factors that can e.g. turn on the expression of the co-clustered genes.

In Section 3, we establish identifiability results for LFCMs, based primarily on the tetrad representation theorem of Spirtes (2013). Based on our identifiability results, in Section 4 we propose a constraint-based method for learning the underlying graph over *both* latent and observed variables. The proposed method has three stages. In the first stage, our method identifies clusters of observed variables with the same latent parent, as well as an ordering over these clusters. The second stage merges clusters from the first stage if necessary. In the third stage, we learn edges from the observed variables to the latent variables, by testing for conditional independence with all children of each latent variable. For each stage, the constraints being checked are equivalent to multiple test statistics vanishing simultaneously, requiring the use of multiple hypothesis testing procedures which we describe in Section 4.1. Finally, in Section 5, we demonstrate the performance of our algorithm in both a completely synthetic and a semi-synthetic setting. In particular, we show that our method is capable of recovering the ground truth clustering with nearly 100% accuracy even at relatively low sample sizes. Our method also recovers the ground truth edges between observed nodes and latent nodes with higher accuracy than a baseline which does not make use of multiple hypothesis testing.

1.1. Related Work

Learning undirected graphical models with clusters. Since clusters of correlated variables are common across many disciplines, including biology (Eisen et al., 1998), economics (Bai and Wang, 2016), neuroscience (Arslan et al., 2018; Pircalabelu and Claeskens, 2020), and the behavioral sciences (van der Linden and Hambleton, 2013), several structure learning methods have been developed which encourage clustering in the estimated graphs, especially in the setting of *undirected* graphical models. For example, Tan et al. (2015) introduced the *cluster graphical lasso* method, which generalizes the traditional graphical lasso method to allow for the incorporation of known clustering information, resulting in denser estimated subgraphs over these clusters. Building on this work, Hosseini and Lee (2016) introduce the *GRAB* algorithm, which does *not* require clusters to be

known beforehand, but instead allows the clustering to be learned simultaneously to network structure. More recently, Pircalabelu and Claeskens (2020) introduced *ComGGL*, a method which also learns clusters and graph structure simultaneously, with the additional benefit of high-dimensional consistency guarantees for both cluster recovery and graph structure in sparse settings.

Latent tree models and factor analysis. Unlike in the undirected settings above, in our setting, the clusters of observed variables are explicitly assumed to be induced by latent variables. As has been observed in previous works, especially in latent tree modeling (Choi et al., 2011; Shiers et al., 2016; Drton et al., 2017; Leung and Drton, 2018) and factor analysis (Drton et al., 2007; Kummerfeld and Ramsey, 2016), these latent variables produce "signatures" or "invariants" in the distribution over the observed variables, which can be exploited for structure learning. One invariant which plays an important role in both settings is the *tetrad* $t_{ij,uv}$, a 2×2 subdeterminant of the correlation matrix which must *vanish* (i.e., equal zero) whenever i and j share a single common latent parent, but have no children. As we will see in Section 3, despite the differences in our model assumptions, tetrads also play an important role in our algorithm when identifying causal clusters.

Traditional causal discovery methods. Within the space of discovering causal models on observational data, there are two categorizations of algorithms. First, there are constraint-based methods, which rely on conditional independence testing to draw conclusions about the structure. The well-known PC-algorithm assumes *causal sufficiency*, which bars unmeasured common cause latent variables and selection variables (Spirtes et al., 2000). There also exist constraint-based methods on directed acyclic graphs with latent and selection variables, such as FCI, RFCI and their variants (Spirtes, 2001; Colombo et al., 2012). These methods all learn Markov equivalence classes of directed acyclic graphs, as represented by completed partially directed acyclic graphs (CPDAGs) in the PC-algorithm or partial ancestral graphs (PAGs) in the FCI algorithm. The second categorization of methods is score-based algorithms, such as GES (Chickering, 2002), which identify underlying structure by optimizing a well-designed score function. These methods, even those that are asymptotically correct in the presence of latent confounders, output equivalence classes of DAGs. In our work, we try to recover more complete causal information.

Learning causal models with latent variables. Existing work for structure recovery in the presence of latent variables can often by characterized by the model structures that the method performs well or poorly upon. Agrawal et al. (2021) considers the model in which latent variables are *pervasive*, influencing many observed nodes. Their method, *DeCAMFounder*, recovers the true causal structure over observed variables by applying spectral decomposition in the non-linear additive noise and pervasive confounding setting (Agrawal et al., 2021), extending the linear setting of Frot et al. (2019). A number of methods, similarly to the current work, also rely on algebraic constraints in the covariance matrix over observable variables to infer graph structure over latent variables. The BPC (Silva et al., 2006) and FOFC (Kummerfeld and Ramsey, 2016) algorithms both leverage rank constraints on the covariance matrix to cluster observed variables, then recover some structure over the inferred latent nodes corresponding to these clusters. Other algorithms, such as those proposed by Shimizu et al. (2009), Cai et al. (2019), and Xie et al. (2020), attempt to improve upon previous algorithms by restricting models to the *linear non-Gaussian* case. In our work, we do not require non-Gaussianity, instead working in the general linear acyclic model regime. Furthermore, all of the above algorithms rely on the *measurement assumption*, which requires that no observed variable is the parent of any latent variable. This assumption, however, is not satisfied in many real-world applications of graphical models with latent variables, and in our work, we attempt to recover causal structures *without* the measurement assumption.

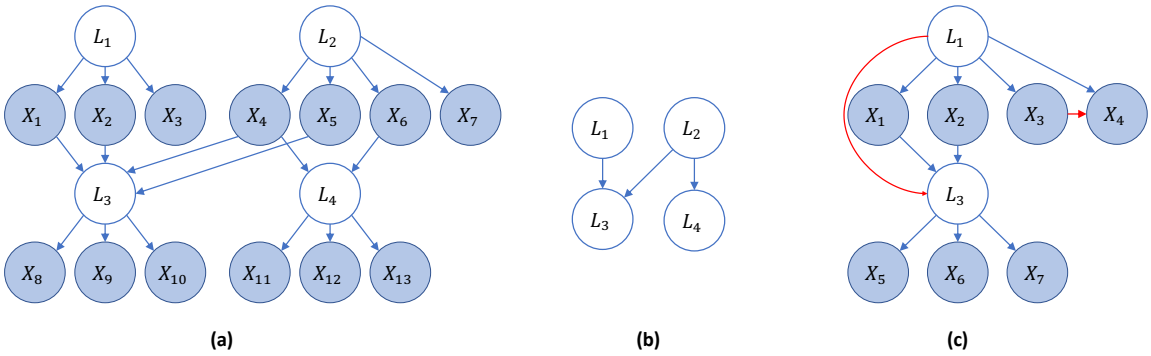


Figure 1: **(a)** $\mathcal{G}$ satisfies our model constraints. **(b)** The latent graph $L(\mathcal{G})$ for $\mathcal{G}$. **(c)** $\mathcal{G}'$ falls outside of the class of models we consider in this paper, with violations shown in red.

