Learning bounded-degree polytrees with known skeleton

Davin Choo*[1], Joy Qiping Yang*[2], Arnab Bhattacharyya[3], and Clément L. Canonne[4]

[1,2,3]National University of Singapore
[2,4]University of Sydney

Editors: Claire Vernade and Daniel Hsu

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 model. Recently, Bhattacharyya et al. (2021) obtained finite-sample guarantees for recovering tree-structured Bayesian networks, i.e., 1-polytrees. We extend their results by providing an efficient algorithm which learns d-polytrees in polynomial time and sample complexity for any bounded d when the underlying undirected graph (skeleton) is known. We complement our algorithm with an information-theoretic sample complexity lower bound, showing that the dependence on the dimension and target accuracy parameters are nearly tight.

Keywords: Bayesian network, Polytree, Sample complexity

1. Introduction

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

Bayesian networks (or *Bayes nets* in short) represent a class of high-dimensional distributions that can be explicitly described by how each variable is 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 graph (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[1] 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 general (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

[*] Equal contribution

[1] In terms of computational efficiency, one can efficiently learn the distribution by following the DAG structure to learn each local probability table. See (Bhattacharyya et al., 2023, Section 6) for a formal analysis of such an approach; there is an exponential dependency on the number of parents but this is unavoidable.

than trying to recover an exact graph structure from the Markov equivalence class of P. That is, prior hardness results on *exact* recovery do not carry over to *approximate* recovery in the PAC sense.

Polytrees are a subclass of Bayesian networks where the undirected graph underlying the DAG is a forest, i.e. there are no cycles in the undirected graph obtained by ignoring edge directions. A polytree with maximum in-degree d is also known as a d-polytree. With an infinite number of samples, one can recover the 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).

Here, 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. We give 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. We also give information-theoretic sample complexity lower bounds that hold even when the ground truth skeleton is known and given to us.

Theorem 1 *Consider a discrete distribution P on n variables, each with alphabet Σ, defined on a polytree G^* with an unknown maximum in-degree d^*. Given m samples from P, accuracy parameter $\varepsilon > 0$, failure probability δ, the skeleton of G^*, and an in-degree upper bound $d \geq d^*$, there exists an algorithm that outputs a d-polytree distribution $\hat{P}$ 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 given only an upper bound d on the true in-degree d^*. In particular, our result yields a sample complexity upper bound of $\tilde{O}(n/\varepsilon)$ for learning $O(1)$-polytrees with constant $|\Sigma|$ and d. Note that this dependence on the dimension n and the accuracy parameter ε is optimal, up to logarithmic factors: indeed, we establish in Theorem 16 an $\Omega(n/\varepsilon)$ sample complexity lower bound for this question, even for $d = 2$ and $|\Sigma| = 2$.[2]

We also state sufficient distributional conditions 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. 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.

[2] 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 $d = 1$.

1.1. Overview of algorithm

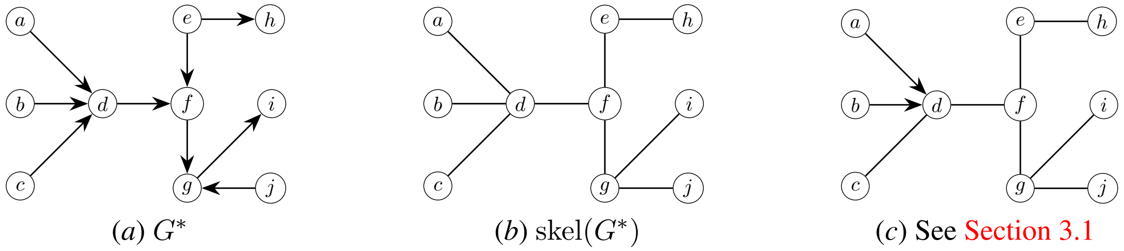
Our algorithm is designed with KL divergence in mind; see Eq. (1). Since there are efficient algorithms for estimating the parameters of a Bayes net with in-degree d once a close-enough graph $\hat{G}$ is recovered (Dasgupta, 1997; Bhattacharyya et al., 2020), it suffices to find a good approximation of the underlying DAG G^*. For any distribution P and DAG G, let P_G be defined as the projection of P unto a DAG G; see Section 2.2 for the formal definition. Then, for a distribution P that is Markov with respect to a DAG G^*, the quality (in terms of KL divergence) of approximating G^* with G is $d_{\mathrm{KL}}(P, P_G) = d_{\mathrm{KL}}(P_{G^*}, P_G) = \sum_{v \in V} I(v; \pi_{G^*}(v)) - I(v; \pi_G(v))$, where $I(\cdot; \cdot)$ refers to *mutual information (MI)*, and $\pi_G(v)$ is the set of parents of v in G. When the true skeleton $\mathrm{skel}(G^*)$ is given to us in advance, what remains is to orient each edge. As such, given error parameter $\varepsilon > 0$ and upper bound on in-degree d, the goal of our algorithm is to judiciously orient the edges of $\mathrm{skel}(G^*)$ such that $d_{\mathrm{KL}}(P_{G^*}, P_G)$ is at most ε while ensuring that every vertex has at most d incoming edges.

Our algorithm relies on estimating MI and conditional MI (CMI) terms involving subsets of variables. A naïve approach of estimating these terms additively would incur an unnecessary overhead on the sample complexity. One of our technical contributions is to show that it suffices to have access to a *tester* that can distinguish between a CMI term being 0 or at least η, for some threshold $\eta > 0$. As shown in Bhattacharyya et al. (2021), the sample complexity for testing (see Corollary 4) is an $O(\eta)$ factor smaller than that for estimating the CMI up to additive error of $\pm\, \eta/2$.

Our algorithm works in three phases. In the first phase, we orient "strong v-structures"; see Section 2 for a precise definition. In the second phase, we locally check if an edge is "forced" to orient in a specific direction. In the third phase, we orient the remaining unoriented edges as a 1-polytree. Throughout the algorithm, we do *not* unorient edges as we will be able to argue that any orientations performed by the first two phases are guaranteed to respect the orientations of the underlying causal graph from which we draw samples from[3].

To explain the intuition behind the first two phases, consider the example of a path on 3 vertices $u - v - w$; see Fig. 1 for a slightly more sophisticated example. If $u \to v \leftarrow w$, then u and w are *dependent* given v. Otherwise, u and w are *independent* given v since G is a polytree. That is, one would expect $I(u; w \mid v)$ to be large if and only if $u \to v \leftarrow w$ was a v-structure. If it is indeed the case that $I(u; w \mid v)$ is "large", then this would be detected by our finite-sample CMI test (i.e. $u \to v \leftarrow w$ was "strong") and so we orient $u \to v$ and $w \to v$ in Phase 1. Now, after Phase 1, some edges in the graph would be partially oriented; say, we have $u \to v - w$ after Phase 1. If $u \to v \to w$ was the ground truth, then $I(u; w \mid v) = 0$ and our tester will detect this term as "small". If $u \to v \leftarrow w$ was the ground truth, then $I(u; w) = 0$ and our tester will detect this term as "small". Via the contrapositive of the previous two statements, if $I(u; w \mid v)$ or $I(u; w)$ is "large", then we know a specific orientation of the edge $v - w$. We may also leave $v - w$ unoriented if neither term was "large". Another form of "forced orientation" is due to the given upper bound d on the number of parents any vertex can have: we should point all remaining incident unoriented edges *away* from a vertex v whenever v already has d incoming arcs. For example, if $d = 1$, then we would immediately orient $v \to w$ if we observe $u \to v - w$ after Phase 1. Given the above intuition, any edge that remains unoriented till the Phase 3 must have been "flexible" in the sense that it could

[3] 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 the underlying distribution is Markov with respect to as the "ground truth".



(*a*) G^* (*b*) $\mathrm{skel}(G^*)$ (*c*) See Section 3.1

Figure 1: 3-polytree example where $I(a; b, c) = I(b; a, c) = I(c; a, b) = 0$ due to deg-3 v-structure centered at d. By Corollary 4, $I(a; f \mid d) = 0$ implies $\hat{I}(a; f \mid d) \leq C \cdot \varepsilon$, and so we will *not* detect $a \to d \to f$ erroneously as a strong deg-2 v-structure $a \to d \leftarrow f$.

be oriented either way. In fact, we later show that "not too much error" has been incurred if the edge orientations from the final phase only increases the incoming degrees of any vertex by at most one.

1.2. Overview of information-theoretic lower bound

Our lower bound shows that $\Omega(n/\varepsilon)$ samples are necessary, *even when a known skeleton is provided*. To show this, we first show that $\Omega(1/\varepsilon)$ samples are required for the case where $n = 3$ by reducing the problem finding an ε-close graph orientation to the problem of *testing* whether samples are drawn from two given distributions. To accomplish this, we designed a pair of distributions P_1 and P_2 and a pair of graphs G_1 and G_2 such that (1) P_1 and P_2 have "small" squared Hellinger distance; (2) P_i has zero KL divergence if projected onto G_i; (3) P_i has "large" KL divergence if projected onto G_j (for $j \neq i$). Since the distributions have small squared Hellinger distance, say less than ε, one needs $\Omega(1/\varepsilon)$ samples to distinguish them, thus showing that $\Omega(1/\varepsilon)$ samples are required for the case where $n = 3$. To obtain a dependency on n, we construct $n/3$ independent copies of the above gadget, à la proof strategy of Bhattacharyya et al. (2021, Theorem 7.6).

1.3. Other related work

Structure learning of Bayesian networks is an old problem in machine learning and statistics that has been intensively studied, e.g. see Koller and Friedman (2009, Chapter 18). Many 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 (2017b); Park and Raskutti (2017); Aragam et al. (2019), with the final three also showing finite sample complexities. Specifically for polytrees, Rebane and Pearl (1988) and Geiger et al. (1990) studied recovery of the DAG for polytrees under the infinite sample regime.

Abbeel et al. (2006) studied the problem of efficiently learning a bounded degree factor graph. Using their method and conversion scheme between factor graphs and Bayes nets, one could efficiently learn polytrees (Bayes nets) with bounded in- *and* out-degrees. However, as we only consider an upper bound on the in-degrees in this work, directly applying their method scales badly in sample complexity (exponential in the number of variables) for even the simple star-like[4] polytree.

More recently, 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 emphasizes 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 also existing works on Bayes net learning with tight bounds in total variation distance with a focus on sample complexity (and not necessarily computational efficiency), e.g. Canonne et al. (2020). Meanwhile, Acharya et al. (2018) consider the problem of learning (in TV distance) a bounded-degree causal Bayes net from interventions, assuming the underlying DAG is known.

1.4. Outline of paper

We introduce 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 (which holds *even when the skeleton is given*). We conclude in Section 6 with some open directions and defer some proofs to the appendix.

2. Preliminaries and tools from previous work

2.1. Preliminary notions and notation

We write $[k]$ to mean $\{1, \ldots, k\}$ and the disjoint union as $\dot\cup$. For any set A, let $|A|$ denote 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, and d to denote an upper bound of d^*.

2.2. Probability distribution definitions

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 $X = \{X_1, \ldots, X_n\}$, 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 $P_G(x_1, \ldots, x_n) = \prod_{x \in X} P(x \mid \pi_G(x))$ where $\pi_G(x)$ are the parents of x in G. Note that P_G is the closest distribution[5] on G to P in d_{KL}, i.e. $P_G =$

[4] A center node v with undirected edges to the rest of the $n-1$ nodes; v's in-degree is d and out-degree is $n-d-1$.

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

$\operatorname{argmin}_{Q \in G} d_{\mathrm{KL}}(P \parallel Q)$. 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.

Our goal is to obtain approximately good graph $\hat{G}$ of P in the sense of $d_{\mathrm{KL}}(P \parallel P_{\hat{G}}) \leq \varepsilon$. With $\hat{G}$, one can employ sample and computational-efficient parameter learning algorithms to output the final hypothesis $\hat{P}$. While one can always model the graph G as a clique (a fully connected DAG) in order to satisfy the ε-close requirement, it takes $\exp(n)$ number of samples in the parameter learning step. Hence, we are interested in obtaining a graph G that is both ε-close and facilitates efficient learning algorithms. For instance, if P was defined on a Bayes net with max in-degree d, then we want G to also be a Bayes net with max in-degree d. This is always possible in $\exp(n)$ time by formulating the search of G as an optimization problem that maximizes the summation of mutual information term (first term in (1)); see (Höffgen, 1993) and why it is NP-hard in general. As each mutual information term can be well-estimated, an ε-close graph could be obtained by optimizing over the empirical mutual information scores. In the case of tree-structured distributions, this can be efficiently solved by using any maximum spanning tree algorithm. 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 (MI) 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 MI is defined as:* $I(X;Y \mid Z) = \sum_{x \in \mathcal{X}, y \in \mathcal{Y}, z \in \mathcal{Z}} P(x,y,z) \cdot \log\left(\frac{P(x,y,z) \cdot P(z)}{P(x,z) \cdot P(y,z)}\right)$.

