Statistical Guarantees for Probabilistic Circuit Parameter Learning

John Leland[1] **YooJung Choi**[1]

[1]School of Computing and Augmented Intelligence, Arizona State University, Tempe, Arizona, USA

Abstract

The expressive efficiency of PC classes has been studied extensively in the exact case and, although to a lesser degree, the approximate case. However, these results are mainly concerned with the theoretical ability of circuits to compactly represent functions. Statistical guarantees, such as sample complexity, for learning PCs remain largely unexplored. In this paper, we first prove a lower bound, linear in the size of the PC, on the number of samples needed for *any learning algorithm* to guarantee bounded statistical distance (Hellinger, reverse Kullback-Leibler, or total variation) with high probability. Next, for learning the parameters of decomposable PCs under known structure, we also show a matching upper bound on the number of samples in the case of the Hellinger distance and reverse KL divergence. Lastly, in the agnostic setting without the known-structure assumption, we provide an instance-specific statistical upper bound on the total variation distance using the Good-Turing denoising estimator, which approaches the irreducible distance due to PC structure as the number of samples increases.

1 INTRODUCTION

Probabilistic circuits (PCs) [Choi et al., 2020], a unifying framework for many families of tractable probabilistic models, can compactly represent probability distributions and support exact and efficient inference of various queries based on the structural properties they satisfy. The computational complexity of inference depends on the size of the circuit, and thus the expressive efficiency of PCs—the size needed to represent target functions—has been studied extensively for both exact representation and, albeit to a lesser degree, approximation [Darwiche, 2003, Bova et al., 2016, Choi et al., 2020, Leland and Choi, 2026, Martens and Medabalimi, 2014]. However, they do not address the setting of learning PCs, which is more common in practice.

Instead, we ask: how does the number of samples needed to learn a PC relate to its size? While many have studied PAC density estimation and sample complexity for various probabilistic models [Dasgupta, 1997, Friedman and Yakhini, 1996, Ashtiani et al., 2020], the only work on the sample complexity for PCs is an upper bound for tree-shaped sum-product networks [Aden-Ali and Ashtiani, 2020]. In this work, we present novel PAC-style bounds for more general classes of PCs over Boolean variables. First, we prove a lower bound on the sample complexity for *any learning algorithm* when the true distribution is a PC of size S (Theorem 1). Next, we show that the parameters of decomposable PCs are efficiently PAC learnable with a tight sample complexity bound of $\Theta(\frac{S+\log(1/\delta)}{\epsilon})$ for KL divergence, and $\Theta(\frac{S+\log(1/\delta)}{\epsilon^2})$ for the Hellinger distance when the true distribution is a decomposable PC with known structure of size S (Theorem 2). We also derive looser lower and upper bounds of $\Omega(\frac{S+\log(1/\delta)}{\epsilon})$ and $O(\frac{S+\log(1/\delta)}{\epsilon^2})$ for the total variation distance. While the realizable setting allows us to prove tight bounds, the assumption does not generally hold when learning PCs from data. Thus, we also consider the agnostic setting and derive an instance-specific bound on the distance between the true data-generating distribution (of unknown structure) and a learned PC, using the Good-Turing estimator (Theorem 3).

2 PRELIMINARIES

Sample Complexity For a given distance measure D, we wish to characterize the necessary and sufficient number of samples from a distribution P to learn it within distance $\epsilon > 0$, with error probability $\delta \in (0, 1]$, i.e., to ensure $\Pr[D(\hat{P}||P) \leq \epsilon] \geq 1 - \delta$. Informally an upper bound on the sample complexity states that *there exists* an algorithm that, given at least M samples, learns $\hat{P}$ such that

$\Pr[D(\hat{P}||P) \leq \epsilon] > 1-\delta$; on the other hand, a lower bound states that *any algorithm* needs at least M samples for the same guarantee. For the distance measure D, we consider the total variation distance (TVD), reverse Kullback-Leibler (KL) divergence, and (squared) Hellinger distance, defined between discrete distributions as follows, respectively:

$$D_{\mathsf{TV}}(\hat{P}\|P) = \frac{1}{2}\sum_{\mathbf{x}} \left|\hat{P}(\mathbf{x}) - P(\mathbf{x})\right|,$$

$$D_{\mathsf{KL}}(\hat{P}\|P) = \sum_{\mathbf{x}} \hat{P}(\mathbf{x}) \log\left(\frac{\hat{P}(\mathbf{x})}{P(\mathbf{x})}\right),$$

$$D_{\mathsf{H}}(\hat{P}\|P)^2 = 1 - \sum_{\mathbf{x}} \sqrt{\hat{P}(\mathbf{x})P(\mathbf{x})}.$$

We also utilize the following inequalities between them.

Lemma 1 (Divergence Inequalities [Canonne, 2020b])**.** *The following hold: (1)* $D_{\mathsf{TV}}(\hat{P}\|P) \leq \sqrt{\frac{1}{2}D_{\mathsf{KL}}(\hat{P}\|P)}$*; (2)* $D_{\mathsf{H}}(\hat{P}\|P)^2 < D_{\mathsf{TV}}(\hat{P}\|P)$*; (3)* $D_{\mathsf{H}}(\hat{P}\|P)^2 < \frac{1}{2}D_{\mathsf{KL}}(\hat{P}\|P)$.

Probabilistic Circuits A probabilistic circuit (PC) recursively encodes a joint probability distribution over random variables through a parameterized directed acyclic graph (DAG), composed of leaf, product, and sum nodes; see Appendix A for detailed background. PCs guarantee tractable inference by enforcing structural conditions such as *decomposability* and *determinism*. As is common in the literature, we assume decomposable PCs, which are tractable for marginal and conditional inference and sampling.

Additionally enforcing determinism enables closed-form maximum-likelihood parameter estimation [Kisa et al., 2014, Liu and Van den Broeck, 2021]. Intuitively, it can be interpreted as learning many categorical distributions, one for each sum node in the circuit with as many categories as there are children. This intuition can be formalized using the following notion: the *context* γ_n of a PC node n is defined recursively in a top-down manner as $\gamma_n = \bigcup_{m \in \mathsf{pa}(n)} \gamma_m \cap \mathsf{supp}(n)$, where the context of the root node is simply its support. Given a sample x, the value of the categorical variable corresponding to a sum node n is given only if $x \in \gamma_n$, with the specific value being the child index i such that $x \in \gamma_{n_i}$ as well; this can be evaluated for all sum nodes by propagating the sample through the circuit.

3 LOWER BOUNDS FOR LEARNING PCS

This section investigates the following question: *how many samples are necessary to ensure that we can learn a PC within a bounded distance to the true distribution with high probability?* While a natural question in learning theory, this has not been answered explicitly for any hypothesis class of PCs. In particular, we consider the classes of deterministic PCs of fixed structure and prove that the required sample complexity is polynomial in the number of circuit parameters S, accuracy $1/\epsilon$, and confidence $1/\delta$.

Theorem 1 (Lower Bound)**.** *Let $\mathcal{C}_S$ be a family of distributions represented by PCs with S parameters. For $\epsilon > 0$, $\delta \in (0, 1]$, any learning algorithm A that takes in M samples from $P \in \mathcal{C}_S$ and outputs $\hat{P}$ such that $D_{\mathsf{H}}(\hat{P}\|P) \leq \epsilon$ with probability $> 1 - \delta$ requires*

$$M \geq \max\left(\frac{S\log(2)}{608\epsilon^2}, \frac{\log(2/\delta)}{5\epsilon^2}\right) = \Omega\left(\frac{S + \log(1/\delta)}{\epsilon^2}\right),$$

$D_{\mathsf{KL}}(\hat{P}\|P) \leq \epsilon$ with prob. $1 - \delta$ requires

$$M \geq \max\left(\frac{S\log(2)}{304\epsilon}, \frac{2\log(2/\delta)}{5\epsilon}\right) = \Omega\left(\frac{S + \log(1/\delta)}{\epsilon}\right),$$

and $D_{\mathsf{TV}}(\hat{P}\|P) \leq \epsilon$ with prob. $1 - \delta$ requires,

$$M \geq \max\left(\frac{S\log(2)}{608\epsilon}, \frac{\log(2/\delta)}{5\epsilon^2}\right) = \Omega\left(\frac{S + \log(1/\delta)}{\epsilon}\right),$$

even if P is in $C_{\mathcal{G}}$, a family of deterministic PCs of a fixed structure $\mathcal{G}$ with S parameters, and $\mathcal{G}$ is known to A.

We emphasize that our result *does not restrict* what the learner A can output and that it applies to *any* family of models that subsumes, or can compactly represent, deterministic PCs of fixed structure. For instance, this bound immediately applies to sum-product networks [Poon and Domingos, 2011], arithmetic circuits [Darwiche, 2003], sum-of-squares circuits [Loconte et al., 2025], InceptionPCs [Wang and Van den Broeck, 2025], and probabilistic generating circuits [Zhang et al., 2021]. All complete proofs can be found in Appendix C. Here, we provide a high level sketch of our proof, which relies on the following lemma.