2. Problem Setup

We now formally define the class of models considered in this paper. A *structural causal model* (SCM) over the variables $\{X_i\}_{i=1}^p$ consists of a set of *structural assignments* of the form $X_i = f_i(X_{\text{pa}(i)}, \epsilon_i)$, and a product distribution $\mathbb{P}_\epsilon$ over mean-zero *exogenous noise* terms $\{\epsilon_i\}_{i=1}^p$. The set $\text{pa}(X_i)$ are called the *parents* of X_i, and the *causal graph* for the SCM is a graph with nodes $\{X_i\}_{i=1}^p$ and directed edges $X_j \to X_i$ for $X_j \in \text{pa}(X_i)$. We assume that the causal graph for the SCM is *acyclic*, in which case the distribution $\mathbb{P}_\epsilon$ induces a unique distribution $\mathbb{P}_X$ over $\{X_i\}_{i=1}^p$.

In this paper, we focus on a class of SCMs with restrictions on both the structural assignments and on the causal graph. First, we assume each f_i is a *linear* function, a common starting point for new methods, which has been the setting of many works (Chickering, 2002; Hauser and Bühlmann, 2012; Solus et al., 2021). Second, we assume that the causal graph is of the following form.

Definition 1 *Let $\mathcal{G}$ be a DAG over latent nodes $L_1, \ldots, L_K$ and* observed nodes *$X = \{X_1, \ldots, X_p\}$. The* clusters *of $\mathcal{G}$ are the sets $C_k = \text{ch}(L_k)$ for $k = 1, \ldots, K$. The* latent graph *for $\mathcal{G}$, denoted $L(\mathcal{G})$, is the graph over $\{L_k\}_{k=1}^K$ with an edge $k \to k'$ if and only if $X_k \to L_{k'}$ for some $X_k \in \text{ch}(L_k)$. $\mathcal{G}$ is called a* latent clustered causal model *if it satisfies the following conditions:*

(a) [Unique cluster assumption] Each observed node has exactly one latent parent.

(b) [Bipartite assumption] There are no edges between pairs of observed nodes or between pairs of latent nodes.

(c) [Triple-child assumption] Each latent node has at least 3 observed children.

(c) [Double-parent assumption] If $k \to k'$ in $L(\mathcal{G})$, then there exist two nodes $X_i, X_j \in \text{ch}(L_k)$ such that $X_i \to L_{k'}$ and $X_j \to L_{k'}$.

See Fig. 1a for an example of a graph that satisfies our model definition, and Fig. 1c for a graph that does not. The importance of each assumption for the purpose of identifying $\mathcal{G}$ will become clear in the proofs of the genericity and identifiability results presented in the next section. For example, we will see that the edge $X_3 \to X_4$ in Fig. 1c prevents the submatrix $\Sigma_{[1,2],[3,4]}$ of the covariance matrix Σ from being low rank, and thus prevents $\{X_1, X_2\}$ and $\{X_3, X_4\}$ from being clustered.

3. Identifiability

We denote the covariance matrix of our model as Σ, and given two subsets of nodes A, B, we use $\Sigma_{A,B}$ to denote the submatrix of Σ with rows in A and columns in B. Our identifiability results rely on a common generalization of d-separation, known as *trek separation*, which relates the causal graph of a SCM to the rank of submatrices of Σ. A *directed path* from node i to node j is a sequence of nodes $p_1 = i, ..., p_k = j$, such that $p_i \to p_{i+1}$ for all i from 1 to $k-1$. In this case, j is called the *sink* of the path and i is called the *source*. A *trek* in the graph $\mathcal{G}$ from i to j is an ordered pair of directed paths (P_1, P_2), such that the sink of P_1 is i and the sink of P_2 is j, and P_1, P_2 share a source k. Now, we define *trek separation*. Given four subsets A, B, C_A, C_B of nodes (note these subsets need not be disjoint), the pair (C_A, C_B) *t-separates* A and B if, for every trek (P_1, P_2) between A and B, P_1 contains a node in C_A or P_2 contains a node from C_B. Finally, the following theorem relates the notion of *t-separation* to the rank of submatrices of the covariance matrix.

Theorem 1 (Trek separation, Sullivant et al. (2010)) *Let A, B be two subset of nodes in $\mathcal{G}$. Then*

$$\textit{rank}(\Sigma_{A,B}) \leq \min\{|C_A| + |C_B| : (C_A, C_B) \textit{ t-separates } A \textit{ from } B \textit{ in } \mathcal{G}\}$$

Moreover, equality holds generically[1] *for Σ consistent with $\mathcal{G}$.*

In this paper, we only need to use information about the rank of 2 x 2 submatrices of Σ. The determinants of these matrices are commonly known as *tetrads*. In particular, we denote $t_{ij,uv} = \det(\Sigma_{[ij],[uv]}) = \Sigma_{iu}\Sigma_{jv} - \Sigma_{iv}\Sigma_{ju}$. Specializing Theorem 1, we obtain the following corollary:

Corollary 1 (Tetrad representation, Spirtes (2013)) *Suppose $A = \{X_i, X_j\}$ and $B = \{X_u, X_v\}$ are t-separated by a single node. Then $t_{ij,uv} = 0$.*

In particular, two observed nodes which share the same latent parent will be t-separated from their non-descendants. In order to use this condition for *identifying* pairs of nodes in the same cluster, we require the following genericity assumptions (see Appendix A for a proof of genericity).

