Learning bounded degree polytrees with samples

Anonymous Author(s)
Affiliation
Address
email

Abstract

We establish finite-sample guarantees for efficient proper learning of bounded-degree *polytrees*, a rich class of high-dimensional probability distributions and a subclass of Bayesian networks, a widely-studied type of graphical models. Very recently, Bhattacharyya et al. [2021] obtained finite-sample guarantees for recovering tree-structured Bayesian networks, i.e., 1-polytrees. We considerably extend their results by providing an efficient algorithm which learns d-polytrees in polynomial time and sample complexity when the in-degree d is constant, provided that the underlying undirected graph (skeleton) is known. We complement our algorithm with an information-theoretic lower bound, showing that the dependence of our sample complexity is nearly tight in both the dimension and target accuracy parameters.

1 Introduction

Distribution learning, or density estimation, is the task of obtaining a good estimate of some unknown underlying probability distribution P from observational samples. Understanding which classes of distributions could be or could not be learnt efficiently is a fundamental problem in both computer science and statistics, where efficiency is measured in terms of *sample* (data) and *computational* (time) complexities.

Bayesian networks (or *Bayes nets* in short) represent a class of high-dimensional distributions that can be explicitly described by how each variable is be generated sequentially in a directed fashion. Being interpretable, Bayes nets have been used to model beliefs in a wide variety of domains (e.g. see [Jensen and Nielsen, 2007, Koller and Friedman, 2009] and references therein). A fundamental problem in computational learning theory is to identify families of Bayes nets which can be learned efficiently from observational data.

Formally, a Bayes net is a probability distribution P, defined over some directed acyclic graphs (DAG) $G = (V, E)$ on $|V| = n$ nodes that factorizes according to G (i.e. Markov with respect to G) in the following sense: $P(v_1, \ldots, v_n) = \prod_{v_1,\ldots,v_n} P(v \mid \pi(v))$, where $\pi(v) \subseteq V$ are the parents of v in G. While it is well-known that given the DAG (structure) of a Bayes net, there exists sample-efficient algorithms that output good hypotheses [Dasgupta, 1997, Bhattacharyya et al., 2020], there is no known computationally efficient algorithms for obtaining the DAG of a Bayes net. In fact, it has long been understood that Bayes net structure learning is computationally expensive, in various general settings [Chickering et al., 2004]. However, these hardness results fall short when the goal is learning the distribution P in the probabilistically approximately correct (PAC) [Valiant, 1984] sense (with respect to, say, KL divergence or total variation distance), rather than trying to recover an exact graph from the Markov equivalence class of P.

Polytrees are a subclass of Bayesian networks where the undirected graph underlying the DAG is a forest, i.e., there is no cycle for the undirected version of the DAG; a polytree with maximum in-degree d is also known as a d-polytree. With an infinite number of samples, one can recover the

Submitted to 37th Conference on Neural Information Processing Systems (NeurIPS 2023). Do not distribute.

DAG of a non-degenerate polytree in the equivalence class with the Chow–Liu algorithm [Chow and Liu, 1968] and some additional conditional independence tests [Rebane and Pearl, 1988]. However, this algorithm does *not* work in the finite sample regime. The only known result for learning polytrees with finite sample guarantees is for 1-polytrees [Bhattacharyya et al., 2021]. Furthermore, in the agnostic setting, when the goal is to find the closest polytree distribution to an arbitrary distribution P, the learning problem becomes NP-hard [Dasgupta, 1999].

In this work, we investigate what happens when the given distribution is a d-polytree, for $d > 1$. *Are d-polytrees computationally hard to learn in the realizable PAC-learning setting?* One motivation for studying polytrees is due to a recent work of Gao and Aragam [2021] which shows that polytrees are easier to learn than general Bayes nets due to the underlying graph being a tree, allowing typical causal assumptions such as faithfulness to be dropped when designing efficient learning algorithms.

Contributions. Our main contribution is a sample-efficient algorithm for proper Bayes net learning in the realizable setting, when provided with the ground truth skeleton (i.e., the underlying forest). Crucially, our result does not require any distributional assumptions such as strong faithfulness, etc.

Theorem 1. *There exists an algorithm which, given m samples from a polytree P over Σ^n, accuracy parameter $\varepsilon > 0$, failure probability δ, as well as its maximum in-degree d and the explicit description of the ground truth skeleton of P, outputs a d-polytree $\hat{P}$ such that $d_{\mathrm{KL}}(P \parallel \hat{P}) \leq \varepsilon$ with success probability at least $1 - \delta$, as long as*

$$m = \tilde{\Omega}\left(\frac{n \cdot |\Sigma|^{d+1}}{\varepsilon} \log \frac{1}{\delta}\right) .$$

Moreover, the algorithm runs in time polynomial in m, $|\Sigma|^d$, and n^d.

We remark that our result holds when even given only an upper bound on the true in-degree d. In particular, our result provides, for constant $|\Sigma|$, d, an upper bound of $\tilde{O}(n/\varepsilon)$ on the sample complexity of learning $O(1)$-polytrees. Note that this dependence on the dimension n and the accuracy parameter ε are optimal, up to logarithmic factors: indeed, we establish in Theorem 15 an $\Omega(n/\varepsilon)$ sample complexity lower bound for this question, even for $d = 2$ and $|\Sigma| = 2$.[1]

We also state sufficient conditions on the distribution that enable recovery of the ground truth skeleton. Informally, we require that the data processing inequality hold in a strong sense with respect to the edges in the skeleton graph. Under these conditions, combining with our main result in Theorem 1, we obtain a polynomial-time PAC algorithm to learn bounded-degree polytrees from samples.

Other related work. Structure learning of Bayesian networks is an old problem in machine learning and statistics that has been intensively studied; see, for example, Chapter 18 of Koller and Friedman [2009]. Many of the early approaches required faithfulness, a condition which permits learning of the Markov equivalence class, e.g. Spirtes and Glymour [1991], Chickering [2002], Friedman et al. [2013]. Finite sample complexity of such algorithms assuming faithfulness-like conditions has also been studied, e.g. Friedman and Yakhini [1996]. An alternate line of more modern work has considered various other distributional assumptions that permits for efficient learning, e.g., Chickering and Meek [2002], Hoyer et al. [2008], Shimizu et al. [2006], Peters and Bühlmann [2014], Ghoshal and Honorio [2017], Park and Raskutti [2017], Aragam et al. [2019], with the latter three also showing analyzing finite sample complexity. Specifically for polytrees, Rebane and Pearl [1988], Geiger et al. [1990] studied recovery of the DAG for polytrees under the infinite sample regime. Gao and Aragam [2021] studied the more general problem of learning Bayes nets, and their sufficient conditions simplified in the setting of polytrees. Their approach emphasize more on the exact recovery, and thus the sample complexity has to depend on the minimum gap of some key mutual information terms. In contrast, we allow the algorithm to make mistakes when certain mutual information terms are too small to detect for the given sample complexity budget and achieve a PAC-type guarantee. As such, once the underlying skeleton is discovered, our sample complexity only depends on the d, n, ε and not on any distributional parameters.

There are existing works on Bayes net learning with tight bounds in total variation distance, with a focus on sample complexity (and not necessarily computational efficiency); for instance, [Canonne et al., 2020]. Acharya et al. [2018] consider the problem of learning (in TV distance) a bounded-degree causal Bayesian network from interventions, assuming the underlying DAG is known.

[1]We remark that [Bhattacharyya et al., 2021, Theorem 7.6] implies an $\Omega(\frac{n}{\varepsilon} \log \frac{n}{\varepsilon})$ sample complexity lower bound for the analogous question when the skeleton is unknown and $d = 1$.

Outline of paper. We begin with some preliminary notions and related work in Section 2. Section 3 then shows how to recover a polytree close in KL divergence, assuming knowledge of the skeleton and maximum in-degree. Section 4 gives sufficient conditions to recover the underlying skeleton from samples, while Section 5 provides our sample complexity lower bound. We conclude in Section 6 with some open directions and defer some full proofs to the appendix.

2 Preliminaries and tools from previous work

2.1 Preliminary notions and notation

We write the disjoint union as $\dot{\cup}$. For any set A, let $|A|$ denotes its size. We use hats to denote estimated quantities, e.g., $\hat{I}(X;Y)$ will be the estimated mutual information of $I(X;Y)$. We employ the standard asymptotic notation $O(\cdot)$, $\Omega(\cdot)$ $\Theta(\cdot)$, and write $\tilde{O}(\cdot)$ to omit polylogarithmic factors. Throughout, we identify probability distributions over discrete sets with their probability mass functions (pmf). We use d^* to denote the true maximum in-degree of the underlying polytree.

2.2 Probability distribution definitions

We begin by defining KL divergence and squared Hellinger distances for a pair of discrete distributions with the same support.

Definition 2 (KL divergence and squared Hellinger distance)**.** For distributions P, Q defined on the same discrete support $\mathcal{X}$, their KL divergence and squared Hellinger distances are defined as $d_{\mathrm{KL}}(P \parallel Q) = \sum_{x \in \mathcal{X}} P(x) \log \frac{P(x)}{Q(x)}$ and $d_{\mathrm{H}}^2(P, Q) = 1 - \sum_{x \in \mathcal{X}} \sqrt{P(x) \cdot Q(x)}$ respectively.

Abusing notation, for a distribution P on variables $X = \{X_1, \dots, X_n\}$, we write P_S to mean the projection of P to the subset of variables $S \subseteq X$ and P_G to mean the projection of P onto a graph G. More specifically, we have $P_G(x_1, \dots, x_n) = \prod_{x \in X} P(x \mid \pi_G(x))$ where $\pi_G(x)$ are the parents of x in G. Note that P_G is the closest distribution[2] on G to P in d_{KL}, i.e. $P_G = \mathrm{argmin}_{Q \in G}\, d_{\mathrm{KL}}(P \parallel Q)$. By Chow and Liu [1968], we also know that

$$d_{\mathrm{KL}}(P, P_G) = -\sum_{i=1}^{n} I(X_i; \pi_G(X_i)) - H(P_X) + \sum_{i=1}^{n} H(P_{X_i}) \,, \tag{1}$$

where H is the entropy function. Note that only the first term depends on the graph structure of G.

By maximizing the sum of mutual information (the negation of the first term in (1)), we can obtain an ε-approximated graph G such that $d_{\mathrm{KL}}(P \parallel P_G) \leq \varepsilon$. In the case of tree-structured distributions, this can be efficiently solved by using any maximum spanning tree algorithm; a natural generalization to bounded degree bayes nets remains open due to the hardness of solving the underlying optimization problem [Höffgen, 1993]. If any valid topological ordering of the target Bayes net P is present, then an efficient greedy approach is able to solve the problem.