Lemma 2 (Assouad's Lemma [Assouad, 1983, Canonne, 2020b])**.** *Let $\mathcal{C}$ be a family of probability distributions and A be the set of learning algorithms that take M samples and output a distribution $\hat{P}$. Suppose there exists a family of $\{P_v\}_{v\in\{-1,1\}^r} \subseteq \mathcal{C}$ of 2^r distributions and constants $\alpha, \beta > 0$ such that $\forall\, v, v' \in \{-1, 1\}^r$:*

1. *The distance between $P_v, P_{v'}$ is at least proportional to the Hamming distance: $D_{\mathsf{H}}(P_v\|P_{v'})^2 \geq \alpha H(v, v')$;*
2. *If $H(v, v') = 1$, then $D_{\mathsf{H}}(P_v\|P_{v'})^2 \leq \beta$.*

Then, for all $M \geq 1$,

$$\inf_{\hat{P}\in A} \sup_{P\in\mathcal{C}} \mathbb{E}_{s_1,\ldots,s_M}[D_{\mathsf{H}}(\hat{P}\|P)^2] \geq \frac{\alpha}{8} r(1-\beta)^{2M}.$$

Informally, Assouad's Lemma interprets the distributions in the constructed family as being determined by r distinct choices (or bit flips), and states that any learning algorithm must learn many choices when the Hamming distance $H(v, v')$ is large and simultaneously cannot easily distinguish two distributions if $H(v, v') = 1$ [Canonne, 2020b] The above statement differs from that found in [Canonne, 2020b], as it is reformulated for the squared Hellinger distance which requires an additional factor of $\frac{1}{2}$ due to the square Hellinger only satisfying the weak triangle inequality, see Appendix B for more details.

Proposition 1 (PC Family Construction)**.** *There exists a family of $2^{\lceil S/2 \rceil}$ distributions in $\mathcal{C}_\mathcal{G} \subseteq \mathcal{C}_S$ for a deterministic PC structure $\mathcal{G}$ of size S, such that if $\eta \leq \min\{\min_n \frac{P(\gamma_n)}{16|\mathsf{ch}(n)|D^2}, \frac{1}{2}(\sum_n \frac{H_n}{P(\gamma_n)})^{-1}\}$, $\forall v, v' \in \{-1,1\}^{S/2}$, $D_\mathsf{H}(P_v \| P_{v'})^2 \geq \frac{27}{64}\eta H(v, v')$ and if $H(v, v')=1$, $D_\mathsf{H}(P_v \| P_{v'})^2 \leq 2\eta$.*

Proof sketch. W.l.o.g. suppose every sum node has an even number of children, and assign an index in $[1, S/2]$ to each pair of edges of the same sum node. Then, define P_v for each $v \in \{-1,1\}^{S/2}$ such that $\theta_{n,c}[v_{n,c}] = \frac{1}{\mathsf{ch}(n)} + v_c \tau_n$ where $\tau_n = \sqrt{\frac{\eta}{|\mathsf{ch}(n)|P(\gamma_n)} - \frac{\eta^2}{4P(\gamma_n)^2}}$. Using our conditions on η acheive the desired result. □

We obtain our lower bound from Assouad's Lemma with this construction and the hardness of learning the bias of a coin.

Proof Sketch of Theorem 1. Use the distribution family constructed in Proposition 1 with Lemma 2. Set η to return $r = \frac{S}{2}, \alpha = 2\epsilon^2$, and $\beta = \frac{76\epsilon^2}{S}$. Then $D_\mathsf{H}(\hat{P} \| P) < \epsilon$ only if M satisfies $(1 - \frac{76}{\epsilon^2})^{2M} < 1/2$, which proves the $\Omega(\frac{S}{\epsilon^2})$ term. Then, the hardness of estimating the bias of a coin proves the $\Omega(\frac{\log(1/\delta)}{\epsilon^2})$ term. Taking the sum of these two proves the lower bound for the Hellinger distance, which immediately extends to the reverse KL divergence by Lemma 1. □

4 TIGHT SAMPLE COMPLEXITY BOUNDS

We next show that this *necessary* number of samples is in fact sufficient to learn the parameters of a given PC structure to guarantee KL divergence and Hellinger distance less than ϵ with probability $1 - \delta$. This proves that the parameters of decomposable PCs are *efficiently* PAC learnable.

Theorem 2 (Upper Bound)**.** *Let $\mathcal{C}_\mathcal{G}$ be a family of distributions represented by PCs of a fixed structure $\mathcal{G}$ of size S (not necessarily deterministic). For $\epsilon > 0$ and $\delta \in (0, 1]$, a closed-form estimator given M samples from $P \in \mathcal{C}_\mathcal{G}$ outputs $\hat{P}$ such that $D_\mathsf{KL}(\hat{P} \| P) \leq \epsilon$ with probability $> 1 - \delta$ if:*

$$M \geq \max\left(\frac{15S}{e\epsilon}, \frac{2\log(1/\delta)}{\epsilon}\right) = O\left(\frac{S + \log(1/\delta)}{\epsilon}\right),$$

and $D_\mathsf{H}(\hat{P} \| P) \leq \epsilon$ and $D_\mathsf{TV}(\hat{P} \| P) \leq \epsilon$ w.p. $> 1 - \delta$ if:

$$M \geq \max\left(\frac{15S}{e\epsilon^2}, \frac{2\log(1/\delta)}{\epsilon^2}\right) = O\left(\frac{S + \log(1/\delta)}{\epsilon^2}\right).$$

Our tight upper bound is achieved by reducing the parameter learning problem for PCs over $\mathbf{X}$ to the closed-form MLE of deterministic PCs over $\mathbf{X} \cup \mathbf{Z}$, using the standard latent-variable ($\mathbf{Z}$) interpretation of PCs [Peharz et al., 2016]. We construct a sampling distribution for latent variables $\mathbf{Z}$ given each sample $\mathbf{x}$, inspired by a similar construction by Dasgupta [1997]. The key observation is that we do not need to know the true distribution $P(\mathbf{X}, \mathbf{Z})$, but rather any distribution $P'(\mathbf{X}, \mathbf{Z})$ that matches its marginal, i.e. $P'(\mathbf{X}) = P(\mathbf{X})$, suffices.

Definition 1. Let P be a PC over $\mathbf{X}$ and $\mathcal{N}$ be the set of sum nodes in P; for an input $\mathbf{x}$ that reaches node $n \in \mathcal{N}$, define $S_n(\mathbf{x}) = \{c \in \mathsf{ch}(n) : \mathbf{x} \in \gamma_c\}$. Then P' is the *deterministic extension* of P over $\mathbf{X}$ and latent variables $\mathbf{Z}$, defined as $P'(\mathbf{x}, \mathbf{z}) = P(\mathbf{x}) \prod_{n \in \mathcal{N}} P'(z_n | \mathbf{x})$ where

$$P'(z_n = c | \mathbf{x}) = \begin{cases} \frac{1}{|S_n(\mathbf{x})|} & \text{if } \mathbf{x} \in \gamma_n \text{ and } c \in S_n(\mathbf{x}), \\ \frac{1}{|\mathsf{ch}(n)|} & \text{if } \mathbf{x} \notin \gamma_n, \\ 0 & \text{otherwise.} \end{cases}$$

We can easily see that P' is a valid probability distribution by checking that $P'(Z_n | \mathbf{x})$ is a valid, normalized distribution for each $n \in \mathcal{N}$ and $\mathbf{x}$, whether $\mathbf{x} \in \gamma_n$ or $\mathbf{x} \notin \gamma_n$. Moreover, for each $\mathbf{x}$, we can evaluate $\mathbf{x} \in \gamma_n$ and compute $S_n(\mathbf{x})$ efficiently for all nodes in the circuit by a single forward and backward pass. Then the closed-form estimator for PCs over $\mathbf{X}$ first "augments" each sample $\mathbf{x}$ with a latent variable instantiation $\mathbf{z}$ sampled from $P'(\mathbf{Z} | \mathbf{x})$ then performs closed-form MLE with this augmented data on the deterministic PC over $\mathbf{X} \cup \mathbf{Z}$.[1]

Given our intuition for MLE of deterministic PCs, it is tempting to assume that the sample complexity can be derived trivially by considering the sample complexity of learning a discrete distribution at each sum node. However, unlike learning discrete distributions where every sample corresponds to some category, not every sample is routed through every sum node. That is, from the total number of samples used to learn the parameters of a PC, only a fraction would be relevant for learning the parameters of each sum node n, directly related to $P(\gamma_n)$. Moving past this naive approach, we next present a series of results to prove Theorem 2.

Lemma 3 (Deterministic reverse KLD Decomposition)**.** *Let $\hat{P}$ and P be two distributions represented by deterministic PCs with the same structure. Then $D_\mathsf{KL}(\hat{P} \| P) = \sum_{\text{sum } n} \hat{P}(\gamma_n) D_\mathsf{KL}(\hat{\theta}_n \| \theta_n)$, where $\hat{P}(\gamma_n)$ is the probability of reaching node n in $\hat{P}$, and $D_\mathsf{KL}(\hat{\theta}_n \| \theta_n)$ is the reverse KL divergence of the categoricals associated with node n.*

That is, the divergence between two deterministic PCs of the same structure is simply a weighted sum of divergence taken locally at each sum node. This allows us to show $\Pr[D_\mathsf{KL}(\hat{P} \| P') \leq \epsilon] \geq 1 - \delta$ by studying each sum node while accounting for its contribution to the overall divergence. Note that we use the *reverse KL divergence*, as enforcing learning guarantees with the standard KL divergence

[1] This closed-form estimator is equivalent to the closed-form MLE solution if the PC is already deterministic but will not necessarily achieve the MLE for non-deterministic PCs.

requires learning arbitrarily small probabilities without approximating by zero [Canonne, 2020a].