Assumption 1 (Cluster tetrad faithfulness) *Suppose X_i and X_j are not in the same cluster. Then there exists some $\{u, v\}$ such that $t_{ij,uv} \neq 0$.*

Assumption 2 (Parent tetrad faithfulness) *Suppose X_i and X_j are in the same cluster, but X_i has at least one child. Then there exists some $\{u, v\}$ such that $t_{ij,uv} \neq 0$.*

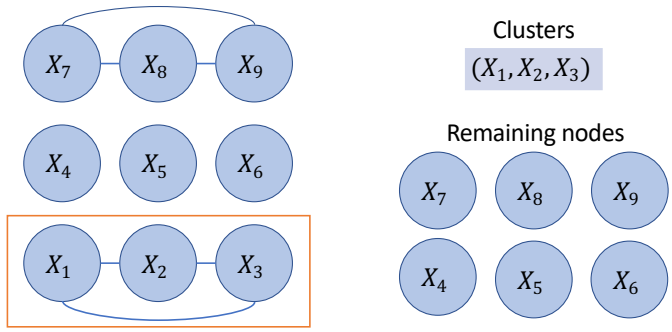
Assumption 3 (Latent adjacency faithfulness) *Suppose $X_i \to L_k$. Let $S_i = \mathrm{ch}(\mathrm{pa}(X_i)) \setminus \{i\}$ and $S' = \cup_{j \leq i} \mathrm{ch}(L_i)$. Then $\rho_{i,k|S_i,S} \neq 0$ for some $X_k \in \mathrm{ch}(L_k)$*

Remark 1 *Since causal structure learning algorithms are always run in a noisy setting,* near *violations of genericity assumptions can degrade the performance of a method, as discussed by Uhler et al. (2013). In particular, the set of parameters which violate a "strong" faithfulness condition is generally a positive measure set, extending from the measure zero set where faithfulness is violated. Fortunately for the current setting, our assumptions require the existence of only a* single *entry of the underlying statistic being far from zero. Thus, the set of parameters violating the "strong" version of our faithfulness assumption is an* intersection *of the sets of parameters for which each $t_{ij,uv}$ is near zero, resulting in a smaller set. In the present work, we will not attempt to quantify the size of this set and the resulting statistical benefits, but note these as interesting directions for future work.*

1. We say a statement holds *generically* if the set of parameters for which it does not hold has Lebesgue measure zero.

	(X_1, X_3)	...	(X_4, X_5)	(X_4, X_6)	...
(X_1, X_2)	-	...	0.09846	0.00212	...
(X_1, X_3)	-	...	0.01911	0.06772	...
(X_1, X_4)	-	...	1.14328	0.83192	...
⋮					
(X_2, X_3)	-	...	0.03328	0.02540	...
⋮					

(a)



(b) (c)

Figure 2: **Phase 1**: **(a)** Compute tetrad scores between all pairs of nodes. **(b)** For each pair of nodes, test the null hypothesis that all tetrads are zero. Construct a graph with the edge $i - j$ for any pair of nodes where we do not reject the null hypothesis. **(c)** Extract a clique from this graph to be a cluster, remove these nodes and repeat with remaining nodes.

4. Methods

Our algorithm, presented in Algorithm 1, consists of three stages. As is common in causal structure learning, we present our algorithm with implementation details of hypothesis testing abstracted away. In particular, we will assume access to two subroutines, whose implementation details will be given in Section 4.1. The first subroutine tests $H_{ci}(X_j, X_A \mid X_B)$, which denotes the null hypothesis that X_j and X_A are conditionally independent given X_B. The second subroutine tests $H_{vt}(X_A, X_B)$, which denotes the null hypothesis that all tetrads of $\Sigma_{A,B}$ vanish.

Algorithm 1 `EstimateLFCM`

Input: Data $\mathbb{X} \in \mathbb{R}^{n \times p}$.
Let π = `FindOrderedClusters`$(\mathbb{X})$
Let π = `MergeClusters`$(\mathbb{X}, \pi)$
Let $\hat{\mathcal{G}}$ = `LearnDAG`$(\mathbb{X}, \pi)$
Output: $\hat{\mathcal{G}}$

In the **first stage** (Algorithm 2, see also Fig. 2), we identify clusters of observed variables with the same latent parent. However, note that since this stage only identifies *leaves* with the same latent parent, it is not guaranteed to identify all nodes with the same latent parent. This stage simultaneously recovers an ordering over these clusters. Thus, in the **second stage** (Algorithm 3, see also Fig. 3a), we iterate over pairs of clusters output from the first stage, identify pairs of clusters with the same latent parent, and merge them, while leaving the ordering of the clusters intact. In the **third stage** (Algorithm 4, see also Fig. 3bc), we use the clustering and ordering information discovered in the previous two stages to learn a DAG over both latent and observed variables. In particular, given a node X_j in cluster C_j which comes before the cluster C_i in our ordering, we wish to determine whether X_j has an edge to the associated latent variable L_i. By Assumption 2, this can be accomplished by checking partial correlations between X_j and the nodes in C_i. These stages compose a consistent algorithm, as established in the following theorem and proven in Appendix B.

Theorem 2 *Algorithm 1 is consistent under Assumptions 1, 2, and 3.*

Next, we outline the complexity of our algorithm, using placeholders for the complexities of hypothesis tests in order to keep our results general. Let $f(d, n)$ denote the cost of performing the hypothesis test H_{vt} on d statistics from n samples, when all sufficient statistics are pre-computed.

