

Causal Structure Discovery from Distributions Arising from Mixtures of DAGs

Basil Saeed¹ Snigdha Panigrahi² Caroline Uhler^{1,3}

Abstract

We consider distributions arising from a mixture of causal models, where each model is represented by a directed acyclic graph (DAG). We provide a graphical representation of such mixture distributions and prove that this representation encodes the conditional independence relations of the mixture distribution. We then consider the problem of structure learning based on samples from such distributions. Since the mixing variable is latent, we consider causal structure discovery algorithms such as FCI that can deal with latent variables. We show that such algorithms recover a “union” of the component DAGs and can identify variables whose conditional distribution across the component DAGs vary. We demonstrate our results on synthetic and real data showing that the inferred graph identifies nodes that vary between the different mixture components. As an immediate application, we demonstrate how retrieval of this causal information can be used to cluster samples according to each mixture component.

1. INTRODUCTION

Determining causal structure from data is a central task in many applications. (Friedman et al., 2000; Robins et al., 2000; Heckerman et al., 1995) Causal structure can be modeled using a *directed acyclic graph* (DAG), where the vertices of the graph represent the variables of interest, and the directed edges represent the direct causal effects between these variables (Pearl, 2009). Assuming that the generating distribution of the data factors according to the DAG structure provides a way to relate the conditional independence relations in the distribution to separation statements in the

DAG (known as *d-separation*) through the *Markov property* (Lauritzen, 1996). When not all variables of interest can be measured, DAGs are not sufficient to represent the observed distribution, since latent variables may introduce confounding effects between the observed variables. Instead, a family of mixed graphs known as a *maximal ancestral graphs* (MAG) can be used to model the observed variables by depicting the presence of latent confounders between pairs of variables through bidirected edges (Richardson and Spirtes, 2002).

With respect to learning the causal graph from data on the nodes, the most ubiquitous methods infer d-separation relations by estimating conditional independence relations from the data; examples include the PC and GSP algorithms in the fully observed setting, and the FCI algorithm in the presence of latent variables (Spirtes et al., 2000; Solus et al., 2017; Colombo et al., 2012). These algorithms require the *faithfulness assumption*, which asserts that every conditional independence relation in the distribution corresponds to a d-separation relation in the graph. Note that even under this assumption, the causal graph is in general not fully identifiable from observational data; it can in general only be identified up to its *Markov equivalence class* (Spirtes et al., 2000).

In various applications, data used for causal structure discovery is *heterogeneous* in that it stems from different causal models on the same set of variables. This is relevant for example in biomedical applications, where the goal is to learn a gene regulatory network based on gene expression data from a disease that consists of multiple not well characterized subtypes (as is the case for many neurological diseases). In such scenarios, the samples stem from a mixture of different causal models on the same set of variables, and the causal effects of the mixture distribution can in general not be faithfully represented by a single DAG. Furthermore, a single DAG inferred from such samples cannot identify differences between the component DAGs in the mixture, which may be critical for personalized biomedical interventions, and may lead to flawed conclusions downstream.

In this work, we consider distributions arising as mixtures of causal DAGs. Our main contributions are as follows:

- We introduce the *mixture graph* to represent such mix-

¹Laboratory for Information and Decision Systems and Institute for Data, Systems and Society, Massachusetts Institute of Technology, Cambridge, MA, USA ²Department of Statistics, University of Michigan, Ann Arbor, MI, USA ³Department of Biosystems Science and Engineering, ETH Zurich, Switzerland. Correspondence to: Caroline Uhler <cuhler@mit.edu>.

ture distributions. We prove that this graph encodes the conditional independence relations in the mixture distribution through separation statements (Theorem 3.2) and show that the separation statements in every such graph can be realized by independence relations in some mixture distribution (Proposition 3.5).

- We introduce the *union graph*, a graph defined from the mixture graph. We prove that, under a faithfulness assumption and an ordering assumption on the DAGs in the mixture, the FCI algorithm applied to data from mixtures of DAGs outputs the union graph (Theorem 4.4).
- We prove that the union graph can identify variables whose conditional distribution across the component DAGs changes (Proposition 4.6), and we demonstrate the implication of this result for identifying critical nodes and for clustering samples according to their mixture component on synthetic data and real genomics data (Section 5).

2. PRELIMINARIES & RELATED WORK

2.1. Graphical representations: DAGs and MAGs

In this paper, we consider two types of graphs: *directed acyclic graphs* (DAGs) and *mixed graphs* with directed ($\rightarrow$) and bidirected ($\leftrightarrow$) edges. We denote the former by $\mathcal{D} = (V, E)$ and the latter by $\mathcal{M} = (V, D, B)$, where V denotes the set of nodes, and D and B the set of directed and bidirected edges, respectively. A mixed graph is said to be *ancestral* if it has no directed cycles, and whenever there is a bidirected edge $u \leftrightarrow v$, then there is no directed path from u to v (Richardson and Spirtes, 2002). While ancestral graphs have been defined more generally to allow also for undirected edges, in this work we will only make use of graphs with directed and bidirected edges.

Throughout, we will use the notation $\text{ch}_{\mathcal{M}}(v)$, $\text{pa}_{\mathcal{M}}(v)$ and $\text{an}_{\mathcal{M}}(v)$ to denote the children, parents and ancestors, respectively, of a node v in the graph $\mathcal{M}$. Furthermore, we use the standard definitions of *path* and *directed path* in a graph; for these definitions, see e.g. Lauritzen (1996). We will use the notation $v \leftrightarrow_{\mathcal{M}} u$ as a shorthand to denote “the edge $v \leftrightarrow u$ between nodes u, v in $\mathcal{M}$ ”, and use similar notations for other types of edges.

The standard notions of d-separation from DAGs can be generalized to ancestral graphs by accounting for the new possible ways to obtain a collider from bidirected edges (Richardson and Spirtes, 2002). In ancestral graphs, unlike in DAGs, it is possible to have a pair of nodes that are not adjacent, but cannot be d-separated given any subset of nodes. An ancestral graph where such pairs of nodes do not occur is called *maximal*, and a non-maximal ancestral graph can be made maximal by adding a bidirected edge

between any such pair of vertices. An ancestral graph that is maximal is called a *Maximal Ancestral Graph* (MAG) (Richardson and Spirtes, 2002).

Ancestral graphs are a useful representation of DAGs with latent variables. Specifically, Richardson and Spirtes (2002) showed that given a DAG $\mathcal{D} = (V \cup L, E)$, with observed nodes V and unobserved nodes L , satisfying a set of d-separation statements of the form “ A d-separated from B given C ” for disjoint $A, B, C \subseteq V$, then there exists an ancestral graph $\mathcal{M} = (V, D, B)$ with the same d-separation statements, called the *marginal ancestral graph of $\mathcal{D}$ with respect to L* . Sadeghi et al. (2013) gave a local graphical criterion to construct the marginal ancestral graph of a DAG $\mathcal{D}$. Throughout our paper, we will make use of this in the special case where L consists of a single node of in-degree 0. The specialization of Sadeghi’s algorithm to this case is provided in Algorithm 1.

Algorithm 1: Construction of the marginal ancestral graph

Input: DAG $\mathcal{D} = (V \cup \{y\}, E)$, where y has in-degree 0.

Output: the marginal ancestral graph of $\mathcal{D}$ w.r.t. y .

- (0) Initialize $D = \emptyset, B = \emptyset$
 - (1) For $u, v \in \text{ch}_{\mathcal{D}}(y)$: add $u \leftrightarrow v$ to B .
 - (2) For t, u, v such that $(t \rightarrow u) \in E$ and $(u \leftrightarrow v) \in B$: if $u \in \text{an}_{\mathcal{D}}(v)$, then add $t \rightarrow v$ to D .
 - (3) For u, v such that $u \leftrightarrow v \in B$: if $u \in \text{an}_{\mathcal{D}}(v)$, then remove $u \leftrightarrow v$ from B and add $u \rightarrow v$ to D .
 - (4) Return the ancestral graph $\mathcal{M} = (V, D, B)$.
-

Although in general the marginal ancestral graph constructed using Sadeghi’s local criterion is not guaranteed to be maximal, the relevant restriction considered in this paper, i.e. when L consists of a single node with in-degree 0, is always a MAG. This is stated in the following proposition; a proof is provided in section A of the appendix.

Proposition 2.1. *The output of Algorithm 1 is a MAG.*

2.2. Markov Properties

Given a graph $\mathcal{M}$ with vertices V , we associate to each vertex $v \in V$ a random variable X_v and denote the joint distribution of $X_V := (x_v : v \in V)$ by p_{X_V} . The *Markov property* associates missing edges in $\mathcal{M}$ with conditional independence statements in p_{X_V} : a distribution p_{X_V} is said to satisfy the Markov property with respect to a graph $\mathcal{M}$ if for any disjoint $A, B, C \subseteq V$ such that A and B are d-separated given C in $\mathcal{M}$, it holds that $X_A \perp\!\!\!\perp X_B \mid X_C$ in p_{X_V} (Lauritzen, 1996). For DAGs, an equivalent condition to the Markov property is for p_{X_V} to factorize as

$$p_{X_V}(x_V) = \prod_{v \in V} p(x_v | x_{\text{pa}_{\mathcal{M}}(v)});$$

see Lauritzen (1996). Considering also latent variables L , Richardson and Spirtes (2002) showed that given a distribution p_{X_V, X_L} that is Markov with respect to a DAG $\mathcal{D} = (V \cup L, E)$, the distribution $p_{X_V}(x_V) = \sum_{x_L} p_{X_V, X_L}(x_V, x_L)$ is Markov with respect to the marginal ancestral graph of $\mathcal{D}$ with respect to L .

It is possible for two different DAGs $\mathcal{D}_1, \mathcal{D}_2$ over the same vertex set to satisfy the same set of d-separation statements. In this case, $\mathcal{G}_1$ and $\mathcal{G}_2$ are said to be *Markov equivalent*, and the set of all DAGs that are Markov equivalent to a DAG $\mathcal{D}$ is called the *Markov Equivalence Class of $\mathcal{D}$* . These definitions trivially extend to MAGs. The Markov equivalence class of a MAG can be represented by a *partial ancestral graph (PAG)*: the edges in such a graph have three types of tips: arrowheads ($\leftarrow$), tails ($-$) and circles $\circ-$, where arrowhead (tail) signifies that this arrowhead exists in all graphs in the Markov equivalence class (Zhang, 2008).

2.3. Causal Structure Discovery

The goal of structure learning is to recover the graph $\mathcal{D}$ or $\mathcal{M}$ from data generated from the distribution p_{X_V} . The Markov condition is not sufficient for this task and a common assumption is the so-called *faithfulness assumption* which states that for any disjoint $A, B, C \subseteq V$, it holds that A and B are d-separated given C whenever $X_A \perp\!\!\!\perp X_B \mid X_C$ in p_{X_V} (Spirtes et al., 2000). The faithfulness assumption allows making inference about the structure of $\mathcal{D}$ or $\mathcal{M}$ from conditional independence tests on the data. Various algorithms have been proposed for this task that are provably consistent, such as the PC, GES or GSP algorithms for learning DAGs (Spirtes et al., 2000; Chickering, 2002; Solus et al., 2017), and the FCI algorithm for learning MAGs (Colombo et al., 2012). Note that even under the faithfulness assumption, it is in general only possible to retrieve the Markov equivalence class of a graph $\mathcal{D}$ or $\mathcal{M}$ from data. In fact, this is the output of the above algorithms. For example, FCI in general does not return a specific MAG, but a PAG representing a Markov equivalence class of MAGs.

2.4. Causal Inference from Mixtures of DAGs

While the problem of learning appropriate representations from data generated from mixtures of DAGs arises in various applications, there exists only little work on theory and methodology in this direction. Spirtes (1994) investigated the conditional independence properties of such mixture distributions; he defined a cyclic graphical model and proved that the mixture distribution is Markov with respect to it. However, this graph does not capture the full set of conditional independence relations for any reasonable mixture. In fact, as we discuss later, this graph is similar to the union graph defined in section 4, which also only provides partial information about the structure of the component DAGs. To

capture the full set of independences in the mixture distribution, a representation sparser than that of (Spirtes, 1994) is necessary. Strobl (2019a;b) built on this work to define a sparser graph, called the *mother graph*. However, we provide examples in section B of the appendix showing that the Markov condition in general does not hold for this graph, i.e., there can be d-separation statements in the graph that do not correspond to conditional independence relations in the mixture distribution. Finally, Thiesson et al. (2013) suggested a heuristic approach based on the Expectation-Maximization (EM) algorithm to learn the component DAGs from the mixture data, when the expectation in the E-step can be computed, as is the case for Gaussian distributions.

3. MIXTURE DAG AND MARKOV PROPERTY

In this section, we provide our first main result: after formally introducing distributions that arise as mixtures of DAGs, we define the *mixture DAG* and prove in Theorem 3.2 and Proposition 3.5 that it is a valid representation of the model, i.e., the DAG encodes the conditional independence relations of the mixture distributions. More precisely, not only is the Markov condition satisfied (i.e., all separation statements in the mixture DAG correspond to conditional independence relations in the mixture distribution), but in addition, every mixture DAG is also realizable by a mixture distribution (meaning that the mixture DAG cannot be made sparser without losing the Markov property).

3.1. Mixture of Causal DAGs

To introduce the mixture model, we consider K DAGs $\{\mathcal{D}^{(1)}, \dots, \mathcal{D}^{(K)}\}$ with $\mathcal{D}^{(j)} = (V^{(j)}, E^{(j)})$ such that $V = V^{(j)}$ for all $j \in \{1, \dots, K\}$, i.e., these K DAGs are defined on the *same* set of vertices.

Associated with each component DAG $\mathcal{D}^{(j)}$ is a random vector X_V whose joint distribution we denote by $p^{(j)}(x_V)$. Let V_{INV} denote the set of nodes that are *invariant* across the K component DAGs, i.e., nodes whose conditional distribution in the factorization does not vary across the K component DAGs; that is

$$V_{\text{INV}} = \left\{ v \in V : p^{(j)}(x_v | x_{\text{pa}_{\mathcal{D}^{(j)}}(v)}) = p^{(k)}(x_v | x_{\text{pa}_{\mathcal{D}^{(k)}}(v)}) \right. \\ \left. \text{for all } j, k \in \{1, 2, \dots, K\} \right\}. \quad (1)$$

Assuming that each distribution $p^{(j)}(x_V)$ admits a factorization according to DAG $\mathcal{D}^{(j)}$, we then obtain:

$$p^{(j)}(x_V) = \prod_{v \in V \setminus V_{\text{INV}}} p^{(j)}(x_v | x_{\text{pa}_{\mathcal{D}^{(j)}}(v)}) \prod_{v \in V_{\text{INV}}} p^{(j)}(x_v | x_{\text{pa}_{\mathcal{D}^{(j)}}(v)}) \\ = \prod_{v \in V \setminus V_{\text{INV}}} p^{(j)}(x_v | x_{\text{pa}_{\mathcal{D}^{(j)}}(v)}) \prod_{v \in V_{\text{INV}}} p^{(1)}(x_v | x_{\text{pa}_{\mathcal{D}^{(1)}}(v)})$$

for all $j = 1, 2, \dots, K$, i.e., each distribution decouples into a component over the variables associated with the invariant nodes V_{INV} that remains the same across all K distributions, and a varying component over the remaining variables which may differ across the K DAGs.

Let J be a discrete variable taking values in $\{1, \dots, K\}$ with probabilities $p_J(j)$ for each $j \in \{1, \dots, K\}$. Defining a joint distribution p_μ over $X_V \cup J$ by

$$p_\mu(x_V, j) := p_J(j) \cdot p^{(j)}(x_V), \quad (2)$$

this joint distribution satisfies $p^{(j)}(x_V) = p_\mu(x_V | J = j)$ and the observed mixture distribution is obtained by marginalizing p_μ over the unobserved index variable J . With a slight abuse of notation, we denote the resulting mixture distribution also by p_μ . Given samples from this distribution, i.e., without knowledge of the membership of each sample to its component DAG, in this paper, we analyze what can still be inferred regarding the structure of the K component DAGs.