Lemma 4. *For P a deterministic PC and $\hat{P}$ learned with MLE with the same underlying structure and S parameters, and for all $\epsilon > \frac{S}{M}$, $\Pr[D_{\mathsf{KL}}(\hat{P}\|P) \geq \epsilon] \leq e^{-\epsilon M}\left(\frac{\epsilon e M}{S}\right)^S$.*

Proof sketch. Using Lemma 3, $\mathbb{E}[\exp(tD_{\mathsf{KL}}(\hat{P}\|P))] = \mathbb{E}[\exp(t\sum_n \hat{P}(\gamma_n)D_{\mathsf{KL}}(\hat{P}_{n,c}\|P_{n,c}))]$. Then, by the law of total expectation, $\mathbb{E}[\exp(tD_{\mathsf{KL}}(\hat{P}\|P))] = \mathbb{E}_m[\prod_n \mathbb{E}[\exp(t\frac{m_n}{M}D_{\mathsf{KL}}(\hat{\theta}_{n,c}\|\theta_{n,c})|m_n]]$. We next use [Agrawal, 2020, Theorem 1.3], which provides an upper bound on $\mathbb{E}[\exp(tD_{\mathsf{KL}}(\hat{P}\|P))]$, where $\hat{P}$ is the empirical estimator and P is a discrete distribution. With this, we can upper bound each $\mathbb{E}[\exp(t\frac{m_n}{M}D_{\mathsf{KL}}(\hat{\theta}_{n,c}\|\theta_{n,c})\ |m_n]$, which then yields the result when combined with a Chernoff bound. □

We are now ready to prove Theorem 2 following a similar argument from Canonne [2020a].

Proof of Theorem 2. Let $P'(\mathbf{X}, \mathbf{Z})$ be the deterministic extension of $P \in \mathcal{C}_\mathcal{G}$ constructed as in Definition 1. If $M \geq \frac{15S}{e\epsilon}$, then applying Lemma 4 to P' and $\hat{P}$ results in:

$$\Pr[D_{\mathsf{KL}}(\hat{P}(\mathbf{X}, \mathbf{Z})\|P'(\mathbf{X}, \mathbf{Z})) \geq \epsilon] \leq e^{-\frac{M\epsilon}{2}}.$$

To guarantee that the above is at most δ, we need $M \geq \frac{2\log(1/\delta)}{\epsilon}$. Combined with our initial requirement of $M \geq \frac{15S}{e\epsilon}$, this shows that given at least $O(\frac{S+\log(1/\delta)}{\epsilon})$ samples, the closed-form estimator $\hat{P}$ satisfies $D_{\mathsf{KL}}(\hat{P}(\mathbf{X})\|P(\mathbf{X})) = D_{\mathsf{KL}}(\hat{P}(\mathbf{X})\|P'(\mathbf{X})) \leq D_{\mathsf{KL}}(\hat{P}(\mathbf{X}, \mathbf{Z})\|P'(\mathbf{X}, \mathbf{Z})) \leq \epsilon$ with probability at least $1 - \delta$. The extension to the total variation and squared Hellinger distance follows from Lemma 1. □

5 RELAXING STRUCTURAL ASSUMPTIONS

Our tight upper bound makes a restrictive assumption that the true distribution can be represented as a PC with a known structure. As a step towards establishing sample complexity in the agnostic setting, we next derive an instance-specific bound on the distance between a learned PC and an unknown distribution P, given a set of samples from P. That is, as opposed to worst-case sample complexity, we show a bound for a given dataset and learned PC, with respective sizes M and S, computable in time $O(M \cdot S)$.

To achieve tight bounds with the least number of samples, we turn to *instance-optimal estimators*: algorithms that exploit the structure present in an instance of a problem *and* outperform all other algorithms on any given instance. Of course, not all structures can be perfectly exploited, resulting in an *irreducible error*—the minimum error that *any* algorithm could achieve—which is inherent to any instance. In particular, Valiant and Valiant [2021] showed that Good-Turing denoising (Algorithm D.1) can be used to obtain an instance-optimal estimator of discrete distributions.

Lemma 5 (Good-Turing Optimality [Valiant and Valiant, 2021])**.** *The GT Algorithm, given m independent draws from any distribution P with discrete support, outputs a labeled vector g, such that with probability $1 - m^{-\omega(1)}$, $\|P - g\|_1 \leq \tilde{O}(m^{-1/6})$.*[2]

Theorem 3. *Let g be the Good-Turing estimator, P be the true distribution, and $\hat{P}$ be a probabilistic circuit learned from a set of m samples $\mathbf{M}$ from P. Denote F_i as the number of domain elements that appear i times in $\mathbf{M}$. Then, with probability $1 - \delta$, $D_{\mathsf{TV}}(P\|\hat{P})$ is at most:*

$$g_\epsilon + \frac{F_1}{2m} + \frac{1}{2}(1 - \sum_{\mathbf{x}\in\mathbf{M}} \hat{P}(\mathbf{x})) + \frac{1}{2}\sum_{\mathbf{x}\in\mathbf{M}} \left|g(\mathbf{x}) - \hat{P}(\mathbf{x})\right|,$$

where g_ϵ is an approximation of $O(m^{-1/6})polylog(m)$ and depends on δ. As $m \to \infty$, the above bound reduces to the structural error of the underlying graph for $\hat{P}$.

Details on g_ϵ and the full proof can be found in Appendix C.5. We empirically test our upper bound using twenty density estimation datasets [Lowd and Davis, 2010, Bekker et al., 2015, Van Haaren and Davis, 2012, Larochelle and Murray, 2011] with circuits learned via the HCLT structure in PyJuice [Liu et al., 2024]. We provide further discussion on the experiments and results in Appendix D.

6 CONCLUSION

We have proved a lower bound on the sample complexity of learning PCs: for example, any learning algorithm requires $\Omega(\frac{S+\log(1/\delta)}{\epsilon})$ samples to ensure KL divergence at most ϵ with probability at least $1 - \delta$, as well as similar results for Hellinger and total variation distance. This holds even when the PC is deterministic with a known structure. Furthermore, we showed efficient PAC learning of decomposable PC parameters, proving a tight upper bound when the true distribution is a PC with a known structure. We further presented an efficiently computable instance-specific upper bound on the distance between learned PC and the true distribution in the agnostic setting.

Acknowledgements

This work was supported in part by the AFOSR grant FA9550-25-1-0320.

[2] Valiant and Valiant [2021, Theorem 1.3] include an additional term $ErrOpt^*(P, m)$ accounting for arbitrary relabeling of the domain; we can ignore this term as our task involves no relabeling.

REFERENCES

Ishaq Aden-Ali and Hassan Ashtiani. On the sample complexity of learning sum-product networks. In *International Conference on Artificial Intelligence and Statistics*, pages 4508–4518. PMLR, 2020.

Rohit Agrawal. Finite-sample concentration of the multinomial in relative entropy. *IEEE Transactions on Information Theory*, 66(10):6297–6302, 2020.

Hassan Ashtiani, Shai Ben-David, Nicholas JA Harvey, Christopher Liaw, Abbas Mehrabian, and Yaniv Plan. Near-optimal sample complexity bounds for robust learning of gaussian mixtures via compression schemes. *Journal of the ACM (JACM)*, 67(6):1–42, 2020.

Patrice Assouad. Deux remarques sur l'estimation. *Comptes rendus des séances de l'Académie des sciences. Série 1, Mathématique*, 296(23):1021–1024, 1983.

Jessa Bekker, Jesse Davis, Arthur Choi, Adnan Darwiche, and Guy Van den Broeck. Tractable learning for complex probability queries. *Advances in Neural Information Processing Systems*, 28, 2015.

Simone Bova, Florent Capelli, Stefan Mengel, and Friedrich Slivovsky. Knowledge compilation meets communication complexity. In *IJCAI*, volume 16, pages 1008–1014, 2016.

Clément L Cannone. A short note on poisson tail bounds. *Columbia. URL: https://www.cs.columbia.edu/ ccanonne/files/misc/2017-poissonconcentration.pdf*, 2017.

Clément L Canonne. A short note on learning discrete distributions. *arXiv preprint arXiv:2002.11457*, 2020a.

Clément L Canonne. A survey on distribution testing: Your data is big. but is it blue? *Theory of Computing*, pages 1–100, 2020b.

Arthur Choi and Adnan Darwiche. On relaxing determinism in arithmetic circuits. In *International Conference on Machine Learning*, pages 825–833. PMLR, 2017.

YooJung Choi, Antonio Vergari, and Guy Van den Broeck. Probabilistic circuits: A unifying framework for tractable probabilistic models. *UCLA. URL: http://starai. cs. ucla. edu/papers/ProbCirc20. pdf*, page 6, 2020.

Adnan Darwiche. A differential approach to inference in bayesian networks. *J. ACM*, 50(3):280–305, May 2003. ISSN 0004-5411. doi: 10.1145/765568.765570. URL https://doi.org/10.1145/765568.765570.

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

Nir Friedman and Zohar Yakhini. On the sample complexity of learning bayesian networks. In *Proceedings of the Twelfth international conference on Uncertainty in artificial intelligence*, pages 274–282, 1996.

Doga Kisa, Guy Van den Broeck, Arthur Choi, and Adnan Darwiche. Probabilistic sentential decision diagrams. In *KR*, 2014.

Hugo Larochelle and Iain Murray. The neural autoregressive distribution estimator. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 29–37. JMLR Workshop and Conference Proceedings, 2011.