Definition 3 ((Conditional) Mutual Information)**.** Given a distribution P, the mutual information of two random variables X and Y, supported on $\mathcal{X}$ and $\mathcal{Y}$ respectively, is defined as

$$I(X;Y) = \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} P(x,y) \cdot \log\left(\frac{P(x,y)}{P(x) \cdot P(y)}\right) .$$

Conditioning on a third random variable Z, supported on $\mathcal{Z}$, the conditional mutual information is defined as:

$$I(X;Y \mid Z) = \sum_{x \in \mathcal{X}, y \in \mathcal{Y}, z \in \mathcal{Z}} P(x,y,z) \cdot \log\left(\frac{P(x,y,z) \cdot P(z)}{P(x,z) \cdot P(y,z)}\right) .$$

By adapting a known testing result from [Bhattacharyya et al., 2021, Theorem 1.3], we can obtain the following corollary, which we will use. We provide the full derivation in the supplementary materials.

[2] One can verify this using Bhattacharyya et al. [2021, Lemma 3.3]: For any distribution Q defined on graph G, we have $d_{\mathrm{KL}}(P \parallel Q) - d_{\mathrm{KL}}(P \parallel P_G) = \sum_{v \in V} P(\pi_G(v)) \cdot d_{\mathrm{KL}}(P(v \mid \pi_G(v)) \parallel Q(v \mid \pi_G(v))) \geq 0$.

Corollary 4 (Conditional Mutual Information Tester, adapted from [Bhattacharyya et al., 2021, Theorem 1.3])**.** *Fix any $\varepsilon > 0$. Let (X, Y, Z) be three random variables over $\Sigma_X, \Sigma_Y, \Sigma_Z$ respectively. Given the empirical distribution $(\hat{X}, \hat{Y}, \hat{Z})$ over a size N sample of (X, Y, Z), there exists a universal constant $0 < C < 1$ so that for any*

$$N \geq \Theta\left(\frac{|\Sigma_X| \cdot |\Sigma_Y| \cdot |\Sigma_Z|}{\varepsilon} \cdot \log \frac{|\Sigma_X| \cdot |\Sigma_Y| \cdot |\Sigma_Z|}{\delta} \cdot \log \frac{|\Sigma_X| \cdot |\Sigma_Y| \cdot |\Sigma_Z| \cdot \log(1/\delta)}{\varepsilon}\right),$$

the following statements hold with probability $1 - \delta$:
(1) If $I(X; Y \mid Z) = 0$, then $\hat{I}(X; Y \mid Z) < C \cdot \varepsilon$.
(2) If $\hat{I}(X; Y \mid Z) \geq C \cdot \varepsilon$, then $I(X; Y \mid Z) > 0$.
(3) If $\hat{I}(X; Y \mid Z) \leq C \cdot \varepsilon$, then $I(X; Y \mid Z) < \varepsilon$.
Unconditional statements involving $I(X; Y)$ and $\hat{I}(X; Y)$ hold similarly by choosing $|\Sigma_Z| = 1$.

2.3 Graph definitions

Let $G = (V, E)$ be a graph on $|V| = n$ vertices and $|E|$ nodes where adjacencies are denoted with dashes, e.g. $u - v$. For any vertex $v \in V$, we use $N(v) \subseteq V \setminus \{v\}$ to denote the neighbors of v and $d(v) = |N(v)|$ to denote v's degree. An undirected cycle is a sequence of $k \geq 3$ vertices such that $v_1 - v_2 - \ldots - v_k - v_1$. For any subset $E' \subseteq E$ of edges, we say that the graph $H = (V, E')$ is the edge-induced subgraph of G with respect to E'.

For oriented graphs, we use arrows to denote directed edges, e.g. $u \to v$. We denote $\pi(v)$ to denote the parents of v and $d^{in}(v)$ to denote v's incoming degree. An interesting directed subgraph on three vertices is the v-structure, where $u \to v \leftarrow w$ and $u \not\sim w$; we say that v is the center of the v-structure. In this work, we study a generalized higher-degree version of v-structures: we define the notion of *deg-ℓ v-structure* as a node v with $\ell \geq 2$ parents $u_1, u_2 \ldots, u_\ell$. We say that a deg-ℓ v-structure is said to be ε-strong if we can reliably identify them in the finite sample regime. In our context, it means that for all $k \in [\ell]$, $I(u_k; \{u_1, u_2 \ldots, u_\ell\} \setminus u_k \mid v) \geq C \cdot \varepsilon$, for the universal constant $0 < C < 1$ appearing in Corollary 4. A directed acyclic graph (DAG) is a fully oriented graph without any directed cycles (a sequence of $k \geq 3$ vertices such that $v_1 \to v_2 \to \ldots \to v_k \to v_1$) and are commonly used to represent the conditional dependencies of a Bayes net.

For any partially directed graph, an *acyclic completion* or *consistent extension* refers to an assignment of edge directions to unoriented edges such that the resulting fully directed graph has no directed cycles; we say that a DAG G is *consistent* with a partially directed graph H if G is an acyclic completion of H. Meek rules are a set of 4 edge orientation rules that are sound and complete with respect to any given set of arcs that has a consistent DAG extension Meek [1995]. Given any edge orientation information, one can always repeatedly apply Meek rules till a fixed point to maximize the number of oriented arcs. One particular orientation rule (Meek $R1$) orients $b \to c$ whenever a partially oriented graph has the configuration $a \to b - c$ and $a \not\sim c$ so as to avoid forming a new v-structure of the form $a \to b \leftarrow c$. In the same spirit, we define Meek $R1(d^*)$ to orient all incident unoriented edges away from v whenever v already has d^* parents in a partially oriented graph.

The *skeleton* $\text{skel}(G)$ of a graph G refers to the resulting undirected graph after unorienting all edges in G, e.g. see Fig. 1. A graph G is a *polytree* if $\text{skel}(G)$ is a forest. For $d \geq 1$, a polytree G is a d-polytree if all vertices in G have at most d parents. Without loss of generality, by picking the minimal d, we may assume that d-polytrees have a vertex with d parents. When we *freely orient* a forest, we pick arbitrary root nodes in the connected components and orient to form a 1-polytree.

3 Recovering a good orientation given a skeleton and degree bound

In this section, we describe and analyze an algorithm for estimating a probability distribution P that is defined on a d^*-polytree G^*. We assume that we are given $\text{skel}(G^*)$ and d as input.

Note that for some distributions there could be more than one ground truth graph, e.g. when the Markov equivalence class has multiple graphs. In such situations, for analysis purposes, we are free to choose any graph that P is Markov with respect to. As the mutual information scores[3] are the same for any graphs that P is Markov with respect to, the choice of G^* does not matter here.

[3]The mutual information score is the sum of the mutual information terms as in Eq. (1).

(a) G^* (b) $\mathrm{skel}(G^*)$

Figure 1: Running polytree example with $d^* = 3$ where $I(a; b, c) = I(b; a, c) = I(c; a, b) = 0$ due to the deg-3 v-structure centered at d. Since $I(a; f \mid d) = 0$, Corollary 4 tells us that $\hat{I}(a; f \mid d) \leq C \cdot \varepsilon$. Thus, we will *not* detect $a \to d \to f$ erroneously as a strong deg-2 v-structure $a \to d \leftarrow f$.

3.1 Algorithm

At any point in the algorithm, let us define the following sets. Let $N(v)$ be the set of all neighbors of v in $\mathrm{skel}(G^*)$. Let $N^{in}(v) \subseteq N(v)$ be the current set of incoming neighbors of v. Let $N^{out}(v) \subseteq N(v)$ be the current set of outgoing neighbors of v. Let $N^{un}(v) \subseteq N(v)$ be the current set of unoriented neighbors of v. That is,

$$N(v) = N^{in}(v) \mathbin{\dot{\cup}} N^{out}(v) \mathbin{\dot{\cup}} N^{un}(v)$$

Algorithm 1 Algorithm for known skeleton and max in-degree.

Input: Skeleton $\mathrm{skel}(G^*) = (V, E)$, max in-degree d^*, threshold $\varepsilon > 0$, universal constant C
Output: A complete orientation of $\mathrm{skel}(G^*)$

Run Phase 1: Orient strong v-structures (Algorithm 3) ▷ $\mathcal{O}(n^{d^*})$ time
Run Phase 2: Local search and Meek $R1(d^*)$ (Algorithm 4) ▷ $\mathcal{O}(n^3)$ time
Run Phase 3: Freely orient remaining unoriented edges (Algorithm 5) ▷ $\mathcal{O}(n)$ time via DFS
return $\hat{G}$

There are three phases to our algorithm. In Phase 1, we orient strong v-structures. In Phase 2, we locally check if an edge is forced to orient one way or another to avoid incurring too much error. In Phase 3, we orient the remaining unoriented edges as a 1-polytree. Since the remaining edges were not forced, we may orient the remaining edges in an arbitrary direction (while not incurring "too much error") as long as the final incoming degrees of any vertex does not increase by more than 1. Subroutine `Orient` (Algorithm 2) performs the necessary updates when we orient $u - v$ to $u \to v$.

Algorithm 2 `Orient`: Subroutine to orient edges

Input: Vertices u and v where $u - v$ is currently unoriented

Orient $u - v$ as $u \to v$.
Update $N^{in}(v)$ to $N^{in}(v) \cup \{u\}$ and $N^{un}(v)$ to $N^{un}(v) \setminus \{u\}$.
Update $N^{out}(u)$ to $N^{out}(u) \cup \{v\}$ and $N^{un}(u)$ to $N^{un}(u) \setminus \{v\}$.

3.2 Analysis

Observe that we perform $\mathcal{O}(n^{d^*})$ (conditional) mutual information tests in Algorithm 1. The following lemma (Lemma 5) ensures us that *all* our tests will behave as expected with sufficient samples. As such, in the rest of our analysis, we analyze under the assumption that our tests are correct.

Lemma 5. *Suppose all variables in the Bayesian network has alphabet Σ. For $\varepsilon' > 0$, with*

$$m \in \mathcal{O}\left(\frac{|\Sigma|^{d^*+1}}{\varepsilon'} \cdot \log \frac{|\Sigma|^{d^*+1} \cdot n^{d^*}}{\delta} \cdot \log \frac{|\Sigma|^{d^*+1} \cdot \log(n^{d^*}/\delta)}{\varepsilon'} \right)$$

empirical samples, $\mathcal{O}(n^{d^*})$ statements of the following forms, where $\mathbf{X}$ and $\mathbf{Y}$ are variable sets of size $|\mathbf{X} \mathbin{\dot{\cup}} \mathbf{Y}| \leq d$ and Z is possibly $\emptyset$, all *succeed with probability at least $1 - \delta$:*

(1) If $I(\mathbf{X};\mathbf{Y} \mid Z) = 0$, then $\hat{I}(\mathbf{X};\mathbf{Y} \mid Z) < C \cdot \varepsilon'$,
(2) If $\hat{I}(\mathbf{X};\mathbf{Y} \mid Z) \geq C \cdot \varepsilon'$, then $I(\mathbf{X};\mathbf{Y} \mid Z) > 0$,
(3) If $\hat{I}(\mathbf{X};\mathbf{Y} \mid Z) \leq C \cdot \varepsilon'$, then $I(\mathbf{X};\mathbf{Y} \mid Z) < \varepsilon'$.