3.2. Mixture DAG and Markov Property

We now present the *mixture DAG*, a DAG that is representative of the independence relations induced amongst the observed variables after marginalizing over the index variable J in (2). Denoting the number of vertices in V by $|V|$, then the mixture DAG is a graph on $K \cdot |V| + 1$ nodes constructed by placing the K component DAGs next to each other (this gives rise to a DAG on $K \cdot |V|$ nodes) and using an additional node to represent the mixture. We now provide the precise definition.

Definition 3.1 (Mixture DAG). *Let $v^{(j)}$ denote vertex v in DAG j and let $[V] := \cup_{1 \leq j \leq K} V^{(j)}$ denote the vertices of the K component DAGs. The mixture DAG, denoted by $\mathcal{D}_\mu$, has nodes $[V] \cup \{y\}$ and edges E_μ consisting of edges in each component DAG, namely*

$$\bigcup_{j=1}^K \left\{ v^{(j)} \rightarrow \tilde{v}^{(j)} : v, \tilde{v} \in V, v \rightarrow \tilde{v} \in E^{(j)} \right\}$$

and additional edges from node y to particular nodes in $[V]$, namely whose incoming directed edges are not the same across all component DAGs, i.e.,

$$\bigcup_{j=1}^K \left\{ y \rightarrow v^{(j)} : v \in V \setminus V_{\text{INV}} \right\}.$$

Figure 1 provides an example of the mixture DAG arising from $K = 2$ component DAGs $\mathcal{D}^{(j)}$ on $|V| = 4$ nodes. Note that, while the results of this section hold even when the DAGs $\mathcal{D}^{(j)}$ have no common topological ordering (meaning that there exists no ordering π such that for all $1 \leq$

$j \leq K$, $v < u$ in π if and only if $u \notin \text{an}_{\mathcal{D}^{(j)}}(v)$), the mixture DAG is sparse (and hence provides information about the component DAGs through separation statements) when a common topological ordering exists (as in the example in Figure 1). When there is no common topological ordering, then the set V_{INV} is generally smaller, since $\text{pa}_{\mathcal{D}^{(j)}} \neq \text{pa}_{\mathcal{D}^{(k)}}$ implies $p^{(j)}(x_v | x_{\text{pa}_{\mathcal{D}^{(j)}}(v)}} \neq p^{(k)}(x_v | x_{\text{pa}_{\mathcal{D}^{(k)}}(v)})$, which implies a denser mixture DAG.

We emphasize here that the DAG in Definition 3.1 is not a graphical model representation of the mixture distribution in the standard sense. This is already clear from the fact that the mixture DAG has $K \cdot |V| + 1$ nodes, whereas the mixture distribution is only $|V|$ -dimensional. Yet, in the following theorem we show that it is possible to read off conditional independence relations that hold in the mixture distribution p_μ from the mixture graph in an intuitive manner. For $A \subseteq V$, we use the notation $[A]$ to denote all K copies of the nodes in A , i.e., $A = \cup_{1 \leq j \leq K} A^{(j)}$.

Theorem 3.2 (Markov Property). *Let $A, B, C \subseteq V$ be disjoint. If $[A]$ and $[B]$ are d -separated given $[C]$ in the mixture DAG $\mathcal{D}_\mu$, then $X_A \perp\!\!\!\perp X_B | X_C$ in the mixture distribution p_μ .*

To illustrate this result, consider the example in Figure E.1. Since $[1] = \{1^{(1)}, 1^{(2)}\}$ and $[4] = \{4^{(2)}, 4^{(2)}\}$ are d -separated given $\emptyset$ in the mixture DAG, then the mixture distribution $p_\mu(x_1, x_2, x_3, x_4)$ satisfies $X_1 \perp\!\!\!\perp X_4$.

We note that while the graphical representation provided by Strobl (2019b) (the *mother graph*) is similar to the mixture DAG, it critically differs in how the component DAGs are connected via the node y . Importantly, we show in Section B in the Appendix that the mixture distribution p_μ is not Markov with respect to the mother graph¹.

In the following, we provide a proof for Theorem 3.2. For each $1 \leq j \leq K$, let $\tilde{\mathcal{D}}^{(j)}$ be the sub-DAG induced by $\mathcal{D}_\mu$ on the vertices $V^{(j)} \cup \{y\}$ for $j = 1, 2, \dots, K$. The main ingredient of the proof is the following lemma, which connects d -separation statements in the mixture DAG to conditional independence relations in the mixture distribution via d -separation statements in $\tilde{\mathcal{D}}^{(j)}$.

Lemma 3.3. *Let $A, B, C \subseteq V$ be disjoint. If for all $1 \leq j \leq K$ it holds that*

- (a) $A^{(j)}$ and $B^{(j)}$ are d -separated given $C^{(j)}$, and;
- (b) $A^{(j)}$ and y are d -separated given $C^{(j)}$ in $\tilde{\mathcal{D}}^{(j)}$,

then $X_A \perp\!\!\!\perp J | X_C$ in p_μ , i.e., the following factorization holds:

$$p^{(j)}(x_A, x_B | x_C) = p^{(1)}(x_A | x_C) p^{(j)}(x_B | x_C)$$

¹Strobl (2019a;b) provides two different constructions; we show that the Markov property does not hold in either.

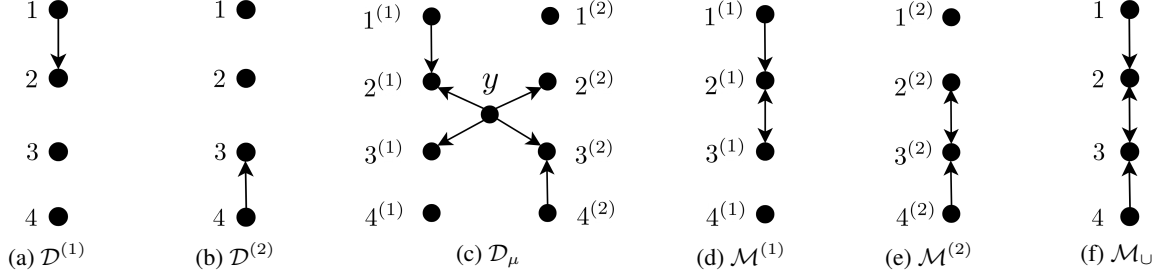


Figure 1: (a)-(b): component DAGs $\mathcal{D}^{(1)}$ and $\mathcal{D}^{(2)}$ for a mixture model with $K = 2$; (c): corresponding mixture DAG $\mathcal{D}_\mu$ (see Definition 3.1); (d)-(e): associated component MAGs $\mathcal{M}^{(1)}$ and $\mathcal{M}^{(2)}$ (see Section 4); (f): associated union graph $\mathcal{M}_U$ (see Definition 4.2).

for all $1 \leq j \leq K$.

We now provide the proof for Theorem 3.2.

Proof. We start by showing that the conditions of Lemma 3.3 are satisfied. First, note that $[A]$ and $[B]$ are d-separated given $[C]$ in $\mathcal{D}_\mu$ implies that $A^{(j)}$ and $B^{(j)}$ are d-separated given $C^{(j)}$ in $\mathcal{D}^{(j)}$ for all $j \in \{1, 2, \dots, K\}$. Second, note that since y has in-degree 0, we cannot have both a d-connecting path between $[A]$ and y given $[C]$ and a d-connecting path between $[B]$ and y given $[C]$ in $\mathcal{D}_\mu$. Hence, we may assume without loss of generality that $[A]$ and y are d-separated given $[C]$ (otherwise, $[B]$ and y are d-separated given $[C]$).

We now use Lemma 3.3 to show that $p_\mu(x_A, x_B | x_C)$ factorizes as $f_A(x_A, x_C)f_B(x_B, x_C)$, which would prove that $X_A \perp\!\!\!\perp X_B | X_C$ in p_μ . By definition of the mixture distribution p_μ in (2),

$$p_\mu(x_A, x_B | x_C) = \sum_{j=1}^K p^{(j)}(x_A, x_B | x_C) p_J(j),$$

and hence as a consequence of Lemma 3.3 we obtain

$$\begin{aligned} p_\mu(x_A, x_B | x_C) &= \sum_{j=1}^K p^{(1)}(x_A | x_C) p^{(j)}(x_B | x_C) p_J(j) \\ &= p^{(1)}(x_A | x_C) \sum_{j=1}^K p^{(j)}(x_B | x_C) p_J(j), \end{aligned}$$

providing a factorization of the desired form and hence completing the proof. $\square$

In Theorem 3.2, we established that every separation statement in the mixture DAG corresponds to a conditional independence relation in the mixture distribution. Next, we show that every mixture DAG is *realizable*, i.e., that for any mixture DAG $\mathcal{D}_\mu$, there exists a distribution whose conditional independence relations are faithfully represented by

the separation statements of the mixture graph. This implies that the mixture DAG is the “correct” graphical representation of a mixture of DAGs and cannot be made sparser without losing the Markov property.

3.3. Faithfulness

We define faithfulness of a mixture distribution p_μ with respect to a mixture DAG $\mathcal{D}_\mu$ analogously to how faithfulness is defined for a distribution with respect to a DAG model.

Definition 3.4 (Mixture Faithfulness). *The mixture distribution p_μ whose density is of the form (2) is faithful with respect to a mixture DAG $\mathcal{D}_\mu$ if for any disjoint $A, B, C \subseteq V$ with $X_A \perp\!\!\!\perp X_B | X_C$ in p_μ it holds that $[A]$ and $[B]$ are d-separated given $[C]$.*

We next provide an example showing that mixture faithfulness is in general not implied by faithfulness of each component distribution $p^{(j)}(\cdot)$ with respect to the corresponding DAG $\mathcal{D}^{(j)}$. Hence, to establish realizability of the mixture graph, it is not sufficient to rely on the fact that for every DAG $\mathcal{D}^{(j)}$, there exists a distribution $p^{(j)}$ that is faithful to $\mathcal{D}^{(j)}$.

Example 1. *Consider the following two distributions $p^{(1)}(x_V)$ and $p^{(2)}(x_V)$ on $V = \{1, 2, 3, 4\}$ that factor according to the DAGs in Figure 1, namely*

$$p^{(1)}(x_V) = p^{(1)}(x_1)p^{(1)}(x_2|x_1)p^{(1)}(x_3)p^{(1)}(x_4),$$

and

$$p^{(2)}(x_V) = p^{(2)}(x_1)p^{(2)}(x_2)p^{(2)}(x_3|x_4)p^{(2)}(x_4),$$

where

$$\begin{aligned} p^{(1)}(x_1) &= \mathcal{N}(x_1; 0, 1), & p^{(2)}(x_1) &= \mathcal{N}(x_1; 0, 1), \\ p^{(1)}(x_2|x_1) &= \mathcal{N}(x_2; x_1, 1), & p^{(2)}(x_2) &= \mathcal{N}(x_2; 0, 2), \\ p^{(1)}(x_3) &= \mathcal{N}(x_3; 0, 1), & p^{(2)}(x_3|x_4) &= \mathcal{N}(x_3; x_4, 1), \\ p^{(1)}(x_4) &= \mathcal{N}(x_4; 0, 1), & p^{(2)}(x_4) &= \mathcal{N}(x_4; 0, 1). \end{aligned}$$

Then, defining the mixture distribution by

$$p_\mu(x_V) = \sum_{j=1}^2 p^{(j)}(x_V) p(j),$$

for some $J \sim p_J(j)$, we obtain that

$$\begin{aligned} p_\mu(x_2, x_3) &= \int p_\mu(x_V) dx_1 dx_4 \\ &= \int p_J(1) p^{(1)}(x_1) p^{(1)}(x_2|x_1) p^{(1)}(x_3) p^{(1)}(x_4) dx_1 dx_4 \\ &\quad + \int p_J(2) p^{(2)}(x_1) p^{(2)}(x_2) p^{(2)}(x_3|x_4) p^{(2)}(x_4) dx_1 dx_4 \\ &= p_J(1) \mathcal{N}(x_2; 0, 2) \mathcal{N}(x_3; 0, 1) \\ &\quad + p_J(2) \mathcal{N}(x_2; 0, 2) \mathcal{N}(x_3; 0, 2) \\ &= \mathcal{N}(x_2; 0, 2) \left(p_J(1) \mathcal{N}(x_3; 0, 1) + p_J(2) \mathcal{N}(x_3; 0, 2) \right) \\ &= f(x_2) g(x_3), \end{aligned}$$

which implies that $X_2 \perp\!\!\!\perp X_3$ in p_μ , although in the mixture DAG corresponding to p_μ shown in Figure 1 the nodes 2 and 3 are d -connected via the path through y . $\square$

This example was carefully crafted; even a slight perturbation such as choosing $p^{(2)}(x_2) = \mathcal{N}(x_2; 0, 2.001)$ would have meant that $p_\mu(x_2, x_3)$ does not factor, indicating that mixture-faithfulness violations are rare. More precisely, consider the family of Gaussian mixture models where each $p^{(j)}$ is a Gaussian distribution that is faithful with respect to $\mathcal{D}^{(j)}$. A violation of mixture-faithfulness occurs if and only if $\sum_j p^{(j)}(x_A, x_B|x_C)$ factors as $p_\mu(x_A|x_C) p_\mu(x_B|x_C)$, i.e.,

$$\sum_j p^{(j)}(x_A, x_B|x_C) = \sum_i p^{(i)}(x_A|x_C) \sum_j p^{(j)}(x_B|x_C),$$

when A and B are d -connected given C in $\mathcal{D}_\mu$. This represents an equality constraint on the parameters of the Gaussians $p^{(j)}$ for $j \in \{1, \dots, K\}$. As a consequence, mixture-faithfulness holds almost surely and any mixture DAG is realizable by a mixture of Gaussians, thereby proving the following result.

Proposition 3.5 (Realizability of $\mathcal{D}_\mu$). *For any mixture DAG $\mathcal{D}_\mu$, there exists a mixture distribution p_μ that is faithful with respect to $\mathcal{D}_\mu$.*

4. LEARNING FROM MIXTURE DATA

Without knowing the membership of each sample to a component DAG, we can in general not hope to learn the structure of each component DAG from the data. Since the mixing variable is latent, an intuitive approach is to apply the FCI algorithm to learn a MAG representation of the mixture distribution. In this section, we will characterize the

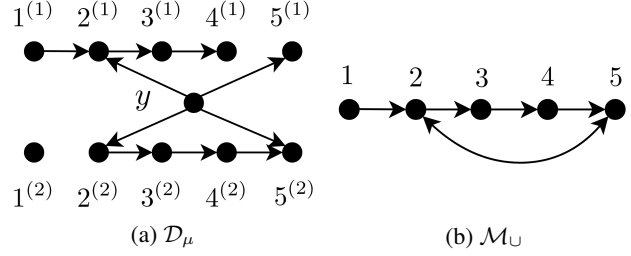


Figure 2: An example of a mixture DAG (a) and its associated union graph (b) as defined in 4.2. In this case, each DAG $\mathcal{D}^{(j)}$ has a common topological ordering, however, the union MAG is not ancestral.

output of the FCI algorithm. In particular, we will show that it identifies critical nodes in the component DAGs, namely those whose conditional distributions across the component DAGs vary.

A difficulty for structure discovery using MAG-based learning algorithms such as FCI, is that even under the mixture-faithfulness assumption the conditional independence relations in a mixture distribution p_μ may not be representable by any MAG. We illustrate this in the following example and then provide conditions to avoid this phenomenon.