John Leland and YooJung Choi. On the hardness of approximating distributions with tractable probabilistic models. *Advances in Neural Information Processing Systems*, 38: 42977–43000, 2026.

Anji Liu and Guy Van den Broeck. Tractable regularization of probabilistic circuits. *Advances in Neural Information Processing Systems*, 34:3558–3570, 2021.

Anji Liu, Kareem Ahmed, and Guy Van den Broeck. Scaling tractable probabilistic circuits: A systems perspective. In *Proceedings of the 41th International Conference on Machine Learning (ICML)*, jul 2024.

Lorenzo Loconte, Stefan Mengel, and Antonio Vergari. Sum of squares circuits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 19077–19085, 2025.

Daniel Lowd and Jesse Davis. Learning markov network structure with decision trees. In *2010 IEEE International Conference on Data Mining*, pages 334–343. IEEE, 2010.

James Martens and Venkatesh Medabalimi. On the expressive efficiency of sum product networks. *arXiv preprint arXiv:1411.7717*, 2014.

Robert Peharz, Robert Gens, Franz Pernkopf, and Pedro Domingos. On the latent variable interpretation in sum-product networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(10):2030–2044, 2016.

Hoifung Poon and Pedro Domingos. Sum-product networks: A new deep architecture. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 689–690. IEEE, 2011.

Alexandre B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer Publishing Company, 2008. ISBN 0387790519.

Gregory Valiant and Paul Valiant. Instance optimal learning. *arXiv preprint arXiv:1504.05321*, 2015.

Gregory Valiant and Paul Valiant. *Instance Optimal Distribution Testing and Learning*, page 506–526. Cambridge University Press, 2021.

Jan Van Haaren and Jesse Davis. Markov network structure learning: A randomized feature generation approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 26, pages 1148–1154, 2012.

Benjie Wang and Guy Van den Broeck. On the relationship between monotone and squared probabilistic circuits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 21026–21034, 2025.

Honghua Zhang, Brendan Juba, and Guy Van den Broeck. Probabilistic generating circuits. In *International Conference on Machine Learning*, pages 12447–12457. PMLR, 2021.

Statistical Guarantees for Probabilistic Circuit Parameter Learning (Supplementary Material)

John Leland[1] **YooJung Choi**[1]

[1]School of Computing and Augmented Intelligence, Arizona State University, Tempe, Arizona, USA

A PC BACKGROUND

Definition 2 (Probabilistic circuits)**.** A probabilistic circuit (PC) $\mathcal{C} := (\mathcal{G}, \theta)$ represents a joint probability distribution $p(\mathbf{X})$ over random variables $\mathbf{X}$ through a directed acyclic graph (DAG) $\mathcal{G}$ parameterized by θ. The DAG is composed of 3 types of nodes: leaf, product $\otimes$, and sum $\oplus$ nodes. Every leaf node in $\mathcal{G}$ is an input, and every internal node receives inputs from its children $\mathsf{ch}(n)$. The number of children at node n is denoted as $\mathsf{ch}(n)$. The scope of a given node, $\phi(n)$, is a recursively defined function which associates to each unit n a subset of $\mathbf{X}$: for each non-input unit n, $\phi(n) = \cup_{c \in \mathsf{ch}(n)} \phi(c)$, and the scope of a leaf node is a single variable in $\mathbf{X}$. The score of the root node is all random variables in the PC, $\mathbf{X}$. Each node n of a PC is defined recursively as:

$$P_n(x) := \begin{cases} l(\mathbf{x}), & \text{if } n \text{ is a leaf} \\ \prod_{c \in \mathsf{ch}(n)} P_c(\mathbf{x}) & \text{if } n \text{ is a } \otimes \\ \sum_{c \in \mathsf{ch}(n)} \theta_{n,c} P_c(\mathbf{x}) & \text{if } n \text{ is a } \oplus \end{cases} \tag{1}$$

where $\theta_{n,c} \in [0, 1]$ is the parameter associated with the edge connecting nodes n, c in $\mathcal{G}$, and $\sum_{c \in \mathsf{ch}(n)} \theta_{n,c} = 1$. We assume $l(\mathbf{x})$ at a leaf node is a Boolean indicator function: i.e., $\mathbb{1}[\mathbf{x} = 1]$ or $\mathbb{1}[\mathbf{x} = 0]$. The distribution represented by the circuit is the output at its root node.

Probabilistic circuits guarantee tractable inference by composing sum and product gates under specific conditions. In this work we focus on circuits that are decomposable, as well as both deterministic and decomposable.[1]

Definition 3 (Smoothness and decomposability)**.** A sum node is *smooth* if its children have identical scopes: $\phi(c) = \phi(n)$, $\forall c \in \mathsf{ch}(n)$. A product node is *decomposable* if its children have disjoint scopes: $\phi(c_i) \cap \phi(c_j) = \emptyset$, $\forall c_i \neq c_j \in \mathsf{ch}(n)$. A PC is smooth and decomposable iff every sum node is smooth and every product node is decomposable.

Definition 4 (Determinism)**.** A sum node is *deterministic* if, for any fully-instantiated input, the output of at most one of its children is nonzero. In other words, the supports of its children are mutually disjoint. A PC is deterministic iff all of its sum nodes are deterministic.

Deterministic and decomposable PCs (see Figure 1 for an example) have the following useful property when looking at their joint probability distribution. See that for a given variable instantiation $\mathbf{X} = \mathbf{x}$, we have that the circuit admits only a single tree-shaped path through the PC [Choi and Darwiche, 2017]. Thus, to compute the probability mass function, we can simply multiply together the parameters associated with the tree defined by $\mathbf{x}$. We denote the tree defined by $\mathbf{x}$ as $\mathsf{tree}(\mathbf{x})$; then the probability of $\mathbf{x}$ according to a deterministic PC can be expressed as:

$$P(\mathbf{x}) = \prod_{e \in \mathsf{tree}(\mathbf{x})} \theta_e.$$

[1]Smoothness is necessary, but we can trivially smooth our circuit in $O(n^2)$ if necessary [Choi et al., 2020].

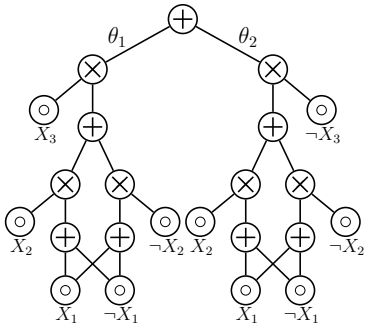

Figure 1: A smooth, decomposable, and deterministic PC (weights shown only for the root for conciseness).

Deterministic and decomposable PCs also admit closed-form maximum-likelihood estimation of their parameters [Liu and Van den Broeck, 2021]. This closed form can be thought of simply as counting the number of samples that reach a child edge c at a sum node n, divided by the total number of samples to reach that same sum node n. Mathematically, this is formalized using the following notion:

Definition 5 (Flows)**.** The flow $F_{n,c}(\mathbf{x})$ of any edge (n, c) in a PC given variables assignments $\mathbf{x}$ is defined as $\mathbb{I}[\mathbf{x} \in \gamma_n \cap \gamma_c]$ where $\mathbb{I}[\cdot]$ is the indicator function and γ_n is the context of node n. The flow $F_{n,c}(D)$ with respect to a dataset $D = \{\mathbf{x}^{(i)}\}_{i=1}^N$ is the sum of the flows of all samples: $F_{n,c}(D) = \sum_{i=1}^N F_{n,c}(\mathbf{x}^{(i)})$.

The MLE parameters [Kisa et al., 2014] are then computed in closed form as the following:

$$\forall (n, c) : \theta^*_{n,c} = \frac{F_{n,c}(D)}{\sum_{c \in \mathsf{ch}(n)} F_{n,c}(D)}.$$

B DERIVATION OF LEMMA 2 FROM ASSOUAD'S LEMMA

Here we explain exactly how our version of Assouad's lemma can be derived from the statement in [Tsybakov, 2008]. Note that this statement differs from that of [Canonne, 2020b] by a factor of $1/2$ as the total variation distance satisfies the triangle inequality, while the squared Hellinger only satisfies $D_{\mathsf{H}}(P\|Q)^2 \leq 2(D_{\mathsf{H}}(P\|Z)^2 + D_{\mathsf{H}}(Z\|Q)^2)$.

Lemma 6 (Assouad's Lemma [Tsybakov, 2008])**.** *Let $\{0,1\}^r$ be the set of all binary sequences of length r. Let $\{P_v\}_{v \in \{0,1\}^r} \subseteq \mathcal{C}$ be a set of 2^r probability measures, and let the corresponding expectations be denoted by $\mathbb{E}_v$. Then,*

$$\inf_{\hat{v}} \max_{v \in \{0,1\}^r} \mathbb{E}_v[H(v, \hat{v})] \geq \frac{r}{2} \min_{v,v' : H(v,v')=1} \inf_{\psi} \left(P_v(\psi \neq 0) + P_{v'}(\psi \neq 1) \right)$$

where $H(v, v')$ is the Hamming distance between v and v', $\inf_{\hat{v}}$ denotes the infimum over all estimators taking values in $\{0,1\}^r$ and where $\inf_{\psi}$ denotes the infimum over all tests taking values in $\{0,1\}$.