Proof. Use Corollary 4 and apply union bound over $\mathcal{O}(n^d)$ tests. □

Recall that $\pi(v)$ is the set of true parents of v in G^*. Let H be the forest induced by the remaining unoriented edges after phase 2. Let $\hat{G}$ be returned graph of the algorithm 1. Let us denote the final $N^{in}(v)$ as $\pi^{in}(v)$ at the end of Phase 2, just before freely orienting, i.e. the vertices pointing into v in $\hat{G} \setminus H$. Let $\pi^{un}(v) = \pi(v) \setminus \pi^{in}(v)$ be the set of ground truth parents that are not identified in Phase 1. Since the algorithm does not make mistakes for orientations in $\hat{G} \setminus H$ (Lemma 6), all edges of in $\pi^{un}(v)$ will be unoriented.

Lemma 6. *Any oriented arc in $\hat{G} \setminus H$ is a ground truth orientation. That is, any vertex parent set in $\hat{G} \setminus H$ is a subset of $\pi(v)$, i.e. $\pi^{in}(v) \subseteq \pi(v)$, and $N^{in}(v)$ at any time during the algorithm will have $N^{in}(v) \subseteq \pi^{in}(v)$.*

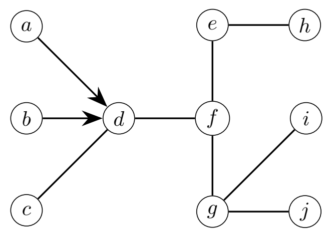


Figure 2: Suppose we have the following partially oriented graph in the execution of Algorithm 4 (after Phase 1). Since $N^{in}(d) = \{a, b\}$, we will check the edge orientations of $c - d$ and $f - d$. Since $I(f; \{a, b\} \mid d) = 0$, we will have $\hat{I}(f; \{a, b\} \mid d) \leq \varepsilon$, so we will *not* erroneously orient $f \to d$. Meanwhile, $I(c; \{a, b\}) = 0$, we will have $\hat{I}(c; \{a, b\}) \leq \varepsilon$, so we will *not* erroneously orient $d \to c$.

Let $\hat{\pi}(v)$ be the proposed parents of v output by Algorithm 1. The KL divergence between the true distribution and our output distribution is essentially $\sum_{v \in V} I(v; \pi(v)) - \sum_{v \in V} I(v; \hat{\pi}(v))$ as the structure independent terms will cancel out.

To get a bound on the KL divergence, we will upper bound $\sum_{v \in V} I(v; \pi(v))$ and lower bound $\sum_{v \in V} I(v; \hat{\pi}(v))$. To upper bound $I(v; \pi(v))$ in terms of $\pi^{in}(v) \subseteq \pi(v)$ and $I(v; u)$ for $u \in \pi^{un}(v)$, we use Lemma 8 which relies on repeated applications of Lemma 7. To lower bound $\sum_{v \in V} I(v; \hat{\pi}(v))$, we use Lemma 9.

Lemma 7. *Fix any vertex v, any $S \subseteq \pi^{un}(v)$, and any $S' \subseteq \pi^{in}(v)$. If $S \neq \emptyset$, then there exists a vertex $u \in S \cup S'$ with*

$$I(v; S \cup S') \leq I(v; S \cup S' \setminus \{u\}) + I(v; u) + \varepsilon \ . \tag{2}$$

Lemma 8. *For any vertex v with $\pi^{in}(v)$, we can show that*

$$I(v; \pi(v)) \leq \varepsilon \cdot |\pi(v)| + I(v; \pi^{in}(v)) + \sum_{u \in \pi^{un}(v)} I(v; u) \ .$$

Algorithm 3 Phase 1: Orient strong v-structures

```
d ← d*
while d ≥ 2 do
    for v ∈ V do                                              ▷ Arbitrary order
        Let 𝒩_d ⊆ 2^{N(v)} be the set of d neighbors of v      ▷ |𝒩_d| = (|N(v)| choose d)
        for S ∈ 𝒩_d s.t. |S| = d, |S ∪ N^in(v)| ≤ d*, and Î(u; S \ {u} | v) ≥ C · ε, ∀u ∈ S do
            for u ∈ S do                                      ▷ Strong deg-d v-structure
                ORIENT(u, v)
    d ← d − 1                                                 ▷ Decrement degree bound
```

Algorithm 4 Phase 2: Local search and Meek $R1(d^*)$

do $\triangleright\ \mathcal{O}(n)$ iterations, $\mathcal{O}(n^2)$ time per iteration
  if $\exists v \in V$ such that $|N^{in}(v)| = d^*$ and $N^{un}(v) \neq \emptyset$ **then** $\triangleright$ Meek $R1(d^*)$
    Orient all unoriented arcs *away* from v
    Update $N^{out}(v) \leftarrow N^{out}(v) \cup N^{un}(v)$; $N^{un}(v) \leftarrow \emptyset$
  for every node $v \in V$ **do**
    if $1 \leq |N^{in}(v)| < d^*$ **then**
      for every $u \in N^{un}(v)$ **do**
        if $\hat{I}(u; N^{in}(v) \mid v) > C \cdot \varepsilon$ **then** ORIENT(u, v)
        else if $\hat{I}(u; N^{in}(v)) > C \cdot \varepsilon$ **then** ORIENT(v, u)
while new edges are being oriented

Algorithm 5 Phase 3: Freely orient remaining unoriented edges

Let H be the forest induced by the remaining unoriented edges.
Freely orient H as a 1-polytree (i.e. maximum in-degree in H is 1).
Let $\hat{G}$ be the combination of the oriented H and the previously oriented arcs.
return $\hat{G}$

In Phase 3, we increase the incoming edges to any vertex by at most one. The following lemma tells us that we lose at most[4] an additive ε error per vertex.

Lemma 9. *Consider an arbitrary vertex v with $\pi^{in}(v)$ at the start of Phase 3. If Phase 3 orients $u \to v$ for some $u - v \in H$, then*

$$I(v; \pi^{in}(v) \cup \{u\}) \geq I(v; \pi^{in}(v)) + I(v; u) - \varepsilon.$$

By using Lemma 8 and Lemma 9, we can show our desired KL divergence bound (Lemma 10).

Lemma 10. *Let $\pi(v)$ be the true parents of v. Let $\hat{\pi}(v)$ be the proposed parents of v output by our algorithm. Then,*

$$\sum_{v \in V} I(v; \pi(v)) - \sum_{v \in V} I(v; \hat{\pi}(v)) \leq n \cdot (d^* + 1) \cdot \varepsilon \ .$$

With these results in hand, we are ready to establish our main theorem:

Proof of Theorem 1. We first combine Lemma 10 and Lemma 5 with $\varepsilon' = \frac{\varepsilon}{2n \cdot (d^*+1)}$ in order to obtain an orientation $\hat{G}$ which is close to G^*. Now, recall that there exist efficient algorithms for estimating the parameters of a Bayes net with in-degree-d (note that this includes d-polytrees) P once a close-enough graph $\hat{G}$ is recovered [Dasgupta, 1997, Bhattacharyya et al., 2020], with sample complexity $\tilde{\mathcal{O}}(|\Sigma|^d n/\varepsilon)$. Denote the final output $\hat{P}_{\hat{G}}$, a distribution that is estimated using the conditional probabilities implied by $\hat{G}$. One can bound the KL divergences as follows:

$$d_{\mathrm{KL}}(P \parallel P_{\hat{G}}) - d_{\mathrm{KL}}(P \parallel P_{G^*}) \leq \varepsilon/2 \quad \text{and} \quad d_{\mathrm{KL}}(P \parallel \hat{P}_{\hat{G}}) - d_{\mathrm{KL}}(P \parallel P_{\hat{G}}) \leq \varepsilon/2 \ .$$

Thus, we see that

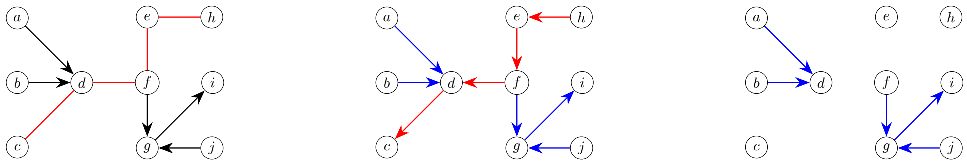
$$d_{\mathrm{KL}}(P \parallel \hat{P}_{\hat{G}}) \leq \varepsilon + d_{\mathrm{KL}}(P \parallel P_{G^*}) = \varepsilon \ .$$

□

4 Skeleton assumption

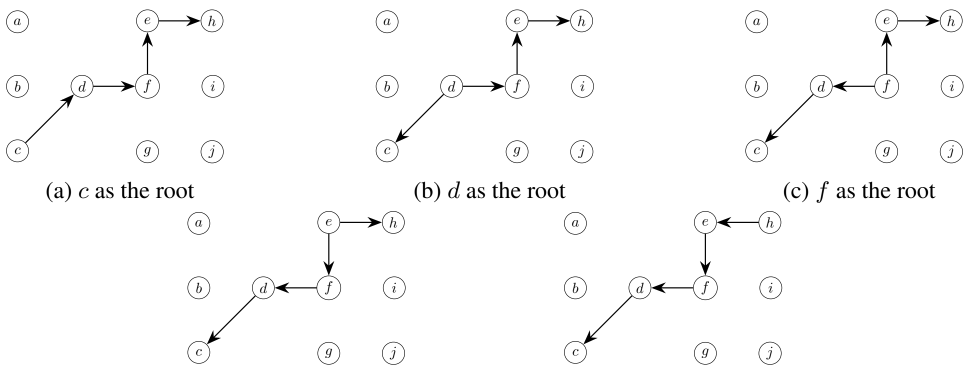
In this section, we present a set of *sufficient* assumptions (Assumption 11) under which the Chow-Liu algorithm will recover the true skeleton even while with finite samples.

[4] Orienting "freely" could also increase the mutual information score and this is considering the worst case.



(a) Before final phase (b) $\hat{G}$ under an arbitrary orientation of H; see Fig. 4 for more. (c) $\hat{G} \setminus H$

Figure 3: Consider the partially oriented graph before the final phase, where H is the edge-induced subgraph on the unoriented edges in red. Since $d^* = 3$ is known, we can conclude that $g \to i$ was oriented due to a local search step and not due to Meek $R1(3)$. We have the following sets before the final phase: $\pi^{in}(c) = \{a, b\}$, $\pi^{in}(g) = \{f, j\}$, $\pi^{i} = \{g\}$, $\pi^{un}(d) = \{c\}$, $\pi^{un}(f) = \{d, e\}$, and $\pi^{un}(e) = \{h\}$. With respect to the chosen orientation of H and the notation in Lemma 10, we have $A = \{c, d, f, e, h\}$, $a_c = d$, $a_d = f$, $a_f = e$, and $a_e = h$. Observe that the π^{un}'s and a's are two different ways to refer to the set of red edges of H.



(a) c as the root (b) d as the root (c) f as the root (d) e as the root (e) h as the root

Figure 4: The five different possible orientations of H. Observe that the ground truth orientation of these edges is inconsistent with all five orientations shown here.