Example 2. Consider the mixture DAG $\mathcal{D}_\mu$ shown in Figure 2a. In the following, we show that there does not exist any MAG $\tilde{\mathcal{M}}$ over the variables $V = \{1, \dots, 5\}$ that satisfies: A d -sep from B given C in $\tilde{\mathcal{M}}$ if and only if $[A]$ d -sep from $[B]$ given $[C]$ in $\mathcal{D}_\mu$. First, note that such a MAG would need to have the skeleton as in the graph shown in Figure 2b to respect the adjacencies in $\mathcal{M}_\mu$. Otherwise it would have an extra or missing d -separation with no analog in $\mathcal{M}_\mu$. In addition, $\tilde{\mathcal{M}}$ would also need to contain the colliders $4 \rightarrow 5 \leftarrow 2$ and $1 \rightarrow 2 \leftarrow 5$ to respect the d -separation statements resulting from $4^{(2)} \rightarrow 5^{(2)} \leftarrow y \rightarrow 2^{(2)}$ and $1^{(1)} \rightarrow 2^{(1)} \leftarrow y \rightarrow 5^{(1)}$ respectively. This implies the existence of $2 \leftrightarrow 5$ in $\tilde{\mathcal{M}}$. Furthermore, note that conditioning on either $[2]$, $[3]$ or $[4]$ (or any subset of these) connects $[5]$ and $[1]$ in $\mathcal{D}_\mu$ which are d -separated given $\emptyset$. The only orientation of arrowheads compatible with the skeleton determined and these d -separation/connection statements is $2 \rightarrow 3 \rightarrow 4$. Hence, 3 and 4 must be descendants of 2 in $\tilde{\mathcal{M}}$. Finally, the existence of an arrowhead $4 \leftarrow * 5$ would violate the separation: $[5]$ d -separated from $[1]$ given $\emptyset$. Hence, $2 \leftrightarrow 5$ in $\tilde{\mathcal{M}}$ and at the same time $2 \in \text{an}_{\tilde{\mathcal{M}}}(5)$, which violates the ancestral property. $\square$

We now identify a class of mixture models for which the d -separations in the mixture DAG are equivalent to d -separation statements in a MAG.

Definition 4.1. Let $\mathcal{M}^{(1)}, \dots, \mathcal{M}^{(K)}$ be the MAGs constructed via Algorithm 1 from the induced sub-DAGs $\tilde{\mathcal{D}}^{(j)} =$

$(V^{(j)} \cup \{y\}, \tilde{E}^{(j)})$ defined in Section 3.2. The MAGs $\mathcal{M}^{(1)}, \dots, \mathcal{M}^{(K)}$ are said to be compatible with the same poset if there exists a partial order π on the set V such that for all $1 \leq j \leq K$ it holds that

- (a) $u \in \text{an}_{\mathcal{M}^{(j)}}(v) \Rightarrow u <_{\pi} v$,
- (b) $u \leftrightarrow_{\mathcal{M}^{(j)}} v \Rightarrow u \not\prec_{\pi} v$.

Figures 1d and 1e show examples of MAGs $\mathcal{M}^{(j)}$ that satisfy this poset compatibility condition. One can further check that the MAGs $\mathcal{M}^{(1)}$ and $\mathcal{M}^{(2)}$ associated with the mixture DAG in Figure 2a do not satisfy this condition. This example shows that there exist DAGs $\mathcal{D}^{(1)}, \dots, \mathcal{D}^{(K)}$ with a common topological ordering whose corresponding MAGs $\mathcal{M}^{(1)}, \dots, \mathcal{M}^{(K)}$ do not satisfy the poset compatibility condition 4.1. On the other hand, it can be readily verified that the poset compatibility assumption on the MAGs implies that the associated component DAGs have a common topological ordering.

In the following, we show that poset compatibility ensures that the set of d-separation statements in a mixture DAG $\mathcal{D}_{\mu}$ is representable by a MAG. We start by defining this MAG, which we call the *union graph*. As the name suggests, this graph is obtained as a union of all the edges of $\mathcal{M}^{(1)}, \dots, \mathcal{M}^{(K)}$.

Definition 4.2 (Union Graph). *The union graph $\mathcal{M}_{\cup} := (V, D_{\cup}, B_{\cup})$ has vertices V , directed edges*

$$D_{\cup} = \{v \rightarrow u : u, v \in V, \exists_j v^j \rightarrow_{\mathcal{M}^{(j)}} u^j\};$$

and the set of bidirected edges

$$B_{\cup} = \{v \leftrightarrow u : v, u \in V, \exists_j v^j \leftrightarrow_{\mathcal{M}^{(j)}} u^j\}.$$

We remark that Spirtes (1994) studied a similar graph and proved the Markov property for a DAG with vertices $V \cup \{y\}$ and directed edges given by the union of $\mathcal{D}^{(1)}, \dots, \mathcal{D}^{(K)}$.

An example of a union graph is given in Figure 1f. In general, the union graph may neither be maximal nor ancestral (see Figure 2b for an example). However, as we show in the following lemma, whose proof is given in Section D in the Appendix, under the poset compatibility assumption, it is guaranteed to be both.

Lemma 4.3. *Under the assumption that the component MAGs $\mathcal{M}^{(1)}, \dots, \mathcal{M}^{(K)}$ are compatible with the same poset, the union graph is maximal and ancestral.*

We now state the main results of this section, characterizing the output of FCI when run on distributions arising as mixtures of DAGs.

Theorem 4.4. *Let $A, B, C \subseteq V$ be disjoint. If the component MAGs satisfy the poset compatibility assumption, then A and B are d-separated given C in $\mathcal{M}_{\cup}$, if and only if $[A]$ and $[B]$ are d-separated given $[C]$ in $\mathcal{D}_{\mu}$.*

The proof is provided in Section E in the Appendix. The following corollary follows directly from the asymptotic consistence of FCI (Colombo et al., 2012).

Corollary 4.5. *If the mixture distribution p_{μ} is faithful with respect to a mixture DAG whose component MAGs satisfy the compatibility assumption, then FCI is consistent, i.e., it outputs the Markov Equivalence Class of the corresponding union MAG $\mathcal{M}_{\cup}$.*

We end this section by pointing out important structural properties of the union graph, which can be used to recover key structural information about component DAGs in the mixture.

Proposition 4.6. *A bidirected edge $u \leftrightarrow v$ in the union graph $\mathcal{M}_{\cup}$ implies that $u \in V \setminus V_{INV}$, i.e., there exist $i, j \in \{1, 2, \dots, K\}$ such that $\text{pa}_{\mathcal{D}^{(j)}}(u) \neq \text{pa}_{\mathcal{D}^{(i)}}(u)$. Additionally, under mixture-faithfulness this implies that $p^{(j)}(x_u | x_{\text{pa}_{\mathcal{D}^{(j)}}(u)}) \neq p^{(i)}(x_u | x_{\text{pa}_{\mathcal{D}^{(i)}}(u)})$.*

Proof. A bidirected edge in $\mathcal{M}_{\cup}$ implies a bidirected edge in $\mathcal{M}^{(j)}$ for some j , which implies that y is connected across in $\mathcal{D}_{\mu}$. $\square$

Hence bidirected edges identify nodes in the component DAGs whose conditional distribution varies across mixture components. As we show in the following section, these nodes are natural candidates for features when clustering the samples.

5. EXPERIMENTS

In this section, we present results of experiments conducted on both synthetic and real data.

5.1. Synthetic Data

In the following, we demonstrate the effectiveness of learning the union graph from mixture data, analyze the performance when estimating $V \setminus V_{INV}$ using Proposition 4.6, and investigate the performance of clustering using mixture data when $V \setminus V_{INV}$ are used as features.

We generated K component DAGs each with $|V| = 10$ nodes and the same topological ordering from an Erdős-Rényi model with expected degree $d = 1.5/K$ so that the nodes in the union graph have an expected degree of less than 1.5. From these DAGs, the corresponding MAGs $\mathcal{M}^{(j)}$ were computed using Algorithm 1. If the MAGs were not compatible with the same poset, the DAGs were discarded to ensure poset-compatibility (2 out of 270 graphs were discarded at this sparsity level).

Data was sampled from each DAG based on a linear structural equation model with additive Gaussian noise, where each edge weight (u, v) was sampled uniformly in

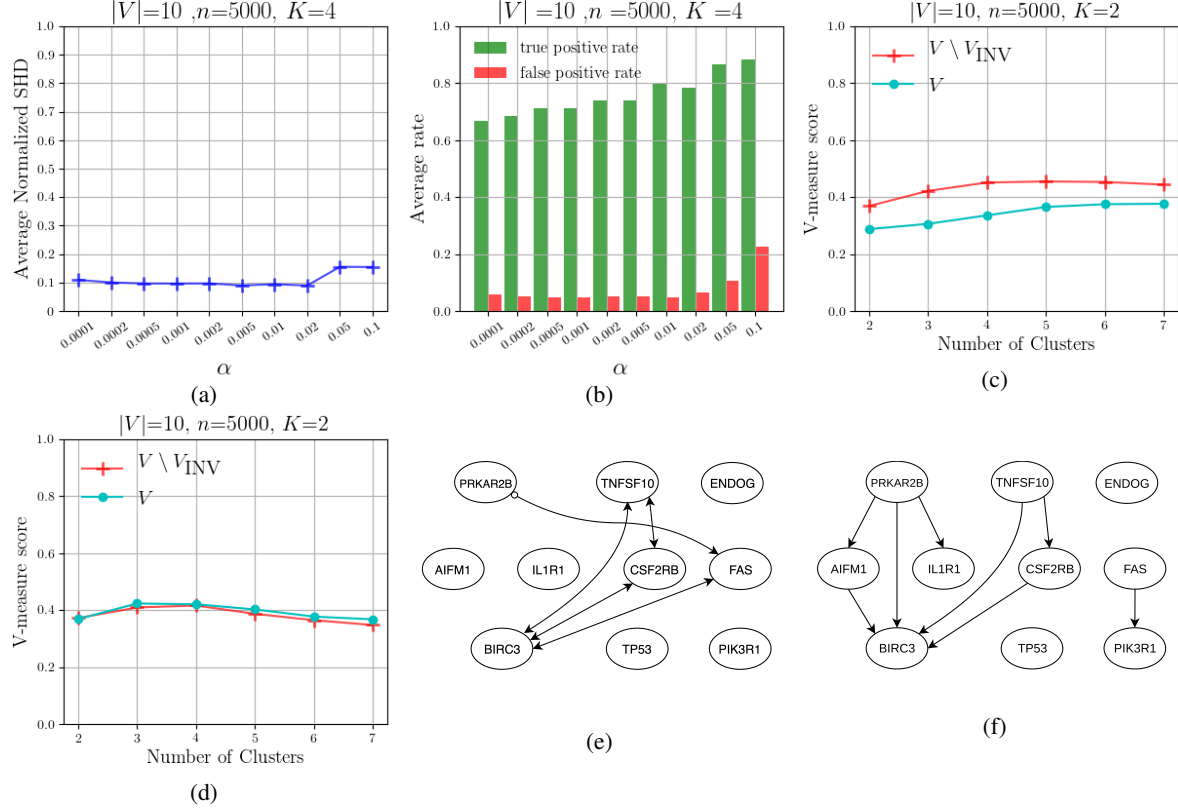


Figure 3: (a) shows the average normalized structural Hamming distance between the PAG $\hat{\mathcal{P}}_{\cup}$ estimated using the mixture data, and $\tilde{\mathcal{P}}_{\cup}$ estimated using data sampled from $\mathcal{M}_{\cup}$; (b) shows the true and false positive rate in estimating $V \setminus V_{INV}$; (c) shows the performance of clustering using $[V \setminus V_{INV}]$ when this set has no descendants in $\mathcal{D}_{\mu}$, while (d) shows the same plot when $[V \setminus V_{INV}]$ has descendants in $\mathcal{D}_{\mu}$; (e) shows the output of FCI on genes in the apoptosis pathway using the ovarian cancer mixture data without knowledge of the cluster membership for each sample, while (f) shows the difference graph of (Wang et al., 2018) on the same set of genes when the cluster membership of each sample is known.

$[-2, -0.25] \cup [0.25, 2]$ (to ensure that it was bounded away from zero) and set to be same for the edges $(u^{(j)}, v^{(j)})$ for all $1 \leq j \leq K$ if this edge existed in DAG $\mathcal{D}^{(j)}$. The mean for the Gaussian noise was sampled uniformly in $[-2, 2]$ with standard deviation 1. From each of the K DAGs, we generated n/K observations, yielding a total of n samples from the mixture distribution.

Learning the Union MAG. To evaluate Corollary 4.5, we ran FCI on this synthetic data using Gaussian conditional independence tests (despite the true distribution being a mixture of Gaussians) with threshold α . The output is a PAG $\hat{\mathcal{P}}_{\cup}$ representing the Markov equivalence class of the union graph. As comparison, we computed the true union graph $\mathcal{M}_{\cup}$ based on the MAGs $\mathcal{M}^{(j)}$, generated n samples from this graph (using a structural equation model with the same parameters as in the mixture) and ran FCI on these samples to obtain an estimate $\tilde{\mathcal{P}}_{\cup}$ for the PAG of the union graph. This offsets the estimation errors that are intrinsic to FCI. The difference between the PAGs $\hat{\mathcal{P}}_{\cup}$ and $\tilde{\mathcal{P}}_{\cup}$ was mea-

sured via a *normalized structural Hamming distance*; the structural Hamming distance (SHD) between PAGs counts the occurrences of $\circ \rightarrow$ in one of the PAGs versus $\circ - \rightarrow$ in the other, plus the number of adjacencies present in one graph but not the other. The normalization is done by dividing over the possible number of errors for the realization at hand to keep the value between 0 and 1 and make the numbers comparable. Figure 3a shows the normalized SHD averaged over 30 realizations of synthetic datasets. We used $K = 4$ and $n = 5000$ in this plot; in Section F in the Appendix, we provide additional plots for $K \in \{2, 6\}$ and $n \in \{1000, 10000\}$.

Identifying Nodes that Vary Across Component DAGs.

To evaluate Proposition 4.6 on synthetic data, we estimated $V \setminus V_{INV}$ by determining all nodes incident to bidirected edges in the PAG $\hat{\mathcal{P}}_{\cup}$ obtained using FCI on the synthetic data. This set was compared to the ground truth; Figure 3b shows true positive and false positive rates for varying sig-

nificance levels,² averaged over 30 realizations. We used $K = 4$ and $n = 5000$ in this plot. In Section F in the Appendix, we show additional plots for $K \in \{2, 6\}$ and $n \in \{1000, 10000\}$.

Clustering. Recall that, under mixture-faithfulness, $X_{V \setminus V_{\text{INV}}}$ represents the set of nodes whose conditionals vary across the component DAGs. This motivates using the nodes $X_{V \setminus V_{\text{INV}}}$ and their descendants as features to cluster the samples since these are the only nodes with different marginals across the mixture components. Since FCI in general cannot identify all the descendants of $X_{V \setminus V_{\text{INV}}}$, we used just the set $X_{V \setminus V_{\text{INV}}}$ for clustering. As a proof-of-concept to demonstrate that these features can be useful, we considered two settings, one in which $[V \setminus V_{\text{INV}}]$ has no descendants in $\mathcal{D}_\mu$ (see Figure 3c), and another one in which this set has descendants (Figure 3d).

In both settings, we used $\tilde{K}$ -means clustering for various values of $\tilde{K}$. To compare the quality of clustering of using $[V \setminus V_{\text{INV}}]$ versus all nodes as features, we used the V-measure score from Rosenberg and Hirschberg (2007) which is based on ground truth cluster assignments; a higher V-measure score represents better performance. As per what is expected from our theoretical results, Figure 3c shows that clustering based on the reduced number of features $[V \setminus V_{\text{INV}}]$ results in higher quality clusters as compared to using all features for clustering in the setting where $[V \setminus V_{\text{INV}}]$ has no descendants in $\mathcal{D}_\mu$, while otherwise both feature sets perform equally well.

5.2. Real Data

Ovarian Cancer. We applied this framework to gene expression data from ovarian cancer in $K = 2$ patient groups (with 93 and 168 observations, respectively) with different survival rates (Tothill et al., 2008). We followed the analysis of Wang et al. (2018), where the difference-DAG was estimated for the two groups based on the apoptosis pathway consisting of $|V| = 10$ genes. The resulting difference-DAG is shown in Figure 3f. While the difference-DAG can identify edges that are different between the two DAGs $\mathcal{D}^{(1)}$ and $\mathcal{D}^{(2)}$ and hence provides more information than the union graph, computing the difference-DAG requires knowledge of the membership of each observation to the two disease subgroups, which is not available for many diseases. The estimated PAG $\tilde{\mathcal{P}}_\cup$ based on the combined samples from the two patient groups is shown in Figure 3e. It was estimated using FCI with stability selection. FCI identified BIRC3 as the node with the highest number of incident bidirected edges; BIRC3 is known to be one of the

major dysregulated genes in ovarian cancer and an inhibitor of apoptosis (Johnstone et al., 2008; Jönsson et al., 2014).