Algorithm 2 `FindOrderedClusters`

Input: Data $\mathbb{X}$.
Initialize $R = [p]$
Initialize π as an empty list
while $|R| > 3$ **do**
  Initialize $\mathcal{G}$ as an empty graph
  for *each pair of nodes* i, j *in* R **do**
    if $H_{vt}(\{i,j\}, [p] \setminus \{i,j\})$, then add $i - j$ to $\mathcal{G}$
  end
  Let C be the largest clique in $\mathcal{G}$
  $R = R \setminus C$
  Remove C from the columns of $\mathbb{X}$
  Append C to π
end
if $|R| > 0$ **then**
  Append R to π
end
Output: An ordered clustering π

Algorithm 3 `MergeClusters`

Input: Data $\mathbb{X}$, ordered clustering π
Repeat the following until convergence:
for *Clusters* c_1 *and* c_2 *such that* $c_1 \prec c_2$ **do**
  if $H_{vt}(c_1 \cup c_2, c_1 \cup c_2)$ **then**
    Add nodes in c_2 to c_1 and delete c_2
  end
end
Output: An ordered clustering π

Algorithm 4 `LearnDAG`

Input: Data $\mathbb{X}$, ordered clustering π
Initialize $\hat{\mathcal{G}}$ as an empty graph
Add L_i to for each C_i in π
Add $L_i \to X_j$ to $\hat{\mathcal{G}}$ for each C_i in π, $X_j \in C_i$
for $X_j \prec L_i$ **do**
  Let $S = \cup_{k | L_k \prec L_i} \mathrm{ch}(L_k)$
  Add $X_j \to L_i$ to $\hat{\mathcal{G}}$ if $H_{ci}(X_j, \mathrm{ch}(L_i) \mid S)$
end
Output: DAG $\hat{\mathcal{G}}$

Let $g(p)$ denote the complexity of an algorithm used to find a clique in a graph on p nodes. While finding the *largest* clique in a graph is in general NP-hard, we can avoid this complexity, since the consistency of our algorithm does not rely on picking the largest clique at each step, only a clique of 3 or more nodes (picking larger cliques is simply a tool for improving statistical accuracy).

In the following, let M be the maximum size of any returned cluster, and let K be the number of clusters discovered by the algorithm. By definition, $M \leq p$ and $K \leq p$, so replacing these quantities by p gives complexities that are only in terms of the known problem parameters. However, such upper bounds can be highly pessimistic. If the true graph has few nodes per cluster, or a small number of latent nodes, then with enough samples, M and K will also be small, respectively.

Theorem 3 *The complexity of each algorithm is:*

(a) Algorithm 2 takes $\mathcal{O}(p^4 + p^3 f(p^2, n) + pg(p))$.

(b) Algorithm 3 takes $\mathcal{O}(p^2 M^4)$.

(c) Algorithm 4 takes $\mathcal{O}(pKM)$.

Proof (a) Computing all tetrads and their associated p-values is $\mathcal{O}(p^4)$. In each round, we perform $\mathcal{O}(p^2)$ hypothesis tests, each on $\mathcal{O}(p^2)$ statistics, so that the complexity at each round is $\mathcal{O}(p^2 f(p^2, n))$. After performing these tests, we identify the largest clique in a graph of $\mathcal{O}(p)$ nodes, so that the total run time per round is $\mathcal{O}(p^2 f(p^2, n) + g(p))$. At most p rounds are required, resulting in the stated complexity.

(b) We must check $\mathcal{O}(p^2)$ pairs of clusters for whether or not they should be merged, and the maximum size of the union of any such pair is $\mathcal{O}(M)$. To check whether $\mathcal{O}(M)$ nodes belong to the same cluster, we require a hypothesis test on $\mathcal{O}(M^4)$ statistics, so that this step takes $\mathcal{O}(p^2 f(M^4, n))$.

(c) We perform $\mathcal{O}(pK)$ hypothesis tests, each based on $\mathcal{O}(M)$ partial correlations. ■

4.1. Implementation Details

The null hypothesis H_{vt} and H_{ci} used in Algorithms 2, 3, and 4 imply that some vector-valued statistic of the covariance matrix is equal to zero. Procedures for *simultaneous hypothesis testing* are designed to (asymptotically) control the false discovery rate (FDR) of such a form of hypothesis test. In practice, we found that computing marginal p-values and performing *Sidak adjustment* Drton and Perlman (2007) yields good performance. In particular, given p-values $\{\pi_m\}_{m=1}^M$, the Sidak-adjusted p-values are

$$\pi_m^{\text{sidak}} = 1 - (1 - \pi_m)^M$$

Remark 2 *The Sidak adjustment uses only the marginal distributions of each tetrad, neglecting potentially important information about the correlations between tetrads. In contrast, the* max-T adjustment *accounts for correlations between the tested statistics by estimating their correlation matrix, and has been shown to outperform the Sidak adjustment both theoretically and in practice (Drton and Perlman, 2007; Chernozhukov et al., 2013). However, the max-T adjustment requires sampling from a potentially high-dimensional multivariate normal distribution, an operation which is $\mathcal{O}(d^3)$ for dimension d. We have found that in practice, max-T adjustment performs similarly to Sidak adjustment while taking substantially longer. Therefore, we use Sidak adjustment for our experimental results, but provide capability for max-T adjustment in our codebase.*

Given a set adjusted p-values and a significance level α, we reject the null hypothesis if *any* of the adjusted p-values are smaller than α. To test conditional independence, recall that $H_{ci}(X_j, X_A \mid X_B)$ holds in a multivariate normal if and only if the vector of partial correlations $\{\rho_{ij|B}\}_{i \in A}$ is zero. To compute p-values, we use a widely used procedure which we call the *Fisher correlation test*. First, given the sample partial correlations $\{\hat{\rho}_{ij|B}\}_{i \in A}$, we apply the *Fisher z-transformation* $\hat{z}_{ij|B} = \sqrt{n - |B| - 3}\,\text{arctanh}(\hat{\rho}_{ij|B})$. Then, we compute the two-tailed p-value of $\hat{z}_{ij|B}$ with respect to $\mathcal{N}(0,1)$, i.e., $\pi_{ij|B} = 2Q(|\hat{z}_{ij|B}|)$, where Q is the tail distribution function of $\mathcal{N}(0,1)$.