Tsybakov [2008] makes the following observation in the continuous case: $\inf_\psi (P_v(\psi \neq 0) + P_{v'}(\psi \neq 1)) = \int \min(dP_v, dP_{v'})$. Thus for the discrete case, this is equivalent to: $\inf_\psi (P_v(\psi \neq 0) + P_{v'}(\psi \neq 1)) = \sum_{\mathbf{x}} \min(P_v(\mathbf{x}), P_{v'}(\mathbf{x}))$. Using Le Cam's inequalities, we have:

$$\sum_{\mathbf{x}} \min(P_v(\mathbf{x}), P_{v'}(\mathbf{x})) \geq \frac{1}{2} (\sum_{\mathbf{x}} \sqrt{P_v(\mathbf{x}) P_{v'}(\mathbf{x})})^2.$$

Note that Lemma 2 uses $v, v' \in \{-1, 1\}^r$ instead of $v, v' \in \{0,1\}^r$ and adds the following conditions:

1. For all $v, v' \in \{-1,1\}^r$, the distance between $P_v, P_{v'}$ is at least proportional to the Hamming distance: $D_{\mathsf{H}}(P_v\|P_{v'})^2 \geq \alpha H(v, v')$.
2. For all $v, v' \in \{-1,1\}^r$ with $H(v, v') = 1$, the squared Hellinger distance of $P_v, P_{v'}$ is small: $D_{\mathsf{H}}(P_v\|P_{v'})^2 \leq \beta$.

We are interested in the squared Hellinger distance, which can be expressed as $D_{\mathsf{H}}(P_v \| P_{v'})^2 = 1 - \sum_{\mathbf{x}} \sqrt{P_v(\mathbf{x}) P_{v'}(\mathbf{x})}$. As such, if condition 2 is satisfied then we have that $\sum_{\mathbf{x}} \sqrt{P_v(\mathbf{x}) P_{v'}(\mathbf{x})} > (1-\beta)$. If we take this over $M \geq 1$ samples, we then have $\sum_{\mathbf{x}} \sqrt{P_v^{\otimes M}(\mathbf{x}) P_{v'}^{\otimes M}(\mathbf{x})} \geq (1-\beta)^M$. Next, using Condition 2,

$$\inf_{\hat{P} \in A} \max_{P \in \{P_v\}_{v \in \{-1,1\}^r}} \mathbb{E}_v[H(v, \hat{v})] \geq \frac{r}{4} \left(\sum_{\mathbf{x}} \sqrt{P_v^{\otimes M}(\mathbf{x}) P_{v'}^{\otimes M}(\mathbf{x})} \right)^2 \geq \frac{r}{4} (1-\beta)^{2M}.$$

Finally, by Condition 1, and the fact that the squared Hellinger satisfies the weak triangle inequality $D_{\mathsf{H}}(P_v \| P_{v'})^2 \leq 2(D_{\mathsf{H}}(P_v \| \hat{P})^2 + D_{\mathsf{H}}(\hat{P} \| P_{v'})^2)$, we can say:

$$\inf_{\hat{P} \in A} \max_{P \in \{P_v\}_{v \in \{-1,1\}^r}} \mathbb{E}_v[D_{\mathsf{H}}(\hat{P} \| P)^2] \geq \frac{\alpha r}{8} (1-\beta)^{2M}.$$

Since $\{P_v\}_{v \in \{-1,1\}^r} \subseteq \mathcal{C}$, this implies exactly the following statement of Lemma 2:

$$\inf_{\hat{P}} \max_{P \in \mathcal{C}} \mathbb{E}_v[D_{\mathsf{H}}(\hat{P} \| P)^2] \geq \frac{\alpha r}{8} (1-\beta)^{2M}.$$

C PROOFS

C.1 PROOF OF PROPOSITION 1

Let $\mathcal{G}$ be a deterministic PC structure of size S. W.l.o.g. suppose every sum node has an even number of children, and assign an index in $[1, S/2]$ to every pair of edges of the same sum node. We parameterize the PC to define P_v for each $v = \{-1, 1\}^{S/2}$, where for every sum node n, the parameters for half of its edges will be perturbed as $\theta_{n,c}[v] = \frac{1}{|\mathsf{ch}(n)|} + v_{n,c}\tau_n$, and the parameters for their counterparts $\theta_{n,c}[v] = \frac{1}{|\mathsf{ch}(n)|} - v_{n,c}\tau_n$. We set $\tau_n = \sqrt{\frac{\eta}{|\mathsf{ch}(n)|P(\gamma_n)} - \frac{\eta^2}{4P(\gamma_n)^2}}$ and $\eta \leq \min\{\min_n \frac{P(\gamma_n)}{16|\mathsf{ch}(n)|D^2}, \frac{1}{2}(\sum_n \frac{H_n}{P(\gamma_n)})^{-1}\}$ where D is the maximum number of parameters that can be in a tree shaped path. Recall that for a deterministic PC, evaluating on an instantiation $\mathbf{x}$ is equivalent to taking the product of the parameters in the tree defined by $\mathbf{x}$, $P_v(\mathbf{x}) = \prod_{(n,c) \in \mathsf{tree}(\mathbf{x})} \theta_{n,c}[v]$ and here we use $P(\gamma_n)$ as the probability of reaching node n under the uniform circuit.

For the first condition of Assouad's Lemma, we lower-bound the squared Hellinger distance between P_v and $P_{v'}$ as a function of the Hamming distance between v and v'. Note that for every edge (n, c): if $v_{n,c} = v'_{n,c}$ then $\theta_{n,c}[v] \cdot \theta_{n,c}[v'] = (\frac{1}{|\mathsf{ch}(n)|} \pm v_{n,c}\tau_n)^2$ where the sign $\pm$ is determined by whether it is a perturbed or dependent edge; if $v_{n,c} \neq v'_{n,c}$ then $\theta_{n,c}[v] \cdot \theta_{n,c}[v'] = (\frac{1}{|\mathsf{ch}(n)|} + \tau_n)(\frac{1}{|\mathsf{ch}(n)|} - \tau_n) = \frac{1}{|\mathsf{ch}(n)|^2} - \tau_n^2$. Then we have:

$$D_{\mathsf{H}}(P_v \| P_{v'})^2 = 1 - \sum_{\mathbf{x}} \sqrt{P_v(\mathbf{x}) P_{v'}(\mathbf{x})} = 1 - \sum_{\mathbf{x}} \sqrt{\prod_{n,c} (\theta_{n,c}[v] \theta_{n,c}[v'])^{\mathbb{I}((n,c) \in \mathsf{tree}(\mathbf{x}))}}$$

$$= 1 - \sum_{\mathbf{x}} \left[\prod_{n,c: v_{n,c} = v'_{n,c}} (\frac{1}{|\mathsf{ch}(n)|} \pm v_{n,c}\tau_n)^{\mathbb{I}((n,c) \in \mathsf{tree}(\mathbf{x}))} \prod_{n,c: v_{n,c} \neq v'_{n,c}} \sqrt{\left(\frac{1}{|\mathsf{ch}(n)|^2} - \tau_n^2 \right)^{\mathbb{I}((n,c) \in \mathsf{tree}(\mathbf{x}))}} \right] \tag{2}$$

For given v, v', we can parameterize the same PC structure using the following:

$$\psi_{n,c} = \begin{cases} \frac{1}{\mathsf{ch}(n)} + v_{n,c}\tau_n & \text{if } v_{n,c} = v'_{n,c}, \\ \sqrt{\frac{1}{\mathsf{ch}(n)^2} - \tau_n^2} & \text{if } v_{n,c} \neq v'_{n,c}. \end{cases}$$

Next, we will normalize $\psi_{n,c}$ to define a valid probability distribution. To that end, we first compute the normalization

constant for each n as follows:

$$\sum_{c\in\mathsf{ch}(n)}\psi_{n,c}=\frac{1}{2}\sum_{c:v_{n,c}=v'_{n,c}}(\frac{1}{|\mathsf{ch}(n)|}+\tau+\frac{1}{|\mathsf{ch}(n)|}-\tau)+\sum_{c:v_{n,c}\neq v'_{n,c}}\sqrt{\frac{1}{\mathsf{ch}(n)^2}-\tau_n^2}$$
$$=\sum_{c:v_{n,c}=v'_{n,c}}\frac{1}{|\mathsf{ch}(n)|}+\sum_{c:v_{n,c}\neq v'_{n,c}}\sqrt{\frac{1}{\mathsf{ch}(n)^2}-\tau_n^2}$$
$$=\sum_{c:v_{n,c}=v'_{n,c}}\frac{1}{|\mathsf{ch}(n)|}+\sum_{c:v_{n,c}\neq v'_{n,c}}\left(\frac{1}{|\mathsf{ch}(n)|}-\rho_n\right)=1-\sum_{c:v_{n,c}\neq v'_{n,c}}\rho_n=1-\rho_nH_n(v,v'),\qquad(3)$$

where $\rho_n=\frac{2}{|\mathsf{ch}(n)|}-2\sqrt{\frac{1}{\mathsf{ch}(n)^2}-\tau_n^2}=\frac{\eta}{P(\gamma_n)}$ to derive Equation 3, and H_n refers to the Hamming distance over only the indices corresponding to edges of n. Then, for each sum node n, we define a normalized distribution $\tilde{P}_{n,c}$ as follows:

$$\tilde{\theta}_{n,c}=\begin{cases}\frac{\frac{1}{|\mathsf{ch}(n)|}+v_n\tau_n}{1-\rho_nH_n(v,v')} & \textit{if } v_{n,c}=v'_{n,c},\\ \\ \frac{\sqrt{\frac{1}{\mathsf{ch}(n)^2}-\tau_n^2}}{1-\rho_nH_n(v,v')} & \textit{if } v_{n,c}\neq v'_{n,c}.\end{cases}$$