We will employ the following result (Corollary 4) about conditional mutual information testers, adapted from Theorem 1.3 of Bhattacharyya et al. (2021); see Appendix A.1 for derivation details and Appendix B for a derivation of a constant C that works.

Corollary 4 (Conditional MI tester) *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 $\hat{I}(X;Y \mid Z) \leq C \cdot \varepsilon$, then $I(X;Y \mid Z) < \varepsilon$.
Unconditional statements involving $I(X;Y)$ and $\hat{I}(X;Y)$ hold similarly by setting $|\Sigma_Z| = 1$.

Using the contrapositive of the first statement of Corollary 4 and non-negativity of conditional mutual information, one can also see that if $\hat{I}(X;Y \mid Z) \geq C \cdot \varepsilon$, then $I(X;Y \mid Z) > 0$.

2.3. Graphical notions

Let $G = (V, E)$ be a graph on $|V| = n$ vertices and $|E|$ edges 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'.

We use arrows to denote directed edges, e.g. $u \to v$, use $\pi(v)$ to denote the parents of v, and use $d^{in}(v)$ to denote v's incoming degree. An interesting directed subgraph on three vertices is the *v-structure*: $u \to v \leftarrow w$, $u \not- w$, and v is called the center of the v-structure.

In this work, we study a generalized version of v-structures (*deg-ℓ v-structure*) where the center has $\ell \geq 2$ 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.

Definition 5 (ε-strong deg-ℓ v-structure) *Let $0 < C < 1$ be the universal constant appearing in Corollary 4. A deg-ℓ v-structure is a subgraph on $\ell + 1$ nodes $v, u_1, \ldots, u_\ell$ such that:*
*1. **deg-ℓ v-structure**: $v \leftarrow u_k$ for all $k \in [\ell]$, and $u_k \not- u_{k'}$ for all $k, k' \in [\ell]$ and $k \neq k'$*
*2. **ε-strong**: $I(u_k; \{u_1, u_2, \ldots, u_\ell\} \setminus u_k \mid v) \geq C \cdot \varepsilon$ for all $k \in [\ell]$*

Meek rules (Meek, 1995) are a set of 4 edge orientation rules that are sound and complete with respect to any partially oriented graph that can be fully oriented in an acyclic fashion. Soundness means that any orientation due to rule invocations are correct while completeness means that any orientation that could theoretically be recovered would be recovered. So, one can always repeatedly apply Meek rules till a fixed point to maximize the number of oriented arcs. One particular orientation rule (Meek $R1$)[6] orients $b \to c$ whenever a partially oriented graph has the configuration $a \to b - c$ and $a \not- c$ 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. When we *freely orient* a forest, we pick arbitrary root nodes in the connected components and orient to form a 1-polytree. In polytrees, there is a unique path in $skel(G)$ between any two vertices in the graph. As such, the ancestors of a vertex v are mutually independent, and typically become mutually dependent when v (or any of v's descendants) are being conditioned over.

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, where $d^* \leq d$.

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[7] are the same for any graphs that P is Markov with respect to, the choice of G^* does not matter here.

[6] The other 3 rules involve undirected cycles in the graph and are thus irrelevant in our context where the underlying undirected graph is a tree.

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

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)                      ▷ $\mathcal{O}(n^{d+1})$ 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 $\hat{G}$
```

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 $N^{\mathrm{in}}(v)$ to $N^{\mathrm{in}}(v) \cup \{u\}$ and $N^{\mathrm{un}}(v)$ to $N^{\mathrm{un}}(v) \setminus \{u\}$.
Update $N^{\mathrm{out}}(u)$ to $N^{\mathrm{out}}(u) \cup \{v\}$ and $N^{\mathrm{un}}(u)$ to $N^{\mathrm{un}}(u) \setminus \{v\}$.
```

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 $N^{\mathrm{in}}(v) \subseteq N(v)$ be the current set of incoming neighbors of v. Let $N^{\mathrm{out}}(v) \subseteq N(v)$ be the current set of outgoing neighbors of v. Let $N^{\mathrm{un}}(v) \subseteq N(v)$ be the current set of unoriented neighbors of v. That is, $N(v) = N^{\mathrm{in}}(v) \,\dot{\cup}\, N^{\mathrm{out}}(v) \,\dot{\cup}\, N^{\mathrm{un}}(v)$.

Our algorithm has three phases. 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.[8] 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 $u \to v$.

Example Suppose we have the partially oriented graph Fig. 1(c) after Phase 1. Since $N^{\mathrm{in}}(d) = \{a, b\}$, 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 $d \to c$.

[8] Note that within the for loop from Line 7 of Algorithm 4, neither condition may hold, in which case we do not orient anything, hence the "missing" else.

Algorithm 3 Phase 1: Orient strong v-structures

```
Input: Skeleton $\mathrm{skel}(G^*) = (V, E)$, max in-degree $d$, threshold $\varepsilon > 0$, universal constant $C$
$\gamma \leftarrow d$
while $\gamma \geq 2$ do
    for $v \in V$ do                                                              ▷ Arbitrary order
        for $T \in \mathcal{N}_\gamma$ do     ▷ $\mathcal{N}_\gamma \subseteq 2^{N(v)}$ are the $\gamma$ neighbors of $v$; $|\mathcal{N}_\gamma| = \binom{|N(v)|}{\gamma}$
            if $|T \cup N^{\mathrm{in}}(v)| \leq d$ and $\hat{I}(u; T \setminus \{u\} \mid v) \geq C \cdot \varepsilon, \forall u \in T$ then
                for $u \in T$ do                                               ▷ Strong deg-$\gamma$ v-structure
                    ORIENT($u$, $v$)
    $\gamma \leftarrow \gamma - 1$                                                    ▷ Decrement degree bound
```

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

```
Input: Skeleton skel($G^*$) = $(V, E)$, max in-degree $d$, threshold $\varepsilon > 0$, universal constant $C$
do                                                    ▷ $\mathcal{O}(n)$ iterations, $\mathcal{O}(n^2)$ time per iteration
    if $\exists v \in V$ such that $|N^{\text{in}}(v)| = d$ and $N^{\text{un}}(v) \neq \emptyset$ then      ▷ Meek $R1(d)$
        Orient all unoriented arcs away from $v$
        Update $N^{\text{out}}(v) \leftarrow N^{\text{out}}(v) \cup N^{\text{un}}(v)$; $N^{\text{un}}(v) \leftarrow \emptyset$
    for every node $v \in V$ do
        if $1 \leq |N^{\text{in}}(v)| < d$ then
            for every $u \in N^{\text{un}}(v)$ do
                if $\hat{I}(u; N^{\text{in}}(v) \mid v) > C \cdot \varepsilon$ then ORIENT$(u, v)$
                else if $\hat{I}(u; N^{\text{in}}(v)) > C \cdot \varepsilon$ then ORIENT$(v, u)$
while new edges are being oriented
```

Algorithm 5 Phase 3: Freely orient remaining unoriented edges

```
Input: Skeleton skel($G^*$) = $(V, E)$, max in-degree $d$, threshold $\varepsilon > 0$, universal constant $C$
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 $\hat{G}$
```

3.2. Analysis

In our subsequent analysis, we rely on the conclusions of Corollary 4 with error tolerance $\varepsilon' = \frac{\varepsilon}{2n\cdot(d+1)}$. Via a union bound over $\mathcal{O}(n^{d+1})$ events, Lemma 6 ensures that *all* our (conditional) MI tests in Algorithm 1 will behave as expected with probability at least $1 - \delta$, with sufficient samples.

Lemma 6 *Suppose all variables in the Bayesian network have alphabet Σ, for $|\Sigma| \geq 2$. For $\varepsilon' > 0$, $\mathcal{O}(n^{d+1})$ statements of the following forms all simultaneously 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 $\hat{I}(\mathbf{X}; \mathbf{Y} \mid Z) \leq C \cdot \varepsilon'$, then $I(\mathbf{X}; \mathbf{Y} \mid Z) < \varepsilon'$.
with m empirical samples, where $Z \in V \cup \{\emptyset\}$, $\mathbf{X}, \mathbf{Y} \subseteq V \setminus \{Z\}$, $|\mathbf{X} \,\dot{\cup}\, \mathbf{Y}| \leq d$, and

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

Proof Set $\varepsilon' = \frac{\varepsilon}{2n\cdot(d+1)}$, then use Corollary 4 and apply union bound over $\mathcal{O}(n^{d+1})$ tests. ■

In the remaining of our analysis, we will analyze under the assumption that all our $\mathcal{O}(n^{d+1})$ tests are correct with the required tolerance level. Full proofs are deferred to Appendix A.2.

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 and $\hat{G}$ be returned graph of Algorithm 1. Let us denote the final $N^{\text{in}}(v)$ as $\pi^{\text{in}}(v)$ at the end of Phase 2, just before freely orienting, i.e. the vertices pointing into v in $\hat{G} \setminus H$. Then, $\pi^{\text{un}}(v) = \pi(v) \setminus \pi^{\text{in}}(v)$ is the set of ground truth parents that are not identified in both Phase 1 and Phase 2. Lemma 7 argues that the algorithm does not make mistakes for orientations in $\hat{G} \setminus H$, so all edges in $\pi^{\text{un}}(v)$ will be unoriented at the end of Phase 2.

Lemma 7 *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. $\pi^{\text{in}}(v) \subseteq \pi(v)$, and $N^{\text{in}}(v)$ at any time during the algorithm will have $N^{\text{in}}(v) \subseteq \pi^{\text{in}}(v)$.*

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 $\sum_{v \in V} I(v; \pi(v)) - \sum_{v \in V} I(v; \hat{\pi}(v))$ 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 $\pi^{\text{in}}(v) \subseteq \pi(v)$ and $I(v; u)$ for $u \in \pi^{\text{un}}(v)$, we use Lemma 9 which relies on repeated applications of Lemma 8. To lower bound $\sum_{v \in V} I(v; \hat{\pi}(v))$, we use Lemma 10.

Lemma 8 *Fix any vertex v, any $S \subseteq \pi^{\text{un}}(v)$, and any $S' \subseteq \pi^{\text{in}}(v)$. 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 9 *For any vertex v with $\pi^{\text{in}}(v)$, we can show that*

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

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[9] an additive ε error per vertex.

Lemma 10 *Consider an arbitrary vertex v with $\pi^{\text{in}}(v)$ at the start of Phase 3. If Phase 3 orients $u \to v$ for some $u - v \in H$, then*

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

By using Lemma 9 and Lemma 10, we can show our desired KL divergence bound (Lemma 11).

Lemma 11 *Let $\pi(v)$ be the true parents of v. Let $\hat{\pi}(v)$ be the proposed parents of v output by our algorithm. Then,*

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

Note that we have a bound with respect to the true max-degree d^* despite only given an upper bound d as input. With these results in hand, we are ready to establish our main theorem.