Next, to test $H_{vt}(X_A, X_B)$, we adopt the widely-used *Wishart test* to compute the p-values (Wishart, 1928; Kummerfeld and Ramsey, 2016), which we now briefly describe. First, we compute the *sample tetrads* $\hat{t}_{ij,uv} = \hat{\Sigma}_{iu}\hat{\Sigma}_{jv} - \hat{\Sigma}_{iv}\hat{\Sigma}_{ju}$ for $\{i,j\} \subset A$ and $\{u,v\} \subset B$ such that i, j, u and v are distinct. Then, we normalize each sample tetrad, dividing by an estimate of its standard deviation to obtain the z-score $\hat{z}_{ij,uv}$. Drton et al. (2008) give the following formula for the variance of sample tetrads in terms of the true covariance matrix Σ:

$$\text{Var}\left(\hat{t}_{ij,uv}\right) = n \cdot (n-1)^{-3} \cdot \left((n+2)|\Sigma_{[ij],[ij]}| \cdot |\Sigma_{[uv],[uv]}| - n|\Sigma_{[ijuv],[ijuv]}| + 3n|\Sigma_{[ij],[uv]}|\right),$$

where $|A| = \det(A)$. To estimate the variance, we use the above formula with the sample covariance $\hat{\Sigma}$ replacing Σ. Finally, we compute the two-tailed p-value of $\hat{z}_{ij,uv}$ with respect to $\mathcal{N}(0,1)$.

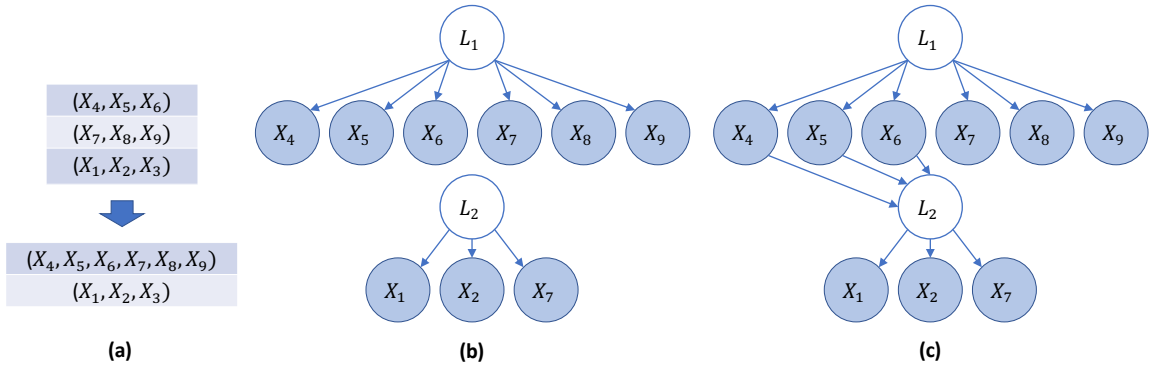


Figure 3: **Phases 2 and 3. (a)** Merge pairs of clusters based on vanishing tetrad tests. **(b)** Introduce latent nodes, and add edges from latent nodes to children. **(c)** Add parents of latent nodes based on conditional independence testing.

5. Empirical Results

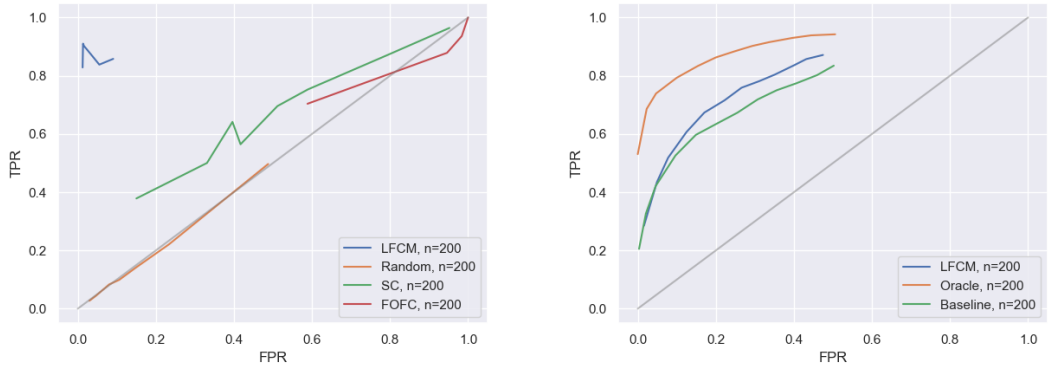
We evaluate our algorithm in two settings. First, we evaluate in a purely synthetic setting, which allows us to generate SCMs which exactly match our proposed model. Then, we evaluate in a semi-synthetic setting, modifying real data to more closely match our proposed model while demonstrating that our approach has promise in real-world biological settings.

5.1. Synthetic data experiments

We begin by briefly describing the simulation settings used for our experiments, before describing the baselines and metrics which we use for evaluation. We generate a graph with 10 latent nodes, we first sample a "latent" skeleton $L(\mathcal{G})$ over $\{1, 2, \ldots, 10\}$ from a directed Erdös-Rényi model with edge probability 0.5. Then, for each latent node L_k, we generate $c_k \sim \mathsf{Unif}(3, 6)$ children. Finally, for each edge $L_k \to L_{k'}$ in $L(\mathcal{G})$, we sample $d_{k,k'} \sim \mathsf{Unif}(2, |\operatorname{ch}(L_k)|)$, then sample $d_{k,k'}$ children of L_k. For each selected child c_k, we add the edge $c_k \to L_{k'}$, giving us a DAG $\mathcal{G}$ over both latent and observed nodes which has latent skeleton $L(\mathcal{G})$ and satisfies Definition 1.

Given this DAG, we generate a linear SCM as follows, proceeding in topological order. For each node X_j, if the node has no parents, its equation is $X_j = \epsilon_j$ for $\epsilon_j \sim \mathcal{N}(0, 1)$. If the node has parents, then for each parent X_i, we sample an "initial" weight $\tilde{w}_{ij} \sim \mathsf{Unif}([-1, -.25] \cup [.25, 1])$. Next, we describe how to normalize these weights in order to avoid the *varsortability* issue described by Reisach et al. (2021), where simulated DAGs are easy to learn because the variance of each node tends to increase according to the topological order. Given these initial weights, we simulate B "parental contributions" $\mu_j^{(b)} = \sum_{i \in \operatorname{pa}_{\mathcal{G}}(j)} \tilde{w}_{ij} X_i^{(b)}$ for $X^{(b)}$ sampled from the linear SEM defined over $i < j$. The sample variance $\hat{\sigma}_j$ of $\mu_j^{(b)}$ serves as an estimate for the variance that the parents of j will contribute to X_j. Finally, we ensure that X_j has variance 1 and that half of its variance is contributed by its parents by setting the final weights as $w_{ij} = (2\hat{\sigma}_j)^{-1/2} \tilde{w}_{ij}$ and $\epsilon_j \sim \mathcal{N}(0, 1/2)$.