Assumption 11. For any given distribution P, there exists a constant $\varepsilon_P > 0$ such that:
(1) For every pair of nodes u and v, if there exists a path $u - \cdots - v$ of length greater than 2 in G^*, then then $I(u; v) + 3 \cdot \varepsilon_P \leq I(a; b)$ for every pair of adjacent vertices $a - b$ in the path.
(2) For every pair of directly connected nodes $a - b$ in G^*, $I(a; b) \geq 3 \cdot \varepsilon_P$.

Suppose there is a large enough gap of ε_P between edges in G^* and edges outside of G^*. Then, with $\mathcal{O}(1/\varepsilon_P^2)$ samples, each estimated mutual information $\hat{I}(a; b)$ will be sufficiently close to the true mutual information $I(a; b)$. Thus, running the Chow-Liu algorithm (which is essentially maximum spanning tree on the estimated mutual information on each pair of vertices) recovers $\mathrm{skel}(G^*)$.

Lemma 12. *Under Assumption 11, running the Chow-Liu algorithm on the m-sample empirical estimates $\{\hat{I}(u; v)\}_{u,v \in V}$ recovers a ground truth skeleton with high probability when $m \geq \Omega(\frac{\log n}{\varepsilon_P^2})$.*

Combining Lemma 12 with our algorithm Algorithm 1, one can learn a polytree that is ε-close in KL with $\tilde{\mathcal{O}}\left(\max\left\{\frac{\log(n)}{\varepsilon_P^2}, \frac{2^d \cdot n}{\varepsilon}\right\}\right)$ samples, where ε_P depends on the distribution P.

5 Lower bound

In this section, we show that $\Omega(n/\varepsilon)$ samples are necessary *even when a known skeleton is provided*. For constant in-degree d, this shows that our proposed algorithm in Section 3 is sample-optimal up to logarithmic factors.

We first begin by showing a lower bound of $\Omega(1/\varepsilon)$ on a graph with three vertices, even when the skeleton is given. Let G_1 be $X \to Z \to Y$ and G_2 be $X \to Z \leftarrow Y$, such that $\mathrm{skel}(G_1) = \mathrm{skel}(G_2)$ is $X - Z - Y$. Now define P_1 and P_2 as follows:

$$P_1 : \begin{cases} X \sim \mathrm{Bern}(1/2) \\ Z = \begin{cases} X & \text{w.p. } 1/2 \\ \mathrm{Bern}(1/2) & \text{w.p. } 1/2 \end{cases} \\ Y = \begin{cases} Z & \text{w.p. } \sqrt{\varepsilon} \\ \mathrm{Bern}(1/2) & \text{w.p. } 1-\sqrt{\varepsilon} \end{cases} \end{cases} \qquad P_2 : \begin{cases} X \sim \mathrm{Bern}(1/2) \\ Y \sim \mathrm{Bern}(1/2) \\ Z = \begin{cases} X & \text{w.p. } 1/2 \\ Y & \text{w.p. } \sqrt{\varepsilon} \\ \mathrm{Bern}(1/2) & \text{w.p. } 1/2-\sqrt{\varepsilon} \end{cases} \end{cases} \tag{3}$$

The intuition is that we keep the edge $X \to Z$ "roughly the same" and tweak the edge $Y - Z$ between the distributions. We define $P_{i,G}$ as projecting P_i onto G. One can check that the following holds (see Supplemental for the detailed calculations):

Lemma 13. *Let G_1 be $X \to Z \to Y$ and G_2 be $X \to Z \leftarrow Y$, such that* $\mathrm{skel}(G_1) = \mathrm{skel}(G_2)$ *is $X - Z - Y$. With respect to Eq. (3), we have the following:*

1. $d_{\mathrm{H}}^2(P_1, P_2) \in \mathcal{O}(\varepsilon)$
2. $d_{\mathrm{KL}}(P_1 \parallel P_{1,G_1}) = 0$ *and* $d_{\mathrm{KL}}(P_1 \parallel P_{1,G_2}) \in \Omega(\varepsilon)$
3. $d_{\mathrm{KL}}(P_2 \parallel P_{2,G_2}) = 0$ *and* $d_{\mathrm{KL}}(P_2 \parallel P_{2,G_1}) \in \Omega(\varepsilon)$

Our hardness result (Lemma 14) is obtained by reducing the problem of finding an ε-close graph orientation of $X - Z - Y$ to the problem of *testing* whether the samples are drawn from P_1 or P_2: To ensure ε-closeness in the graph orientation, one has to correctly determine whether the samples come from P_1 or P_2 and then pick G_1 or G_2 respectively. However, it is well-known that distinguishing two distributions whose squared Hellinger distance is ε requires $\Omega(1/\varepsilon)$ samples (see, e.g., [Bar-Yossef, 2002, Theorem 4.7]).

Lemma 14. *Even when given* $\mathrm{skel}(G^*)$*, it takes $\Omega(1/\varepsilon)$ samples to learn an ε-close graph orientation of G^* for distributions on $\{0,1\}^3$.*

Using the above construction as a gadget, we can obtain a dependency on n in our lower bound by constructing $n/3$ independent copies of the above gadget, à la proof strategy of Bhattacharyya et al. [2021, Theorem 7.6]. For some constant $c > 0$, we know that a constant $1/c$ fraction of the gadgets will incur an error or more than ε/n if less than cn/ε samples are used. The desired result then follows from the tensorization of KL divergence, i.e., $d_{\mathrm{KL}}\left(\prod_i P_i \parallel \prod_i Q_i\right) = \sum_i d_{\mathrm{KL}}(P_i \parallel Q_i)$.

Theorem 15. *Even when given* $\mathrm{skel}(G^*)$*, it takes $\Omega(n/\varepsilon)$ samples to learn an ε-close graph orientation of G^* for distributions on $\{0,1\}^n$.*

6 Conclusion

In this work, we studied the problem of estimating a distribution defined on a d-polytree P with graph structure G^* using finite observational samples. We designed and analyzed an efficient algorithm that produces an estimate $\hat{P}$ such that $d_{\mathrm{KL}}(P \parallel \hat{P}) \leq \varepsilon$ assuming access to $\mathrm{skel}(G^*)$ and d. The skeleton $\mathrm{skel}(G^*)$ is recoverable under Assumption 11 and we show that there is an inherent hardness in the learning problem even under the assumption that $\mathrm{skel}(G^*)$ is given. For constant d, our hardness result shows that our proposed algorithm is sample-optimal up to logarithmic factors.

An interesting open question is whether one can extend the hardness result to arbitrary $d \geq 1$, or design more efficient learning algorithms for d-polytrees.

References

J. Acharya, A. Bhattacharyya, C. Daskalakis, and S. Kandasamy. Learning and testing causal models with interventions. In *NeurIPS*, pages 9469–9481, 2018.

B. Aragam, A. Amini, and Q. Zhou. Globally optimal score-based learning of directed acyclic graphs in high-dimensions. *Advances in Neural Information Processing Systems*, 32, 2019.

Z. Bar-Yossef. *The Complexity of Massive Data Set Computations*. PhD thesis, UC Berkeley, 2002. Adviser: Christos Papadimitriou. Available at http://webee.technion.ac.il/people/zivby/index_files/Page1489.html.

A. Bhattacharyya, S. Gayen, K. S. Meel, and N. V. Vinodchandran. Efficient distance approximation for structured high-dimensional distributions via learning. In *NeurIPS*, 2020.

A. Bhattacharyya, S. Gayen, E. Price, and N. V. Vinodchandran. Near-optimal learning of tree-structured distributions by Chow-Liu. In *STOC '21—Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 147–160. ACM, New York, 2021. doi: 10.1145/3406325.3451066. URL https://doi.org/10.1145/3406325.3451066.

C. L. Canonne, I. Diakonikolas, D. M. Kane, and A. Stewart. Testing bayesian networks. *IEEE Trans. Inf. Theory*, 66(5):3132–3170, 2020. Preprint available at arXiv:1612.03156.

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

D. M. Chickering and C. Meek. Finding optimal bayesian networks. In *Proceedings of the Eighteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI 02)*, pages 94–102, 2002.

D. M. Chickering, D. Heckerman, and C. Meek. Large-sample learning of bayesian networks is np-hard. *J. Mach. Learn. Res.*, 5:1287–1330, 2004.

C. K. Chow and C. N. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Trans. Inf. Theory*, 14(3):462–467, 1968.

S. Dasgupta. The sample complexity of learning fixed-structure bayesian networks. *Mach. Learn.*, 29(2-3):165–180, 1997.

S. Dasgupta. Learning polytrees. In *UAI*, pages 134–141. Morgan Kaufmann, 1999.

N. Friedman and Z. Yakhini. On the sample complexity of learning bayesian networks. In *Proceedings of the Twelfth Annual Conference on Uncertainty in Articial Intelligence (UAI {96}), San Francisco, CA*, pages 274–282, 1996.

N. Friedman, I. Nachman, and D. Pe'er. Learning bayesian network structure from massive datasets: The" sparse candidate" algorithm. *arXiv preprint arXiv:1301.6696*, 2013.

M. Gao and B. Aragam. Efficient bayesian network structure learning via local markov boundary search. *Advances in Neural Information Processing Systems*, 34:4301–4313, 2021.

D. Geiger, A. Paz, and J. Pearl. Learning causal trees from dependence information. In *AAAI*, pages 770–776. AAAI Press / The MIT Press, 1990.

A. Ghoshal and J. Honorio. Learning identifiable gaussian bayesian networks in polynomial time and sample complexity. *Advances in Neural Information Processing Systems*, 30, 2017.

K.-U. Höffgen. Learning and robust learning of product distributions. In *Proceedings of the sixth annual conference on Computational learning theory*, pages 77–83, 1993.

P. Hoyer, D. Janzing, J. M. Mooij, J. Peters, and B. Schölkopf. Nonlinear causal discovery with additive noise models. *Advances in Neural Information Processing Systems*, 21, 2008.

F. V. Jensen and T. D. Nielsen. *Bayesian networks and decision graphs*, volume 2. Springer, 2007.

D. Koller and N. Friedman. *Probabilistic Graphical Models - Principles and Techniques*. MIT Press, 2009.

C. Meek. Causal Inference and Causal Explanation with Background Knowledge. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, UAI'95, page 403–410, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc. ISBN 1558603859.

G. Park and G. Raskutti. Learning quadratic variance function (qvf) dag models via overdispersion scoring (ods). *J. Mach. Learn. Res.*, 18:224–1, 2017.

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

G. Rebane and J. Pearl. The recovery of causal poly-trees from statistical data. *Int. J. Approx. Reason.*, 2(3):341, 1988.

S. Shimizu, P. O. Hoyer, A. Hyvärinen, A. Kerminen, and M. Jordan. A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7(10), 2006.

P. Spirtes and C. Glymour. An algorithm for fast recovery of sparse causal graphs. *Social science computer review*, 9(1):62–72, 1991.

L. G. Valiant. A theory of the learnable. In *STOC*, pages 436–445. ACM, 1984.

A Deferred proofs

A.1 Adapting the known tester result of Bhattacharyya et al. [2021]

Corollary 4 is adapted from Theorem 1.3 of Bhattacharyya et al. [2021].