T cell activation. We also applied our framework to single-cell gene expression data of naive and activated T cells (i.e. $K = 2$, with 298 and 377 samples, respectively) from Singer et al. (2016). Following the analysis in Wang et al. (2018), we performed the analysis on 60 genes that had a fold expression change above 10. The FCI output on these 60 nodes is shown in Section F.2 in the Appendix. The following nodes have the highest number of incident bidirected edges, indicating that they may play important roles in T cell activation: CDC6, CDC20, SHCBP1, NKG2A, GZMB4 and KIF2C. All these genes have been described before as critical. CDC6 and CDC20 are essential regulators of the cell division cycle. Shorter cell cycle time for increased proliferation is a hallmark of T cell activation (Qiao et al., 2016; Borlado and Méndez, 2008). SHCBP1 has been shown to be tightly linked to cell proliferation and strongly correlates with proliferative stages of T cell development (Schmandt et al., 1999; Buckley et al., 2014). NKG2A functions to limit excessive activation, prevent apoptosis, and preserve the specific T cell response (Rapaport et al., 2015). GZMB4 has been shown to regulate antiviral T cell response (Salti et al., 2011). Finally, the gene KIF2C encodes a Kinesin-like protein that functions as a microtubule-dependent molecular motor. It is over-expressed in a variety of solid tumors and induces frequent T cell responses (Gnjatic et al., 2010).

6. DISCUSSION

In this paper, we provided a graphical representation (via the mixture DAG) of distributions that arise as mixtures of causal DAGs. We showed that the mixture DAG not only satisfies the Markov property with respect to such mixture distributions, but is also always realizable by a mixture distribution, meaning that it cannot be made sparser without losing the Markov property. In addition, we characterized the output of the prominent FCI algorithm when applied to data from such mixture distributions. The FCI algorithm is a natural candidate in this setting due to the presence of the latent mixing variable. In particular, we proved that the FCI algorithm can identify nodes whose conditionals vary across the different component DAGs and showed how this property can be used to infer the cluster membership for each sample. This is relevant for many applications, as for example when studying diseases consisting of multiple not well characterized subtypes. In such studies, genomic perturbation experiments can not be performed relatively routinely, leading to high-throughput interventional data. In future work it would be interesting to study how interventional data could be used to enhance causal inference based on mixtures of DAGs or which interventions to perform in order to enhance identifiability of pathways that are

²We do not use an ROC since while increasing the threshold monotonically increases the true positive rate of the estimated adjacencies, it in general does not monotonically increase the number of correctly inferred edge orientations.

shared among the different subtypes as well as those that are different across the subtypes for personalized interventions.

ACKNOWLEDGEMENTS

Basil Saeed was partially supported by the Abdul Latif Jameel Clinic for Machine Learning in Health at MIT. Caroline Uhler was partially supported by NSF (DMS-1651995), ONR (N00014-17-1-2147 and N00014-18-1-2765), IBM, and a Simons Investigator Award.

REFERENCES

- Borlado, L. R. and Méndez, J. (2008). CDC6: from DNA replication to cell cycle checkpoints and oncogenesis. *Carcinogenesis*, 29(2):237–243.
- Buckley, M. W., Arandjelovic, S., Trampont, P. C., Kim, T. S., Braciale, T. J., and Ravichandran, K. S. (2014). Unexpected phenotype of mice lacking SHCBP1, a protein induced during T cell proliferation. *PloS ONE*, 9(8).
- Chickering, D. M. (2002). Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3(Nov):507–554.
- Colombo, D., Maathuis, M. H., Kalisch, M., and Richardson, T. S. (2012). Learning high-dimensional directed acyclic graphs with latent and selection variables. *The Annals of Statistics*, pages 294–321.
- Friedman, N., Linial, M., Nachman, I., and Pe’er, D. (2000). Using Bayesian networks to analyze expression data. *Journal of Computational Biology*, 7(3-4):601–620.
- Gnjatic, S., Cao, Y., Reichelt, U., Yekebas, E. F., Nölker, C., Marx, A. H., Erbersdobler, A., Nishikawa, H., Hildebrandt, Y., Bartels, K., et al. (2010). NY-CO-58/KIF2C is overexpressed in a variety of solid tumors and induces frequent T cell responses in patients with colorectal cancer. *International Journal of Cancer*, 127(2):381–393.
- Heckerman, D., Mamdani, A., and Wellman, M. P. (1995). Real-world applications of Bayesian networks. *Communications of the ACM*, 38(3):24–26.
- Johnstone, R. W., Frew, A. J., and Smyth, M. J. (2008). The TRAIL apoptotic pathway in cancer onset, progression and therapy. *Nature Reviews Cancer*, 8(10):782–798.
- Jönsson, J.-M., Bartuma, K., Dominguez-Valentin, M., Harbst, K., Ketabi, Z., Malander, S., Jönsson, M., Carneiro, A., Måsbäck, A., Jönsson, G., et al. (2014). Distinct gene expression profiles in ovarian cancer linked to Lynch syndrome. *Familial Cancer*, 13(4):537–545.
- Lauritzen, S. L. (1996). *Graphical Models*, volume 17. Clarendon Press.
- Pearl, J. (2009). *Causality*. Cambridge university press.
- Qiao, R., Weissmann, F., Yamaguchi, M., Brown, N. G., VanderLinden, R., Imre, R., Jarvis, M. A., Brunner, M. R., Davidson, I. F., Litos, G., et al. (2016). Mechanism of APC/CCDC20 activation by mitotic phosphorylation. *Proceedings of the National Academy of Sciences*, 113(19):E2570–E2578.
- Rapaport, A. S., Schriewer, J., Gilfillan, S., Hembrador, E., Crump, R., Plougastel, B. F., Wang, Y., Le Friec, G., Gao, J., Cella, M., et al. (2015). The inhibitory receptor NKG2A sustains virus-specific CD8+ T cells in response to a lethal poxvirus infection. *Immunity*, 43(6):1112–1124.
- Richardson, T. and Spirtes, P. (2002). Ancestral graph Markov models. *The Annals of Statistics*, 30(4):962–1030.
- Robins, J. M., Hernan, M. A., and Brumback, B. (2000). Marginal structural models and causal inference in epidemiology.
- Rosenberg, A. and Hirschberg, J. (2007). V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 410–420.
- Sadeghi, K. et al. (2013). Stable mixed graphs. *Bernoulli*, 19(5B):2330–2358.
- Sadeghi, K. and Lauritzen, S. (2014). Markov properties for mixed graphs. *Bernoulli*, 20(2):676–696.
- Salti, S. M., Hammelev, E. M., Grewal, J. L., Reddy, S. T., Zemple, S. J., Grossman, W. J., Grayson, M. H., and Verbsky, J. W. (2011). Granzyme B regulates antiviral CD8+ T cell responses. *The Journal of Immunology*, 187(12):6301–6309.
- Schmandt, R., Liu, S. K., and McGlade, C. J. (1999). Cloning and characterization of mPAL, a novel Shc SH2 domain-binding protein expressed in proliferating cells. *Oncogene*, 18(10):1867–1879.
- Singer, M., Wang, C., Cong, L., Marjanovic, N. D., Kowalczyk, M. S., Zhang, H., Nyman, J., Sakuishi, K., Kurtulus, S., Gennert, D., et al. (2016). A distinct gene module for dysfunction uncoupled from activation in tumor-infiltrating T cells. *Cell*, 166(6):1500–1511.
- Solus, L., Wang, Y., Matejovicova, L., and Uhler, C. (2017). Consistency guarantees for permutation-based causal inference algorithms. *arXiv preprint arXiv:1702.03530*.

- Spirtes, P. (1994). Conditional independence properties in directed cyclic graphical models for feedback. Technical report, Technical report, Carnegie Mellon University.
- Spirtes, P., Glymour, C. N., Scheines, R., and Heckerman, D. (2000). *Causation, Prediction, and Search*. MIT press.
- Strobl, E. V. (2019a). The global Markov property for a mixture of dags. *arXiv preprint arXiv:1909.05418*.
- Strobl, E. V. (2019b). Improved causal discovery from longitudinal data using a mixture of dags. In Le, T. D., Li, J., Zhang, K., Cui, E. K. P., and Hyvärinen, A., editors, *Proceedings of Machine Learning Research*, volume 104 of *Proceedings of Machine Learning Research*, pages 100–133, Anchorage, Alaska, USA. PMLR.
- Thiesson, B., Meek, C., Chickering, D. M., and Heckerman, D. (2013). Learning mixtures of dag models. *arXiv preprint arXiv:1301.7415*.
- Tothill, R. W., Tinker, A. V., George, J., Brown, R., Fox, S. B., Lade, S., Johnson, D. S., Trivett, M. K., Etemadmoghadam, D., Locandro, B., et al. (2008). Novel molecular subtypes of serous and endometrioid ovarian cancer linked to clinical outcome. *Clinical cancer Research*, 14(16):5198–5208.
- Wang, Y., Squires, C., Belyaeva, A., and Uhler, C. (2018). Direct estimation of differences in causal graphs. In *Advances in Neural Information Processing Systems*, pages 3770–3781.
- Zhang, J. (2008). Causal reasoning with ancestral graphs. *Journal of Machine Learning Research*, 9(Jul):1437–1474.

APPENDIX

A. Proof of Proposition 2.1

We begin by recalling the definition of an inducing path from Richardson and Spirtes (2002), specialized to ancestral graphs.

Definition A.1. A path $v_1, \dots, v_n$ in an ancestral graph $\mathcal{G}$ is inducing if v_1 and v_n are not adjacent in $\mathcal{G}$ and for all $i \in \{2, \dots, n-1\}$, we have

$$v_{i-1} \leftrightarrow v_i \leftrightarrow v_{i+1} \quad \text{and} \quad v_i \in \text{an}_{\mathcal{G}}(\{v_1, v_n\}).$$

Richardson and Spirtes (2002) showed the following condition for an ancestral graph to be maximal.

Lemma A.2 ((Richardson and Spirtes, 2002)). *An ancestral graph $\mathcal{M}$ is maximal if and only if $\mathcal{G}$ does not contain any inducing paths.*

This allows us to prove Proposition 2.1.

Proof of Proposition 2.1. We show that the graph resulting from Algorithm 1 does not contain inducing paths. Let $\mathcal{M}$ be the output of the algorithm. Suppose we have vertices $v_1, \dots, v_n$ where $v_{i-1} \leftrightarrow v_i \leftrightarrow v_{i+1}$ for all $i \in \{2, \dots, n-1\}$ in $\mathcal{M}$. Then by step 1 of the algorithm, we must have $v_1, \dots, v_n \in \text{ch}_{\mathcal{D}_\mu}(y)$, implying that $v_1 \leftrightarrow_{\mathcal{M}} v_n$, and hence the path is not inducing. $\square$

B. Counter-example for the Markov property of the mother graph

In the following, we provide a counter-example for the Markov property of the mother-graph representation introduced by Strobl (2019b;a). We first remark that the Markov property in Strobl (2019a) generalizes that of Strobl (2019b) in the following sense: if the Markov property of the latter is satisfied, then the former is satisfied. Hence, we here provide a counter-example for the former, which can serve as a counter-example for both.

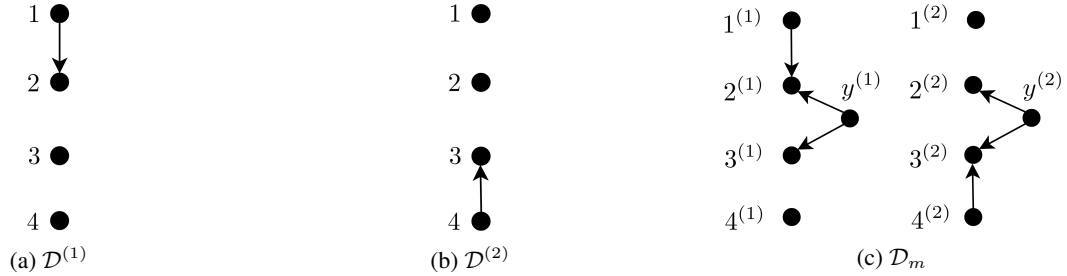


Figure B.1: (c) shows the mother graph $\mathcal{D}_m$ associated with the DAGs $\mathcal{D}^{(1)}$ and $\mathcal{D}^{(2)}$ in (a) and (b).

We start by recalling a few definitions from Strobl (2019b) using notation native to our development. Given a mixture of DAGs with distribution p_μ where $p^{(j)}$ factorizes according to $\mathcal{D}^{(j)}$, the mother graph $\mathcal{D}_m = (V_m, D_m)$ has nodes $V_m := [V] \cup \{y^{(1)}, \dots, y^{(K)}\}$ and directed edges

$$D_m := \bigcup_{1 \leq j \leq K} \{y^{(j)} \rightarrow v^{(j)} : v \in V \setminus V_{\text{INV}}\} \cup \{u^{(j)} \rightarrow v^{(j)} : u \rightarrow_{\mathcal{D}^{(j)}} v\}.$$

An example of the mother graph is shown in Figure B.1. A variable $c^{(j)} \in [V]$ in the mother graph is called an m -collider if and only if at least one of the following conditions hold:

- $a^{(j)} \rightarrow c^{(j)} \leftarrow b^{(j)}$, where $a, b \in V \cup \{y\}$
- $a^{(j)} \rightarrow c^{(j)} \leftarrow y^{(j)}$ and $y^{(k)} \rightarrow c^{(k)} \leftarrow b^{(k)}$ where $a, b \in V$.

An m -path exists between $[A]$ and $[B]$ in the mother graph if and only if there exists a sequence of triples between $[A]$ and $[B]$ such that at least one of the following two conditions is true for each triple in the sequence:

- $a^{(j)} * \rightarrow c^{(j)} * \rightarrow b^{(j)}$ with $a, b, c \in V \cup \{y\}$
- $a^{(j)} \rightarrow c^{(j)} \leftarrow y^{(j)}$ and $y^{(k)} \rightarrow c^{(k)} \leftarrow b^{(k)}$ where $a, b, c \in V$.

Finally, $[A]$ and $[B]$ are said to be *m-d-connected given $[C]$* if and only if there exists an m-path between $[A]$ and $[B]$ such that the following two conditions hold:

- $c^{(j)} \in [C]$ for every m-collider on the path, where $c \in V$
- $a^{(j)} \notin [C]$ for every non-m-collider on the path, where $c \in V \cup \{y\}$.

Now, the Markov property for the mother graph states that if $[A]$ and $[B]$ are not m-d connected given $[C]$ in the mother graph, then $X_A \perp\!\!\!\perp X_B \mid X_C$ in p_μ (Strobl, 2019b;a).

We now provide a counter example for this Markov property. For this, consider the mother graph in Figure B.1c over $V = \{1, 2, 3, 4\}$. Note that according to the definition of m-d-connection, $\{1\}$ and $\{4\}$ are not m-d-connected given $\{2, 3\}$. Hence, the Markov property should imply that $X_1 \perp\!\!\!\perp X_4 \mid X_2, X_3$ in any mixture distribution whose mother graph is as shown. In the following, construct a mixture distribution where this is not satisfied.