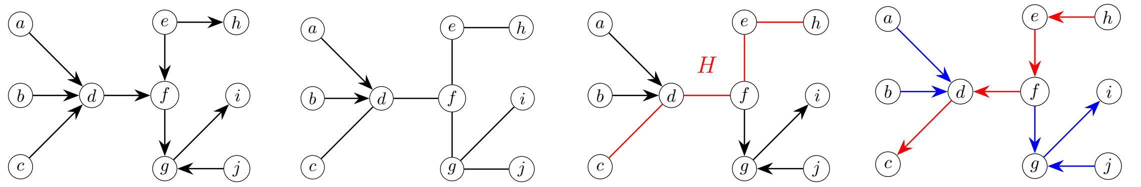
Proof of Theorem 1 We first combine Lemma 11 and Lemma 6 with $\varepsilon' = \frac{\varepsilon}{2n \cdot (d+1)} \leq \frac{\varepsilon}{2n \cdot (d^*+1)}$ in order to obtain an orientation $\hat{G}$ which is close to G^*. Now, much similar to the proof of Bhattacharyya et al. (2021, Theorem 1.4), we recall that there exist efficient algorithms for estimating

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

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 $\tilde{\mathcal{O}}(n \cdot |\Sigma|^d/\varepsilon)$. 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:

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

The first inequality follows from our graph learning guarantees on $\hat{G}$ while the second is due to performing parameter learning algorithms on $\hat{G}$. Thus, $d_{\mathrm{KL}}(P \parallel \hat{P}_{\hat{G}}) \leq \varepsilon + d_{\mathrm{KL}}(P \parallel P_{G^*}) = \varepsilon$. ■



(a) Ground truth G^*. $\pi(d) = \{a, b, c\}$

(b) Midway of Phase 1. $N^{in}(d) = \{a, b\}$

(c) Before final phase. $\pi^{in}(d) = \{a, b\}$

(d) Proposed graph $\hat{G}$. $\hat{\pi}(d) = \{a, b, f\}$

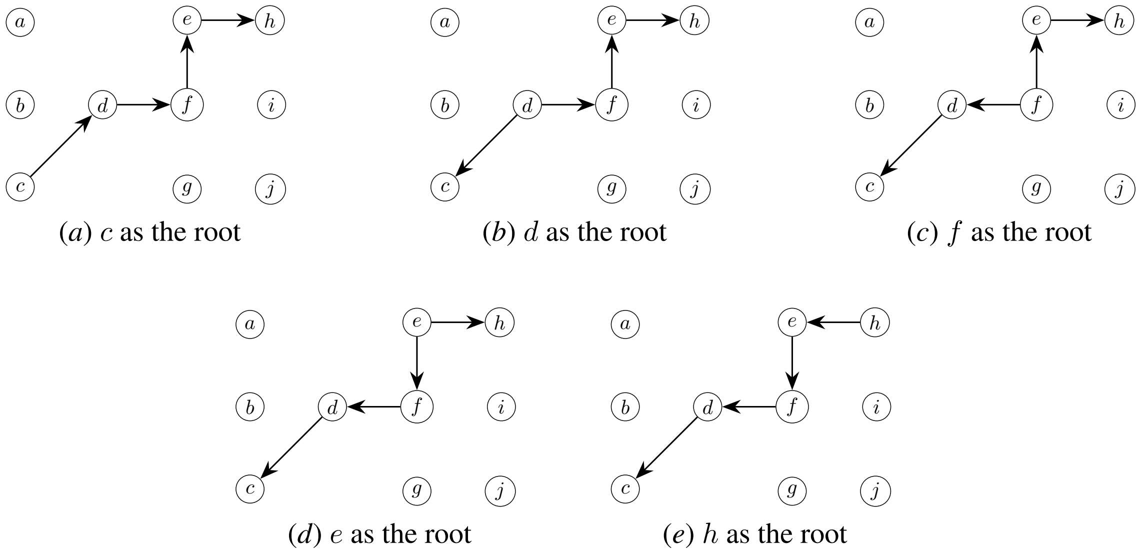
Figure 2: An example run to illusrate notations. In G^*, vertex d has parents $\pi(d) = \{a, b, c\}$. While the algorithm executes, we track a tentative parent set $N^{in}(d)$ of d and fix it to $\pi^{in}(d)$ right before the final phase. Since $d = 3$, observe that $g \to i$ must have been oriented due to a local search step and *not* due to Meek $R1(3)$ in Phase 2. At the end, in $\hat{G}$, the proposed parent set of d is $\hat{\pi}(d) = \{a, b, f\}$. Note that $\hat{G}$ only shows one possible orientation of the red unoriented subgraph H before the final phase; see Fig. 3 for others.

4. Skeleton assumption

In this section, we present a set of *sufficient* assumptions (Assumption 12) under which the Chow-Liu algorithm will recover the true skeleton even with finite samples. We note that the conditions listed here are in spirit very similar to the assumptions made to recover exact graphical structures in other works (Gao and Aragam, 2021; Ghoshal and Honorio, 2017a; Gao et al., 2022), i.e., assuming a sufficiently detectable gap on an edge or from an alternate graph. Otherwise, it is not hard to find counter examples to thwart learners from recovering the correct network structure with finite sample access.[10] As such, it is often necessary to make these assumptions for exact structure recovery. Aside from the ones presented here, Barik and Honorio (2020); Chatterjee and Vidyasagar (2022) study other sufficient conditions for recovering the skeleton of Polytrees (or Bayes nets).

Nevertheless, we would like to highlight that our paper has made progress in polytree PAC-learning in the following statisical sense: it suffices to have exact first order mutual information and approximate higher order mutual information to learn (most) bounded in-degree polytrees in polynomial time. For prior works, it is only known that one can recover polytrees efficiently with exact first and second order mutual information (Rebane and Pearl, 1988) or exponential time algorithm for approximating bounded in-degree Bayes nets (Kitson et al., 2023).

[10] E.g., a distribution on $X \to Y$ with infinitely small $I(X;Y)$ and given finite sample access no algorithm can distinguish the actual graph from the empty graph.



(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 3: 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 12 *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 $I(u; v) + \varepsilon_P \leq I(a; b)$ for every pair of adjacent vertices $a - b$ in the path.
(2) For every pair of directly connected nodes $a - b$ in G^*, $I(a; b) \geq \varepsilon_P$.

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 maximum spanning tree on the estimated mutual information on each pair of vertices) recovers $\mathrm{skel}(G^*)$. See Appendix A.3 for the full proof.

Lemma 13 *Under Assumption 12, 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 13 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 $X - Z - Y$. Letting $Bern(1/2)$ denote the Bernoulli distribution with parameter 1/2, i.e. a fair coin flip, we 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. By defining $P_{i,G}$ as projecting P_i onto G, one can show Lemma 14; see Appendix A.4 for its proof.

Lemma 14 (Key lower bound lemma) *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 15) 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. Put differently, if one can solve the problem in Lemma 15, then one can use that algorithm to solve the problem in Lemma 14. However, it is well-known that distinguishing two distributions whose squared Hellinger distance is ε requires $\Omega(1/\varepsilon)$ samples (e.g. see (Bar-Yossef, 2002, Theorem 4.7)).

Lemma 15 *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 16 *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 and discussion

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 12 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.

It is natural to ask whether what we can do with access to a false skeleton that is approximately correct (i.e. has some orientation close in KL to the ground truth) produced by running the Chow-Liu algorithm on the sample statistics. However, it is unclear to us why we can hope to design efficient algorithms with provable guarantees in this case for two reasons:

- The Chow-Liu algorithm only uses order-1 mutual information while the KL divergence of Equation (1) requires information from order-d mutual information. It is unclear why one can hope that this false skeleton would yield provable guarantees with respect to Eq. (1).
- An "approximately correct" skeleton may have potentially unknown number of edges in the skeleton being wrong and we do not see how to design efficient global orientation algorithms using only statistics from the ground truth samples.

Without the true skeleton, a "local algorithm" (such as ours) can be tricked into some "local optima" and it is hard to argue why the output would obtain "global guarantees" with respect to the parent sets of Eq. (1).

Another 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. In particular, we are unaware of any obstruction a lower bound for $|\Sigma| > 2$ and $d > 2$. While we do not know an optimal construction, the following construction (emulating Appendix A.2 of Canonne et al. (2020)) yields $\Omega(\frac{n2^d}{(d+1)\varepsilon^2})$, showing that the exponential dependence on d is unavoidable. Consider $\frac{n}{d+1}$ stars with binary alphabets, where each star center has d incoming parents. Each parental node is set to be an independent uniform coin flip over the binary alphabet and so it takes $\Omega(2^d/\varepsilon^2)$ to learn each star to accuracy ε. As KL is additive, one would require any constant fraction of the stars to incur less than $\frac{\varepsilon(d+1)}{n}$ error. To do so, one would need $\Omega(\frac{n2^d}{(d+1)\varepsilon^2})$ samples.

Acknowledgments

This research/project is supported by the National Research Foundation, Singapore under its AI Singapore Programme (AISG Award No: AISG-PhD/2021-08-013). CC was supported by an ARC DECRA (DE230101329) and an unrestricted gift from Google Research. Joy is supported by the JD Technology Scholarship. We would like to thank Vipul Arora for valuable feedback and discussions.

References

Pieter Abbeel, Daphne Koller, and Andrew Y. Ng. Learning factor graphs in polynomial time and sample complexity. *J. Mach. Learn. Res.*, 7:1743–1788, 2006.

Jayadev Acharya, Arnab Bhattacharyya, Constantinos Daskalakis, and Saravanan Kandasamy. Learning and testing causal models with interventions. In *NeurIPS*, pages 9469–9481, 2018.

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

Ziv 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.

Adarsh Barik and Jean Honorio. Provable efficient skeleton learning of encodable discrete bayes nets in poly-time and sample complexity. In *ISIT*, pages 2486–2491. IEEE, 2020.

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

Arnab Bhattacharyya, Sutanu Gayen, Eric 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.

Arnab Bhattacharyya, Sutanu Gayen, Eric Price, Vincent Y. F. Tan, and N. V. Vinodchandran. Near-optimal learning of tree-structured distributions by chow and liu. *SIAM J. Comput.*, 52(3):761–793, 2023.

Clément L. Canonne, Ilias Diakonikolas, Daniel M. Kane, and Alistair Stewart. Testing bayesian networks. *IEEE Trans. Inf. Theory*, 66(5):3132–3170, 2020. Preprint available at arXiv:1612.03156.

Sourav Chatterjee and Mathukumalli Vidyasagar. Estimating large causal polytree skeletons from small samples. *CoRR*, abs/2209.07028, 2022.

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

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

David Maxwell Chickering, David Heckerman, and Christopher 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.

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

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

Nir Friedman and Zohar 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.

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

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

Ming Gao, Wai Ming Tai, and Bryon Aragam. Optimal estimation of gaussian DAG models. In *AISTATS*, volume 151 of *Proceedings of Machine Learning Research*, pages 8738–8757. PMLR, 2022.

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

Asish Ghoshal and Jean Honorio. Learning identifiable gaussian bayesian networks in polynomial time and sample complexity. In *NIPS*, pages 6457–6466, 2017a.

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

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

Patrik Hoyer, Dominik Janzing, Joris M Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear causal discovery with additive noise models. *Advances in Neural Information Processing Systems*, 21, 2008.

Finn V Jensen and Thomas Dyhre Nielsen. *Bayesian networks and decision graphs*, volume 2. Springer, 2007.

Neville Kenneth Kitson, Anthony C. Constantinou, Zhigao Guo, Yang Liu, and Kiattikun Chobtham. A survey of bayesian network structure learning. *Artif. Intell. Rev.*, 56(8):8721–8814, 2023.

Daphne Koller and Nir Friedman. *Probabilistic Graphical Models - Principles and Techniques*. MIT Press, 2009.

Christopher 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.

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

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

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

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

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

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

Appendix 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). See Appendix B for a derivation of a constant C that works.

Theorem 17 (Conditional MI 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 $I(\hat{X}; \hat{Y} \mid \hat{Z}) > C \cdot I(X; Y \mid Z)$.*

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

Corollary 4 (Conditional MI tester) *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 $\hat{I}(X; Y \mid Z) \leq C \cdot \varepsilon$, then $I(X; Y \mid Z) < \varepsilon$.
Unconditional statements involving $I(X; Y)$ and $\hat{I}(X; Y)$ hold similarly by setting $|\Sigma_Z| = 1$.

Proof In the original proof of (1) in (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, yielding our first statement.

Now, suppose $\hat{I}(X; Y \mid Z) \leq C \cdot \varepsilon$. Assume, for a contradiction, that $I(X; Y \mid Z) \geq \varepsilon$. Then, statement 2 of Theorem 17 tells us that $\hat{I}(X; Y \mid Z) > C \cdot I(X; Y \mid Z) \geq C \cdot \varepsilon$. This contradicts 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 18) of mutual information and two properties about (conditional) mutual information on a polytree (Lemma 19) which will be helpful in our proofs later.

Lemma 18 (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 19 *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 18 by observing 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 7 *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. $\pi^{\text{in}}(v) \subseteq \pi(v)$, and $N^{\text{in}}(v)$ at any time during the algorithm will have $N^{\text{in}}(v) \subseteq \pi^{\text{in}}(v)$.*

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. 1(c) 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 $d^* \leq d$ incoming arcs to any vertex, the forced orientations due to Meek $R1(d)$ will always be correct.