Theorem 16 (Conditional Mutual Information Tester, [Bhattacharyya et al., 2021, Theorem 1.3])**.** *Fix any $\varepsilon > 0$. Let (X, Y, Z) be three random variables over $\Sigma_X, \Sigma_Y, \Sigma_Z$ respectively. Given the empirical distribution $(\hat{X}, \hat{Y}, \hat{Z})$ over a size N sample of (X, Y, Z), there exists a universal constant $0 < C < 1$ so that for any*

$$N \geq \Theta\left(\frac{|\Sigma_X| \cdot |\Sigma_Y| \cdot |\Sigma_Z|}{\varepsilon} \cdot \log \frac{|\Sigma_X| \cdot |\Sigma_Y| \cdot |\Sigma_Z|}{\delta} \cdot \log \frac{|\Sigma_X| \cdot |\Sigma_Y| \cdot |\Sigma_Z| \cdot \log(1/\delta)}{\varepsilon}\right),$$

the following results hold with probability $1 - \delta$:

1. *If $I(X; Y \mid Z) = 0$, then $I(\hat{X}; \hat{Y} \mid \hat{Z}) < \varepsilon$.*
2. *If $I(X; Y \mid Z) \geq \varepsilon$, then $I(\hat{X}; \hat{Y} \mid \hat{Z}) > C \cdot I(X; Y \mid Z)$.*

Unconditional mutual information statements involving $I(X; Y)$ and $I(\hat{X}; \hat{Y})$ hold similarly by treating $|\Sigma_Z| = 1$.

In our notation, we use $\hat{I}(X; Y \mid Z)$ to mean the mutual information of the empirical distribution $I(\hat{X}; \hat{Y} \mid \hat{Z})$.

Corollary 4 (Conditional Mutual Information Tester, adapted from [Bhattacharyya et al., 2021, Theorem 1.3])**.** *Fix any $\varepsilon > 0$. Let (X, Y, Z) be three random variables over $\Sigma_X, \Sigma_Y, \Sigma_Z$ respectively. Given the empirical distribution $(\hat{X}, \hat{Y}, \hat{Z})$ over a size N sample of (X, Y, Z), there exists a universal constant $0 < C < 1$ so that for any*

$$N \geq \Theta\left(\frac{|\Sigma_X| \cdot |\Sigma_Y| \cdot |\Sigma_Z|}{\varepsilon} \cdot \log \frac{|\Sigma_X| \cdot |\Sigma_Y| \cdot |\Sigma_Z|}{\delta} \cdot \log \frac{|\Sigma_X| \cdot |\Sigma_Y| \cdot |\Sigma_Z| \cdot \log(1/\delta)}{\varepsilon}\right),$$

the following statements hold with probability $1 - \delta$:
(1) If $I(X; Y \mid Z) = 0$, then $\hat{I}(X; Y \mid Z) < C \cdot \varepsilon$.
(2) If $\hat{I}(X; Y \mid Z) \geq C \cdot \varepsilon$, then $I(X; Y \mid Z) > 0$.
(3) If $\hat{I}(X; Y \mid Z) \leq C \cdot \varepsilon$, then $I(X; Y \mid Z) < \varepsilon$.
Unconditional statements involving $I(X; Y)$ and $\hat{I}(X; Y)$ hold similarly by choosing $|\Sigma_Z| = 1$.

Proof. Notice in the original proof (1) of the CMI tester [Bhattacharyya et al., 2021, Theorem 1.3], it is possible to change ε to $C \cdot \varepsilon$ by paying a factor $1/C$ more in sample complexity. We obtain the first statement by doing so.

Suppose $\hat{I}(X; Y \mid Z) \geq C \cdot \varepsilon$. By the contrapositive of the first statement, we have $I(X; Y \mid Z) \neq 0$. Since conditional mutual information is non-negative, we must have $I(X; Y \mid Z) > 0$.

Suppose $\hat{I}(X; Y \mid Z) \leq C \cdot \varepsilon$. Suppose, for a contradiction, that $I(X; Y \mid Z) \geq \varepsilon$. Then, statement 2 of Theorem 16 tells us that $\hat{I}(X; Y \mid Z) > C \cdot I(X; Y \mid Z) \geq C \cdot \varepsilon$. This is a contradiction to the assumption that $\hat{I}(X; Y \mid Z) \leq C \cdot \varepsilon$. Therefore, we must have $I(X; Y \mid Z) < \varepsilon$. □

A.2 Algorithm analysis

The following identity (Lemma 17) of mutual information and two properties about (conditional) mutual information on a polytree (Lemma 18) which will be helpful in our proofs later.

Lemma 17 (A useful identity)**.** *For any variable v and sets $A, B \subseteq V \setminus \{v\}$, we have*

$$I(v; A \cup B) = I(v; A) + I(v; B) + I(A; B \mid v) - I(A; B).$$

Proof. By the chain rule for mutual information, we can express $I(v, A; B)$ in the following two ways:

1. $I(v, A; B) = I(v; B) + I(A; B \mid v)$;
2. $I(v, A; B) = I(A; B) + I(v; B \mid A)$.

So,

$$\begin{aligned} I(v; A \cup B) &= I(v; A) + I(v; B \mid A) \\ &= I(v; A) + I(v, A; B) - I(A; B) \\ &= I(v; A) + I(v; B) + I(A; B \mid v) - I(A; B). \end{aligned}$$

□

Lemma 18. *Let v be an arbitrary vertex in a Bayesian polytree with parents $\pi(v)$. Then, we have*

1. *For any disjoint subsets $A, B \subseteq \pi(v)$,*

$$I(v; A \cup B) = I(v; A) + I(v; B) + I(A; B \mid v)$$

2. *For any subset $A \subseteq \pi(v)$,*

$$I(v; A) \geq \sum_{u \in A} I(v; u)$$

Proof. For the first equality, apply Lemma 17 with the observation that $I(A; B) = 0$ since $A, B \subseteq \pi(v)$.

For the second inequality, apply the first equality $|A|$ times with the observation that conditional mutual information is non-negative. Suppose $A = \{a_1, \ldots, a_k\}$. Then,

$$\begin{aligned} I(v; A) &= I(v; \{a_1\}) + I(v; A \setminus \{a_1\}) + I(\{a_1\}; A \setminus \{a_1\} \mid v) \\ &\geq I(v; \{a_1\}) + I(v; A \setminus \{a_1\}) \\ &\ldots \\ &\geq \sum_{u \in A} I(v; u) \end{aligned}$$

□

Lemma 6. *Any oriented arc in $\hat{G} \setminus H$ is a ground truth orientation. That is, any vertex parent set in $\hat{G} \setminus H$ is a subset of $\pi(v)$, i.e. $\pi^{in}(v) \subseteq \pi(v)$, and $N^{in}(v)$ at any time during the algorithm will have $N^{in}(v) \subseteq \pi^{in}(v)$.*

Proof. We consider the three cases in which we orient edges within the while loop:

1. Strong v-structures (in Phase 1)
2. Forced orientation due to local checks (in Phase 2)
3. Forced orientation due to Meek $R1(d^*)$ (in Phase 2)

Case 1: Strong v-structures Consider an arbitrary strong deg-d v-structure with center v. That is, there is a set S (all neighbors of v) with size $|S| = d$, such that $\hat{I}(u; S \setminus \{u\} \mid v) \geq C \cdot \varepsilon$ for any $u \in S$. So, by Corollary 4, we know that $I(u; S \setminus \{u\} \mid v) > 0$ for all $u \in S$.

Consider an arbitrary vertex $u_0 \in S$. Suppose, for a contradiction, that the ground truth orients *some* edge outwards from v, say $v \to u_0$ for some $u_0 \in S$. This would imply that $I(u_0; S \setminus \{u_0\} \mid v) = 0$. This contradicts the fact that we had $I(u_0; S \setminus \{u_0\} \mid v) > 0$ for any $u \in S$. Therefore, for all $u \in S$, orienting $u \to v$ is a ground truth orientation.

Case 2: Forced orientation due to local checks Consider an arbitrary vertex v. Suppose it currently has incoming oriented arcs $N^{in}(v)$ and we are checking for the orientation for an unoriented neighbor u of v. By induction, the existing incoming arcs to v are ground truth orientations.

If the ground truth orients $u \to v$, then $I(u; N^{in}(v)) = 0$ and we should have $\hat{I}(u; N^{in}(v)) < C \cdot \varepsilon \leq \varepsilon$ via Corollary 4. Hence, if we detect $\hat{I}(N^{in}(v); u) > \varepsilon$, it must be the case that the ground truth orientation is $u \leftarrow v$, which is what we also orient.

Meanwhile, if the ground truth orients $u \leftarrow v$, then $I(u; N^{in}(v) \mid v) = 0$ and we should have $\hat{I}(u; N^{in}(v) \mid v) \leq C \cdot \varepsilon \leq \varepsilon$ via Corollary 4. Hence, if we detect $\hat{I}(u; N^{in}(v) \mid v) > \varepsilon$, it must be the case that the ground truth orientation is $u \to v$, which is what we also orient.

See Fig. 2 for an illustration. Note that we may possibly detect both $\hat{I}(u; N^{in}(v)) \leq \varepsilon$ and $\hat{I}(u; N^{in}(v) \mid v) \leq \varepsilon$. In that case, we leave the edge $u \sim v$ unoriented.

Case 3: Forced orientation due to Meek $R1(d^*)$ Meek $R1(d^*)$ only triggers when there are d^* incoming arcs to a particular vertex. Since oriented arcs are inductively ground truth orientations and there are at most d^* incoming arcs to any vertex, the forced orientations due to Meek $R1(d^*)$ will always be correct. □

Lemma 7. *Fix any vertex v, any $S \subseteq \pi^{un}(v)$, and any $S' \subseteq \pi^{in}(v)$. If $S \neq \emptyset$, then there exists a vertex $u \in S \cup S'$ with*

$$I(v; S \cup S') \leq I(v; S \cup S' \setminus \{u\}) + I(v; u) + \varepsilon \ . \tag{2}$$

Proof. Since $S \cup S' \subseteq \pi(v)$, we see that $I(u; S \cup S' \setminus \{u\}) = 0$. Furthermore, since $S \neq \emptyset$, Phase 1 guarantees that there exists a vertex $u \in S$ such that $\hat{I}(u; S \cup S' \setminus \{u\} \mid v) \leq \varepsilon$. Since $0 < C < 1$, this implies that $\hat{I}(u; S \cup S' \setminus \{u\} \mid v) \leq C \cdot \varepsilon$ and so Corollary 4 tells us that $I(u; S \cup S' \setminus \{u\}) < \varepsilon$. Thus, we get

$$\begin{aligned}
&I(v; S \cup S') \\
&= I(v; S \cup S' \setminus \{u\}) + I(v; u) + I(u; S \cup S' \setminus \{u\} \mid v) - I(u; S \cup S' \setminus \{u\}) \\
&= I(v; S \cup S' \setminus \{u\}) + I(v; u) + I(u; S \cup S' \setminus \{u\} \mid v) \\
&\leq I(v; S \cup S' \setminus \{u\}) + I(v; u) + \varepsilon
\end{aligned}$$