For simplicity, let $p_J(1) = p_J(2) = \frac{1}{2}$. Define $p^{(1)}(x_V)$ as

$$\begin{aligned} p^{(1)}(x_1) &= \mathcal{N}(x_1; 0, 1), \\ p^{(1)}(x_2|x_1) &= \mathcal{N}(x_2; x_1, 1), \\ p^{(1)}(x_3) &= \mathcal{N}(x_3; 0, 1), \\ p^{(1)}(x_4) &= \mathcal{N}(x_4; 0, 1), \end{aligned}$$

and $p^{(2)}(x_V)$ as

$$\begin{aligned} p^{(2)}(x_1) &= \mathcal{N}(x_1; 0, 1), \\ p^{(2)}(x_2) &= \mathcal{N}(x_2; 0, 1), \\ p^{(2)}(x_3|x_4) &= \mathcal{N}(x_3; x_4, 1), \\ p^{(2)}(x_4) &= \mathcal{N}(x_4; 0, 1). \end{aligned}$$

Clearly, $p^{(1)}(x_V)$ and $p^{(2)}(x_V)$ factorize according to $\mathcal{D}^{(1)}$ of Figure B.1a and $\mathcal{D}^{(2)}$ of Figure B.1b, respectively. Now,

$$\begin{aligned} p_\mu(x_1, x_2, x_3, x_4) &= \sum_{j \in \{1, 2\}} p_J(j) p^{(j)}(x_1, x_2, x_3, x_4) \\ &= \frac{1}{2} \frac{1}{(2\pi)^2} \left(e^{-\frac{x_1^2}{2}} e^{-\frac{x_2^2}{2}} e^{-\frac{x_3^2}{2}} e^{-\frac{(x_2-x_1)^2}{2}} + e^{-\frac{x_1^2}{2}} e^{-\frac{x_2^2}{2}} e^{-\frac{x_3^2}{2}} e^{-\frac{(x_3-x_4)^2}{2}} \right) \\ &= \frac{1}{2} \frac{1}{(2\pi)^2} e^{-\frac{x_1^2}{2}} e^{-\frac{x_2^2}{2}} e^{-\frac{x_3^2}{2}} e^{-\frac{x_4^2}{2}} \left(e^{x_2 x_1} e^{-\frac{x_1^2}{2}} + e^{x_3 x_4} e^{-\frac{x_4^2}{2}} \right), \end{aligned}$$

which cannot be written as

$$f(x_1, x_2, x_3) g(x_2, x_3, x_4)$$

for any f, g , implying that $X_1 \not\perp\!\!\!\perp X_4 \mid X_2, X_3$ in p_μ . □

C. Proof of Lemma 3.3

Proof of Lemma 3.3. By the assumption, $p^{(j_1)}(x_V)$ factors according to $\mathcal{D}^{(j_1)}$. Hence, it is sufficient to define a distribution $\tilde{p}_{X_V, J}(x_V, j)$ over $X_V \cup \{J\}$ that factors according to $\tilde{\mathcal{D}}^{(j)}$, with $J \in \{j_1, j_2\}$ for an arbitrarily chosen $j_2 \in \{1, \dots, K\} \setminus \{j_1\}$, such that

$$\tilde{p}_{X_V | J}(x_V | j_1) = p^{(j_1)}(x_V).$$

Then, the factorization with respect to $\tilde{\mathcal{D}}^{(j_1)}$ along with the two d-separation statements in the hypothesis of the lemma would imply

$$\begin{aligned} p^{(j_1)}(x_A, x_B | x_C) &= \frac{\sum_{x_{V \setminus (A \cup B \cup C)}} p^{(j_1)}(x_V)}{\sum_{x_{V \setminus C}} p^{(j_1)}(x_V)} \\ &= \frac{\sum_{x_{V \setminus (A \cup B \cup C)}} \tilde{p}(x_V | j_1)}{\sum_{x_{V \setminus C}} \tilde{p}(x_V | j_1)} \\ &= \tilde{p}(x_A, x_B | x_C, j_1) \\ &= \tilde{p}(x_A | x_C) \tilde{p}(x_B | x_C, j_1). \end{aligned}$$

To complete the proof, we define such a distribution $\tilde{p}$. First let $V_y := \text{ch}_{\tilde{\mathcal{D}}^{(j_1)}}(y)$ and note that

$$\begin{aligned} \tilde{p}(x_V, j) &= \tilde{p}_J(j) \prod_{v \in V} \tilde{p}(x_v | x_{\text{pa}_{\tilde{\mathcal{D}}^{(j_1)}}(v)}, j) \\ &= \tilde{p}_J(j) \prod_{v \in V_y} \tilde{p}(x_v | x_{\text{pa}_{\mathcal{D}}^{(j_1)}}(v), j) \prod_{v \in V \setminus V_y} \tilde{p}(x_v | x_{\text{pa}_{\mathcal{D}}^{(j_1)}}(v)). \end{aligned}$$

Define

$$\tilde{p}_J(j) := \begin{cases} p_J(j_1) & j = j_1 \\ 1 - p_J(j_1) & j = j_2 \end{cases},$$

$$\tilde{p}(x_v | x_{\text{pa}_{\mathcal{D}}^{(j)}}(v)) := p(x_v | x_{\text{pa}_{\mathcal{D}}^{(j)}}(v)) \quad \forall v \in V \setminus V_y.$$

Now, for each $v \in V_y$, define

$$U(v) := \text{pa}_{\mathcal{D}^{(j_1)}}(v) \cap \text{pa}_{\mathcal{D}^{(j_2)}}(v)$$

and

$$D(v) := \text{pa}_{\mathcal{D}^{(j_2)}}(v) \setminus \text{pa}_{\mathcal{D}^{(j_1)}}(v),$$

and choose an arbitrary fixed value for $x_{\text{pa}_{\mathcal{D}^{(i)}}(v) \setminus \text{pa}_{\mathcal{D}^{(j)}}(v)}$ and denote it by $x'_d(v)$.

Then define for all $v \in V_y$,

$$\tilde{p}(x_v | x_{\text{pa}_{\mathcal{D}}^{(j)}}(v), j) := \begin{cases} p_{X_v | X_{\text{pa}_{\mathcal{D}}^{(j_1)}}(v), J}(x_v | x_{\text{pa}_{\mathcal{D}}^{(j_1)}}(v), j_1) & j = j_1 \\ p_{X_v | X_{U(v)}, X_{D(v)}, J}(x_v | x_{U(v)}, x'_d(v), j_2) & j = j_2 \end{cases}.$$

Now, one easily checks that this distribution indeed satisfies the factorization property, which completes the proof. $\square$

D. Proof of Lemma 4.3

The ancestral property follows directly since we impose the order compatibility assumption of Definition 4.1. In the following, we show maximality using the definition of inducing path and the associated maximality condition in Section A.

Proof of Lemma 4.3. Suppose we have a path $v_1 \leftrightarrow v_2 \leftrightarrow \dots \leftrightarrow v_{n-1} \leftrightarrow v_n$ in $\mathcal{M}_\cup$. Then, for all $m \in \{1, \dots, n-1\}$, we must have some $j \in \{1, \dots, K\}$ such that $v_m^{(j)} \leftrightarrow v_{m+1}^{(j)}$ in $\mathcal{M}^{(j)}$, implying that for all m , we must have a j such that $v_m^{(j)}, v_{m+1}^{(j)} \in \text{ch}_{\mathcal{D}^{(j)}}(y)$ and hence a j such that $v_m^{(j)}, v_{m+1}^{(j)} \in \text{ch}_{\mathcal{D}_\mu}(y)$. But by construction of $\mathcal{D}_\mu$, this implies that $v_m^{(j)} v_{m+1}^{(j)} \in \text{ch}_{\mathcal{D}_\mu}(y)$ for all $j \in \{1, \dots, K\}$. Therefore, for any j , we have $v_1^{(j)} \dots, v_n^{(j)} \in \text{ch}_{\mathcal{D}^{(j)}}(y)$, and hence Algorithm 1 adds an edge between $v_1^{(j)}$ and $v_n^{(j)}$ in $\mathcal{M}^{(j)}$, resulting in an edge between v_1 and v_n in $\mathcal{M}_\cup$. Therefore, the path $v_1, \dots, v_n$ is not inducing in $\mathcal{M}_\cup$. $\square$

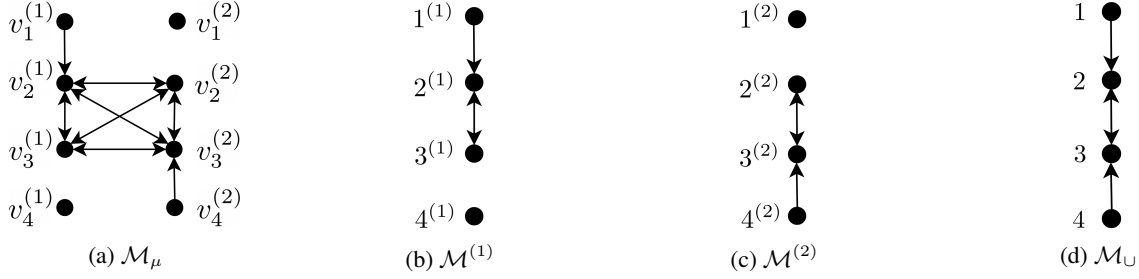


Figure E.1

E. Proof of Theorem 4.4

Since we assume that $A, B, C \subseteq V$, i.e., these sets do not contain y , then $[A]$ and $[B]$ are d-separated in $\mathcal{D}_\mu$ given $[C]$ if and only if they are d-separated in the marginal MAG of $\mathcal{D}_\mu$ w.r.t. $\{y\}$ obtained from Algorithm 1. We refer to this MAG as the *mixture MAG* and denote it by $\mathcal{M}_\mu$. We will make use of this MAG in parts of the following proof since it simplifies the arguments.

One thing to note about $\mathcal{M}_\mu$ is that if we remove the edges of the form $u^{(j)} \circ \circ v^{(i)}$ for $u, v \in V$ and $i \neq j$, then we obtain a bijection between the edges of $\mathcal{M}_\mu$ and the union of all the edges of $\mathcal{M}^{(j)}$ for all j . Figure E.1 illustrates this for an example. Hence, we can alternatively think of the union graph as having directed edges

$$D_\cup := \{u \rightarrow v : u, v \in V, \exists_i u^{(i)} \rightarrow_{\mathcal{M}_\mu} v^{(i)}\},$$

and bidirected edges

$$B_\cup := \{u \leftrightarrow v : u, v \in V, \exists_i u^{(i)} \leftrightarrow_{\mathcal{M}_\mu} v^{(i)}\}.$$

We prove Theorem 4.4 in 3 main steps. *First*, in Lemma E.5 we show that for any d-connecting path between a and b given C in $\mathcal{M}_\cup$, we can find a d-connecting path between $a^{(i)}$ and $b^{(k)}$ given $[C]$ in $\mathcal{M}_\mu$. *Second*, in Lemma E.7 we show the converse: that for any d-connecting path $a^{(i)}$ and $b^{(k)}$ given $[C]$ in $\mathcal{M}_\mu$, we can find a d-connecting path between a and b given C in $\mathcal{M}_\cup$. *Finally*, in Lemma E.8 we show that this equivalence implies that for any disjoint sets $A, B, C \subseteq V$, A and B are d-separated in $\mathcal{M}_\cup$ if and only if $[A]$ and $[B]$ are d-separated in $\mathcal{M}_\mu$ given $[C]$.

The proof strategy in Lemmas E.5 and E.7 relies on concatenating d-connecting paths given C of the form $P_1 = \langle v_1, \dots, v_n \rangle$ and $P_2 = \langle v_n, \dots, v_m \rangle$ together to create longer d-connecting paths given C of the form $P = \langle v_1, \dots, v_m \rangle$. When doing so, we must take care to ensure that v_n is active on the longer path, i.e., we must ensure that v_n is a collider on the path P if and only if $v_n \in C$.

E.1. A connecting path in $\mathcal{M}_\cup$ implies an analogous one in $\mathcal{M}_\mu$

We begin by proving some auxiliary results for step 1.

Lemma E.1 (Bidirected Connections). *If $a^{(i)} \leftrightarrow_{\mathcal{M}_\mu} b^{(k)}$ for any $i, k \in \{1, \dots, K\}$, then $a^{(i)} \leftrightarrow_{\mathcal{M}_\mu} b^{(j)}$ for all $j \in \{1, \dots, K\} \setminus \{i\}$.*

Proof. $a^{(i)} \leftrightarrow_{\mathcal{M}_\mu} b^{(k)}$ implies that $a^{(i)}, b^{(k)} \in \text{ch}_{\mathcal{D}_\mu}(y)$. By construction of $\mathcal{D}_\mu$, this implies $a^{(j)}, b^{(j)} \in \text{ch}_{\mathcal{D}_\mu}(y)$ for all $j \in \{1, \dots, K\}$, and hence step 1 of Algorithm 1 will add the bidirected edges $a^{(i)} \leftrightarrow b^{(j)}$ for all $j \in \{1, \dots, K\}$. Step 3 will only remove it if $a^{(i)}$ and $a^{(j)}$ are ancestors of one another in $\mathcal{D}_\mu$, which could happen only if $j = i$. Hence, $a^{(i)} \leftrightarrow_{\mathcal{M}_\mu} b^{(j)}$ for all $j \in \{1, \dots, K\} \setminus \{i\}$. $\square$

Lemma E.2 (Bidirected district). *Assume $a^{(i)} \leftrightarrow_{\mathcal{M}_\mu} b^{(j)}$ and $c^{(k)} \leftrightarrow_{\mathcal{M}_\mu} d^{(l)}$.*

- If $i \neq l$, then $a^{(i)} \leftrightarrow_{\mathcal{M}_\mu} d^{(l)}$.
- If $i = l$, then
 - $a^{(i)} \leftrightarrow_{\mathcal{M}_\mu} d^{(l)}$ if neither $a^{(i)}, d^{(l)}$ is an ancestor of another in $\mathcal{M}_\mu$,

- $a^{(i)} \rightarrow_{\mathcal{M}_\mu} d^{(l)}$ if $a^{(i)} \in \text{an}_{\mathcal{M}_\mu}(d^{(l)})$; or
- $a^{(i)} \leftarrow_{\mathcal{M}_\mu} d^{(l)}$ if $d^{(l)} \in \text{an}_{\mathcal{M}_\mu}(a^{(i)})$.

Proof. $a^{(i)} \leftrightarrow_{\mathcal{M}_\mu} b^{(j)}$ and $c^{(k)} \leftrightarrow_{\mathcal{M}_\mu} d^{(l)}$ implies that $a^{(\iota)}, b^{(\iota)}, c^{(\iota)}, d^{(\iota)} \in \text{ch}_{\mathcal{D}_\mu}(y)$ for all $\iota \in \{1, \dots, K\}$. Hence, step 1 of Algorithm 1 will add $a^{(i)} \leftrightarrow_{\mathcal{M}_\mu} d^{(l)}$. If $i \neq l$, then $a^{(i)}$ and $d^{(l)}$ cannot be ancestors of one another, implying that step 3 will not remove this bidirected edge. If $i = l$, then the edge will be removed and replaced with the appropriate directed edge if one of $a^{(i)}$ or $d^{(l)}$ is an ancestor of the other. Otherwise, the bidirected edge will remain. $\square$

Lemma E.3 (Arrow tip lemma). *Under the ordering assumption in Definition 4.1, if a directed edge $a \rightarrow_{\mathcal{M}_\cup} b$ exists in $\mathcal{M}_\cup$, then we must have $a^j \rightarrow_{\mathcal{M}_\mu} b^j$ for some j in $\mathcal{M}_\mu$. If a bidirected edge $a \leftrightarrow_{\mathcal{M}_\cup} b$ exists in $\mathcal{M}_\cup$, then we must have $a^j \leftrightarrow_{\mathcal{M}_\mu} b^j$ for some j in $\mathcal{M}_\mu$.*

Proof. The proof follows directly from the definition of the union graph. $\square$

Lemma E.4 (Changing Arrowtips Lemma). *Under the ordering assumption in Definition 4.1, if $a^{(j)} \ast \rightarrow_{\mathcal{M}_\mu} b^{(j)}$ but not $a^{(k)} \ast \rightarrow_{\mathcal{M}_\mu} b^{(k)}$ (same type of edge) for some $j \neq k$, then we must have $b^{(j)} \leftrightarrow b^{(k)}$.*

Proof. The ordering assumption does not allow $a^{(j)} \rightarrow_{\mathcal{M}_\mu} b^{(j)}$ and $a^{(k)} \leftrightarrow_{\mathcal{M}_\mu} b^{(k)}$ (and vice versa). Hence, we must only look at the existence of $a^{(j)} \ast \rightarrow_{\mathcal{M}_\mu} b^{(j)}$ and the in-existence of an edge between $a^{(k)}$ and $b^{(k)}$.