■

Lemma 8 *Fix any vertex v, any $S \subseteq \pi^{\mathrm{un}}(v)$, and any $S' \subseteq \pi^{\mathrm{in}}(v)$. 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 $u \in S \cup S'$ such that $\hat{I}(u; S \cup S' \setminus \{u\} \mid v) \leq C \cdot \varepsilon$. To see why, we need to look at Line 5 of Algorithm 3 where we check all subsets T of $\pi(v)$ (as well as some other sets) to see if *every* $u \in T$ satisfies $\hat{I}(u; T \setminus \{u\} | v) \geq C \cdot \varepsilon$. From here, we can see that if a subset T of $\pi(v)$ is not *all* oriented into v, then we know that from Line 5 of Algorithm 3 that there exists some $u \in T$ such that $\hat{I}(u; T \setminus \{u\} | v) < C \cdot \varepsilon$. Applying this to $T = S \cup S'$, where the set of unoriented neighboring nodes S is non-empty, we have our claim. As $\hat{I}(u; S \cup S' \setminus \{u\} \mid v) \leq C \cdot \varepsilon$, Corollary 4 tells us that $I(u; S \cup S' \setminus \{u\}) < \varepsilon$, 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 9 *For any vertex v with $\pi^{\mathrm{in}}(v)$, we can show that*

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

Proof Initializing $S' = \pi^{in}(v)$ and $S = \pi(v) \setminus \pi^{in}(v) = \pi^{un}(v)$, we can repeatedly apply Lemma 8 to remove vertices one by one, until $S = \emptyset$. Without loss of generality, by relabelling the vertices, we may assume that Lemma 8 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 $S' = \pi^{in}(v)$.

$$\begin{aligned}
I(v;\pi(v)) &\leq I(v;\pi(v) \setminus \{u_1\}) + I(v;u_1) + \varepsilon && \text{By Lemma 8}\\
&\leq I(v;\pi(v) \setminus \{u_1,u_2\}) + I(v;u_1) + I(v;u_2) + 2\varepsilon && \text{By Lemma 8}\\
&\leq \ldots\\
&\leq I(v;\pi(v) \setminus U) + \sum_{u \in U} I(v;u) + \varepsilon \cdot |U| && \text{By Lemma 8}
\end{aligned}$$

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

$$\begin{aligned}
&I(v;\pi^{in}(v))\\
&= I(v;\pi^{in}(v) \setminus U) + I(v;\pi^{in}(v) \cap U) + I(\pi^{in}(v) \cap U;\pi^{in}(v) \setminus U \mid v) && \text{By Lemma 19}\\
&\geq I(v;\pi^{in}(v) \setminus U) + I(v;\pi^{in}(v) \cap U)\\
&\geq I(v;\pi^{in}(v) \setminus U) + \sum_{u \in \pi^{in}(v) \cap U} I(v;u) && \text{By Lemma 19}
\end{aligned}$$

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

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

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,

$$\begin{aligned}
&|I(v; \pi^{in}(v) \cup u) - I(v; \pi^{in}(v)) - I(v; u)| \\
&= |I(u; \pi^{in}(v) \mid v) - I(u; \pi^{in}(v))| && \text{By Lemma 18} \\
&= \max\{I(u; \pi^{in}(v) \mid v), I(u; \pi^{in}(v))\} && \text{At most one of these term can be non-zero} \\
&\leq \varepsilon
\end{aligned}$$

■

Lemma 11 *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. 4 for an illustration.

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

$$\begin{aligned}
&\sum_{v \in V} I(v; \hat{\pi}(v)) \\
&= \sum_{v \in A} I(v; \hat{\pi}(v)) + \sum_{v \in V \setminus A} I(v; \hat{\pi}(v)) \\
&\geq \sum_{v \in A} \Big(I(v; \hat{\pi}(v) \setminus \{a_v\}) + I(v; a_v) - \varepsilon \Big) + \sum_{v \in V \setminus A} I(v; \hat{\pi}(v)) && \text{By Lemma 10} \\
&= \sum_{v \in A} I(v; \pi^{in}(v)) + \sum_{v \in A} I(v; a_v) + \sum_{v \in V \setminus A} I(v; \pi^{in}(v)) - |A| \cdot \varepsilon \\
&= \sum_{v \in V} I(v; \pi^{in}(v)) + \sum_{v \in A} I(v; a_v) - |A| \cdot \varepsilon \\
&\geq \sum_{v \in V} I(v; \pi^{in}(v)) + \sum_{v \in A} I(v; a_v) - n\varepsilon && \text{Since } A \subseteq V \text{ and } |V| = n
\end{aligned}$$

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

$$\sum_{v \in V} I(v; \pi(v))$$

$$
\begin{aligned}
&= \sum_{\substack{v \in V \\ \pi^{un}(v) \neq \emptyset}} I(v; \pi(v)) + \sum_{\substack{v \in V \\ \pi^{un}(v) = \emptyset}} I(v; \pi(v)) \\
&\leq \sum_{\substack{v \in V \\ \pi^{un}(v) \neq \emptyset}} \left(\varepsilon \cdot |\pi(v)| + I(v; \pi^{in}(v)) + \sum_{u \in \pi^{un}(v)} I(v; u) \right) + \sum_{\substack{v \in V \\ \pi^{un}(v) = \emptyset}} I(v; \pi(v)) \quad \text{By Lemma 9} \\
&= \sum_{v \in V} I(v; \pi^{in}(v)) + \sum_{\substack{v \in V \\ \pi^{un}(v) \neq \emptyset}} \left(\varepsilon \cdot |\pi(v)| + \sum_{u \in \pi^{un}(v)} I(v; u) \right)
\end{aligned}
$$

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

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

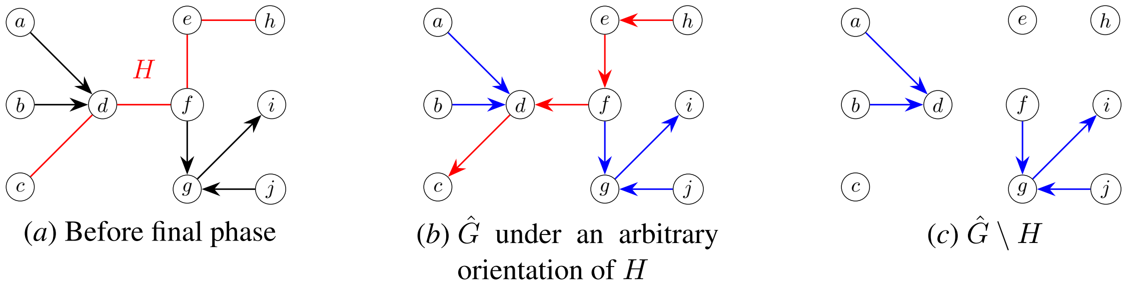
where the last equality is because the last two terms are two different ways to enumerate the edges of H, e.g. see Fig. 4. ■

A.3. Skeleton assumption

Lemma 13 *Under Assumption 12, 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 12, one can estimate $\hat{I}(u; v)$ up to $(\varepsilon_P)/3$-closeness with high probability in n, i.e. $|I(u; v) - \hat{I}(u; v)| \leq \varepsilon_P/3$ for any pair of vertices $u, v \in V$.

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



(a) Before final phase

(b) $\hat{G}$ under an arbitrary orientation of H

(c) $\hat{G} \setminus H$

Figure 4: Illustration of notation used in proof of Lemma 11. Suppose (a) is the partial orientation of Fig. 1 after Phase 2, with H as the edge-induced subgraph on the unoriented edges in red. Before the final phase, we have $\pi^{\text{in}}(d) = \{a, b\}$, $\pi^{\text{in}}(g) = \{f, j\}$, $\pi^{\text{in}}(i) = \{g\}$, $\pi^{\text{un}}(c) = \{d\}$, $\pi^{\text{un}}(d) = \{c, f\}$, $\pi^{\text{un}}(f) = \{d, e\}$, $\pi^{\text{un}}(e) = \{h, f\}$, and $\pi^{\text{un}}(h) = \{e\}$. With respect to H's orientation in (b), 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 $a_\square$s are two different ways to refer to the red edges and (b) only shows one possible orientation of H (see Fig. 3 for others).

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,

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

for any $i \in \{1, \ldots, k\}$. Since $\hat{I}(u; v) \leq \hat{I}(z_i, z_{i+1}) - \varepsilon_P/3$ for each $i \in \{1, \ldots, k\}$, 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

$$\hat{I}(u; v) \leq I(u; v) + \varepsilon_P/3 = \varepsilon_P/3 < 2 \cdot \varepsilon_P/3 \leq I(a; b) - \varepsilon_P/3 \leq \hat{I}(a; b)$$

That is, the Chow-Liu algorithm will always consider edges crossing different components *after* all true edges have been considered. ■

A.4. Proof of key lower bound lemma

We will use the following inequality in our proofs.

Fact 20 *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.

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	$\frac{3}{16} \cdot (1+\alpha)$	$\frac{1}{16} \cdot (3+2\alpha)$

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	$\frac{1}{8} \cdot (2+\alpha)$

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 1: Explicit (conditional) probability tables for our lower bound construction.

Lemma 21 $d_H^2(P_1, P_2) \leq \alpha^2$

Proof From Table 1, we see that

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

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

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

Hence,

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

■

Lemma 22 $d_{\mathrm{KL}}(P_1 \parallel P_{1,G_1}) = 0$ *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

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

where $(\dagger)$ is by applying Lemma 18 with $v = Z$, $A = \{X\}$, $B = \{Y\}$ and $(*)$ is because $I(X;Y\mid Z) = 0$ in P_1.

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

$$\begin{aligned}
I(X;Y) &= \sum_{(x,y)\in\{0,1\}^2} P_1(x,y)\cdot\log\frac{P_1(x,y)}{P_1(x)\cdot P_1(y)} \\
&= \frac{1}{4}\cdot\left((2+\alpha)\cdot\log\left(1+\frac{\alpha}{2}\right)+(2-\alpha)\cdot\log\left(1-\frac{\alpha}{2}\right)\right) && \text{From Table 1} \\
&\geq \frac{1}{4}\cdot\log_2(e)\cdot\frac{\alpha}{2}\cdot\left((2+\alpha)\cdot\left(1-\frac{\alpha}{4}\right)-(2-\alpha)\cdot\left(1+\frac{\alpha}{4}\right)\right) && \text{By Fact 20}
\end{aligned}$$

$$\in \Omega(\alpha^2)$$

■

Lemma 23 *$d_{\mathrm{KL}}(P_2 \parallel P_{2,G_2}) = 0$ 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

$$\begin{aligned} & d_{\mathrm{KL}}(P_2 \parallel P_{2,G_1}) \\ &= I(Z;X,Y) - I(X;Z) - I(Z;Y) \\ &= (I(Z;X) + I(Z;Y) + I(X;Y \mid Z) - I(X;Y)) - I(X;Z) - I(Z;Y) && (\dagger) \\ &= I(X;Y \mid Z) - I(X;Y) \\ &= I(X;Y \mid Z) && (*) \end{aligned}$$

where $(\dagger)$ is by applying Lemma 18 with $v = Z$, $A = \{X\}$, $B = \{Y\}$ and $(*)$ is because $I(X;Y) = 0$ in P_2.

We will now show that $I(X;Y \mid Z) \in \Omega(\alpha^2)$. 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, 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 $I(X;Y \mid Z=0) \in \Omega(\alpha^2)$.

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

Using Taylor series expansion, one can verify that for $0 \leq \alpha \leq 1/2$,

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

Thus, using Fact 20, we get

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

■

Lemma 14 (Key lower bound lemma) *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 Theorem 21, Theorem 22, and Theorem 23 with α as $\sqrt{\varepsilon}$. ■

Lemma 15 *Even when given* $\mathrm{skel}(G^*)$*, it takes $\Omega(1/\varepsilon)$ samples to learn an ε-close graph orientation of G^* for distributions on $\{0,1\}^3$.*

Proof Our result 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 (sharing the same skeleton but having different orientations), with respect to the construction in Lemma 14. 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. Put differently, if one can learn an ε-close graph orientation, then one can use that algorithm to distinguish between P_1 and P_2 in Lemma 14. However, it is well-known that distinguishing two distributions whose squared Hellinger distance is ε requires $\Omega(1/\varepsilon)$ samples (e.g. see (Bar-Yossef, 2002, Theorem 4.7)). ■

Theorem 16 *Even when given* $\mathrm{skel}(G^*)$*, it takes $\Omega(n/\varepsilon)$ samples to learn an ε-close graph orientation of G^* for distributions on $\{0,1\}^n$.*

Proof Consider a distribution P on $n/3$ independent copies of the lower bound construction from Lemma 15, where each copy is indexed by P_i for $i \in \{1, \ldots, n/3\}$. Suppose, for a contradiction, that the algorithm draws cn/ε samples for sufficiently small $c > 0$, and manages to output Q that is ε-close to P with probability at least $2/3$. From Lemma 15 with error tolerance $\Omega(\varepsilon/n)$, we know that each copy is *not* $\Omega(\varepsilon/n)$-close with probability at least $1/5$. By Chernoff bound, at least $\Omega(n)$ copies are *not* $\Omega(\varepsilon/n)$-close with probability at least $2/3$. Then, by the tensorization of KL divergence, we see that $d_{\mathrm{KL}}\left(\prod_{i=1}^{n/3} P_i \parallel \prod_{i=1}^{n/3} Q_i\right) = \sum_{i=1}^{n/3} d_{\mathrm{KL}}(P_i \parallel Q_i) > \Omega(\varepsilon)$. This contradicts the assumption that Q is ε-close to P with probability at least $2/3$. ■

Appendix B. Constant calculation for Bhattacharyya et al. (2023, Theorem 1.3)

In the original proof of the conditional mutual information tester in Bhattacharyya et al. (2023, Theorem 1.3), the constants were implicit. Here, we provide a reference proof of computing the constants. Our proof approach mirrors that of Bhattacharyya et al. (2023, Theorem 1.3) except that we are explicit in the constants involved. As the calculations are quite involved, we color coded some manipulation steps for easier equation matching and verification. We will also use the following two results (Lemma 24 and Corollary 25) from Bhattacharyya et al. (2023) without proof.