Therefore, the squared Hellinger distance via Equation 2 can be expressed as:

$$1-\sum_{\mathbf{x}}\prod_{(n,c)\in\mathsf{tree}(\mathbf{x})}\psi_{n,c}=1-\sum_{\mathbf{x}}\prod_{(n,c)\in\mathsf{tree}(\mathbf{x})}\tilde{\theta}_{n,c}(1-\rho_nH_n)=\mathbb{E}_{t\sim\tilde{P}}[1-\prod_{n\in t}(1-\rho_nH_n)].\qquad(4)$$

We lower- and upper-bound the above using Bonferrnoni and Weierstrass' inequality, respectively, as our choice of τ_n ensures the necessary conditions to apply them—$\sum_{n,c}\rho_nH_n\leq w$ and $\rho_nH_n\leq 1$:

$$(1-\frac{w}{2})\sum_{n\in t}\rho_nH_n\leq 1-\prod_{n\in t}(1-\rho_nH_n)\leq\sum_{n\in t}\rho_nH_n.$$

Thus, we can bound the expectation in Equation 4 from below using $\mathbb{E}_{t\sim\tilde{P}}[(1-\frac{w}{2})\sum_{n\in t}\rho_nH_n]=(1-\frac{w}{2})\sum_n\rho_nH_n\mathbb{E}_{t\sim\tilde{P}}[\mathbb{I}(n\in t)]=(1-\frac{w}{2})\sum_n\rho_nH_n\tilde{P}(\gamma_n)$ and similarly from above using $\mathbb{E}_{t\sim\tilde{P}}[\sum_{n,c\in t}\rho_nH_n]=\sum_n\rho_nH_n\tilde{P}(\gamma_n))$. Note that

The following can be verified with a short computation: for all n,c, and v,v' (W.l.o.g write v),

$$\tilde{\theta}_{n,c}\geq\sqrt{\frac{\frac{1}{|\mathsf{ch}(n)|}-\tau_n}{\frac{1}{|\mathsf{ch}(n)|}+\tau_n}}\cdot\theta_{n,c}[v]$$

Then let D be the number of parameterized edges in the tree t, and let $\varphi=\min_n\sqrt{\frac{\frac{1}{|\mathsf{ch}(n)|}-\tau_n}{\frac{1}{|\mathsf{ch}(n)|}+\tau_n}}$, $\tilde{P}(\gamma_n))\geq v^DP(\gamma_n))$. Therefore,

$$D_{\mathsf{H}}(P_v\|P_{v'})^2\geq(1-\frac{w}{2})\varphi^d\sum_n\rho_nH_nP_v(\gamma_n).$$

Now, by our choice of η, we know that $\rho_n\leq\frac{1}{16|\mathsf{ch}(n)|D^2}$, which implies $\tau_n\leq\sqrt{\frac{1}{|\mathsf{ch}(n)|}\rho_n}\leq\frac{1}{4|\mathsf{ch}(n)|D}$ and $\tau_n|\mathsf{ch}(n)|\leq\frac{1}{4D}$.

Remember that for all n,c,

$$\frac{1}{|\mathsf{ch}(n)|}(1-\tau_n|\mathsf{ch}(n)|)\leq\theta_{n,c}[v]\leq\frac{1}{|\mathsf{ch}(n)|}(1+\tau_n|\mathsf{ch}(n)|)$$

Suppose n has a path t which collects k parameters from root to node n, then:

$$\prod_{n\in t}(\frac{1}{|\mathsf{ch}(n)|}(1-\tau_n|\mathsf{ch}(n)|))\leq P_v(\gamma_n)\leq\prod_{n\in t}(\frac{1}{|\mathsf{ch}(n)|}(1+\tau_n|\mathsf{ch}(n)|))$$

Plugging in the bounds from above:

$$\prod_{n\in t}(\frac{1}{|\mathsf{ch}(n)|}(1-\frac{1}{4D}))^k \le \theta_{n,c}[v] \le \prod_{n\in t}(\frac{1}{|\mathsf{ch}(n)|}(1+\frac{1}{4D})^k$$

Now using Bernoulli's inequality with the maximum number of parameters that can be collected, D,

$$(1-\frac{1}{4D})^D \ge (1-\frac{1}{4D}D) = 3/4$$

Conversely,

$$(1+\frac{1}{4D})^D \le (1-\frac{1}{4D})^{-D} \le \frac{4}{3}.$$

Therefore,

$$\frac{3}{4}P(\gamma_n) \le P_v(\gamma_n) \le \frac{4}{3}P(\gamma_n).$$

Using the same trick to get an upper bound on $\tilde{P}(\gamma_n)$ see that $\tilde{\theta}_{n,c} = \frac{\psi_{n,c}}{1-\rho_n H_n} \le \frac{1}{\frac{|\mathsf{ch}(n)|(1-\tau_n|\mathsf{ch}(n)|)}{(1-\frac{1}{32D^2})}} \le \frac{3}{2}$. Next, see that $v = \min_n \sqrt{\frac{\frac{1}{|\mathsf{ch}(n)|}-\tau_n}{\frac{1}{|\mathsf{ch}(n)|}+\tau_n}} \ge \min_n 1-\tau_n|\mathsf{ch}(n)|$. From the above $\tau_n|\mathsf{ch}(n)| \le \frac{1}{4d}$, which implies again that $\varphi \ge (1-\frac{1}{4d})$. From above, $\varphi^d \ge \frac{3}{4}$.

Putting this together, our constructed family of distributions meets the first condition of Lemma 2 as

$$\begin{aligned} D_{\mathsf{H}}(P_v\|P_{v'})^2 &\ge (1-\frac{w}{2})v^d\sum_n \rho_n H_n P_v(\gamma_n)) \ge (1-\frac{1}{4})\frac{3}{4}\sum_{n,c}\frac{\eta}{P(\gamma_n)}H_n P_v(\gamma_n)) \\ &\ge \frac{9}{16}\sum_{n,c}\frac{\eta}{P(\gamma_n)}H_n P(\gamma_n)\frac{3}{4} = \frac{27}{64}\eta\sum_n H_n \end{aligned}$$

and the second condition since $\sum_n H_n = 1$:

$$D_{\mathsf{H}}(P_v\|P_{v'}) \le \sum_n \rho_n\tilde{P}(\gamma_n) \le \frac{\eta}{P(\gamma)}\tilde{P}(\gamma_n)) \le \frac{3}{2}\eta < 2\eta.$$

□

C.2 PROOF OF THEOREM 1

We first apply Lemma 2 using the family of distributions constructed in Proposition 1 to obtain:

$$\begin{aligned} \inf_{A\in A_M}\sup_{P\in\mathcal{C}_S}\mathbb{E}_{s_1,\ldots,s_M\sim P}[D_{\mathsf{H}}(\hat{P}\|P)^2] &\ge \inf_{A\in A_M}\sup_{P\in\mathcal{C}_\mathcal{G}}\mathbb{E}_{s_1,\ldots,s_M\sim P}[D_{\mathsf{H}}(\hat{P}\|P)^2] \ge \frac{1}{8}\alpha\frac{S}{2}(1-\beta)^{2M} \\ &\ge \frac{1}{8}\frac{27\eta}{64}\frac{S}{2}(1-2\eta)^{2M} = \frac{27\eta S}{1024}(1-2\eta)^{2M}. \end{aligned}$$

To make $\frac{27\eta S}{1024} = 2\epsilon^2$, first define $\omega \ge |\mathsf{ch}(n)|$ for all n and let $\mathcal{G}$ have depth d. Then, set $\eta = \frac{76\epsilon^2}{S}$ by forcing $\epsilon \le \sqrt{\frac{1}{76\omega^d}}$. Finally, $2\epsilon^2(1-2\frac{76\epsilon^2}{S})^{2M}$. Thus we must set, $(1-2\frac{76\epsilon^2}{S})^{2M} < 1/2$ to guarantee $2\epsilon^2(1-2\frac{76\epsilon^2}{S})^{2M} \le \epsilon^2$. Using Bernoulli,

$$1/2 > (1-2\frac{76\epsilon^2}{S})^{2M} \ge 1-4\frac{76\epsilon^2}{S}M \quad\Longrightarrow\quad \frac{S}{608\epsilon^2} \le M,$$

which proves that $M = \Omega(\frac{S}{\epsilon^2})$. Additionally, under our construction any learning algorithm must also learn the bias of a coin corresponding to the perturbations v, v'. Learning the bias of a coin is a well studied task in learning theory which requires at least $M = \frac{\log(2/\delta)}{5\epsilon^2} = \Omega(\frac{\log(1/\delta)}{\epsilon^2})$ to learn within probability $1-\delta$.[2] Combining the two complexity results, we have:

$$M = \Omega(\max(\frac{S}{\epsilon^2}, \frac{\log(1/\delta)}{\epsilon^2})) = \Omega(\frac{S+\log(1/\delta)}{\epsilon^2}).$$

[2]For a simple proof of the upper and lower bounds, see Dasgupta [1997]