Accuracy of learning clusters. In our first set of experiments, we evaluate the accuracy of the learned clusters. To measure the accuracy over the learned clustering compared to the underlying clustering, we use the following criteria: the pair (X_i, X_j) is a *true positive* if X_i and X_j are in the same underlying cluster and are in the same learned cluster, the pair is a *false positive* if X_i and X_j



(a) **Cluster recovery performance.** (b) **Edge recovery performance.**

Figure 4: **Performance on synthetic data.** The first two phases of our algorithm almost perfectly recover the ground truth clusters, while the third phase of our algorithm demonstrates the utility of multiple hypothesis testing for recovering edges between observed nodes and latent nodes.

are not in the same underlying cluster but are in the same learned cluster, and so on. We generate 50 different SEMs via the process described above, and from each SEM we generate $n = 200$ samples. We run our algorithm using significance levels ranging from .05 and .5. The results are shown in Fig. 4a. Due to interactions between the hypothesis tests used by our algorithm (denoted "LFCM", shown in blue), the ROC curve is highly non-monotonic over larger ranges of values, so that the curve occupies only a small range of the plot, though it clearly drastically outperforms the competing methods, achieving almost perfect performance. In particular, we consider two baselines. First, we compare to *spectral clustering* (denoted "SC", shown in green), as implemented in `sklearn` in Python, a widely used clustering technique in genomics (Higham et al., 2007), with a varying number of estimated clusters from 2 to 30. Second, we compare to the FindOneFactorCluster algorithm of Kummerfeld and Ramsey (2016) (denoted "FOFC", shown in red). We found that spectral clustering performs slightly better than random guessing, while FOFC performs about the same as random guessing, reflecting the drastic deviation from the measurement assumption on which it relies. Finally, we verified that randomly picking K clusters of equal size (denoted "Random", shown in orange), for K varying from 2 to 30, matched the diagonal random guessing line.

Accuracy of learning edges from observed nodes to latent nodes. In our second set of experiments, we evaluate the accuracy of learning the edges from observed nodes to latent nodes, when the true clusters and their ordering is known. In particular, for $L_i \prec L_j$ in the ordering, and $X_i \in \mathrm{ch}(L_i)$, the pair (X_i, L_j) is considered a *true positive* if $X_i \to L_j$ in the true LFCM as well as in the estimated LFCM, a *false positive* if it is not in the true LFCM but does appear in the estimated LFCM, and so on. In Fig. 4b, we compare the third phase of our algorithm (denoted "LFCM", shown in blue) to a *baseline* which simply uses a single child of each latent node for the conditional independence test (denoted "Baseline", shown in green), as well as an *oracle* which is able to observe the values of the latent nodes and is thus infeasible (denoted "Oracle", shown in orange). As expected, our algorithm does not perform as well as this unrealizable case, but still performs significantly better than random (the diagonal line) and noticeably better than the baseline.

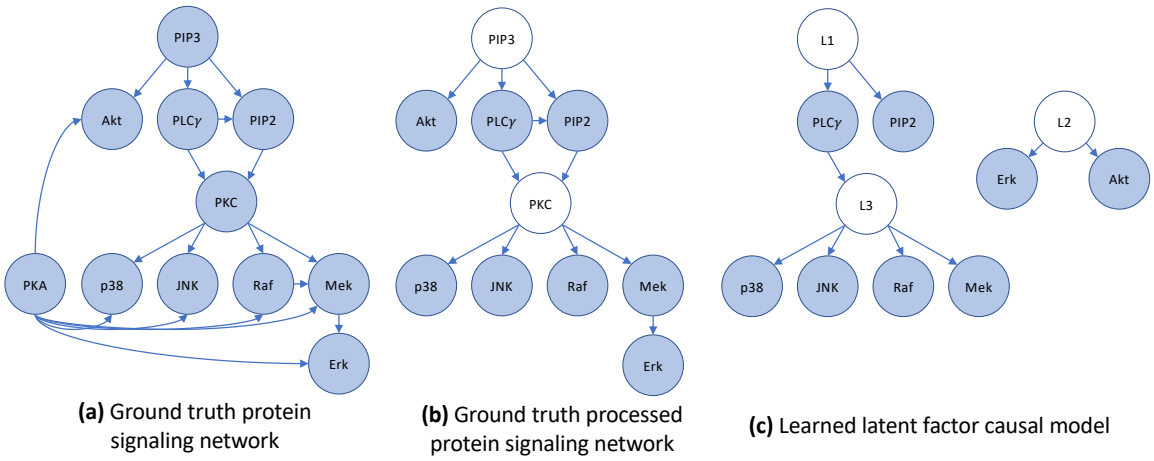


Figure 5: **Learning a latent factor causal model for protein signaling**. Our recovered model in **(c)** nearly captures the ground truth network in **(b)**.

5.2. Semi-synthetic experiments on protein signaling data

In this section, we demonstrate the applicability of our method to a real-world dataset in a semi-synthetic setting. The Sachs protein mass spectroscopy dataset (Sachs et al., 2005) is a widely used benchmark for causal discovery, in part due to the existence of a commonly accepted ground truth network over the 11 measured protein expression values, shown in Fig. 5a. We use the 1,755 "observational" samples, where the experimental conditions involve only perturbing receptor enzymes, and not any signaling molecules, as described in Wang et al. (2017). To make the ground truth network more similar to a latent factor causal model, we perform three data-processing steps: (1) we "condition" on PKA, by regressing it out of the dataset, (2) we "remove" the direct effect of Raf on Mek, and (3) we "marginalize" out PIP3 and PKC by removing the corresponding columns from the dataset. The resulting network is shown in Fig. 5b.

Running our method with significance level $\alpha = 0.01$ for H_{vt} and $\alpha = 0.1$ for H_{ci}, we obtain the network shown in Fig. 5c. The clustering by our algorithm closely matches the clustering (*Akt, PLCγ, PIP2*), (*p38, JNK, Raf, Mek, Erk*) induced by the true network, with the exception that *Akt* from the first cluster and *Erk* from the second cluster are pulled out into a cluster with one another, which may indicate that the effect of PKA on Akt and Erk cannot be completely removed using a purely linear approach. The ordering between the clusters (*PLCγ, PIP2*) and (*p38, JNK, Raf, Mek*) is preserved, but the edge $PIP2 \to L3$ is missing.