Lemma 24 (Lemma 4.2 of Bhattacharyya et al. (2023)) *For any $a \geq -b$ and $b \geq 0$,*

$$\frac{1}{3} \min\left(\frac{a^2}{b}, |a| \log\left(2 + \frac{|a|}{b}\right)\right) \leq f(a, b) \leq \min\left(\frac{a^2}{b}, |a| \log\left(2 + \frac{|a|}{b}\right)\right) . \tag{4}$$

Corollary 25 (Corollary 4.8 of Bhattacharyya et al. (2023)) *Let $\hat{P}$ be empirical distribution over $N \geq 1$ samples. If P is a product distribution with marginals P_x and P_y, then with probability $1-\delta$,*

$$I(\hat{X}; \hat{Y} | \hat{Z} = z) \leq \frac{1}{N} \cdot \log\left(\frac{(N+1)^{|\Sigma|^2}}{\delta}\right) . \tag{5}$$

Claim 26 (c.f. Claim 4.3 of Bhattacharyya et al. (2023)) *For any x, y, the following holds:*

1. $f(\Delta_{xy}, P_x P_y) \leq 1$.
2. $\min(P_x, P_y, |\Delta_{xy}|) \geq \frac{1}{2} \frac{f(\Delta_{xy}, P_x P_y)}{\log(6/f(\Delta_{xy}, P_x P_y))}$.

Proof As the first statement has no constants involved, we will focus only on the second statement.

By definition,

$$|\Delta_{xy}| = \max\{P_{xy} - P_x P_y, P_x P_y - P_{xy}\} \leq \max(P_{xy}, P_x P_y) \leq \max\{P_x, P_y\} \tag{6}$$

So, by (4), we have

$$f(\Delta_{xy}, P_x P_y) \leq |\Delta_{xy}| \cdot \log\left(2 + \frac{|\Delta_{xy}|}{P_x P_y}\right) \tag{7}$$

$$
\begin{aligned}
&\le P_x \cdot \log\left(2 + \frac{1}{P_x}\right) && \text{By (6)} \\
&\le P_x \cdot \log\left(\frac{3}{P_x}\right) \; . && (8)
\end{aligned}
$$

Taking log on both sides, we see that

$$
\begin{aligned}
& \log f(\Delta_{xy}, P_x P_y) && \le \log P_x + \log\log(3/P_x) \\
\iff \quad & \log 3 - \log f(\Delta_{xy}, P_x P_y) && \ge \log 3 - \log P_x - \log\log(3/P_x) \\
\iff \quad & \tfrac{1}{\log(3/f(\Delta_{xy}, P_x P_y))} && \le \frac{1}{\log(3/P_x) - \log\log(3/P_x)} \; .
\end{aligned}
$$

Multiplying the inequality of (8) in the numerator, we get

$$
\begin{aligned}
\frac{f(\Delta_{xy}, P_x P_y)}{\log(3/f(\Delta_{xy}, P_x P_y))} &\le \frac{P_x \log(3/P_x)}{\log(3/P_x) - \log\log(3/P_x)} \\
&= \frac{P_x}{1 - \frac{\log\log(3/P_x)}{\log(3/P_x)}} && \text{Divide by } \log(3/P_x) \text{ on top and bottom} \\
&\le \frac{P_x}{1 - \frac{1}{e}} && \text{Since } \frac{\log x}{x} \le \frac{1}{e} \text{ for } x > 0 \\
&\le \frac{P_x}{0.5} \\
&= 2 \cdot P_x
\end{aligned}
$$

By repeating the same argument above for P_y instead of P_x, we get

$$
\min(P_x, P_y) \ge \frac{1}{2} \frac{f(\Delta_{xy}, P_x P_y)}{\log(3/f(\Delta_{xy}, P_x P_y))} \; . \qquad (9)
$$

Let us abbreviate f as $f(\Delta_{xy}, P_x P_y)$ for the following calculations.

$$
\begin{aligned}
|\Delta_{xy}| &\ge \frac{f}{\log(3/P_y)} && \text{By (7) and (6)} \\
&\ge \frac{f}{\log\left(\frac{6\log(3/f)}{f}\right)} && \text{By (9)} \\
&= \frac{f}{\log 2 + \log\log(3/f) + \log(3/f)} \\
&\ge \frac{f}{2\log(2) + 2\log(3/f)} && \text{Since } x \ge \log x \\
&\ge \frac{1}{2} \cdot \frac{f}{\log(6/f)} && (10)
\end{aligned}
$$

Combining (9) and (10), we get $\min(P_x, P_y, |\Delta_{xy}|) \ge \frac{1}{2} \frac{f(\Delta_{xy}, P_x P_y)}{\log(6/f(\Delta_{xy}, P_x P_y))}$. ■

Claim 27 (c.f. Claim 4.4 of Bhattacharyya et al. (2023)) *Fix any $x \in \Sigma, y \in \Sigma$. Let $\hat{P}_x$, $\hat{P}_y$, and $\hat{P}_{xy}$ be empirical estimates over $N > 1$ samples. Then, each of the following bound holds with $1 - 3\delta$ probability:*

$$|\hat{P}_x - P_x| \leq 3 \left(\sqrt{\frac{P_x \log \frac{2}{\delta}}{N}} + \frac{\log \frac{2}{\delta}}{N} \right) \tag{11}$$

$$|\hat{P}_{xy} - P_{xy}| \leq 3 \left(\sqrt{\frac{P_{xy} \log \frac{2}{\delta}}{N}} + \frac{\log \frac{2}{\delta}}{N} \right) \tag{12}$$

$$|\hat{P}_x \hat{P}_y - P_x P_y| \leq 6 \sqrt{P_x P_y \frac{\log \frac{2}{\delta}}{N}} + 18(P_x + P_y) \frac{\log \frac{2}{\delta}}{N} + 18 \frac{\log^2 \frac{2}{\delta}}{N^2} \tag{13}$$

$$|\hat{\Delta}_{xy} - \Delta_{xy}| \leq 3 \sqrt{|\Delta_{xy}| \frac{\log \frac{2}{\delta}}{N}} + 9 \sqrt{P_x P_y \frac{\log \frac{2}{\delta}}{N}} + 39 \frac{\log \frac{2}{\delta}}{N} + 18 \frac{\log^2 \frac{2}{\delta}}{N^2} \tag{14}$$

Proof Calculating the constant of the multiplicative Chernoff bound, we can show that

$$\Pr\left[|\hat{P}_x - P_x| > \varepsilon P_x \right] < 2 \exp\left(-\frac{1}{3} \cdot \min\{\varepsilon, \varepsilon^2\} \cdot P_x \cdot N \right) . \tag{15}$$

Upper bounding the right hand side by δ gives

$$\min\{\varepsilon, \varepsilon^2\} \geq \frac{3 \log\left(\frac{2}{\delta}\right)}{P_x \cdot N} \iff \varepsilon \geq \frac{3 \log\left(\frac{2}{\delta}\right)}{P_x \cdot N} \quad \text{and} \quad \varepsilon^2 \geq \frac{3 \log\left(\frac{2}{\delta}\right)}{P_x \cdot N} .$$

So,

$$\Pr\left[|\hat{P}_x - P_x| > \max\left\{ \frac{3 \log \frac{2}{\delta}}{P_x N}, \sqrt{\frac{3 \log \frac{2}{\delta}}{P_x N}} \right\} \cdot P_x \right] \leq \delta .$$

Thus, with probability $1 - \delta$, we get (11):

$$|\hat{P}_x - P_x| \leq 3 \left(\sqrt{\frac{P_x \log \frac{2}{\delta}}{N}} + \frac{\log \frac{2}{\delta}}{N} \right) .$$

Repeating the same argument for P_{xy} instead of P_x yields (12).
As for (13), observe that triangle inequality gives us

$$|\hat{P}_x \hat{P}_y - P_x P_y| \leq |\hat{P}_x \hat{P}_y - \hat{P}_x P_y| + |\hat{P}_x P_y - P_x P_y| = \hat{P}_x \cdot |\hat{P}_y - P_y| + P_y \cdot |\hat{P}_x - P_x|$$

To simplify notation in the following, let us define $\alpha = \frac{\log \frac{2}{\delta}}{N}$. Then, applying (11) for the terms $\hat{P}_x$ and $\hat{P}_y$ gives

$$\hat{P}_x \cdot |\hat{P}_y - P_y| \leq 3 \left(P_x + 3 \left(\sqrt{\frac{P_x \log \frac{2}{\delta}}{N}} + \frac{\log \frac{2}{\delta}}{N} \right) \right) \left(\sqrt{\frac{P_y \log \frac{2}{\delta}}{N}} + \frac{\log \frac{2}{\delta}}{N} \right)$$

$$
\begin{aligned}
&= 3\left(P_x + 3\left(\sqrt{P_x \cdot \alpha} + \alpha\right)\right)\left(\sqrt{P_y \cdot \alpha} + \alpha\right) \\
&= 3P_x\left(\sqrt{P_y \cdot \alpha} + \alpha\right) + 9\left(\sqrt{P_x \cdot \alpha} + \alpha\right)\left(\sqrt{P_y \cdot \alpha} + \alpha\right) \\
&= 3P_x\sqrt{P_y \cdot \alpha} + 3P_x \cdot \alpha + 9\sqrt{P_x P_y} \cdot \alpha + 9(\sqrt{P_x} + \sqrt{P_y}) \cdot \alpha^{3/2} + 9\alpha^2 \\
&\leq 3P_x\sqrt{P_y \cdot \alpha} + 3P_x \cdot \alpha + 9\sqrt{P_x P_y} \cdot \alpha + 18(\sqrt{P_x + P_y}) \cdot \alpha^{3/2} + 9\alpha^2 \\
&= 3P_x\sqrt{P_y \cdot \alpha} + 3P_x \cdot \alpha + 9\sqrt{P_x P_y} \cdot \alpha + 18 \cdot \sqrt{(P_x + P_y) \cdot \alpha} \cdot \alpha + 9\alpha^2 \\
&\leq 3P_x\sqrt{P_y \cdot \alpha} + 3P_x \cdot \alpha + 9 \cdot \frac{P_x + P_y}{2} \cdot \alpha + 18 \cdot \frac{P_x + P_y + \alpha}{2} \cdot \alpha + 9\alpha^2 \\
&= 3P_x\sqrt{P_y \cdot \alpha} + 3P_x \cdot \alpha + \frac{27}{2} \cdot (P_x + P_y) \cdot \alpha + 18\alpha^2
\end{aligned}
$$

where the second last inequality is because $P_x, P_y \geq 0$ implies that $\max\{P_x, P_y\} \leq P_x + P_y$ while the last inequality is due to two applications of the AM-GM inequality. Similarly,

$$
P_y \cdot |\hat{P}_x - P_x| \leq 3P_y\left(\sqrt{\frac{P_x \log \frac{2}{\delta}}{N}} + \frac{\log \frac{2}{\delta}}{N}\right) = 3P_y\left(\sqrt{P_x \cdot \alpha} + \alpha\right) = 3P_y\sqrt{P_x \cdot \alpha} + 3P_y \cdot \alpha
$$

Combining the above chain of inequalities, we have

$$
\begin{aligned}
&|\hat{P}_x \hat{P}_y - P_x P_y| \\
&\leq \hat{P}_x \cdot |\hat{P}_y - P_y| + P_y \cdot |\hat{P}_x - P_x| \\
&\leq 3 \cdot (P_x\sqrt{P_y} + P_y\sqrt{P_x}) \cdot \sqrt{\alpha} + \left(3 + \frac{27}{2}\right) \cdot (P_x + P_y) \cdot \alpha + 18\alpha^2 \\
&= 3 \cdot (\sqrt{P_x} + \sqrt{P_y}) \cdot \sqrt{P_x P_y} \cdot \sqrt{\alpha} + \left(3 + \frac{27}{2}\right) \cdot (P_x + P_y) \cdot \alpha + 18\alpha^2 \\
&\leq 6 \cdot \sqrt{P_x P_y} \cdot \sqrt{\alpha} + 18 \cdot (P_x + P_y) \cdot \alpha + 18\alpha^2 && \text{Since } \sqrt{P_x}, \sqrt{P_y} \leq 1
\end{aligned}
$$

Replacing α back with $\frac{\log \frac{2}{\delta}}{N}$ yields (13).