Then taking the Hellinger distance as the distance measure, since $D_{\mathsf{H}}(\hat{P}\|P)^2 \leq \epsilon^2 \implies D_{\mathsf{H}}(\hat{P}\|P) \leq \epsilon$, we have that ensuring $D_{\mathsf{H}}(\hat{P}\|P) \leq \epsilon$ with probability $> 1-\delta$ requires:

$$M \geq \max(\frac{S\log(2)}{608\epsilon^2}, \frac{\log(2/\delta)}{5\epsilon^2}),$$

Similarly, using $D_{\mathsf{H}}(\hat{P}\|P)^2 < \frac{1}{2}D_{\mathsf{KL}}(\hat{P}\|P)$, ensuring $D_{\mathsf{KL}}(\hat{P}\|P) \leq \epsilon$ with probability $1-\delta$ requires:

$$M \geq \max(\frac{S\log(2)}{304\epsilon}, \frac{2\log(2/\delta)}{5\epsilon}),$$

and finally using $D_{\mathsf{H}}(\hat{P}\|P)^2 \leq \epsilon \implies D_{\mathsf{TV}}(\hat{P}\|P) < \epsilon$, ensuring $D_{\mathsf{TV}}(\hat{P}\|P) \leq \epsilon$ with probability $1-\delta$ requires:

$$M \geq \max(\frac{S\log(2)}{608\epsilon}, \frac{\log(2/\delta)}{5\epsilon^2}).$$

□

C.3 PROOF OF LEMMA 3

Suppose that $\hat{P}$ and P are deterministic and decomposable PCs with the same structure over variables $\mathbf{Y}$. Then for all $\mathbf{y}$:

$$\log\left(\frac{\hat{P}(\mathbf{y})}{P(\mathbf{y})}\right) = \log\left(\frac{\prod_{e\in\mathsf{tree}(\mathbf{y})}\hat{\theta}_e}{\prod_{e\in\mathsf{tree}(\mathbf{y})}\theta_e}\right) = \sum_{e\in\mathsf{tree}(\mathbf{y})}\log\left(\frac{\hat{\theta}_e}{\theta_e}\right) = \sum_{n,c}\log\left(\frac{\hat{\theta}_{n,c}}{\theta_{n,c}}\right)\mathbb{I}((n,c)\in\mathsf{tree}(\mathbf{y})).$$

Next, use the definition of the reverse KL divergence: We can then express the reverse KL divergence as the following:

$$\begin{aligned}
&D_{\mathsf{KL}}(\hat{P}\|P) = \sum_{\mathbf{y}}\hat{P}(\mathbf{y})\log\left(\frac{\hat{P}(\mathbf{y})}{P(\mathbf{y})}\right) = \sum_{\mathbf{y}}\hat{P}(\mathbf{y})\sum_{n,c}\log\left(\frac{\hat{\theta}_{n,c}}{\theta_{n,c}}\right)\mathbb{I}((n,c)\in\mathsf{tree}(\mathbf{y}))\\
&= \sum_{n,c}\log\left(\frac{\hat{\theta}_{n,c}}{\theta_{n,c}}\right)\sum_{\mathbf{y}}\hat{P}(\mathbf{y})\mathbb{I}((n,c)\in\mathsf{tree}(\mathbf{y})) = \sum_{n,c}\log\left(\frac{\hat{\theta}_{n,c}}{\theta_{n,c}}\right)\hat{P}(\gamma_n\wedge\gamma_c) = \sum_{n,c}\log\left(\frac{\hat{\theta}_{n,c}}{\theta_{n,c}}\right)\hat{P}(\gamma_n)\hat{P}(\gamma_c|\gamma_n)\\
&= \sum_{n,c}\log\frac{\hat{\theta}_{n,c}}{\theta_{n,c}}\hat{P}(\gamma_n)\hat{\theta}_{n,c} = \sum_{n}\hat{P}(\gamma_n)D_{\mathsf{KL}}(\hat{\theta}_n\|\theta_n).
\end{aligned}$$

□

C.4 PROOF OF LEMMA 4

Let $\hat{P}$ be a deterministic and decomposable PC with MLE parameters obtained using samples from the true distribution P, which is a PC of the same structure. For all n, let m_n be the number of samples that reach node n, and $0 < t\frac{m_n}{M} < m_n$. Then by Lemma 3, we have that $\mathbb{E}[\exp(tD_{\mathsf{KL}}(\hat{P}\|P))] = \mathbb{E}[\exp(t\sum_n\hat{P}(\gamma_n)D_{\mathsf{KL}}(\hat{\theta}_{n,c}\|\theta_{n,c}))]$. Next, use the law of total expectation and condition to fix the m_n's.

$$\begin{aligned}
&\mathbb{E}[\exp(t\sum_n\hat{P}(\gamma_n)D_{\mathsf{KL}}(\hat{P}_{n,c}\|P_{n,c}))] = \mathbb{E}_m[\mathbb{E}[\exp(t\sum_n\frac{m_n}{M}D_{\mathsf{KL}}(\hat{\theta}_{n,c}\|\theta_{n,c}))|(m_1,\ldots,m_s)]]\\
&= \mathbb{E}_m\left[\mathbb{E}\left[\prod_n\exp(t\frac{m_n}{M}D_{\mathsf{KL}}(\hat{\theta}_{n,c}\|\theta_{n,c})|(m_1,\ldots,m_s)\right]\right] = \mathbb{E}_m\left[\prod_n\mathbb{E}\left[\exp(t\frac{m_n}{M}D_{\mathsf{KL}}(\hat{\theta}_{n,c}\|\theta_{n,c})|m_n\right]\right] \qquad (5)
\end{aligned}$$

As such, it is easy to see that we can view each $D_{\mathsf{KL}}(\hat{\theta}_{n,c}\|\theta_{n,c})$ as P being a categorical distribution over $|\mathsf{ch}(n)|$ categories conditioned on having m_n samples each. Thus, applying [Agrawal, 2020, Theorem 1.3] results in:

$$\mathbb{E}[\exp(t\frac{m_n}{M}D_{\mathsf{KL}}(\hat{\theta}_{n,c}\|\theta_{n,c})|m_n] \leq \left(\frac{1}{1-\frac{t\frac{m_n}{M}}{m_n}}\right)^{|\mathsf{ch}(n)|-1} = \left(\frac{1}{1-\frac{t}{M}}\right)^{|\mathsf{ch}(n)|-1}.$$

This allows us to bound Equation 5 as the following:

$$\mathbb{E}_m\left[\prod_n \mathbb{E}\left[\exp(t\frac{m_n}{M}D_{\mathsf{KL}}(\hat{\theta}_{n,c}\|\theta_{n,c})|m_n\right]\right] \leq \mathbb{E}_m\left[\prod_n\left(\frac{1}{1-\frac{t}{M}}\right)^{|\mathsf{ch}(n)|-1}\right] = \left(\frac{1}{1-\frac{t}{M}}\right)^S.$$

Using a standard Chernoff bound, we then have that for $0 \leq t < M$:

$$\Pr[D_{\mathsf{KL}}(\hat{P}\|P) \geq \epsilon] \leq \inf_{t\in[0,M)]} e^{-t\epsilon}\mathbb{E}[e^{tD_{\mathsf{KL}}(\hat{P}\|P)}] \leq \inf_{t\in[0,M)]} e^{-t\epsilon}\left(\frac{1}{1-\frac{t}{M}}\right)^S.$$

Finally, choose $\frac{t}{M} = 1 - \frac{S}{\epsilon M}$, which completes our proof as follows for all $\epsilon > \frac{S}{M}$:

$$\Pr[D_{\mathsf{KL}}(\hat{P}\|P) \geq \epsilon] \leq e^{-t\epsilon}\left(\frac{1}{1-\frac{t}{M}}\right)^S \leq e^{-\epsilon M}\left(\frac{\epsilon e M}{S}\right)^S.$$

□

C.5 PROOF OF THEOREM 3

First, by the definition of total variation distance and the triangle inequality, we have:

$$\begin{aligned} D_{\mathsf{TV}}(P\|\hat{P}) &\leq D_{\mathsf{TV}}(P\|g) + D_{\mathsf{TV}}(g\|\hat{P}) \leq D_{\mathsf{TV}}(P\|g) + \frac{1}{2}\sum_{\mathbf{x}\notin\mathbf{M}}\left|g(\mathbf{x}) - \hat{P}(\mathbf{x})\right| + \frac{1}{2}\sum_{\mathbf{x}\in\mathbf{M}}\left|g(\mathbf{x}) - \hat{P}(\mathbf{x})\right| \\ &\leq D_{\mathsf{TV}}(P\|g) + \frac{1}{2}\sum_{\mathbf{x}\notin\mathbf{M}} g(\mathbf{x}) + \frac{1}{2}\sum_{\mathbf{x}\notin\mathbf{M}}\hat{P}(\mathbf{x}) + \frac{1}{2}\sum_{\mathbf{x}\in\mathbf{M}}\left|g(\mathbf{x}) - \hat{P}(\mathbf{x})\right|. \end{aligned}$$

From the Good-Turing Algorithm, $\sum_{\mathbf{x}\notin\mathbb{M}} g(\mathbf{x}) = \frac{F_1}{m}$. Therefore,

$$D_{\mathsf{TV}}(P\|\hat{P}) \leq D_{\mathsf{TV}}(P\|g) + \frac{F_1}{2m} + \frac{1}{2}\left(1 - \sum_{\mathbf{x}\in\mathbf{M}}\hat{P}(\mathbf{x})\right) + \frac{1}{2}\sum_{\mathbf{x}\in\mathbf{M}}\left|g(\mathbf{x}) - \hat{P}(\mathbf{x})\right|. \tag{6}$$

Note that given the set of samples $\mathbf{M}$, we can compute the last three terms on the RHS above by simply iterating over each sample and evaluating its probability according to $\hat{P}$ and g. Before we describe our estimate for the first term $D_{\mathsf{TV}}(P\|g)$ in detail, first observe that as $m \to \infty$, the first two terms $D_{\mathsf{TV}}(P\|g)$ and $\frac{F_1}{2m}$ vanish to 0. Moreover, as $m \to \infty$, the distribution of $\mathbf{M}$ approaches P. Therefore, the RHS of Equation 6 approaches the following as $m \to \infty$:

$$\frac{1}{2}\sum_{\mathbf{x}\notin\mathsf{supp}(P)}\hat{P}(\mathbf{x}) + \frac{1}{2}\sum_{\mathbf{x}\in\mathsf{supp}(P)}\left|g(\mathbf{x}) - \hat{P}\right| = D_{\mathsf{TV}}(P\|\hat{P}).$$

In other words, as $m \to \infty$, our bound approaches the irreducible error due to the given PC structure.