6. Discussion

In this paper, we introduce a method (Algorithm 1) for learning *latent factor causal models* (LFCMs), a novel, biologically-motivated class of causal models with latent variables. We showed that these models are identifiable in the linear setting using rank constraints on submatrices of the covariance matrix, and that our method provides a consistent estimator for these models. We also showed that our method outperforms existing clustering algorithms on synthetic data, and almost perfectly recovers a widely-accepted ground truth network in a semi-synthetic biological setting. These results serve as a proof-of-concept, suggesting that our algorithm may be able to shed biological insight

on the problem of identifying the spatial clustering of genes in the cell nucleus given data on the expression of the genes. Interestingly, since it is possible (although expensive) to measure the 3D organization of the genome in the cell nucleus (Lieberman-Aiden et al., 2009), there is a meaningful avenue to validate our method on the important biological application of connecting 3D genome organization with gene expression (Uhler and Shivashankar, 2017). We conclude with a discussion of the limitations of our model, which suggests a number of other directions for future work.

Limitations. The latent factor causal model (LFCM) class considered in this paper has two obvious limitations. First, we make the strong parametric assumption of a linear Gaussian SEM. While many nonparametric conditional independence tests have been proposed (Gretton et al., 2007; Zhang et al., 2011), we are not aware of nonparametric tests for shared latent factors that would generalize H_{vt}. Thus, extending our algorithm to a nonparametric setting would require development of such tests. In particular, generalizing tetrad constraints to the nonlinear setting is an interesting direction for future research. Second, we make two strong structural assumptions. The "unique cluster" assumption is well-motivated by our biological setting of interest, and is likely the easiest assumption to remove since Theorem 1 already provides a generalization of the rank constraint we leverage. Indeed, generalized rank constraints have already been explored in prior work on factor analysis (Drton et al., 2007; Kummerfeld et al., 2014). The "bipartite assumption" has two components which may be separately examined. First, the assumption that there are no edges between observed variables is most reasonable for systems such as gene regulatory networks where a different, unobserved entity class (in this case, proteins) mediates *all* interactions between the observed variables (i.e., genes). This assumption may also be expendable, for instance by allowing for a *small number* of edges between observed variables, akin to the low-rank plus sparse literature in previous work on learning with exogenous latent variables (Frot et al., 2019; Agrawal et al., 2021). Similarly, existing techniques (Cai et al., 2019; Xie et al., 2020) may help to eliminate the assumption that there are no edges between latent variables.

References

Raj Agrawal, Chandler Squires, Neha Prasad, and Caroline Uhler. The decamfounder: Non-linear causal discovery in the presence of hidden variables. *arXiv preprint arXiv:2102.07921*, 2021.

Salim Arslan, Sofia Ira Ktena, Antonios Makropoulos, Emma C Robinson, Daniel Rueckert, and Sarah Parisot. Human brain mapping: A systematic comparison of parcellation methods for the human cerebral cortex. *NeuroImage*, 170:5–30, 2018.

Jushan Bai and Peng Wang. Econometric analysis of large factor models. *Annual Review of Economics*, 8:53–80, 2016.

Anastasiya Belyaeva, Saradha Venkatachalapathy, Mallika Nagarajan, G.V. Shivashankar, and Caroline Uhler. Network analysis identifies chromosome intermingling regions as regulatory hotspots for transcription. *PNAS*, 114(52):13714–13719, 2017.

Daniel Bernstein, Basil Saeed, Chandler Squires, and Caroline Uhler. Ordering-based causal structure learning in the presence of latent variables. In *International Conference on Artificial Intelligence and Statistics*, pages 4098–4108. PMLR, 2020.

Jr Blalock. *Causal models in the social sciences*. Routledge, 2017.

Ruichu Cai, Feng Xie, Clark Glymour, Zhifeng Hao, and Kun Zhang. Triad constraints for learning causal structure of latent variables. *Advances in Neural Information Processing Systems*, 32: 12883–12892, 2019.

Victor Chernozhukov, Denis Chetverikov, and Kengo Kato. Gaussian approximations and multiplier bootstrap for maxima of sums of high-dimensional random vectors. *The Annals of Statistics*, 41 (6):2786–2819, 2013.

David Maxwell Chickering. Optimal structure identification with greedy search. *Journal of machine learning research*, 3(Nov):507–554, 2002.

Myung Jin Choi, Vincent YF Tan, Animashree Anandkumar, and Alan S Willsky. Learning latent tree graphical models. *Journal of Machine Learning Research*, 12:1771–1812, 2011.

Diego Colombo, Marloes H Maathuis, Markus Kalisch, and Thomas S Richardson. Learning high-dimensional directed acyclic graphs with latent and selection variables. *The Annals of Statistics*, pages 294–321, 2012.

Mathias Drton and Michael D Perlman. Multiple testing and error control in gaussian graphical model selection. *Statistical Science*, 22(3):430–449, 2007.

Mathias Drton, Bernd Sturmfels, and Seth Sullivant. Algebraic factor analysis: tetrads, pentads and beyond. *Probability Theory and Related Fields*, 138(3-4):463–493, 2007.

Mathias Drton, Hélène Massam, and Ingram Olkin. Moments of minors of wishart matrices. *The Annals of Statistics*, 36(5):2261–2283, 2008.

Mathias Drton, Shaowei Lin, Luca Weihs, and Piotr Zwiernik. Marginal likelihood and model selection for gaussian latent tree and forest models. *Bernoulli*, 23(2):1202–1232, 2017.

Michael B Eisen, Paul T Spellman, Patrick O Brown, and David Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863–14868, 1998.

Nir Friedman, Michal Linial, Iftach Nachman, and Dana Pe'er. Using bayesian networks to analyze expression data. *Journal of computational biology*, 7(3-4):601–620, 2000.

Benjamin Frot, Preetam Nandy, and Marloes H Maathuis. Robust causal structure learning with some hidden variables. *Journal of the Royal Statistical Society*, 2019.