Finally, for (14), we will apply the inequalities from (12) and (13) to get

$$
\begin{aligned}
&|\hat{\Delta}_{xy} - \Delta_{xy}| \\
&\leq |\hat{P}_{xy} - P_{xy}| + |\hat{P}_x \hat{P}_y - P_x P_y| \\
&\leq 3\left(\sqrt{\frac{P_{xy} \log \frac{2}{\delta}}{N}} + \frac{\log \frac{2}{\delta}}{N}\right) + 6\sqrt{P_x P_y}\sqrt{\frac{\log \frac{2}{\delta}}{N}} + 18 \cdot (P_x + P_y)\frac{\log \frac{2}{\delta}}{N} + 18\left(\frac{\log \frac{2}{\delta}}{N}\right)^2 \\
&\leq 3\sqrt{\frac{P_{xy} \log \frac{2}{\delta}}{N}} + 6\sqrt{P_x P_y}\sqrt{\frac{\log \frac{2}{\delta}}{N}} + 39\frac{\log \frac{2}{\delta}}{N} + 18\left(\frac{\log \frac{2}{\delta}}{N}\right)^2 \\
&= 3\sqrt{(\Delta_{xy} + P_x P_y)\frac{\log \frac{2}{\delta}}{N}} + 6\sqrt{P_x P_y \frac{\log \frac{2}{\delta}}{N}} + 39\frac{\log \frac{2}{\delta}}{N} + 18\frac{\log^2 \frac{2}{\delta}}{N^2} \\
&\leq 3\sqrt{(|\Delta_{xy}| + P_x P_y)\frac{\log \frac{2}{\delta}}{N}} + 6\sqrt{P_x P_y \frac{\log \frac{2}{\delta}}{N}} + 39\frac{\log \frac{2}{\delta}}{N} + 18\frac{\log^2 \frac{2}{\delta}}{N^2}
\end{aligned}
$$

$$
\begin{aligned}
&\leq 3\sqrt{|\Delta_{xy}|\frac{\log\frac{2}{\delta}}{N}} + 3\sqrt{P_xP_y\frac{\log\frac{2}{\delta}}{N}} + 6\sqrt{P_xP_y\frac{\log\frac{2}{\delta}}{N}} + 39\frac{\log\frac{2}{\delta}}{N} + 18\frac{\log^2\frac{2}{\delta}}{N^2} \\
&= 3\sqrt{|\Delta_{xy}|\frac{\log\frac{2}{\delta}}{N}} + 9\sqrt{P_xP_y\frac{\log\frac{2}{\delta}}{N}} + 39\frac{\log\frac{2}{\delta}}{N} + 18\frac{\log^2\frac{2}{\delta}}{N^2}
\end{aligned}
$$

■

Lemma 28 (c.f. Lemma 4.5 of Bhattacharyya et al. (2023)) *Fix any $x \in \Sigma, y \in \Sigma$, $\delta > 0$, and constant $C \geq 90000$. Let $\hat{P}$ be the empirical distribution over $N \geq 3 > e$ samples. If $f(\Delta_{xy}, P_xP_y) \geq C\frac{\log N}{N}\log\frac{2}{\delta}$, then*

$$
\Pr\left[f(\hat{\Delta}_{xy}, \hat{P}_x\hat{P}_y) > \frac{1}{50}f(\Delta_{xy}, P_xP_y)\right] > 1 - 3\delta\ .
$$

Proof In this proof, we have the following assumption:

$$
f(\Delta_{xy}, P_xP_y) \geq C\frac{\log N}{N}\log\frac{2}{\delta} \tag{16}
$$

By Claim 26, we have $f(\Delta_{xy}, P_xP_y) \leq 1$ and

$$
\min(P_x, P_y, |\Delta_{xy}|) \geq \frac{1}{2}\frac{f(\Delta_{xy}, P_xP_y)}{\log(6/f(\Delta_{xy}, P_xP_y))} \geq \frac{C}{4}\cdot\frac{1}{N}\log\frac{2}{\delta}\ , \tag{17}
$$

where the final inequality is due to (16). Meanwhile,

$$
\begin{aligned}
\frac{f(\Delta_{xy}, P_xP_y)}{\log(6/f(\Delta_{xy}, P_xP_y))} &= \frac{f(\Delta_{xy}, P_xP_y)}{\log 6 - \log f(\Delta_{xy}, P_xP_y)} \\
&\geq \frac{C\frac{\log N}{N}\log\frac{2}{\delta}}{\log(6/C) + \log N - \log\log N - \log\log\frac{2}{\delta}} && \text{By (16)} \\
&\geq \frac{C\frac{\log N}{N}\log\frac{2}{\delta}}{\log(6/C) + \log N} \\
&\geq \frac{C\frac{\log N}{N}\log\frac{2}{\delta}}{2\log N} && \text{Holds when } C \geq \frac{6}{N} \\
&= \frac{C}{2}\frac{\log\frac{2}{\delta}}{N}
\end{aligned}
$$

Recall that $\Delta_{xy} = P_{xy} - P_xP_y$. We will split the remaining analysis into two cases, depending on the size of $|\Delta_{xy}|$.

Case 1 (Big $|\Delta_{xy}|$): $|\Delta_{xy}| = \Delta_{xy} \geq 8P_xP_y$

Case 2 (Small $|\Delta_{xy}|$): $-P_xP_y \leq \Delta_{xy} \leq 8P_xP_y$

Case 1: $|\Delta_{xy}| = \Delta_{xy} \geq 8P_xP_y$

$$\begin{aligned}
&|\hat{\Delta}_{xy} - \Delta_{xy}| \\
&\leq 3\sqrt{|\Delta_{xy}|\frac{\log\frac{2}{\delta}}{N}} + 9\sqrt{P_xP_y\frac{\log\frac{2}{\delta}}{N}} + 39\frac{\log\frac{2}{\delta}}{N} + 18\frac{\log^2\frac{2}{\delta}}{N^2} && \text{By Claim 27, Eqn (14)} \\
&\leq 3\sqrt{|\Delta_{xy}|\frac{4}{C}|\Delta_{xy}|} + 9\sqrt{\frac{|\Delta_{xy}|}{8}\frac{4}{C}|\Delta_{xy}|} + 39\frac{4}{C}|\Delta_{xy}| + 18\left(\frac{4}{C}|\Delta_{xy}|\right)^2 && \text{By (17) with } |\Delta_{xy}| \\
&\leq \sqrt{\frac{36}{C}}|\Delta_{xy}| + \sqrt{\frac{81}{2C}}|\Delta_{xy}| + \left(\frac{160}{C} + \frac{300}{C^2}\right)|\Delta_{xy}| && \text{Since } |\Delta_{xy}|^2 \leq |\Delta_{xy}| \\
&\leq \frac{1}{10}|\Delta_{xy}| && \text{Since } C \geq 40000
\end{aligned}$$

There are two ways to bound $|\hat{P}_x\hat{P}_y - P_xP_y|$ after applying Claim 27 and (17) with $|\Delta_{xy}|$. We can either use the bound of Case 1 ($|\Delta_{xy}| = \Delta_{xy} \geq 8P_xP_y$), or the bound of (17) with $\min\{P_x, P_y\}$.

Using $|\Delta_{xy}| = \Delta_{xy} \geq 8P_xP_y$, we get

$$\begin{aligned}
&|\hat{P}_x\hat{P}_y - P_xP_y| \\
&\leq 6\sqrt{P_xP_y\frac{\log\frac{2}{\delta}}{N}} + 18(P_x + P_y)\frac{\log\frac{2}{\delta}}{N} + 18\frac{\log^2\frac{2}{\delta}}{N^2} && \text{By Claim 27, Eqn (13)} \\
&\leq 6\sqrt{P_xP_y\frac{4}{C}|\Delta_{xy}|} + 18\cdot\frac{4}{C}|\Delta_{xy}| + 18\cdot\left(\frac{4}{C}|\Delta_{xy}|\right)^2 && \text{By (17) with } |\Delta_{xy}| \\
&\leq 6\sqrt{\frac{1}{8}|\Delta_{xy}|\frac{4}{C}|\Delta_{xy}|} + 18\cdot\frac{4}{C}|\Delta_{xy}| + 18\cdot\left(\frac{4}{C}|\Delta_{xy}|\right)^2 && \text{Case 1} \\
&\leq \sqrt{\frac{18}{C}}|\Delta_{xy}| + \frac{72}{C}|\Delta_{xy}| + \frac{300}{C^2}|\Delta_{xy}| && \text{Since } |\Delta_{xy}|^2 \leq |\Delta_{xy}| \\
&\leq \frac{1}{10}|\Delta_{xy}| && \text{Since } C \geq 3600
\end{aligned}$$

Using the bound of (17) with $\min\{P_x, P_y\}$, we get

$$\begin{aligned}
&|\hat{P}_x\hat{P}_y - P_xP_y| \\
&\leq 6\sqrt{P_xP_y\frac{\log\frac{2}{\delta}}{N}} + 18(P_x + P_y)\frac{\log\frac{2}{\delta}}{N} + 18\frac{\log^2\frac{2}{\delta}}{N^2} && \text{By Claim 27, Eqn (13)} \\
&\leq 6\sqrt{P_xP_y\frac{4}{C}|\Delta_{xy}|} + 18(P_x + P_y)\frac{\log\frac{2}{\delta}}{N} + 18\frac{\log^2\frac{2}{\delta}}{N^2} && \text{By (17) with } |\Delta_{xy}| \\
&= 6\sqrt{P_xP_y\frac{4}{C}|\Delta_{xy}|} + 18\left(P_x\frac{\log\frac{2}{\delta}}{N} + P_y\frac{\log\frac{2}{\delta}}{N}\right) + 18\frac{\log^2\frac{2}{\delta}}{N^2} \\
&\leq 6\sqrt{P_xP_y\frac{4}{C}|\Delta_{xy}|} + \left(144\frac{1}{C} + 288\frac{1}{C^2}\right)P_xP_y && \text{By (17) with } \min\{P_x, P_y\}
\end{aligned}$$

$$= \frac{12}{\sqrt{C}}\sqrt{P_x P_y \Delta_{xy}} + \left(144\frac{1}{C} + 288\frac{1}{C^2}\right) P_x P_y$$

Recall that in Case 1, we have $|\Delta_{xy}| = \Delta_{xy} \geq 8P_x P_y$. Therefore, from the above, we get:

$$\hat{\Delta}_{xy} \geq \frac{9}{10}|\Delta_{xy}| = 0.9\Delta_{xy} \tag{18}$$

$$\hat{P}_x\hat{P}_y \leq P_x P_y + \frac{1}{10}|\Delta_{xy}| \leq 0.23\Delta_{xy} \tag{19}$$

$$\hat{P}_x\hat{P}_y \leq \frac{12}{\sqrt{C}}\sqrt{P_x P_y |\Delta_{xy}|} + \left(144\frac{1}{C} + 288\frac{1}{C^2} + 1\right) P_x P_y \tag{20}$$

The first two inequalities (18) and (19) together yield

$$\frac{\hat{\Delta}_{xy}}{\hat{P}_x\hat{P}_y} \geq \frac{0.9}{0.23}\frac{\Delta_{xy}}{\Delta_{xy}} \geq 3.9$$

Since $x \geq \log(2+x)$ for any $x \geq 2$, we see that

$$\min\left\{\frac{\hat{\Delta}^2_{xy}}{\hat{P}_x\hat{P}_y}, |\hat{\Delta}_{xy}|\log\left(2 + \frac{|\hat{\Delta}_{xy}|}{\hat{P}_x\hat{P}_y}\right)\right\} = |\hat{\Delta}_{xy}|\log\left(2 + \frac{|\hat{\Delta}_{xy}|}{\hat{P}_x\hat{P}_y}\right) \geq |\hat{\Delta}_{xy}|\log\frac{|\hat{\Delta}_{xy}|}{\hat{P}_x\hat{P}_y} \tag{21}$$