First, we note that if step 1 of Algorithm 1 defining $\mathcal{M}_\mu$ adds $b^{(j)} \leftrightarrow b^{(k)}$, then it will remain since step 2 does not modify edges but only adds them, while step 3 will never remove an edge $b^{(j)} \leftrightarrow b^{(k)}$ since neither can be an ancestor or a descendant of the other in $\mathcal{D}_\mu$.

Now, if $a^{(j)} \rightarrow_{\mathcal{D}(j)} b^{(j)}$ but not $a^{(k)} \rightarrow_{\mathcal{D}(k)} b^{(k)}$ for some k , then we must have $b \in V \setminus V^I$ and hence $b^{(\iota)} \in \text{ch}_{\mathcal{D}_\mu}(y)$ for all $\iota \in \{1, \dots, K\}$ by construction of $\mathcal{D}_\mu$. Therefore, step 1 of Algorithm 1 will add $b^{(j)} \leftrightarrow b^{(k)}$.

For the other case we must check that $a^{(j)} \circ \rightarrow_{\mathcal{M}_\mu} b^{(j)}$ was added by the algorithm that created $\mathcal{M}_\mu$. In all steps, the algorithm will only add such an edge if $b \in V \setminus V_{\text{INV}}$ and hence $b^{(j)} \leftrightarrow b^{(k)}$ must have been added in step 1. $\square$

Lemma E.5 (Step 1). *Under the ordering compatibility assumption in Definition 4.1, if there is a connecting path between a and b given some $C \subseteq V \setminus \{a, b\}$ in $\mathcal{M}_\cup$ ending in an arrow head (or tail respectively) incident to b , then there is a connecting path between $a^{(i)}$ and $b^{(k)}$ given $[C]$ in $\mathcal{M}_\mu$ for some $i, k \in \{1, \dots, K\}$ that also ends in an arrow head (or tail respectively) towards $b^{(k)}$.*

Proof. We use induction on the number of edges in the connecting path in $\mathcal{M}_\cup$. The base case for 1 edge follows directly from Lemma E.3.

Now assume we have a d-connecting path given C consisting of $m + 1$ edges in $\mathcal{M}_\cup$: $P_\cup = \langle a, \dots, d, b \rangle$ ending in an arrow head (or tail respectively). Consider the sub-path $\langle a, \dots, d \rangle$ with m edges. By the inductive hypothesis, there is a path $P_\mu = \langle a^{(i)}, \dots, d^{(j)} \rangle$ in $\mathcal{M}_\mu$ that is d-connecting given $[C]$, for some i, j , ending in the same tip. In the following, we show that we can always find a path of the form $\langle d^{(j)}, \dots, b^{(k)} \rangle$ for some k that can be joined together with P_μ to create a path $\langle a^{(i)}, b^{(k)} \rangle$ that is d-connecting given $[C]$. We do this by considering all the different cases for the tips of the edges $c \ast \rightarrow_{\mathcal{M}_\cup} d$ and $d \ast \rightarrow_{\mathcal{M}_\cup} b$.

Before discussing the different cases, note that if the edge $d^{(j)} \ast \rightarrow_{\mathcal{M}_\mu} b^{(k)}$ exists and is of the same type as the edge $d \ast \rightarrow_{\mathcal{M}_\cup} b$, then we can create the desired d-connecting path $\tilde{P}_\mu$ from $a^{(j)}$ to $b^{(i)}$ given $[C]$ by concatenating this edge with P_μ , since:

$$\begin{aligned} d \text{ is active on } Q_\cup &\Rightarrow (d \text{ is a collider on } Q_\cup \Leftrightarrow d \in C) \\ &\Rightarrow (d^{(j)} \text{ is a collider on } \tilde{P}_\mu \Leftrightarrow d^{(j)} \in [C]) \\ &\Rightarrow d^{(j)} \text{ is active on } \tilde{P}_\mu, \end{aligned}$$

where the second implication follows because the path P_μ ends in the correct type of arrow tip by the inductive hypothesis (I.H.). Hence, in what follows, it is sufficient to

$$\text{assume either } d^{(j)} \ast \ast b^{(k)} \text{ is not in } \mathcal{M}_\mu \text{ or is not the same edge type as } d \ast \ast b \text{ in } \mathcal{M}_\cup. \quad (\text{S.3})$$

(i) case $c \ast \rightarrow d \leftarrow \ast b$ in $\mathcal{M}_\cup$:

Since d is a collider on the path $Q_\cup$, we must have $d \in C$, and hence $d^{(j)} \in [C]$ for all j . Hence the path P_μ is of the form $\langle a^{(i)}, \dots, \gamma^{(\iota)}, d^{(j)} \rangle$, where $\gamma^{(\iota)} \ast \rightarrow d^{(j)}$ for some $\gamma \in V$ and $\iota \in \{1, \dots, K\}$ by the I.H. Furthermore, by Lemma E.3, we must have $d^{(\iota)} \leftarrow \ast_{\mathcal{M}_\mu} b^{(k)}$ for some $\iota, k \in \{1, \dots, K\}$. Since we assumed in (S.3) that this isn't true for $\iota = j$, then by Lemma E.4 we must have $d^{(j)} \leftrightarrow d^{(\iota)}$, creating the path $\ast \rightarrow d^{(j)} \leftrightarrow d^{(\iota)} \leftarrow \ast b^{(k)}$ that is d-connected given $[C]$ (recall $d^{(j)}, d^{(\iota)} \in [C]$). Concatenating $d^{(j)} \leftrightarrow d^{(\iota)} \leftarrow \ast b^{(k)}$ to P_μ gives the desired d-connecting path $\langle a^{(i)}, \dots, d^{(j)}, d^{(\iota)}, b^{(k)} \rangle$.

For the remaining cases, we begin by recalling that the edge $d \ast \ast_{\mathcal{M}_\cup} b$ must exist since $d^{(k)} \ast \ast_{\mathcal{M}_\mu} b^{(k)}$ for some k by Lemma E.3. Now, let $\alpha^{(j)}$ be the node on the path P_μ closest to $a^{(i)}$ such that all nodes between $\alpha^{(j)}$ and $d^{(j)}$ have the same index j , i.e., all of these are contained in the same MAG $\mathcal{M}^{(j)}$. This means that the node preceding $\alpha^{(j)}$ on this path, call it $\gamma^{(\kappa)}$, either has a different index (i.e., a part of a different $\mathcal{M}^{(\kappa)}$), or $\alpha^{(j)} = a^{(i)}$.

Call $P_\mu^{(j)} = \langle \alpha^{(j)}, \dots, d^{(j)} \rangle$ the subpath of P_μ from $\alpha^{(j)}$ to $d^{(j)}$. This path is completely contained in $\mathcal{M}^{(j)}$. If it is possible to find a path $P_\mu^{(k)} = \langle \alpha^{(k)}, \dots, d^{(k)} \rangle$ in $\mathcal{M}^{(k)}$ that is analogous to $P_\mu^{(j)}$ (same types of edges), then we can replace the segment $P_\mu^{(j)}$ of P_μ with $P_\mu^{(k)}$ to obtain a connecting path between $a^{(i)}$ and $d^{(k)}$ given $[C]$. Then, concatenating $d^{(k)} \ast \ast b^{(k)}$ gives us the desired connecting path from $a^{(i)}$ to $b^{(k)}$ given $[C]$ in $\mathcal{M}_\mu$.

Hence, in checking the remaining cases, we further

$$\text{assume that it is not possible to find a path } P_\mu^{(k)} \text{ in } \mathcal{M}^{(k)}. \quad (\text{S.4})$$

Therefore, walking along the path $P_\mu^{(j)}$ backwards starting at $d^{(j)}$ until $\alpha^{(j)}$, we will eventually find an edge $\beta^{(j)} \ast \ast \delta^{(j)}$ such that $\beta^{(k)} \ast \ast \delta^{(k)}$ is not an edge. Take the first such edge. Now, if this edge was $\beta^{(j)} \leftrightarrow \delta^{(j)}$, then by Lemma E.1, we must have $\beta^{(j)} \leftrightarrow \delta^{(k)}$, implying that we can concatenate the subpath of P_μ of the form $\langle a^{(i)}, \dots, \beta^{(j)} \rangle$ with $\beta^{(j)} \leftrightarrow \delta^{(k)}$ and the subpath of $P_\mu^{(k)}$ of the form $\langle \delta^{(k)}, \dots, b^{(k)} \rangle$ to create the desired d-connecting path given $[C]$. Next we look at the situations where we do not have $\beta^{(j)} \leftrightarrow \delta^{(j)}$, considering each remaining case on the arrowheads of $c \ast \ast d \ast \ast b$ in $\mathcal{M}_\cup$ separately.

(ii) case $c \leftarrow d \rightarrow b$ in $\mathcal{M}_\cup$: This case is depicted in Figure E.2a. If the first edge found is of the form $\beta^{(j)} \leftarrow \delta^{(j)}$ where $\beta^{(k)} \leftarrow \delta^{(k)}$ is not present (see Figure E.2b), then by Lemmas E.4 and E.2, we must have $\beta^{(j)} \leftrightarrow b^{(k)}$ (Figure E.2d). Replacing the segment $\langle \beta^{(j)}, \dots, d^{(j)} \rangle$ of P_μ with $\beta^{(j)} \leftrightarrow b^{(k)}$ gives the desired path.

Otherwise, if we have $\beta^{(j)} \rightarrow \delta^{(j)}$ instead (Figure E.2c), then Lemmas E.4 and E.2 again say that we must have $\delta^{(j)} \leftrightarrow b^{(k)}$ (Figure E.2e). The subpath of P_μ of the form $\langle \delta^{(j)}, c^{(j)} \rangle$ shown in Figure E.2e is connecting given $[C]$ by the I.H. Starting at $\delta^{(j)}$ and walking towards $c^{(j)}$, we can find a collider that is in $[C]$ (shown in Figure E.2f). This collider must be a descendant of $\delta^{(j)}$. Hence, $\delta^{(j)}$ is active given $[C]$ on the path $\beta^{(j)} \rightarrow \delta^{(j)} \leftrightarrow b^{(k)}$ since it is a collider whose descendant is in $[C]$. Replacing the segment $\langle \beta^{(j)}, \dots, d^{(j)} \rangle$ in P_μ with this path gives the desired connecting path given $[C]$.

(iii) case $c \rightarrow d \rightarrow b$ in $\mathcal{M}_\cup$: Proceeding similarly, if the edge found is of the form $\beta^{(j)} \leftarrow \delta^{(j)}$, then we must have $\beta^{(j)} \leftrightarrow b^{(k)}$ similar to before and for the same reasons. Furthermore, we can find a d-connecting path by performing a concatenation similar to the one we did before: replace the segment $\langle \beta^{(j)}, \dots, d^{(j)} \rangle$ of P_μ with $\beta^{(j)} \leftrightarrow_{\mathcal{M}_\mu} b^{(k)}$. This is illustrated in Figure E.3a,

If, otherwise, the edge found is of the form $\beta^{(j)} \rightarrow \delta^{(j)}$. We can conclude that we have the bidirected edge $\delta^{(j)} \leftrightarrow d^{(k)}$ by applying the Lemmas E.2 and E.4 again.

If there is a collider on the subpath between $\langle \delta^{(j)}, \dots, c^{(j)} \rangle$, then any such collider must be in $[C]$ since P_μ is d-connecting given $[C]$ (see Figure E.3b). Furthermore, one of these colliders will be a descendant of $\delta^{(j)}$, and we can apply similar logic to that in Case (ii) to show that the path obtained by replacing the segment $\langle \beta^{(j)}, \dots, d^{(j)} \rangle$ of P_μ with $\delta^{(j)} \leftrightarrow d^{(k)}$ is d-connecting given $[C]$.

Otherwise, no such collider exists between $\delta^{(j)}$ and $c^{(j)}$ and hence $c^{(j)}$ is a descendant of $\delta^{(j)}$ (see Figure E.3c). Therefore, $b^{(k)}$ is a descendant of $\delta^{(k)}$ by the ordering compatibility assumption, and Algorithm 1 adds the directed edge $\delta^{(k)} \rightarrow b^{(k)}$ since $\delta^{(k)}$ and $b^{(k)}$ will both be in $\text{ch}_{\mathcal{D}_\mu}(y)$. This further implies that $\delta^{(j)}, b^{(j)} \in \text{ch}_{\mathcal{D}_\mu}(y)$, so Algorithm 1 will add an edge between these two nodes. The ordering assumption once again ensures that this edge is of the form $\delta^{(j)} \rightarrow b^{(j)}$.

- (iv) case $c \leftarrow d \leftarrow b$ in $\mathcal{M}_\cup$. Proceeding similarly, if we have the edge $\beta^{(j)} \rightarrow \delta^{(j)}$, then we can follow the same logic to create the d-connecting path (see Figure E.4a).

Otherwise, $\beta^{(j)} \leftarrow \delta^{(j)}$, and we have the bidirected edge $\beta^{(j)} \leftrightarrow d^{(k)}$, and we again check for colliders between $\beta^{(j)}$ and $d^{(j)}$.

If there is a collider, it will be both in $[C]$ and a descendant of $d^{(k)}$ in $\mathcal{M}_\mu$, and we can find the desired d-connecting path with the same logic followed previously (see Figure E.4b).

If there is no such collider, then $\beta^{(j)}$ will be a descendant of $d^{(j)}$, and using a similar argument to that used for Figure E.3c, we can conclude that we have directed edges $\beta^{(j)} \leftarrow_{\mathcal{M}_\mu} d^{(j)}$ and $\beta^{(k)} \leftarrow_{\mathcal{M}_\mu} d^{(k)}$ (see Figure E.4c). In such a scenario, we can repeat the logic for the node β in place of the node c : we continue walking along the path $P_\mu^{(j)}$ starting from $\beta^{(j)}$ until $\alpha^{(j)}$ is reached or until we find another edge along this path that does not exist on $P_\mu^{(k)}$. If the former happens first, we deal with the case like we would have if $P_\mu^{(k)}$ and $P_\mu^{(j)}$ had identical edges. If the latter happens first, then we recursively repeat the logic of case (iv).

This completes the proof. $\square$

E.2. A d-connecting path in $\mathcal{M}_\mu$ implies an analogous d-connecting path in $\mathcal{M}_\cup$

Again, we begin with some auxiliary results.