□

Lemma 8. *For any vertex v with $\pi^{in}(v)$, we can show that*

$$I(v; \pi(v)) \leq \varepsilon \cdot |\pi(v)| + I(v; \pi^{in}(v)) + \sum_{u \in \pi^{un}(v)} I(v; u) \ .$$

Proof. Initializing $S' = \pi^{in}(v)$ and $S = \pi(v) \setminus \pi^{in}(v) = \pi^{un}(v)$, we can repeatedly apply Lemma 7 to remove vertices one by one, until $S = \emptyset$. Without loss of generality, by relabelling the vertices, we may assume that Lemma 7 removes u_1, then u_2, and so on. Let us denote the set of all removed vertices by U and note that some of the removed vertices may come from $S' = \pi^{in}(v)$.

$$\begin{aligned}
I(v; \pi(v)) &\leq I(v; \pi(v) \setminus \{u_1\}) + I(v; u_1) + \varepsilon && \text{By Lemma 7} \\
&\leq I(v; \pi(v) \setminus \{u_1, u_2\}) + I(v; u_1) + I(v; u_2) + 2\varepsilon && \text{By Lemma 7} \\
&\leq \dots \\
&\leq I(v; \pi(v) \setminus U) + \sum_{u \in U} I(v; u) + \varepsilon \cdot |U| && \text{By Lemma 7}
\end{aligned}$$

Since $I(A; B) = 0$ for any $A \mathbin{\dot{\cup}} B \subseteq \pi^{in}(v)$, we have

$$
\begin{aligned}
& I(v; \pi^{in}(v)) \\
&= I(v; \pi^{in}(v) \setminus U) + I(v; \pi^{in}(v) \cap U) + I(\pi^{in}(v) \cap U; \pi^{in}(v) \setminus U \mid v) && \text{By Lemma 18} \\
&\geq I(v; \pi^{in}(v) \setminus U) + I(v; \pi^{in}(v) \cap U) \\
&\geq I(v; \pi^{in}(v) \setminus U) + \sum_{u \in \pi^{in}(v) \cap U} I(v; u) && \text{By Lemma 18}
\end{aligned}
$$

where the second last inequality is because $I(\pi^{in}(v); \pi^{in}(v) \cap U \mid v) \geq 0$.

$$
\begin{aligned}
I(v; \pi(v)) &\leq I(v; \pi^{in}(v) \setminus U) + \sum_{u \in U} I(v; u) + \varepsilon \cdot |U| && \text{From above} \\
&= I(v; \pi^{in}(v) \setminus U) + \sum_{u \in \pi^{in}(v) \cap U} I(v; u) + \sum_{u \in \pi^{un}(v)} I(v; u) + \varepsilon \cdot |U| && \text{Since } \pi^{un} \subseteq U \\
&\leq I(v; \pi^{in}(v)) + \sum_{u \in \pi^{un}(v)} I(v; u) + \varepsilon \cdot |U| && \text{From above} \\
&\leq I(v; \pi^{in}(v)) + \sum_{u \in \pi^{un}(v)} I(v; u) + \varepsilon \cdot |\pi(v)| && \text{Since } U \subseteq \pi(v)
\end{aligned}
$$

□

Lemma 9. *Consider an arbitrary vertex v with $\pi^{in}(v)$ at the start of Phase 3. If Phase 3 orients $u \to v$ for some $u - v \in H$, then*

$$I(v; \pi^{in}(v) \cup \{u\}) \geq I(v; \pi^{in}(v)) + I(v; u) - \varepsilon.$$

Proof. Since $u \sim v \in E(H)$ remained unoriented, Phase 2 guarantees that $\hat{I}(u; \pi^{in}(v) \mid v) \leq \varepsilon$ and $\hat{I}(u; \pi^{in}(v)) \leq \varepsilon$. Since $0 < C < 1$, we also see that $\hat{I}(u; \pi^{in}(v) \mid v) \leq C \cdot \varepsilon$ and $\hat{I}(u; \pi^{in}(v)) \leq C \cdot \varepsilon$ and so Corollary 4 tells us that $I(u; \pi^{in}(v) \mid v) \leq \varepsilon$ and $I(u; \pi^{in}(v)) \leq \varepsilon$. So,

$$
\begin{aligned}
& |I(v; \pi^{in}(v) \cup u) - I(v; \pi^{in}(v)) - I(v; u)| \\
&= |I(u; \pi^{in}(v) \mid v) - I(u; \pi^{in}(v))| && \text{By Lemma 17} \\
&= \max\{I(u; \pi^{in}(v) \mid v), I(u; \pi^{in}(v))\} && \text{At most one of these term can be non-zero} \\
&\leq \varepsilon
\end{aligned}
$$

□

Lemma 10. *Let $\pi(v)$ be the true parents of v. Let $\hat{\pi}(v)$ be the proposed parents of v output by our algorithm. Then,*

$$\sum_{v \in V} I(v; \pi(v)) - \sum_{v \in V} I(v; \hat{\pi}(v)) \leq n \cdot (d^* + 1) \cdot \varepsilon\ .$$

Proof. We will argue that this summation is bounded by individually bounding each term in the summation. The main argument of the proof is that once we identified all the strong v-structures (and thus cancel out the scores of every strong v-structures), the rest should be roughly the score of a tree (up to additive ε error). Then, since we are guaranteed to be given $\mathrm{skel}(G^*)$, the tree scores will match.

Let $A \subseteq V$ be the set of vertices which receive an additional incoming neighbor in the final phase, which we denote by $a_v \in V$, i.e. $\hat{\pi}(v) = \pi^{in}(v) \cup \{a_v\}$. Note that the set of edges $\{a_v \to v\}_{v \in A}$ is exactly the edges of the undirected graph H in the final phase. See Fig. 3 for an illustration.

To lower bound $\sum_{v \in V} I(v; \hat{\pi}(v))$, we can show

$$
\begin{aligned}
&\sum_{v \in V} I(v; \hat{\pi}(v)) \\
&= \sum_{v \in A} I(v; \hat{\pi}(v)) + \sum_{v \in V \setminus A} I(v; \hat{\pi}(v)) \\
&\geq \sum_{v \in A} \Big(I(v; \hat{\pi}(v) \setminus \{a_v\}) + I(v; a_v) - \varepsilon \Big) + \sum_{v \in V \setminus A} I(v; \hat{\pi}(v)) && \text{By Lemma 9} \\
&= \sum_{v \in A} I(v; \pi^{in}(v)) + \sum_{v \in A} I(v; a_v) + \sum_{v \in V \setminus A} I(v; \pi^{in}(v)) - |A| \cdot \varepsilon \\
&= \sum_{v \in V} I(v; \pi^{in}(v)) + \sum_{v \in A} I(v; a_v) - |A| \cdot \varepsilon \\
&\geq \sum_{v \in V} I(v; \pi^{in}(v)) + \sum_{v \in A} I(v; a_v) - n\varepsilon && \text{Since } A \subseteq V \text{ and } |V| = n
\end{aligned}
$$

Meanwhile, to upper bound $\sum_{v \in V} I(v; \pi(v))$, we can show

$$
\begin{aligned}
&\sum_{v \in V} I(v; \pi(v)) \\
&= \sum_{\substack{v \in V \\ \pi^{un}(v) \neq \emptyset}} I(v; \pi(v)) + \sum_{\substack{v \in V \\ \pi^{un}(v) = \emptyset}} I(v; \pi(v)) \\
&\leq \sum_{\substack{v \in V \\ \pi^{un}(v) \neq \emptyset}} \left(\varepsilon \cdot |\pi(v)| + I(v; \pi^{in}(v)) + \sum_{u \in \pi^{un}(v)} I(v; u) \right) + \sum_{\substack{v \in V \\ \pi^{un}(v) = \emptyset}} I(v; \pi(v)) && \text{By Lemma 8} \\
&= \sum_{v \in V} I(v; \pi^{in}(v)) + \sum_{\substack{v \in V \\ \pi^{un}(v) \neq \emptyset}} \left(\varepsilon \cdot |\pi(v)| + \sum_{u \in \pi^{un}(v)} I(v; u) \right)
\end{aligned}
$$

where the final equality is because $\pi^{in}(v) = \pi(v)$ when $\pi^{un}(v) = \emptyset$. Since $|\pi(v)| \leq d^*$ and $|V| = n$, we get

$$
\sum_{v \in V} I(v; \pi(v)) \leq nd^* \varepsilon + \sum_{v \in V} I(v; \pi^{in}(v)) + \sum_{\substack{v \in V \\ \pi^{in}(v) \neq \emptyset}} \sum_{u \in \pi^{un}(v)} I(v; u)
$$

Putting together, we get

$$
\begin{aligned}
&\sum_{v \in V} I(v; \pi(v)) - \sum_{v \in V} I(v; \hat{\pi}(v)) \\
&\leq \left(nd^* \varepsilon + \sum_{v \in V} I(v; \pi^{in}(v)) + \sum_{\substack{v \in V \\ \pi^{in}(v) \neq \emptyset}} \sum_{u \in \pi^{un}(v)} I(v; u) \right) && \text{From above} \\
&\quad - \left(\sum_{v \in V} I(v; \pi^{in}(v)) + \sum_{v \in A} I(v; a_v) - n\varepsilon \right) \\
&= n \cdot (d^* + 1) \cdot \varepsilon + \sum_{v \in V} \sum_{u \in \pi^{un}(v)} I(v; u) - \sum_{v \in A} I(v; a_v) \\
&= n \cdot (d^* + 1) \cdot \varepsilon
\end{aligned}
$$

where the last equality is because the last two terms are two different ways to enumerate the edges of H, e.g. see Fig. 3.

$\square$

A.3 Skeleton assumption

Lemma 12. *Under Assumption 11, running the Chow-Liu algorithm on the m-sample empirical estimates $\{\hat{I}(u;v)\}_{u,v\in V}$ recovers a ground truth skeleton with high probability when $m \geq \Omega(\frac{\log n}{\varepsilon_P^2})$.*

Proof. Fix a graph G^*. Recall that the Chow-Liu algorithm can be thought of as running maximum spanning tree with the edge weights as the estimated mutual information between any pair of vertices. With $m \geq \Omega(\log(n)/\varepsilon_P^2)$ samples and Assumption 11, one can estimate $\hat{I}(u;v)$ up to ε_P-closeness with high probability in n, i.e. $|I(u;v) - \hat{I}(u;v)| \leq \varepsilon_P$ for any pair of vertices $u, v \in V$.

Now, consider two arbitrary vertices u and v that are *not* neighbors in G^*.

Case 1 (u and v belong in the same connected component in G^*): Let $P_{u,v} = z_0 - z_1 - \ldots - z_k - z_{k+1}$ be the unique path between $u = z_0$ and $v = z_{k+1}$, where $k \geq 1$. Then,

$$\hat{I}(u;v) - \varepsilon_P \leq I(u;v) \leq I(z_i, z_{i+1}) - 3 \cdot \varepsilon_P \leq \hat{I}(z_i, z_{i+1}) - 2 \cdot \varepsilon_P$$

for any $i \in \{1, \ldots, k\}$. That is, the Chow-Liu algorithm will *not* add the edge $u \sim v$ in the output tree.