Putting everything together,

$$\begin{aligned}
& f(\hat{\Delta}_{xy}, \hat{P}_x\hat{P}_y) \\
&\geq \frac{1}{3}\min\left\{\frac{\hat{\Delta}^2_{xy}}{\hat{P}_x\hat{P}_y}, |\hat{\Delta}_{xy}|\log\left(2 + \frac{|\hat{\Delta}_{xy}|}{\hat{P}_x\hat{P}_y}\right)\right\} && \text{By Lemma 24} \\
&\geq \frac{1}{3}|\hat{\Delta}_{xy}|\log\frac{|\hat{\Delta}_{xy}|}{\hat{P}_x\hat{P}_y} && \text{By (21)} \\
&\geq \frac{3}{10}\Delta_{xy}\left(\log\frac{9}{10} + \log\left(\frac{\Delta_{xy}}{\hat{P}_x\hat{P}_y}\right)\right) && \text{Since } \hat{\Delta}_{xy} \geq 0.9\Delta_{xy} \\
&\geq \frac{3}{10}\Delta_{xy}\left(\log\frac{9}{10} + \log\left(\frac{\sqrt{\Delta_{xy}}}{\frac{12}{\sqrt{C}}\sqrt{P_x P_y} + \left(144\frac{1}{C} + 288\frac{1}{C^2} + 1\right)\frac{P_x P_y}{\sqrt{\Delta_{xy}}}}\right)\right) && \text{By (20)} \\
&\geq \frac{3}{10}\Delta_{xy}\left(\log\frac{9}{10} + \log\left(\frac{\sqrt{\Delta_{xy}}}{\frac{12}{\sqrt{C}}\sqrt{P_x P_y} + \left(144\frac{1}{C} + 288\frac{1}{C^2} + 1\right)\frac{\sqrt{P_x P_y}}{\sqrt{8}}}\right)\right) && \text{Case 1} \\
&\geq \frac{3}{10}\Delta_{xy}\left(\log\left(\frac{9}{10\sqrt{2}}\sqrt{\frac{\Delta_{xy}}{P_x P_y}}\right)\right) && \text{Since } C \geq 288 \\
&\geq \frac{3}{10}\Delta_{xy}\left(\log\left(\frac{9(8)^{1/4}}{10\sqrt{2}}\left(\frac{\Delta_{xy}}{P_x P_y}\right)^{1/4}\right)\right) && \text{Case 1} \\
&\geq \frac{3}{40}\Delta_{xy}\log\left(\frac{\Delta_{xy}}{P_x P_y}\right) && \text{Since } \frac{9(8)^{1/4}}{10\sqrt{2}} > 1.07\ldots
\end{aligned}$$

$$\begin{aligned}
&\geq \frac{3}{40} f(\Delta_{xy}, P_x P_y) && \text{By Lemma 24} \\
&\geq \frac{1}{50} f(\Delta_{xy}, P_x P_y)
\end{aligned}$$

Case 2: $-P_x P_y \leq \Delta_{xy} \leq 8 P_x P_y$

In this case,

$$\begin{aligned}
C \frac{\log N}{N} \log \frac{2}{\delta} &\leq f(\Delta_{xy}, P_x P_y) && \text{By (16)} \\
&\leq \min \left\{ \frac{\Delta_{xy}^2}{P_x P_y}, |\Delta_{xy}| \log \left(2 + \frac{|\Delta_{xy}|}{P_x P_y} \right) \right\} && \text{By Lemma 24} \\
&\leq \frac{\Delta_{xy}^2}{P_x P_y} \\
&\leq 64 P_x P_y && \text{Case 2} \\
&\leq 64 \min\{P_x, P_y\} && \text{Since } P_x, P_y \leq 1
\end{aligned}$$

There are two observations we can make here. Firstly, we have $C \frac{\log N}{N} \log \frac{2}{\delta} \leq \frac{\Delta_{xy}^2}{P_x P_y}$, and so

$$\sqrt{C P_x P_y \frac{\log N}{N} \log \frac{2}{\delta}} \leq |\Delta_{xy}| \tag{22}$$

Secondly, since $N \geq 3 > e$, we see that

$$C \frac{\log N}{N} \log \frac{2}{\delta} \leq 64 \min\{P_x, P_y\} \iff \frac{\log \frac{2}{\delta}}{N} \leq \frac{64}{C} \frac{\min\{P_x, P_y\}}{\log N}$$

That is,

$$\frac{\log \frac{2}{\delta}}{N} \leq \frac{64}{C} \min\{P_x, P_y\} \tag{23}$$

Combining (23) with (11) gives the following:

$$|\hat{P}_x - P_x| \leq 3 \left(\sqrt{\frac{P_x \log \frac{2}{\delta}}{N}} + \frac{\log \frac{2}{\delta}}{N} \right) \leq 3 \left(\frac{8}{\sqrt{C}} + \frac{64}{C} \right) P_x \leq \frac{P_x}{10}$$

Likewise, we have

$$|\hat{P}_y - P_y| \leq \frac{P_y}{10}$$

$$\begin{aligned}
\Big| |\Delta_{xy}| - |\hat{\Delta}_{xy}| \Big| &\leq |\hat{\Delta}_{xy} - \Delta_{xy}| && \text{Reverse triangle inequality} \\
&\leq 3 \sqrt{|\Delta_{xy}| \frac{\log \frac{2}{\delta}}{N}} + 9 \sqrt{P_x P_y \frac{\log \frac{2}{\delta}}{N}} + 39 \frac{\log \frac{2}{\delta}}{N} + 18 \frac{\log^2 \frac{2}{\delta}}{N^2} && \text{By (14)}
\end{aligned}$$

$$\begin{aligned}
&\leq 3\sqrt{|\Delta_{xy}|\frac{\log\frac{2}{\delta}}{N}} + 9\sqrt{P_x P_y \frac{\log\frac{2}{\delta}}{N}} + 60\frac{\log\frac{2}{\delta}}{N} && \text{Since } \frac{\log\frac{2}{\delta}}{N} \leq 1\\
&\leq 3\sqrt{|\Delta_{xy}|\frac{\log\frac{2}{\delta}}{N}} + 9\sqrt{P_x P_y \log N \frac{\log\frac{2}{\delta}}{N}} + 60\frac{\log\frac{2}{\delta}}{N} && \text{Since } N > e\\
&\leq 3\sqrt{|\Delta_{xy}|\frac{\log\frac{2}{\delta}}{N}} + 9\frac{|\Delta_{xy}|}{\sqrt{C}} + 60\frac{\log\frac{2}{\delta}}{N} && \text{By (22)}\\
&\leq 3\sqrt{\frac{8}{C}|\Delta_{xy}|} + 9\frac{|\Delta_{xy}|}{\sqrt{C}} + 60\frac{4|\Delta_{xy}|}{C} && \text{By (17) with } |\Delta_{xy}|\\
&\leq \frac{1}{10}|\Delta_{xy}| && \text{Since } C \geq 40000
\end{aligned}$$

The above bounds give us the following:

$$\frac{9}{10}P_x \leq \hat{P}_x \leq \frac{11}{10}P_x \quad \text{and} \quad \frac{9}{10}P_y \leq \hat{P}_y \leq \frac{11}{10}P_y \quad \text{and} \quad \frac{9}{10}|\Delta_{xy}| \leq |\hat{\Delta}_{xy}| \leq \frac{11}{10}|\Delta_{xy}| \tag{24}$$

So,

$$0 \leq \frac{|\hat{\Delta}_{xy}|}{\hat{P}_x \hat{P}_y} \leq \frac{11/10}{(9/10)^2}\frac{|\Delta_{xy}|}{P_x P_y} \leq \frac{110}{81} \cdot 8\,.$$

Furthermore[11], $\frac{81}{880}x < \log(2+x)$ for $0 < x < \frac{110}{81} \cdot 8$. Thus,

$$\frac{81}{880}\frac{\hat{\Delta}_{xy}^2}{\hat{P}_x \hat{P}_y} < |\hat{\Delta}_{xy}| \log\left(2 + \frac{|\hat{\Delta}_{xy}|}{\hat{P}_x \hat{P}_y}\right) \tag{25}$$

Therefore,

$$\begin{aligned}
&f(\hat{\Delta}_{xy}, \hat{P}_x \hat{P}_y)\\
&\geq \frac{1}{3}\min\left\{\frac{\hat{\Delta}_{xy}^2}{\hat{P}_x \hat{P}_y}, |\hat{\Delta}_{xy}| \log\left(2 + \frac{|\hat{\Delta}_{xy}|}{\hat{P}_x \hat{P}_y}\right)\right\} && \text{By Lemma 24}\\
&\geq \frac{1}{3} \cdot \frac{81}{880} \cdot |\hat{\Delta}_{xy}|^2 \cdot \frac{1}{\hat{P}_x} \cdot \frac{1}{\hat{P}_y} && \text{By (25)}\\
&\geq \frac{27}{880}\left(\frac{9}{10}|\Delta_{xy}|\right)^2 \cdot \frac{10}{11}\frac{1}{P_x} \cdot \frac{10}{11}\frac{1}{P_y} && \text{By (24)}\\
&\geq \frac{1}{50}\frac{|\Delta_{xy}|^2}{P_x P_y}\\
&\geq \frac{1}{50} f(\Delta_{xy}, P_x P_y) && \text{By Lemma 24}
\end{aligned}$$

The claim holds since we showed $f(\hat{\Delta}_{xy}, \hat{P}_x \hat{P}_y) \geq \frac{1}{50} f(\Delta_{xy}, P_x P_y)$ in both cases. ■

[11] e.g. see https://www.wolframalpha.com/input?i=solve+log%282%2Bx%29%3E81%2F880x

Corollary 29 (c.f. Corollary 4.6 of Bhattacharyya et al. (2023)) *Let $\hat{P}$ be the empirical distribution over $N > 1$ samples. Then there exist universal constants $C_1 = \frac{1}{50}$ and $C_2 = 1800$ such that for every $\delta > 0$:*

$$I(\hat{X};\hat{Y}) \geq C_1 \cdot I(X;Y) - C_2 \cdot |\Sigma|^2 \frac{\log N}{N} \log\left(\frac{6|\Sigma|^2}{\delta}\right)$$

with probability at least $1-\delta$.

Proof Since $f \geq 0$ on any input, whenever $f(\Delta_{xy}, P_x P_y) < 90000 \cdot \frac{\log N}{N} \log\left(\frac{6|\Sigma|^2}{\delta}\right)$, we have

$$f(\hat{\Delta}_{xy}, \hat{P}_x\hat{P}_y) \geq 0 > \frac{1}{50}\left(f(\Delta_{xy}, P_xP_y) - 90000 \cdot \frac{\log N}{N} \log\left(\frac{6|\Sigma|^2}{\delta}\right)\right)$$

Meanwhile, by Lemma 28, $f(\Delta_{xy}, P_xP_y) \geq 90000 \cdot \frac{\log N}{N} \log\left(\frac{6|\Sigma|^2}{\delta}\right)$ with probability $1 - \frac{\delta}{|\Sigma|^2}$.

$$f(\hat{\Delta}_{xy}, \hat{P}_x\hat{P}_y) \geq \frac{1}{50} f(\Delta_{xy}, P_xP_y) \ .$$

Summing up over the $|\Sigma|^2$ possible values of x and y, and then taking a union bound over these $|\Sigma|^2$ events, we have

$$I(\hat{X};\hat{Y}) \geq \frac{1}{50}\left(I(X;Y) - 90000 \cdot \frac{\log N}{N} |\Sigma|^2 \log\left(\frac{6|\Sigma|^2}{\delta}\right)\right) \ .$$

That is, we have $C_1 = \frac{1}{50}$ and $C_2 = 90000 \cdot \frac{1}{50} = 1800$. ■

Theorem 30 (c.f. Theorem 1.3 of Bhattacharyya et al. (2023)) *Let (X, Y, Z) be three random variables over Σ, and $(\hat{X}, \hat{Y}, \hat{Z})$ be the empirical distribution over a size N sample of (X, Y, Z). For any*

$$N \geq 6.48 \times 10^6 \cdot \frac{|\Sigma|^3 \left(\log\left(\frac{|\Sigma|}{\varepsilon\delta}\right) + \log(7.2 \times 10^5)\right) \cdot \log\left(\frac{12|\Sigma|^2}{\delta}\right)}{\varepsilon} ,$$

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

1. *If $I(X;Y|Z) = 0$, then $I(\hat{X};\hat{Y}|\hat{Z}) \leq \frac{1}{400} \cdot \varepsilon$*
2. *If $I(X;Y|Z) \geq \varepsilon$, then $I(\hat{X};\hat{Y}|\hat{Z}) \geq \frac{1}{400} \cdot I(X;Y|Z)$*

Proof Recall that $|\Sigma| \geq 2$ and $\varepsilon, \delta \in (0, 1]$. Let $N = 9\frac{|\Sigma|^3 \log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right) \cdot \log\left(\frac{12|\Sigma|^3}{\delta}\right)}{C\varepsilon}$ where $C^{-1} = 720000$. In particular, $N \geq 9\frac{|\Sigma|^3 \log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)}{C\varepsilon}$, $C < 1/6 < 1$, and $\frac{\log(N+1)}{N}$ monotonically decreases.

Proving the first statement

We will prove a stronger conclusion: $I(\hat{X};\hat{Y}|\hat{Z}) \leq C \cdot \varepsilon$. Since $C \leq \frac{1}{400}\cdot$, we get $I(\hat{X};\hat{Y}|\hat{Z}) \leq \frac{1}{400} \cdot \varepsilon$ as desired.