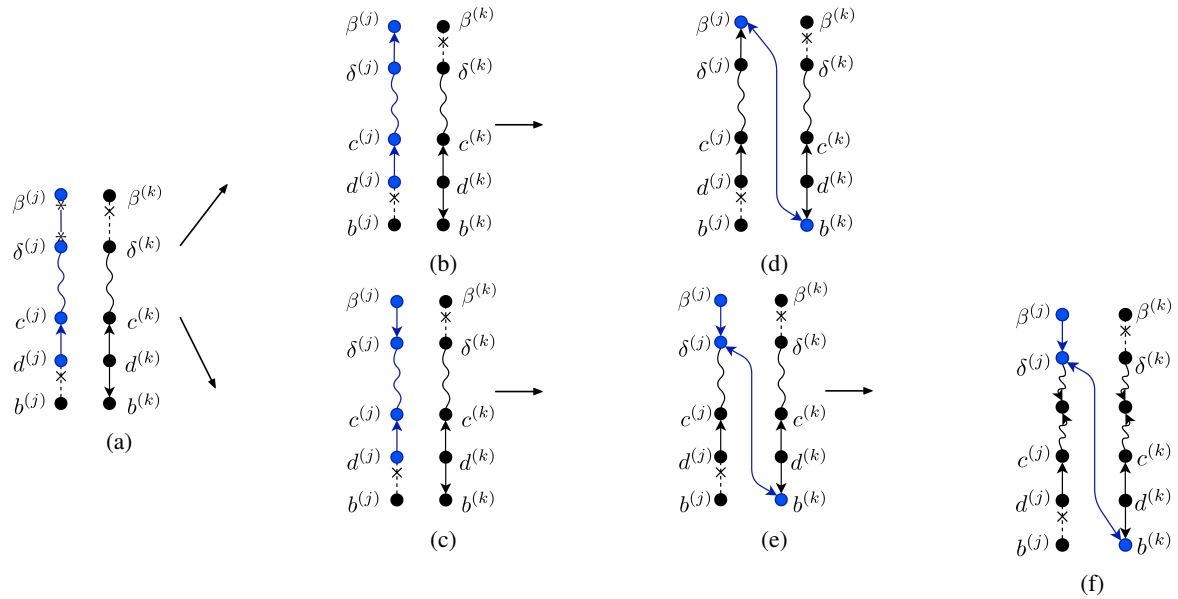


Figure E.2: An illustration of the logic in the proof of Lemma E.5, case (ii). We do not plot all possible edges in order to reduce clutter. Instead, we plot non-edges using an $\times$ superimposed on a dashed line. Furthermore, we indicate paths between two nodes with a squiggly line. (a), (b) and (c) show the relevant segment of the path P_μ in blue; (d), (e) and (f) show the segment that replaces $\langle \beta^{(j)}, \dots, d^{(j)} \rangle$ on P_μ to create the desired d-connecting path in blue.

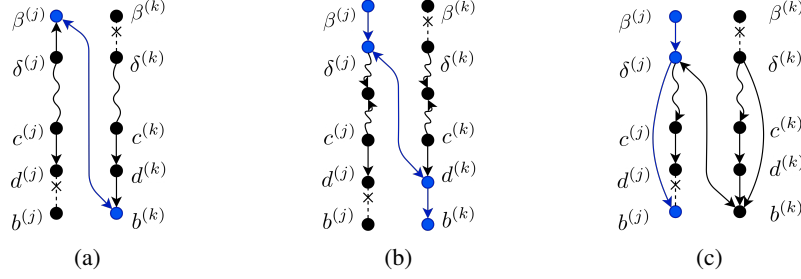


Figure E.3: An illustration of the d-connecting paths constructed by following the logic of case (iii) in the proof of Lemma E.5. In each of (a), (b) and (c), the segment that replaces $\langle \beta^{(j)}, \dots, d^{(j)} \rangle$ on P_μ to create the desired d-connecting path is colored in blue.

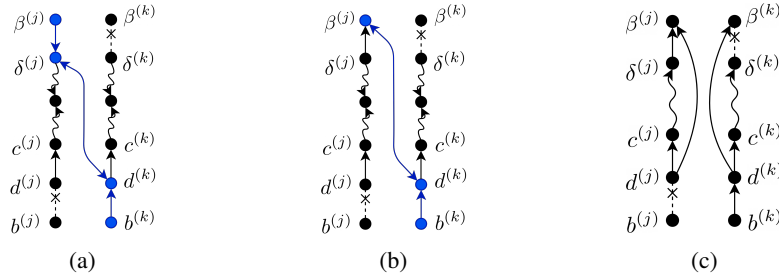


Figure E.4: An illustration of the d-connecting paths constructed by following the logic of case (iv) in the proof of lemma E.5. In each of (a) and (b), the segment that replaces $\langle \beta^{(j)}, \dots, d^{(j)} \rangle$ on P_μ to create the desired d-connecting path is colored in blue.

Lemma E.6 (At most 1 bidirected edge). *If there exists a connecting path between $a^{(i)}$ and $b^{(k)}$ given some $[C]$ where $a, b \in V$ and $C \subseteq V \setminus \{a, b\}$ in $\mathcal{M}_\mu$, then there must exist a path $\tilde{P}_\mu$ between $a^{(i)}$ and $b^{(k)}$ that is also connecting given $[C]$ that contains at most one bidirected edge.*

Proof. Since $a^{(i)}$ and $b^{(k)}$ are connected given $[C]$ in $\mathcal{M}_\mu$, then they must also be connected given $[C]$ in $\mathcal{D}_\mu$. Let P_μ denote the path connecting $a^{(i)}$ to $b^{(k)}$ given $[C]$ in $\mathcal{D}_\mu$. Let $P_\mu = \langle a^{(i)}, u_1, \dots, u^{(l)} \rangle$ and let u_x, u_z be the first and last occurrences of the vertex y on P_μ , respectively, if any. Since y has an in-degree of 0, neither u_x nor u_z can be a collider. Hence, we can concatenate the paths $P_1 = \langle a^{(i)}, \dots, u_x \rangle$ and $P_2 = \langle u_z, \dots, b^{(k)} \rangle$ to get a connecting path given $[C]$ in $\mathcal{D}_\mu$.

Now, if u_{x-1} is neither an ancestor nor a descendant of u_{z+1} , then in $\mathcal{M}_\mu$, we will have the path $a^{(i)}, \dots, u_{x-1}, u_{z+1}, \dots, b^{(k)}$ by virtue of Algorithm 1, since it adds a bidirected edge between any pair of children of y . This is a path from $a^{(i)}$ to $b^{(k)}$ that is also connected given $[C]$ that contains only 1 bidirected edge.

Otherwise, (W.L.O.G) $u_{x-1} \in \text{an}_{\mathcal{D}_\mu}(u_{z+1})$, i.e., there is a directed path from u_{x-1} to u_{z+1} in $\mathcal{D}_\mu$. Step 3 of Algorithm 1 adds the edge $u_{x-1} \rightarrow u_{z+1}$ to $\mathcal{M}_\mu$ to create the path $\tilde{P}_\mu := \langle a^{(i)}, \dots, u_{x-1}, u_{z+1}, \dots, b^{(k)} \rangle$. This path is from $a^{(i)}$ to $b^{(k)}$ and passes through no bidirected edges. If this path is active, then we are done. If this path is not active, then, since $\langle a^{(i)}, \dots, u_{x-1} \rangle$ and $\langle u_{z+1}, \dots, b^{(k)} \rangle$ are active, $\tilde{P}_\mu$ must be inactive by virtue of $u_{x-1} \in [C]$. But since P_μ in $\mathcal{D}_\mu$ is connecting, this implies that u_{x-1} must have been a collider on that path, hence we have the edge $u_{x-2} \rightarrow u_{x-1}$ in $\mathcal{D}_\mu$ and $\mathcal{M}_\mu$. Step 2 of Algorithm 1 adds $u_{x-2} \rightarrow u_{z+1}$ in such a case. Then, the path $\langle a^{(i)}, \dots, u_{x-1}, u_{z+1}, \dots, b^{(k)} \rangle$ must be connecting from $a^{(i)}$ to $b^{(k)}$ given $[C]$, which completes the proof. $\square$

Lemma E.7 (A Connecting Path in $\mathcal{M}_\mu$ implies a connecting path in $\mathcal{M}_\cup$). *Under the assumption in Definition 4.1, if there is a connecting path between $a^{(i)}$ and $b^{(k)}$ given some $[C]$ in $\mathcal{M}_\mu$ for some $i, k \in \{1, \dots, K\}$, where $C \subseteq V \setminus \{a, b\}$, then there is a connecting path between a and b given C in $\mathcal{M}_\cup$.*

Proof. By Lemma E.6, we must have a connecting path in $\mathcal{M}_\mu$ between $a^{(i)}$ and $b^{(k)}$ given $[C]$ that passes through at most 1 bidirected edge. If there exist paths that pass through no bidirected edges, take any such path. Otherwise, take any path

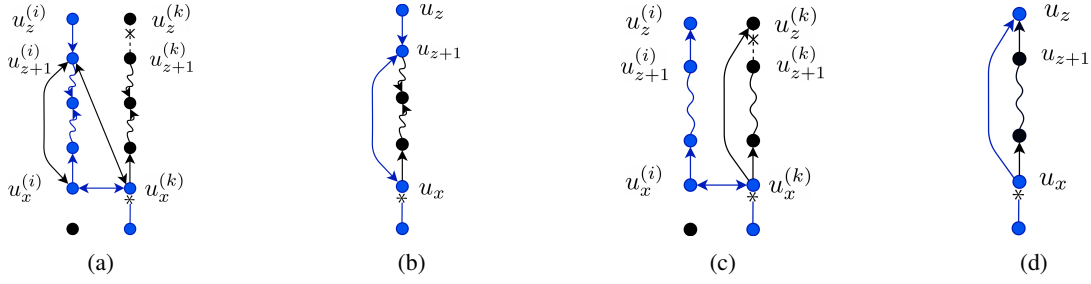


Figure E.5: An illustration of the logic for case (i) for the proof of Lemma E.7. In (a) and (c), we color in blue the relevant segments of the d-connecting path in $\mathcal{M}_\mu$, while in (b) and (d), we color in blue the relevant segments of the constructed d-connecting path in $\mathcal{M}_\cup$.

that passes through 1 bidirected edge. Call this path $P_\mu = \langle a^{(i)} =: u_0^{(i)}, u_1^{(i)}, \dots, u_m^{(i)} =: b^{(k)} \rangle$.

By the structure of $\mathcal{M}_\mu$ discussed in the beginning of this section, only a bidirected edge can connect a node $u_x^{(i)}$ to a node $u_{x+1}^{(k)}$ in $\mathcal{M}_\mu$ for $i \neq k$. Hence, if there is no bidirected edge on this path, then all the nodes $u_0^{(i)}, \dots, u_m^{(i)}$ will be contained in the same MAG $\mathcal{M}^{(i)}$. Each edge along this d-connecting path given $[C]$ will show up in $\mathcal{M}_\cup$, and hence we can create a path $\langle u_0, \dots, u_m \rangle$ that is d-connecting given C in $\mathcal{M}_\cup$.

In the case where P_μ contains a bidirected edge, let us label the nodes incident as $u_x^{(i)} \leftrightarrow u_{x+1}^{(k)}$. The segments $\langle u_0^{(i)}, \dots, u_x^{(i)} \rangle$ and $\langle u_{x+1}^{(k)}, \dots, u_m^{(k)} \rangle$ will each be contained in $\mathcal{M}^{(i)}$ and $\mathcal{M}^{(k)}$ respectively, and hence we can find d-connecting paths $\langle u_0, \dots, u_x \rangle$ and $\langle u_{x+1}, \dots, u_m \rangle$ in $\mathcal{M}_\cup$ that are each d-connecting given C . We must now show that we can connect these paths to create a d-connecting path given C from $u_0 = a$ to $u_m = b$ in $\mathcal{M}_\cup$.

Of course, there is no difficulty if the bidirected edge $u_x \leftrightarrow u_{x+1}$ appears in $\mathcal{M}_\cup$, since we can connect these two subpaths with this bidirected edge and have the desired connecting path. The difficulty is when this edge does not appear. From the definition of $\mathcal{M}_\cup$, we can see that this only happens when the bidirected edge connects $u_x^{(i)}$ and $u_{x+1}^{(k)}$ for $i \neq k$, i.e., the bidirected edge is not contained in any MAG $\mathcal{M}^{(j)}$ for any j . We split the remainder into two cases.

- (i) case $u_x = u_{x+1}$. If $u_x^{(i)}$ and $u_{x+1}^{(k)}$ are both colliders on P_μ , then we must have $u_x, u_{x+1} \in C$. Then $c = d$ will be an active collider given C in $\mathcal{M}_\cup$ on the path obtained by concatenating $\langle u_0, \dots, u_x \rangle$ and $\langle u_{x+1}, \dots, u_m \rangle$ in $\mathcal{M}_\cup$, and hence we have our d-connecting path given C . We therefore assume, W.L.O.G., that $u_x^{(i)}$ is not a collider on P_μ .

If there is a path $\langle u_0^{(k)}, \dots, u_x^{(k)} \rangle$ in $\mathcal{M}_\mu$ where every pair of adjacent vertices $u_n^{(k)}, u_{n+1}^{(k)}$ on this path are connected by the same edge type as the pair $u_n^{(i)}, u_{n+1}^{(i)}$ in P_μ , then we can replace the segment of $\langle u_0^{(i)}, \dots, u_x^{(i)} \rangle$ of P_μ with $\langle u_0^{(k)} \dots u_x^{(k)} \rangle$ to obtain a path that is d-connecting given $[C]$ and contained completely in $\mathcal{M}^{(k)}$, meaning that we can find the desired d-connecting path given C in $\mathcal{M}_\cup$. If no such path exists in $\mathcal{M}_\mu$, then starting at $u_x^{(i)}$ and walking backwards along P_μ towards $u_0^{(i)}$, we will find an edge $u_z^{(i)} \leftrightarrow_{\mathcal{M}_\mu} u_{z+1}^{(i)}$ where $u_z^{(k)} \leftrightarrow_{\mathcal{M}_\mu} u_{z+1}^{(k)}$ is not an edge. Take the first such edge found (i.e., the edge closest to $u_x^{(i)}$ that satisfies this; see Figure E.5a).

If $u_z^{(i)} \rightarrow_{\mathcal{M}_\mu} u_{z+1}^{(i)}$, then by Lemmas E.4 and E.2, there is a bidirected edge $u_{z+1}^{(i)} \leftrightarrow_{\mathcal{M}_\mu} u_x^{(k)}$, implying that step 1 of Algorithm 1 adds another bidirected edge $u_{z+1}^{(k)} \leftrightarrow u_x^{(k)}$. If $u_{z+1}^{(i)}$ is not a descendant of $u_x^{(i)}$, then the bidirected edge $u_{z+1}^{(k)} \leftrightarrow_{\mathcal{M}_\mu} u_x^{(k)}$ would not be removed by step 3 of Algorithm 1 and hence will appear in $\mathcal{M}_\mu$. Furthermore, we will have colliders $\alpha^{(i)}$ and $\gamma^{(i)}$ between $u_{z+1}^{(i)}$ and $u_x^{(i)}$ that are in $[C]$ that will be descendants of $u_{z+1}^{(i)}$ and $u_x^{(i)}$ respectively. The ordering assumption ensures that α and γ are descendants of u_{z+1} and u_x in $\mathcal{M}_\cup$, respectively. Hence, the path $\langle u_0, \dots, u_{z+1}, u_x, \dots, u_m \rangle$ in $\mathcal{M}_\cup$ is d-connecting in $\mathcal{M}_\cup$ given C . Figures E.5a and E.5b illustrate this.

Now we check the case where $u_z^{(i)} \leftarrow u_{z+1}^{(i)}$. If $u_z^{(i)}$ is not a descendant of $u_x^{(i)}$, then we can construct a path in $\mathcal{M}_\cup$ by a similar argument to the above. If $u_z^{(i)}$ is a descendant of $u_x^{(i)}$, then by Lemma E.2, there is a directed edge $u_z^{(i)} \leftarrow_{\mathcal{M}_\mu} u_x^{(i)}$, which appears as $u_z \leftarrow_{\mathcal{M}_\cup} u_x$. We can use this to construct a path in $\mathcal{M}_\cup$ as shown in Figures E.5c

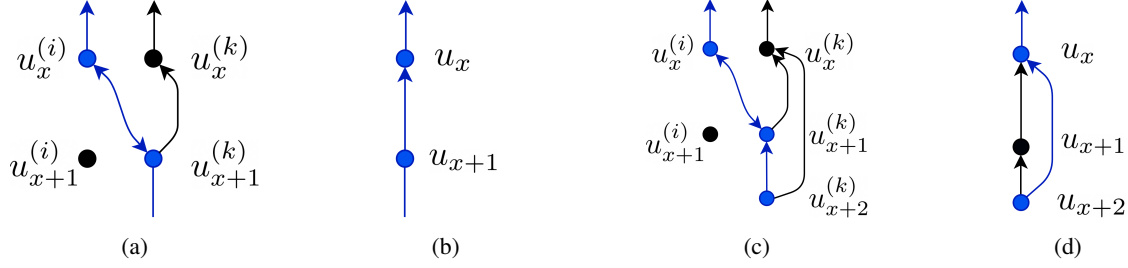


Figure E.6: An illustration of the logic for case (ii) for the proof of Lemma E.7. In (a) and (c), we color in blue the relevant segments of the d-connecting path in $\mathcal{M}_\mu$, while in (b) and (d), we color in blue the relevant segments of the constructed d-connecting path in $\mathcal{M}_\cup$.

and E.5d. This path is active since $u_x^{(i)}$ is not a collider, and hence $u_x^{(i)} \notin [C]$, implying that $u_x \notin C$.