Case 2 (u and v belong in the different connected components in G^*): Since u and v belong in the different connected components in G^*, we have $I(u;v) = 0$. With m samples, for any two edge $a \sim b$ in G^*, we have

$$\hat{I}(u;v) \leq I(u;v) + \varepsilon_P = \varepsilon_P < 2 \cdot \varepsilon_P \leq I(a;b) - \varepsilon_P \leq \hat{I}(a;b)$$

That is, the Chow-Liu algorithm will always add edges crossing different components *after* all true edges have been considered. $\square$

A.4 Lower bounds

We will use the following inequality in our proofs.

Fact 19. For $x > 0$, we have $\log_2(1+x) \geq \log_2(e) \cdot \left(x - \frac{x^2}{2}\right) = \log_2(e) \cdot x \cdot \left(1 - \frac{x}{2}\right)$.

Recall the lower bound distributions from Section 5, but we replace $\sqrt{\varepsilon}$ with α for notational convenience:

$$P_1 : \begin{cases} X \sim \mathrm{Bern}(1/2) \\ Z = \begin{cases} X & \text{w.p. } 1/2 \\ \mathrm{Bern}(1/2) & \text{w.p. } 1/2 \end{cases} \\ Y = \begin{cases} Z & \text{w.p. } \alpha \\ \mathrm{Bern}(1/2) & \text{w.p. } 1-\alpha \end{cases} \end{cases} \qquad P_2 : \begin{cases} X \sim \mathrm{Bern}(1/2) \\ Y \sim \mathrm{Bern}(1/2) \\ Z = \begin{cases} X & \text{w.p. } 1/2 \\ Y & \text{w.p. } \alpha \\ \mathrm{Bern}(1/2) & \text{w.p. } 1/2-\alpha \end{cases} \end{cases}$$

By construction, we have

$$P_1(x,y,z) = \frac{1}{2} \cdot \left(\frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \mathbb{1}_{x=z}\right) \cdot \left(\alpha \cdot \mathbb{1}_{y=z} + (1-\alpha) \cdot \frac{1}{2}\right)$$

and

$$P_2(x,y,z) = \frac{1}{2} \cdot \frac{1}{2} \cdot \left(\frac{1}{2} \cdot \mathbb{1}_{x=z} + \alpha \cdot \mathbb{1}_{y=z} + \left(\frac{1}{2} - \alpha\right) \cdot \frac{1}{2}\right)$$

which corresponds to the probability tables given in Table 1.

Lemma 20. $d_H^2(P_1, P_2) \leq \alpha^2$

x	y	z	$P_1(x,y,z)$	$P_2(x,y,z)$
0	0	0	$\frac{3}{16}\cdot(1+\alpha)$	$\frac{1}{16}\cdot(3+2\alpha)$
0	0	1	$\frac{1}{16}\cdot(1-\alpha)$	$\frac{1}{16}\cdot(1-2\alpha)$
0	1	0	$\frac{3}{16}\cdot(1-\alpha)$	$\frac{1}{16}\cdot(3-2\alpha)$
0	1	1	$\frac{1}{16}\cdot(1+\alpha)$	$\frac{1}{16}\cdot(1+2\alpha)$
1	0	0	$\frac{1}{16}\cdot(1+\alpha)$	$\frac{1}{16}\cdot(1+2\alpha)$
1	0	1	$\frac{3}{16}\cdot(1-\alpha)$	$\frac{1}{16}\cdot(3-2\alpha)$
1	1	0	$\frac{1}{16}\cdot(1-\alpha)$	$\frac{1}{16}\cdot(1-2\alpha)$
1	1	1	$\frac{3}{16}\cdot(1+\alpha)$	$\frac{1}{16}\cdot(3+2\alpha)$

Table 1: Explicit probability tables for our lower bound construction

Proof. From Table 1, we see that

$$\begin{aligned}
&\sum_{(x,y,z)\in\{0,1\}^3} \sqrt{P_1(x,y,z)\cdot P_2(x,y,z)} \\
&= \frac{1}{16}\cdot\Big(\sqrt{3\cdot(1+\alpha)\cdot(3+2\alpha)} + \sqrt{(1-\alpha)\cdot(1-2\alpha)} \\
&\quad + \sqrt{3\cdot(1-\alpha)\cdot(3-2\alpha)} + \sqrt{(1+\alpha)\cdot(1+2\alpha)} \\
&\quad + \sqrt{(1+\alpha)\cdot(1+2\alpha)} + \sqrt{3\cdot(1-\alpha)\cdot(3-2\alpha)} \\
&\quad + \sqrt{(1-\alpha)\cdot(1-2\alpha)} + \sqrt{3\cdot(1+\alpha)\cdot(3+2\alpha)}\Big) \\
&= \frac{1}{8}\cdot\Big(\sqrt{3\cdot(1+\alpha)\cdot(3+2\alpha)} + \sqrt{(1-\alpha)\cdot(1-2\alpha)} \\
&\quad + \sqrt{3\cdot(1-\alpha)\cdot(3-2\alpha)} + \sqrt{(1+\alpha)\cdot(1+2\alpha)}\Big)
\end{aligned}$$

Consider the Taylor expansion of each of the four terms at $\alpha = 0$:

$$\begin{aligned}
\sqrt{3\cdot(1+\alpha)\cdot(3+2\alpha)} &= 3 + \frac{5\alpha}{2} - \frac{\alpha^2}{24} + \frac{5\alpha^3}{144} - \ldots \\
\sqrt{(1-\alpha)\cdot(1-2\alpha)} &= 1 - \frac{3\alpha}{2} - \frac{\alpha^2}{8} - \frac{3\alpha^3}{16} - \ldots \\
\sqrt{3\cdot(1-\alpha)\cdot(3-2\alpha)} &= 3 - \frac{5\alpha}{2} - \frac{\alpha^2}{24} - \frac{5\alpha^3}{144} - \ldots \\
\sqrt{(1+\alpha)\cdot(1+2\alpha)} &= 1 + \frac{3\alpha}{2} - \frac{\alpha^2}{8} + \frac{3\alpha^3}{16} - \ldots
\end{aligned}$$

So,

$$\begin{aligned}
&\sum_{(x,y,z)\in\{0,1\}^3} \sqrt{P_1(x,y,z)\cdot P_2(x,y,z)} \\
&= \frac{1}{8}\cdot\Big(\sqrt{3\cdot(1+\alpha)\cdot(3+2\alpha)} + \sqrt{(1-\alpha)\cdot(1-2\alpha)} \\
&\quad + \sqrt{3\cdot(1-\alpha)\cdot(3-2\alpha)} + \sqrt{(1+\alpha)\cdot(1+2\alpha)}\Big) \\
&\geq \frac{1}{8}\cdot\left(8 - \frac{\alpha^2}{3} - \mathcal{O}(\alpha^4)\right) \\
&\geq 1 - \frac{\alpha^2}{24} - \mathcal{O}(\alpha^4)
\end{aligned}$$

Hence,

$$d_H^2(P_1, P_2) = 1 - \sum_{(x,y,z)\in\{0,1\}^3} \sqrt{P_1(x,y,z)\cdot P_2(x,y,z)} \leq \frac{\alpha^2}{24} + \mathcal{O}(\alpha^4) \in \mathcal{O}(\alpha^2)$$

$\square$

Lemma 21. *$d_{\mathrm{KL}}(P_1 \parallel P_{1,G_1}) = 0$ and $d_{\mathrm{KL}}(P_1 \parallel P_{1,G_2}) \in \Omega(\alpha^2)$*

Proof. We have $d_{\mathrm{KL}}(P_1 \parallel P_{1,G_1}) = 0$ by definition of P_1: Z depends on X and Y depends on Z. Observe that

$$\begin{aligned}
&d_{\mathrm{KL}}(P_1 \parallel P_{1,G_2}) \\
&= I(X;Z) + I(X;Y) - I(Z;X,Y) \\
&= I(X;Z) + I(X;Y) - (I(Z;X) + I(Z;Y) + I(X;Y \mid Z) - I(X;Y)) \quad (\dagger) \\
&= I(X;Y) - I(X;Y \mid Z) \\
&= I(X;Y) && \text{Since } I(X;Y \mid Z) = 0 \text{ in } P_1
\end{aligned}$$

where $(\dagger)$ is by applying Lemma 17 with $v = Z$, $A = \{X\}$, $B = \{Y\}$.

We will now show that $I(X;Y) \in \Omega(\alpha^2)$.

x	y	$P_1(x,y)$
0	0	$\frac{1}{8}\cdot(2+\alpha)$
0	1	$\frac{1}{8}\cdot(2-\alpha)$
1	0	$\frac{1}{8}\cdot(2-\alpha)$
1	1	$\frac{1}{8}\cdot(2+\alpha)$

Table 2: Probability table for $P_1(x,y)$

From Table 1, one can verify that $P_1(x=0) = P_1(x=1) = P_1(y=0) = P_1(y=1) = 1/2$. So,

$$\begin{aligned}
I(X;Y) &= \sum_{(x,y)\in\{0,1\}^2} P_1(x,y)\cdot\log\frac{P_1(x,y)}{P_1(x)\cdot P_1(y)} \\
&= \frac{1}{8}\cdot\left((2+\alpha)\cdot\log\left(\frac{(2+\alpha)/8}{1/4}\right) + (2-\alpha)\cdot\log\left(\frac{(2-\alpha)/8}{1/4}\right)\right. && \text{From Table 2} \\
&\quad \left. + (2+\alpha)\cdot\log\left(\frac{(2+\alpha)/8}{1/4}\right) + (2-\alpha)\cdot\log\left(\frac{(2-\alpha)/8}{1/4}\right)\right) \\
&= \frac{1}{4}\cdot\left((2+\alpha)\cdot\log\left(1+\frac{\alpha}{2}\right) + (2-\alpha)\cdot\log\left(1-\frac{\alpha}{2}\right)\right) \\
&\geq \frac{1}{4}\cdot\log_2(e)\cdot\frac{\alpha}{2}\cdot\left((2+\alpha)\cdot\left(1-\frac{\alpha}{4}\right) - (2-\alpha)\cdot\left(1+\frac{\alpha}{4}\right)\right) && \text{By Fact 19} \\
&\geq \frac{1}{8}\cdot\log_2(e)\cdot\alpha\cdot\left(\left(2+\alpha-\frac{\alpha}{2}-\frac{\alpha^2}{4}\right) - \left(2-\alpha+\frac{\alpha}{2}-\frac{\alpha^2}{4}\right)\right) \\
&= \frac{1}{8}\cdot\log_2(e)\cdot\alpha^2 \\
&\in \Omega(\alpha^2)
\end{aligned}$$

$\square$

Lemma 22. *$d_{\mathrm{KL}}(P_2 \parallel P_{2,G_2}) = 0$ and $d_{\mathrm{KL}}(P_2 \parallel P_{2,G_1}) \in \Omega(\alpha^2)$*

Proof. We have $d_{\mathrm{KL}}(P_2 \parallel P_{2,G_2}) = 0$ by definition of P_2: Z depends on both X and Y. Observe that

$$\begin{aligned}
&d_{\mathrm{KL}}(P_2 \parallel P_{2,G_1}) \\
&= I(Z; X, Y) - I(X; Z) - I(Z; Y) \\
&= \left(I(Z; X) + I(Z; Y) + I(X; Y \mid Z) - I(X; Y)\right) - I(X; Z) - I(Z; Y) \quad (\dagger) \\
&= I(X; Y \mid Z) - I(X; Y) \\
&= I(X; Y \mid Z) \qquad \text{Since } I(X;Y) = 0 \text{ in } P_2
\end{aligned}$$

where ($\dagger$) is by applying Lemma 17 with $v = Z$, $A = \{X\}$, $B = \{Y\}$.