Arthur Gretton, Kenji Fukumizu, Choon Hui Teo, Le Song, Bernhard Schölkopf, Alexander J Smola, et al. A kernel statistical test of independence. In *Nips*, volume 20, pages 585–592. Citeseer, 2007.

Alain Hauser and Peter Bühlmann. Characterization and greedy learning of interventional markov equivalence classes of directed acyclic graphs. *The Journal of Machine Learning Research*, 13 (1):2409–2464, 2012.

Christina Heinze-Deml, Marloes H Maathuis, and Nicolai Meinshausen. Causal structure learning. *Annual Review of Statistics and Its Application*, 5:371–391, 2018.

Desmond J Higham, Gabriela Kalna, and Milla Kibble. Spectral clustering and its use in bioinformatics. *Journal of computational and applied mathematics*, 204(1):25–37, 2007.

Mohammad Javad Hosseini and Su-In Lee. Learning sparse gaussian graphical models with overlapping blocks. In *Advances in Neural Information Processing Systems*, pages 3808–3816, 2016.

Erich Kummerfeld and Joseph Ramsey. Causal clustering for 1-factor measurement models. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1655–1664, 2016.

Erich Kummerfeld, Joe Ramsey, Renjie Yang, Peter Spirtes, and Richard Scheines. Causal clustering for 2-factor measurement models. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 34–49. Springer, 2014.

Dennis Leung and Mathias Drton. Algebraic tests of general gaussian latent tree models. *arXiv preprint arXiv:1810.11526*, 2018.

Erez Lieberman-Aiden, Nynke L. van Berkum, Louise Williams, Maxim Imakaev, Tobias Ragoczy, Agnes Telling, Ido Amit, Bryan R. Lajoie, Peter J. Sabo, Michael O. Dorschner, Richard Sandstrom, Bradley Bernstein, M.A. Bender, Mark Groudine, Andreas Gnirke, John Stamatoyannopoulos, Leonid A. Mirny, Eric S. Lander, and Job Dekker. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science*, 326:289–293, 2009.

Preetam Nandy, Alain Hauser, and Marloes H Maathuis. High-dimensional consistency in score-based and hybrid structure learning. *The Annals of Statistics*, 46(6A):3151–3183, 2018.

Jonas Peters and Peter Bühlmann. Identifiability of gaussian structural equation models with equal error variances. *Biometrika*, 101(1):219–228, 2014.

Eugen Pircalabelu and Gerda Claeskens. Community-based group graphical lasso. *J. Mach. Learn. Res.*, 21:64–1, 2020.

Alexander G Reisach, Christof Seiler, and Sebastian Weichwald. Beware of the simulated dag! varsortability in additive noise models. *arXiv preprint arXiv:2102.13647*, 2021.

Thomas Richardson and Peter Spirtes. Ancestral graph markov models. *The Annals of Statistics*, 30(4):962–1030, 2002.

James M Robins, Miguel Angel Hernan, and Babette Brumback. Marginal structural models and causal inference in epidemiology, 2000.

Karen Sachs, Omar Perez, Dana Pe'er, Douglas A Lauffenburger, and Garry P Nolan. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721): 523–529, 2005.

Nathaniel Shiers, Piotr Zwiernik, John AD Aston, and James Q Smith. The correlation space of gaussian latent tree models and model selection without fitting. *Biometrika*, 103(3):531–545, 2016.

Shohei Shimizu, Patrik O Hoyer, Aapo Hyvärinen, Antti Kerminen, and Michael Jordan. A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7(10), 2006.

Shohei Shimizu, Patrik O Hoyer, and Aapo Hyvärinen. Estimation of linear non-gaussian acyclic models for latent factors. *Neurocomputing*, 72(7-9):2024–2027, 2009.

Ricardo Silva, Richard Scheines, Clark Glymour, Peter Spirtes, and David Maxwell Chickering. Learning the structure of linear latent variable models. *Journal of Machine Learning Research*, 7 (2), 2006.

Liam Solus, Yuhao Wang, and Caroline Uhler. Consistency guarantees for greedy permutation-based causal inference algorithms. *Biometrika*, 2021.

Peter Spirtes. An anytime algorithm for causal inference. In *International Workshop on Artificial Intelligence and Statistics*, pages 278–285. PMLR, 2001.

Peter Spirtes, Clark N Glymour, Richard Scheines, and David Heckerman. *Causation, prediction, and search*. MIT press, 2000.

Peter L Spirtes. Calculation of entailed rank constraints in partially non-linear and cyclic models. *arXiv preprint arXiv:1309.7004*, 2013.

Ralph Stadhouders, Guillaume J Filion, and Thomas Graf. Transcription factors and 3d genome conformation in cell-fate decisions. *Nature*, 569(7756):345–354, 2019.

Seth Sullivant, Kelli Talaska, and Jan Draisma. Trek separation for gaussian graphical models. *The Annals of Statistics*, 38(3):1665–1685, 2010.

Kean Ming Tan, Daniela Witten, and Ali Shojaie. The cluster graphical lasso for improved estimation of gaussian graphical models. *Computational statistics & data analysis*, 85:23–36, 2015.

Caroline Uhler and G.V. Shivashankar. Regulation of genome organization and gene expression by nuclear mechanotransduction. *Nature Reviews Molecular Cell Biology*, 18:717–727, 2017.

Caroline Uhler, Garvesh Raskutti, Peter Bühlmann, and Bin Yu. Geometry of the faithfulness assumption in causal inference. *The Annals of Statistics*, pages 436–463, 2013.

Wim J van der Linden and Ronald K Hambleton. *Handbook of modern item response theory*. Springer Science & Business Media, 2013.

Yuhao Wang, Liam Solus, Karren D. Yang, and Caroline Uhler. Permutation-based causal inference algorithms with interventions. In *Neural Information Processing Systems*, volume 31, 2017.

John Wishart. Sampling errors in the theory of two factors. *British Journal of Psychology*, 19(2): 180, 1928.

Feng Xie, Ruichu Cai, Biwei Huang, Clark Glymour, Zhifeng Hao, and Kun Zhang. Generalized independent noise condition for estimating latent variable causal graphs. *arXiv preprint arXiv:2010.04917*, 2020.

Kun Zhang, Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. Kernel-based conditional independence test and application in causal discovery. In *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, pages 804–813, 2011.