- (ii) case $u_x \neq u_{x+1}$: Step 1 of Algorithm 1 adds the bidirected edge $u_x^{(k)} \leftrightarrow u_{x+1}^{(k)}$, which will show up in $\mathcal{M}_\cup$ as an edge $u_x \leftrightarrow u_{x+1}$ unless it is removed by step 3; so this is the only case we must check. Assume W.L.O.G. that this edge is removed by step 3 because $u_x^{(k)}$ is a descendant of $u_{x+1}^{(k)}$ in $\mathcal{D}_\mu$ and therefore in $\mathcal{M}_\mu$. Then a directed edge $u_x^{(i)} \leftarrow u_{x+1}^{(i)}$ will be added instead, which appears in $\mathcal{M}_\cup$ as $u_x \leftarrow u_{x+1}$. The only case where we cannot join $\langle u_0, \dots, u_x \rangle$ and $\langle u_{x+1}, \dots, u_m \rangle$ in $\mathcal{M}_\cup$ together using this directed edge $u_x \leftarrow u_{x+1}$ to create a d-connected path given C is when $u_{x+1}^{(k)}$ is in $[C]$, and hence is a collider on P_μ . This implies that we have $u_{x+1}^{(k)} \leftarrow_{\mathcal{M}_\mu} u_{x+2}^{(k)}$, in which case step 2 of Algorithm 1 would have added the edge $u_{x+1}^{(k)} \leftarrow u_{x+2}^{(k)}$, which appears as $u_{x+1} \leftarrow_{\mathcal{M}_\cup} u_{x+2}$. This edge can be used to create the d-connecting path given C given by $\langle u_0, \dots, u_x, u_{x+2}, \dots, u_m \rangle$ in $\mathcal{M}_\cup$. This is illustrated in Figure E.6 and completes the proof. $\square$

E.3. The main result

Finally, we use the results of the first two steps to prove the following.

Theorem E.8. *Under the assumption in Definition 4.1, for any disjoint $A, B, C \subseteq V$, $[A]$ and $[B]$ are d-separated given $[C]$ in $\mathcal{D}_\mu$ if and only if A and B are d-separated given C in $\mathcal{M}_\cup$.*

Proof. Since $\mathcal{M}_\mu$ is the marginal MAG in $\mathcal{D}_\mu$ with respect to the vertex y , the d-separation statements involving subsets not including y are the same in both. By proposition 2.1, $\mathcal{M}_\mu$ is a MAG, hence d-separation in $\mathcal{M}_\mu$ is compositional (Sadeghi and Lauritzen, 2014); therefore for $A, B, C \subseteq V$ disjoint it holds that

$$\begin{aligned} & \left\{ [A] \text{ sep from } [B] \text{ in } \mathcal{M}_\mu \text{ given } [C] \right\} \\ & \Leftrightarrow \left\{ a^i \text{ sep from } b^k \text{ in } \mathcal{M}_\mu \text{ given } [C] \text{ for all } a^i \in [A], b^k \in [B] \right\}. \end{aligned}$$

Now Lemmas E.5 and E.7 imply

$$\begin{aligned} & \left\{ a^i \text{ sep from } b^k \text{ in } \mathcal{M}_\mu \text{ given } [C] \text{ for all } a^i \in [A], b^k \in [B] \right\} \\ & \Leftrightarrow \left\{ a \text{ sep from } b \text{ given } C \text{ for all } a \in A, b \in B \right\}. \end{aligned}$$

Finally, since $\mathcal{M}_\cup$ is a MAG, applying compositionality gives

$$\left\{ a \text{ sep from } b \text{ in } \mathcal{M}_\cup \text{ given } C \text{ for all } a \in A, b \in B \right\} \Leftrightarrow \left\{ A \text{ sep from } B \text{ given } C \text{ in } \mathcal{M}_\cup \right\},$$

which completes the proof. $\square$

F. Additional Experimental Results

F.1. Synthetic Data

In the following, we present figures for the experiments described in Section 5 for additional values of K and n . Figure F.1 shows the normalized SHD plot in evaluating the union graph as described in the main paper, while Figure F.2 shows the true and false positives in predicted $V \setminus V_{\text{INV}}$. Finally, Figure F.3 shows the result of K -means clustering.

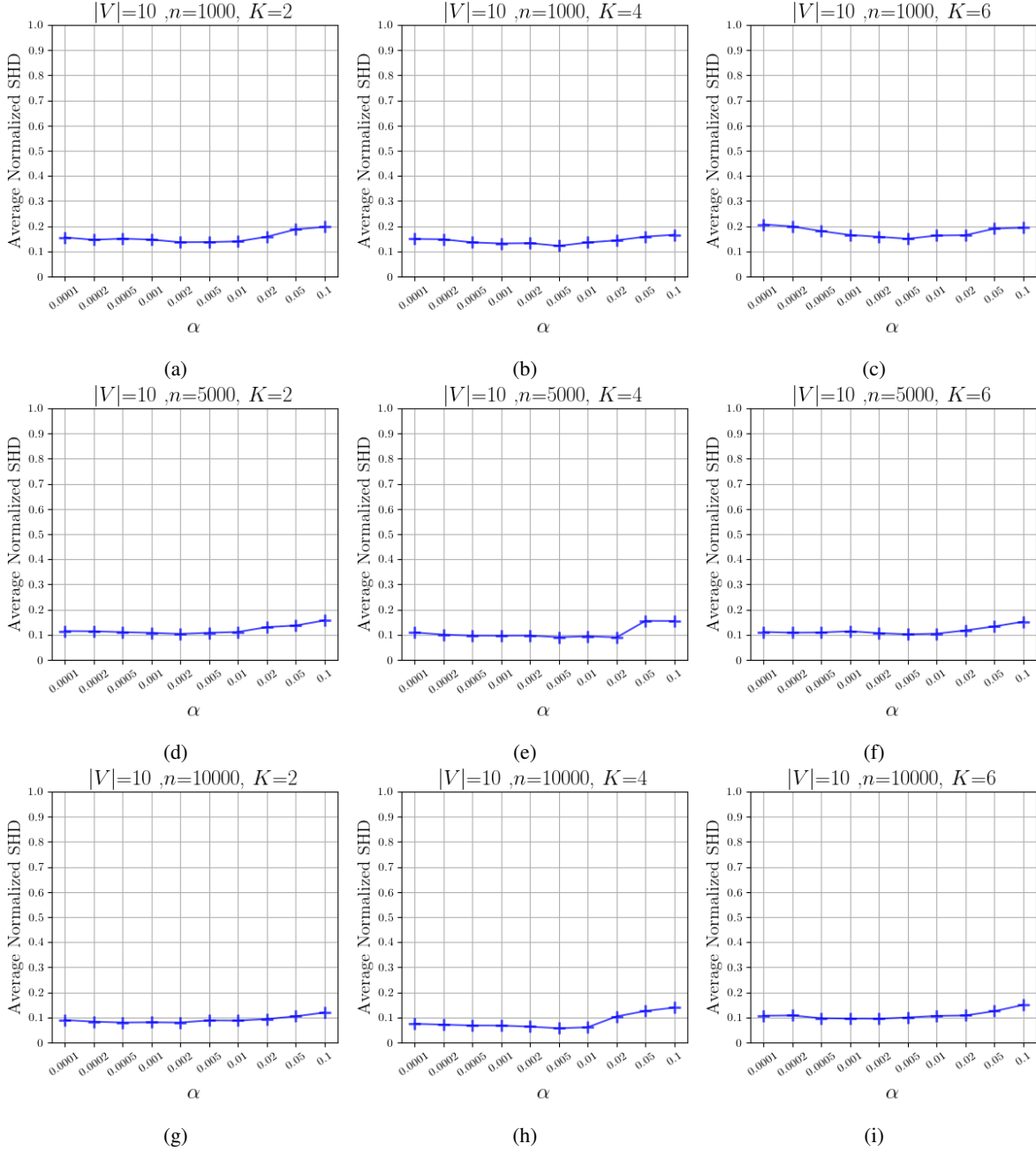


Figure F.1: Normalized SHD evaluating the estimation of the union graph from mixture data using FCI for $K \in \{2, 4, 6\}$ and $n \in \{1000, 5000, 10000\}$.



Figure F.2: True and false positive rates in estimating $V \setminus V_{INV}$ using Proposition 4.6 applied to the PAG $\hat{\mathcal{P}}_{\cup}$ estimated by running FCI on the mixture data. The figures show the results for $K \in \{2, 4, 6\}$ and $n \in \{1000, 5000, 10000\}$.

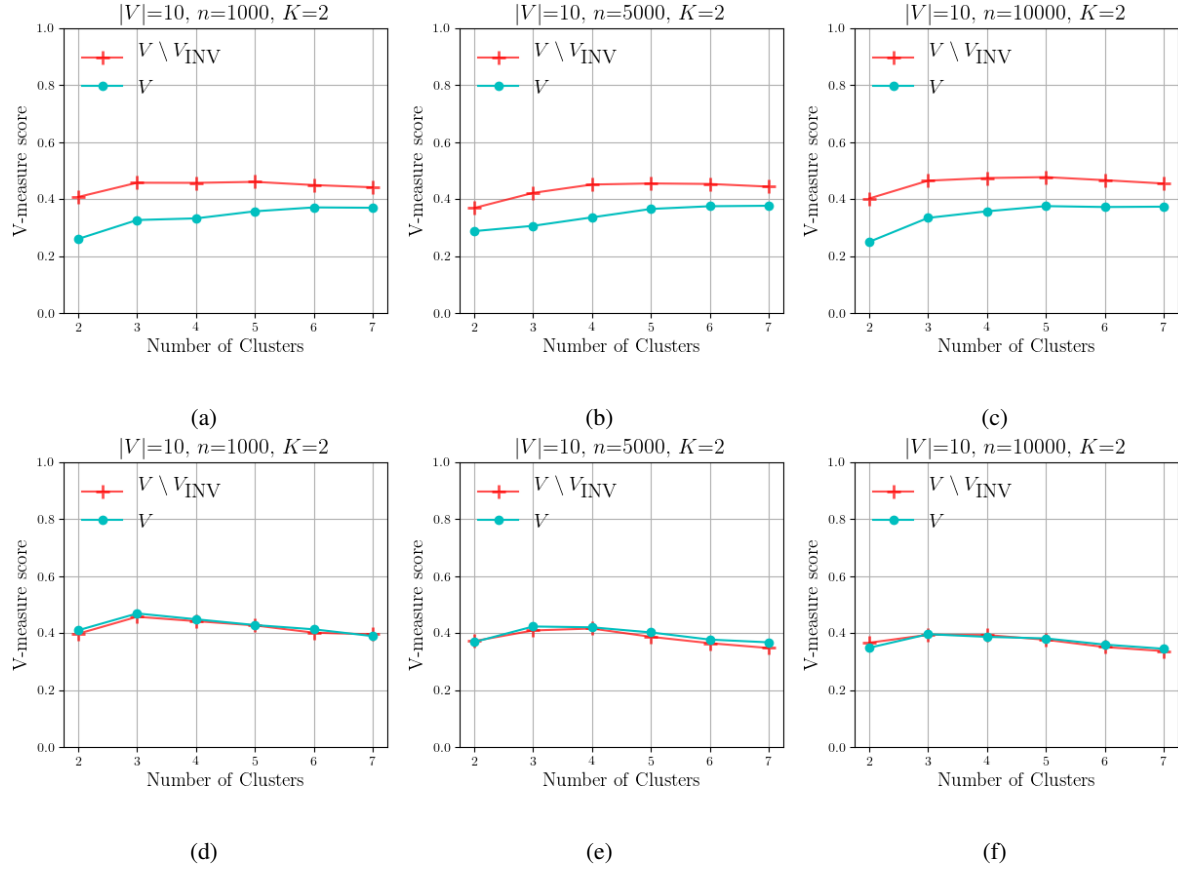


Figure F.3: A comparison of clustering when all the variables are used as features vs. when only the variables in the estimated set $V \setminus V_{INV}$ are used as features. In generating figures (a), (b) and (c), $V \setminus V_{INV}$ has descendants in the generating model, while in figures (d), (e) and (f), $V \setminus V_{INV}$ has no descendants.

F.2. Real Data

Here, we present the output of FCI on the T cell mixture data referenced in section 5.2.

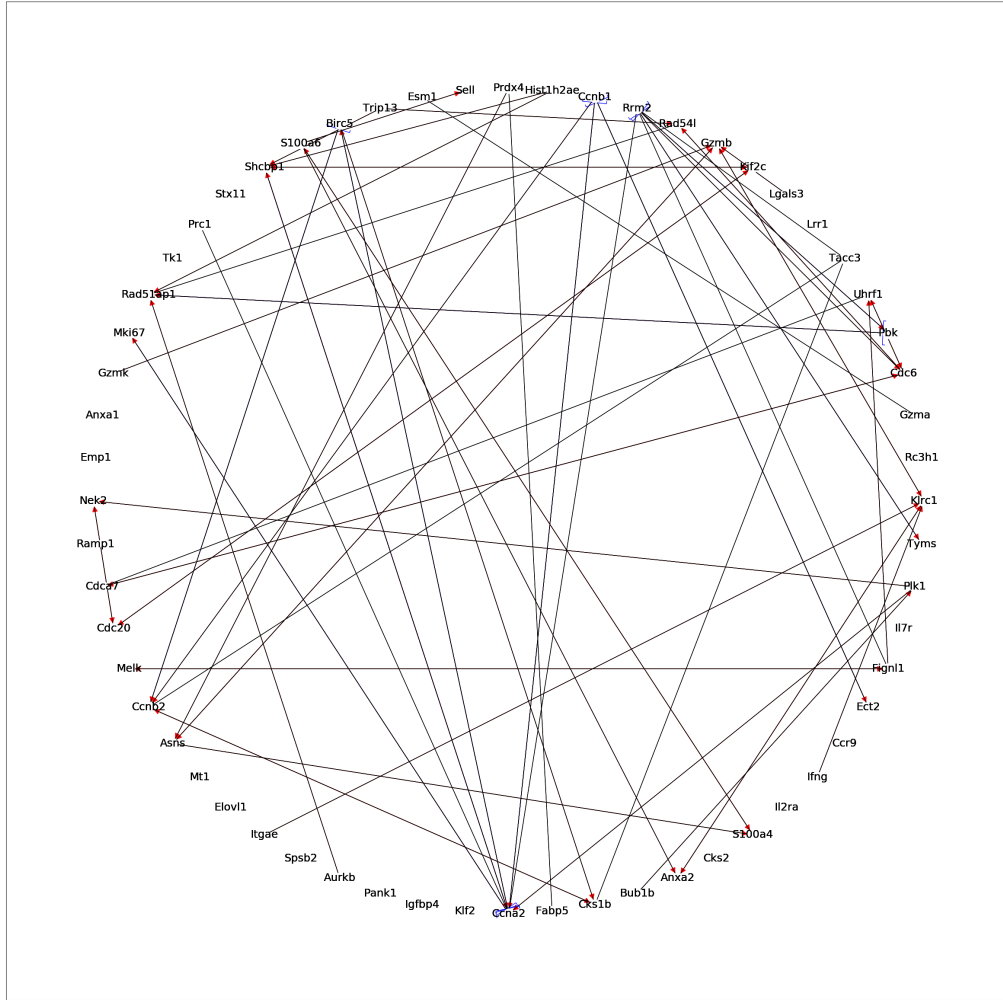


Figure F.4: The PAG learned using FCI on the T cell mixture data. The inferred arrowheads are shown in red, while the inferred arrowtails are shown as blue brackets.