We will now show that $I(X; Y \mid Z) \in \Omega(\alpha^2)$.

x	y	$P_2(x, y \mid z = 0)$	$P_2(x, y \mid z = 1)$
0	0	$\frac{1}{8} \cdot (3 + 2\alpha)$	$\frac{1}{8} \cdot (1 - 2\alpha)$
0	1	$\frac{1}{8} \cdot (3 - 2\alpha)$	$\frac{1}{8} \cdot (1 + 2\alpha)$
1	0	$\frac{1}{8} \cdot (1 + 2\alpha)$	$\frac{1}{8} \cdot (3 - 2\alpha)$
1	1	$\frac{1}{8} \cdot (1 - 2\alpha)$	$\frac{1}{8} \cdot (3 + 2\alpha)$

x	$P_2(x \mid z = 0)$	$P_2(x \mid z = 1)$
0	$\frac{3}{4}$	$\frac{1}{4}$
1	$\frac{1}{4}$	$\frac{3}{4}$

y	$P_2(y \mid z = 0)$	$P_2(y \mid z = 1)$
0	$\frac{1+\alpha}{2}$	$\frac{1-\alpha}{2}$
1	$\frac{1-\alpha}{2}$	$\frac{1+\alpha}{2}$

Table 3: Conditional probability tables for P_2.

By definition,

$$\begin{aligned}
I(X; Y \mid Z) = \sum_{(x,y,z) \in \{0,1\}^3} P_2(x, y \mid z) \cdot \log\left(\frac{P_2(x, y \mid z)}{P_2(x \mid z) \cdot P_2(y \mid z)}\right) \\
= I(X; Y \mid Z = 0) + I(X; Y \mid Z = 1)
\end{aligned}$$

From Table 1 and Table 3, one can verify that $P_2(z = 0) = P_2(z = 1) = 1/2$ and $I(X; Y \mid Z = 0) = I(X; Y \mid Z = 1)$. So, it suffices to show that $I(X; Y \mid Z = 0) \in \Omega(\alpha^2)$.

$$\begin{aligned}
&I(X; Y \mid Z = 0) \\
&= \sum_{(x,y) \in \{0,1\}^2} P_2(x, y \mid z = 0) \cdot \log\left(\frac{P_2(x, y \mid z = 0)}{P_2(x \mid z = 0) \cdot P_2(y \mid z = 0)}\right) \\
&= \frac{3 + 2\alpha}{8} \cdot \log\left(\frac{\frac{1}{8} \cdot (3 + 2\alpha)}{\frac{3}{4} \cdot \frac{1+\alpha}{2}}\right) + \frac{3 - 2\alpha}{8} \cdot \log\left(\frac{\frac{1}{8} \cdot (3 - 2\alpha)}{\frac{3}{4} \cdot \frac{1-\alpha}{2}}\right) \\
&+ \frac{1}{8} \cdot (1 + 2\alpha) \cdot \log\left(\frac{\frac{1}{8} \cdot (1 + 2\alpha)}{\frac{1}{4} \cdot \frac{1+\alpha}{2}}\right) + \frac{1}{8} \cdot (1 - 2\alpha) \cdot \log\left(\frac{\frac{1}{8} \cdot (1 - 2\alpha)}{\frac{1}{4} \cdot \frac{1-\alpha}{2}}\right) \\
&= \frac{3 + 2\alpha}{8} \cdot \log\left(\frac{3 + 2\alpha}{3 + 3\alpha}\right) + \frac{3 - 2\alpha}{8} \cdot \log\left(\frac{3 - 2\alpha}{3 - 3\alpha}\right) + \frac{1 + 2\alpha}{8} \cdot \log\left(\frac{1 + 2\alpha}{1 + \alpha}\right) + \frac{1 - 2\alpha}{8} \cdot \log\left(\frac{1 - 2\alpha}{1 - \alpha}\right) \\
&= \frac{3}{8} \cdot \log\left(\frac{3 + 2\alpha}{3 + 3\alpha} \cdot \frac{3 - 2\alpha}{3 - 3\alpha}\right) + \frac{1}{8} \cdot \log\left(\frac{1 + 2\alpha}{1 + \alpha} \cdot \frac{1 - 2\alpha}{1 - \alpha}\right) + \frac{\alpha}{4} \cdot \log\left(\frac{3 + 2\alpha}{3 + 3\alpha} \cdot \frac{3 - 3\alpha}{3 - 2\alpha} \cdot \frac{1 + 2\alpha}{1 + \alpha} \cdot \frac{1 - \alpha}{1 - 2\alpha}\right)
\end{aligned}$$

By Taylor series at $\alpha = 0$, one can verify that

- By Taylor series[5] of $\frac{3+2\alpha}{3+3\alpha} \cdot \frac{3-2\alpha}{3-3\alpha} = 1 + \frac{5}{9}\alpha^2 + \frac{5}{9}\alpha^4 + \mathcal{O}(\alpha^6)$
- By Taylor series[6] of $\frac{1+2\alpha}{1+\alpha} \cdot \frac{1-2\alpha}{1-\alpha} = 1 - 3\alpha^2 - 3\alpha^4 + \mathcal{O}(\alpha^6)$
- By Taylor series[7] of $\frac{3+2\alpha}{3+3\alpha} \cdot \frac{3-3\alpha}{3-2\alpha} \cdot \frac{1+2\alpha}{1+\alpha} \cdot \frac{1-\alpha}{1-2\alpha}$
  $= 1 + \frac{4}{3}\alpha + \frac{8}{9}\alpha^2 + \frac{124}{27}\alpha^3 + \frac{464}{81}\alpha^4 + \frac{3844}{243}\alpha^5 + \mathcal{O}(\alpha^6)$

Motivated by the Taylor series, for $0 \leq \alpha \leq 1/2$, one can verify that

- $\frac{3+2\alpha}{3+3\alpha} \cdot \frac{3-2\alpha}{3-3\alpha} \geq 1 + \frac{5}{9}\alpha^2$
- $\frac{1+2\alpha}{1+\alpha} \cdot \frac{1-2\alpha}{1-\alpha} \geq 1 - 3\alpha^2 - 4\alpha^4$
- $\frac{3+2\alpha}{3+3\alpha} \cdot \frac{3-3\alpha}{3-2\alpha} \cdot \frac{1+2\alpha}{1+\alpha} \cdot \frac{1-\alpha}{1-2\alpha} \geq 1 + \frac{4}{3}\alpha$

Thus, using Fact 19, we get

$$
\begin{aligned}
&I(X; Y \mid Z = 0) \\
&= \frac{3}{8} \cdot \log\left(\frac{3+2\alpha}{3+3\alpha} \cdot \frac{3-2\alpha}{3-3\alpha}\right) + \frac{1}{8} \cdot \log\left(\frac{1+2\alpha}{1+\alpha} \cdot \frac{1-2\alpha}{1-\alpha}\right) + \frac{\alpha}{4} \cdot \log\left(\frac{3+2\alpha}{3+3\alpha} \cdot \frac{3-3\alpha}{3-2\alpha} \cdot \frac{1+2\alpha}{1+\alpha} \cdot \frac{1-\alpha}{1-2\alpha}\right) \\
&\geq \frac{3}{8} \cdot \log\left(1 + \frac{5}{9}\alpha^2\right) + \frac{1}{8} \cdot \log\left(1 - 3\alpha^2 - 4\alpha^4\right) + \frac{\alpha}{4} \cdot \log\left(1 + \frac{4}{3}\alpha\right) \\
&\geq \log_2(e) \cdot \left(\frac{3}{8} \cdot \left(\frac{5}{9}\alpha^2 - \left(\frac{5}{9}\alpha^2\right)^2\right) - \frac{1}{8} \cdot \left(3\alpha^2 + 4\alpha^4 + \left(3\alpha^2 + 4\alpha^4\right)^2\right) + \frac{\alpha}{4} \cdot \left(\frac{4}{3}\alpha - (\frac{4}{3}\alpha)^2\right)\right) \\
&= \log_2(e) \cdot \left(\frac{\alpha^2}{6} - \mathcal{O}(\alpha^3)\right) \\
&\in \mathcal{O}(\alpha^2)
\end{aligned}
$$

□

Lemma 13. *Let G_1 be $X \to Z \to Y$ and G_2 be $X \to Z \leftarrow Y$, such that* $\mathrm{skel}(G_1) = \mathrm{skel}(G_2)$ *is $X - Z - Y$. With respect to Eq. (3), we have the following:*

1. $d_{\mathrm{H}}^2(P_1, P_2) \in \mathcal{O}(\varepsilon)$
2. $d_{\mathrm{KL}}(P_1 \parallel P_{1,G_1}) = 0$ *and* $d_{\mathrm{KL}}(P_1 \parallel P_{1,G_2}) \in \Omega(\varepsilon)$
3. $d_{\mathrm{KL}}(P_2 \parallel P_{2,G_2}) = 0$ *and* $d_{\mathrm{KL}}(P_2 \parallel P_{2,G_1}) \in \Omega(\varepsilon)$

Proof. Combine Lemma 20, Lemma 21, and Lemma 22 with α as $\sqrt{\varepsilon}$. □

[5] e.g. see https://www.wolframalpha.com/input?i=taylor+series+of+%5Cfrac%7B3+%2B+2+%5Calpha%7D%7B3+%2B+3+%5Calpha%7D+%5Ccdot+%5Cfrac%7B3+-+2+%5Calpha%7D%7B3+-+3+%5Calpha%7D+at+alpha+%3D+0

[6] e.g. see https://www.wolframalpha.com/input?i=taylor+series+of+%5Cfrac%7B1+%2B+2+%5Calpha%7D%7B1+%2B+%5Calpha%7D+%5Ccdot+%5Cfrac%7B1+-+2+%5Calpha%7D%7B1+-+%5Calpha%7D+at+alpha+%3D+0

[7] e.g. see https://www.wolframalpha.com/input?i=taylor+series+of+%5Cfrac%7B3+%2B+2+%5Calpha%7D%7B3+%2B+3+%5Calpha%7D+%5Ccdot+%5Cfrac%7B3+-+3+%5Calpha%7D%7B3+-+2+%5Calpha%7D+%5Ccdot+%5Cfrac%7B1+%2B+2+%5Calpha%7D%7B1+%2B+%5Calpha%7D+%5Ccdot+%5Cfrac%7B1+-+%5Calpha%7D%7B1+-+2+%5Calpha%7D+at+0