If $I(X;Y|Z) = 0$ then $I(X;Y|Z = z) = 0$ for any z. Let $N_z \geq 1$ be the count of $\hat{Z} = z$ and $\hat{P}_z = N_z/N$. Then, Corollary 25 tells us that, with probability at least $1 - \delta'$,

$$I(\hat{X};\hat{Y}|\hat{Z} = z) \leq \frac{1}{N_z} \cdot \log\left(\frac{(N_z+1)^{|\Sigma|^2}}{\delta'}\right) \leq \frac{|\Sigma|^2}{N_z} \cdot \log\left(\frac{N+1}{\delta'}\right) . \tag{26}$$

Using (26) and a union bound over $|\Sigma|$ values of z, the following holds with probability at least $1 - |\Sigma|\delta'$:[12]

$$I(\hat{X};\hat{Y}|\hat{Z}) = \sum_z \hat{P}_z \cdot I(\hat{X};\hat{Y}|\hat{Z} = z) \leq \sum_z \frac{N_z}{N} \cdot \frac{|\Sigma|^2}{N_z} \cdot \log\left(\frac{N+1}{\delta'}\right) = \frac{|\Sigma|^3}{N} \cdot \log\left(\frac{N+1}{\delta'}\right) .$$

Rescaling $\delta' = \frac{\delta}{|\Sigma|}$, we get $I(\hat{X};\hat{Y}|\hat{Z}) \leq \frac{|\Sigma|^3}{N} \log\left(\frac{(N+1)|\Sigma|}{\delta}\right)$. For $I(\hat{X};\hat{Y}|\hat{Z})$ to be at most $C \cdot \varepsilon$, it suffices to argue that

$$\frac{|\Sigma|^3}{N} \log\left(\frac{(N+1)|\Sigma|}{\delta}\right) \leq C\varepsilon .$$

To see this, consider

$$\begin{aligned}
&\frac{|\Sigma|^3}{N} \log\left(\frac{(N+1)|\Sigma|}{\delta}\right) \\
&= \frac{|\Sigma|^3}{N}\left(\log(N+1) + \log\left(\frac{|\Sigma|}{\delta}\right)\right) \\
&\leq \frac{C\varepsilon}{9\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)}\left(\log\left(9\frac{|\Sigma|^3\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)}{C\varepsilon}+1\right) + \log\left(\frac{|\Sigma|}{\delta}\right)\right) && \text{Since } N \geq 9\frac{|\Sigma|^3\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)}{C\varepsilon} \\
&= \frac{C\varepsilon\log\left(9\frac{|\Sigma|^3\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)}{C\varepsilon}+1\right)}{9\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)} + \frac{C\varepsilon}{9}\frac{\log\left(\frac{|\Sigma|}{\delta}\right)}{\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)} \\
&\leq \frac{C\varepsilon\log\left(10\frac{|\Sigma|^3\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)}{C\varepsilon}\right)}{9\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)} + \frac{C\varepsilon}{9}\cdot 1 && \text{Since } |\Sigma| \geq 2,\ C < 1, \text{ and } \varepsilon,\delta \in (0,1] \\
&= \frac{C\varepsilon\log 10}{9\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)} + \frac{C\varepsilon\log\left(\frac{|\Sigma|^3}{C\varepsilon}\right)}{9\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)} + \frac{C\varepsilon\log\left(\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)\right)}{9\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)} + \frac{C\varepsilon}{9} \\
&\leq \frac{C\varepsilon\log 10}{9\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)} + \frac{C\varepsilon\log\left(\frac{|\Sigma|^3}{C\varepsilon}\right)}{9\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)} + \frac{C\varepsilon}{9}\cdot\frac{1}{e} + \frac{C\varepsilon}{9} && \text{Since } \frac{\log\log x}{\log x} \leq \frac{1}{e}
\end{aligned}$$

[12] By definition, the empirical estimator $I(\hat{X};\hat{Y}) = \mathrm{KL}(\hat{P}_{xy}\|\hat{P}_x\hat{P}_y) < \infty$, as $\hat{P}_x = 0$ implies $\hat{P}_{xy} = 0$. So, if $\hat{P}_z = N_z = 0$, then we have $\hat{P}_z I(\hat{X};\hat{Y}|\hat{Z} = z) = 0$.

$$
\begin{aligned}
&= C\varepsilon \frac{\log 10}{\frac{9}{3.4}\log(|\Sigma|^{3.4})} + C\varepsilon \frac{\log\left(\frac{|\Sigma|^3}{C\varepsilon}\right)}{3\log\left(\left(\frac{|\Sigma|}{C\varepsilon}\right)^3\right)} + \frac{C\varepsilon}{9}\cdot\frac{1}{e} + \frac{C\varepsilon}{9} \\
&\leq C\varepsilon \cdot \left(\frac{3.4}{9} + \frac{1}{3} + \frac{1}{9e} + \frac{1}{9}\right) && \text{Since } |\Sigma| \geq 2,\ C < 1, \text{ and } \varepsilon, \delta \in (0,1] \\
&\leq C\varepsilon
\end{aligned}
$$

Therefore, with probability at least $1-\delta$, $I(X;Y|Z) = 0$ implies that $I(\hat{X};\hat{Y}|\hat{Z}) \leq C \cdot \varepsilon$.

Proving the second statement

Consider the set S of values $z \in \Sigma$ which satisfy $P_z \cdot I(X;Y|Z=z) \geq \frac{I(X;Y|Z)}{2|\Sigma|} \geq \frac{\varepsilon}{2|\Sigma|}$. Since $I(X;Y|Z=z) \leq \log(|\Sigma|)$, this implies that $P_z \geq \frac{\varepsilon}{2|\Sigma|\log|\Sigma|}$ for $z \in S$.

We also have

$$
\begin{aligned}
&\sum_{z\in S} P_z \cdot I(X;Y|Z=z) \\
&= \sum_{z\in\Sigma} P_z \cdot I(X;Y|Z=z) - \sum_{z\notin S} P_z \cdot I(X;Y|Z=z) \\
&\geq I(X;Y|Z) - |\Sigma| \cdot \frac{I(X;Y|Z)}{2|\Sigma|} && z \notin S \implies P_z \cdot I(X;Y|Z=z) < \frac{\varepsilon}{2|\Sigma|} \\
&= \frac{I(X;Y|Z)}{2} && (27)
\end{aligned}
$$

Taking a union bound over $|S| \leq |\Sigma|$ values of z, Corollary 29 states that the following holds for all $z \in S$, with probability $1 - |\Sigma|\delta'$, $C_1 = \frac{1}{50}$, and $C_2 = 1800$:

$$
I(\hat{X};\hat{Y}|\hat{Z}=z) \geq C_1 \cdot I(X;Y|Z=z) - C_2 \cdot |\Sigma|^2 \frac{\log(N_z)}{N_z} \log\left(\frac{6|\Sigma|^2}{\delta'}\right) . \tag{28}
$$

Calculating the constant of the multiplicative Chernoff bound, we can show that

$$
\Pr\left[|\hat{P}_x - P_x| > \varepsilon P_x\right] < 2\exp\left(-\frac{1}{3}\cdot\min\{\varepsilon,\varepsilon^2\}\cdot P_x \cdot N\right) . \tag{29}
$$

Plugging in $\varepsilon = 1/2$ in (29), we get $\Pr[|\hat{P}_z - P_z| > P_z/2] \leq 2\exp\left(-\frac{1}{3}\cdot\frac{1}{4}\cdot P_z \cdot N\right)$. Now, since $N \geq 24\frac{|\Sigma|\log|\Sigma|\log\frac{2}{\delta'}}{\varepsilon}$ and $P_z \geq \frac{\varepsilon}{2|\Sigma|\log|\Sigma|}$ for any $z \in S$, we see that $P_z \geq 12\frac{\log(2/\delta')}{N}$ and so the following holds for any $z \in S$:

$$
\Pr[|\hat{P}_z - P_z| > P_z/2] \leq 2\exp(-\log(2/\delta')) = \delta' .
$$

Recall that $N_z = N\hat{P}_z$ is the empirical count of $\hat{Z} = z$. So, by a union bound over $|S| \leq |\Sigma|$ values of z, we have that

$$
N_z = N\hat{P}_z \geq NP_z/2 \tag{30}
$$

for all $z \in S$, with probability $1 - |\Sigma|\delta'$.

Notice that we performed union bound twice, once in (28) and once in (30). Combining these expressions, we get that at least with probability $1 - 2|\Sigma|\delta'$,

$$I(\hat{X};\hat{Y}|\hat{Z}=z) \geq C_1 \cdot I(X;Y|Z=z) - C_2 \cdot |\Sigma|^2 \frac{\log(NP_z/2)}{NP_z/2} \log\left(\frac{6|\Sigma|^2}{\delta'}\right) . \tag{31}$$

Set $\delta = 2|\Sigma|\delta'$ so that everything holds with probability $1-\delta$. Multiplying $\hat{P}_z$ and summing over $z \in S$ gives:

$$\begin{aligned}
&I(\hat{X};\hat{Y}|\hat{Z}) \\
&\geq \sum_{z \in S} \hat{P}_z \cdot I(\hat{X};\hat{Y}|\hat{Z}=z) && \text{Since } S \subseteq \Sigma \\
&\geq \sum_{z \in S} \frac{P_z}{2} \left(C_1 \cdot I(X;Y|Z=z) - C_2 \cdot |\Sigma|^2 \frac{\log(NP_z/2)}{NP_z/2} \log\left(\frac{6|\Sigma|^2}{\delta'}\right)\right) && \text{By (30) and (31)} \\
&= \sum_{z \in S} \frac{1}{2} \left(C_1 \cdot P_z \cdot I(X;Y|Z=z) - 2 \cdot C_2 \cdot |\Sigma|^2 \frac{\log(NP_z/2)}{N} \log\left(\frac{6|\Sigma|^2}{\delta'}\right)\right) \\
&\geq \frac{C_1}{4} \cdot I(X;Y|Z) - \sum_{z \in S} C_2 \cdot |\Sigma|^2 \frac{\log(N/2)}{N} \log\left(\frac{6|\Sigma|^2}{\delta'}\right) && \text{By (27) and } P_z \leq 1 \\
&\geq \frac{C_1}{4} \cdot I(X;Y|Z) - C_2 \cdot |\Sigma|^3 \frac{\log(N/2)}{N} \log\left(\frac{6|\Sigma|^2}{\delta'}\right) && \text{Since } S \subseteq \Sigma \\
&= \frac{C_1}{4} \cdot I(X;Y|Z) - C_2 \cdot |\Sigma|^3 \frac{\log(N/2)}{N} \log\left(\frac{12|\Sigma|^3}{\delta}\right) && \text{Since } \delta = 2|\Sigma|\delta'
\end{aligned}$$

Recall that $N = 9\frac{|\Sigma|^3 \log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right) \cdot \log\left(\frac{12|\Sigma|^3}{\delta}\right)}{C\varepsilon}$ and $C < 1/6$. Continuing from above,

$$\begin{aligned}
&I(\hat{X};\hat{Y}|\hat{Z}) \\
&\geq \frac{C_1}{4} \cdot I(X;Y|Z) - C_2 \cdot |\Sigma|^3 \frac{\log(N/2)}{N} \log\left(\frac{12|\Sigma|^3}{\delta}\right) && \text{From above} \\
&= \frac{C_1}{4} \cdot I(X;Y|Z) - C_2 \cdot C\varepsilon \frac{\log\left(\frac{9}{2}\frac{|\Sigma|^3}{C\varepsilon} \log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right) \log\left(\frac{12|\Sigma|^3}{\delta}\right)\right)}{9 \log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)} && \text{Definition of } N \\
&= \frac{C_1}{4} \cdot I(X;Y|Z) \\
&\qquad - C_2 \cdot C\varepsilon \left(\frac{\log\left(\frac{9}{2}\frac{|\Sigma|^3}{C\varepsilon}\right)}{3\log\left(\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)^3\right)} + \frac{\log\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)}{9\log\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)} + \frac{\log\log\left(\frac{12|\Sigma|^3}{\delta}\right)}{3\log\left(\left(\frac{|\Sigma|}{C\varepsilon\delta}\right)^3\right)} \right) \\
&\geq \frac{C_1}{4} \cdot I(X;Y|Z) - C_2 \cdot C\varepsilon \left(\frac{1}{3} + \frac{1}{9} + \frac{1}{3}\right) && (\dagger) \\
&\geq \frac{C_1}{4} \cdot I(X;Y|Z) - C_2 \cdot C\varepsilon
\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{4} \cdot \frac{1}{50} \cdot I(X;Y|Z) - 1800C\varepsilon && \text{Since } C_1 = \frac{1}{50}, C_2 = 1800 \\
&\geq \frac{1}{200}\varepsilon - 1800C\varepsilon && \text{Since } I(X;Y|Z) \geq \varepsilon \\
&= \frac{1}{400}\varepsilon && \text{Since } C^{-1} = 720000
\end{aligned}
$$

where (†) is because $|\Sigma| \geq 2$, $C^{-1} = 720000$, and $\varepsilon, \delta \in (0,1]$.

Therefore, with probability at least $1-\delta$, $I(X;Y|Z) \geq \varepsilon$ implies that $I(\hat{X};\hat{Y}|\hat{Z}) \geq \frac{1}{400}\varepsilon$. ■