We now describe our choice of g_ϵ, which approximates an upper bound on $D_{\mathsf{TV}}(P\|g)$, to obtain a computable (approximate) bound on Equation 6. Valiant and Valiant [2021, Lemma 1.9] upper-bounds the L1 distance between P and g (equivalent to $2D_{\mathsf{TV}}(P\|g)$) with the following, by separately analyzing the error due to elements (i.e. assignments $\mathbf{x}$) occurring more than $m^{1/3}$ times and the error due to less frequently occurring elements:

$$\sum_{\mathbf{x}:\mathit{freq}(\mathbf{x})\geq m^{1/3}}\frac{\sqrt{\mathit{freq}(\mathbf{x})}}{m}\mathit{polylog}(m) + \sum_{i\in\{1,\ldots m^{1/3}\}}\frac{1}{m}\sqrt{1+\mathbb{E}[F_i]}\mathit{polylog}(m).$$

Poissonization is a common technique for analyzing the histograms of Good-Turing estimation [Valiant and Valiant, 2015]. First, the problem of frequency estimation can be modeled as taking m balls and having n bins for them to go into; the number of balls in the i^{th} bin corresponds to the number of domain elements seen i times in our dataset. Thus, $F_i \sim Binom(m, p_i)$. Now, poissonization allows us to approximate this random variable F_i as $Poisson(\lambda = mp_i)$. As such, we treat F_i to be

approximately Poisson distributed, which allows us to treat each F_i as independent, and thus use a Poisson tail bound from [Cannone, 2017]: $\Pr[|F_i - \mathbb{E}[F_i]| \geq \mathbf{x}] \leq \exp(\frac{-\mathbf{x}^2}{2(\lambda+\mathbf{x})})$. Thus, for a $c > 0$,

$$\Pr\left(|F_i - \mathbb{E}[F_i]| \geq c\sqrt{1+\mathbb{E}[F_i]}\right) \leq \exp\left(\frac{-c^2(1+\mathbb{E}[F_i])}{2(\mathbb{E}[F_i]+c\sqrt{1+\mathbb{E}[F_i]})}\right).$$

To guarantee that the above probability is at most δ for all $|\mathbf{M}| = m$ samples, we bound it by δ/m

$$c^2(1+\mathbb{E}[F_i]) - 2\log(\frac{m}{\delta})\mathbb{E}[F_i] - 2c\log(\frac{m}{\delta})\sqrt{1+\mathbb{E}[F_i]} \geq 0$$

After some algebraic manipulation, we can show that the above holds if:

$$c \geq \frac{2\log(\frac{m}{\delta})}{\sqrt{1+\mathbb{E}[F_i]}} + \sqrt{2\log(\frac{m}{\delta})\frac{\mathbb{E}[F_i]}{1+\mathbb{E}[F_i]}}.$$

For instance, this is satisfied by $c = \frac{2\log(\frac{m}{\delta})}{\sqrt{1+\mathbb{E}[F_i]}} + \sqrt{2\log(\frac{m}{\delta})}$, which we *approximate* for a given set of samples by:

$$\frac{2\log(\frac{m}{\delta})}{\sqrt{1+F_i}} + \sqrt{2\log(\frac{m}{\delta})}.$$

Finally, we convert the bound on L1 distance into a bound on the total variation distance by dividing by 2 to get:

$$g_\epsilon = \frac{1}{2m}\left(\frac{2\log(\frac{m}{\delta})}{\sqrt{1+F_i}} + \sqrt{2\log(\frac{m}{\delta})}\right)\left(\sum_{\mathbf{x}:\mathit{freq}(\mathbf{x})>m^{1/3}} \sqrt{\mathit{freq}(\mathbf{x})} + \sum_{i=1}^{\lceil m^{1/3}\rceil} \sqrt{1+F_i}\right).$$

□

D ADDITIONAL DETAILS FOR INSTANCE-SPECIFIC BOUNDS

Algorithm D.1 Good-Turing Denoising Estimator

1: **Input:** A set of m independent draws from an unknown distribution.
2: **Output:** A labeled vector of probabilities, g.
3: For each $i \geq m^{1/3}$, for every domain element observed exactly i times in the m samples, assign probability i/m.
4: For each $i < m^{1/3}$, assign probability $\frac{i+1}{m-i}\frac{F_{i+1}}{F_i}$ to each of the F_i elements observed exactly i times, where F_j denotes the number of domain elements observed exactly j times in the m samples.

We compute our proposed bound for $\delta = 0.1$ on the train and test sets of density estimation benchmark datasets. Tables 1 and 2 show that we are able to achieve a number of non-trivial results, with the majority of mass from the unseen elements, and the bound improving as the number of samples grows larger. Interestingly, see also that the final two terms of our upper bound, $\frac{1}{2}(1 - \sum_{\mathbf{x}\in\mathbf{M}} \hat{P}(\mathbf{x})) + \frac{1}{2}\sum_{\mathbf{x}\in\mathbf{M}} \left|g(\mathbf{x}) - \hat{P}(\mathbf{x})\right|$ are an upper bound on the irreducible error for the choice of circuit structure and dataset pair.

Dataset	# Vars	m	S	$g_\epsilon^\downarrow$	$g \notin \mathbf{M}^\downarrow$	$\hat{P} \notin \mathbf{M}^\mapsto$	$D_{\mathsf{TV}}(g\|\hat{P})^\mapsto$	Structural Error	TVD Bound
accidents	111	12758	1612	0.072760	0.499843	0.499997	0.000312	0.500310	1.072913
ad	1556	2461	21332	0.402369	0.263714	0.415625	0.217823	0.633448	1.299531
baudio	100	15000	1748	0.067885	0.498133	0.499825	0.001959	0.501784	1.067803
bbc	1058	1670	15332	0.214602	0.449102	0.500000	0.101827	0.601827	1.265531
bnetflix	100	15000	1652	0.066552	0.499900	0.500000	0.500000	1.000000	1.566452
book	500	8700	7172	0.097033	0.474368	0.494330	0.023415	0.517744	1.089145
c20ng	910	11293	12052	0.079834	0.495528	0.500000	0.004738	0.504738	1.080100
connect4	126	16000	1780	0.064770	0.500000	0.496494	0.496494	0.992987	1.557757
cr52	889	6532	11612	0.111057	0.474127	0.499989	0.027245	0.527234	1.112418
cwebkb	839	2803	11900	0.181791	0.487335	0.500000	0.013203	0.513203	1.182329
dna	180	1600	3268	0.220691	0.467500	0.500000	0.033774	0.533774	1.221966
jester	100	9000	1636	0.094808	0.496500	0.499976	0.003771	0.503747	1.095055
kdd	64	180092	1108	0.168850	0.015034	0.017610	0.077322	0.094932	**0.278816**
kosarek	190	33375	2580	0.390466	0.148539	0.161051	0.101616	0.262667	**0.801672**
moviereview	1001	1600	15020	0.206633	0.498125	0.500000	0.003751	0.503751	1.208509
msnbc	17	291326	236	0.429670	0.007188	0.084884	0.258789	0.343673	**0.780531**
mushrooms	112	2000	1652	0.183440	0.500000	0.498316	0.498316	0.996632	1.680072
nips	500	400	8212	0.410135	0.500000	0.500000	0.500000	1.000000	1.910135
nltcs	16	16181	260	0.621526	0.046845	0.060453	0.168531	0.228984	**0.897356**
ocrletters	128	32152	2820	0.062890	0.470266	0.499997	0.030559	0.530556	1.063712
plants	69	17412	1324	0.454917	0.192166	0.403076	0.274294	0.677371	1.324454
pumsbstar	163	12262	2396	0.080820	0.483812	0.499536	0.016081	0.515618	1.080249
rcv1	150	40000	1972	0.062701	0.442688	0.499810	0.057836	0.557646	1.063034
tmovie	500	4524	7876	0.277771	0.429929	0.456490	0.070548	0.527038	1.234738
tretail	135	22041	1948	0.306068	0.209836	0.220528	0.109755	0.330283	**0.846187**
voting	1359	1214	16572	0.380127	0.366557	0.500000	0.134268	0.634268	1.380952

Table 1: Upper bounds w.r.t. the **train** set for PCs learned on density estimation benchmark datasets. $\downarrow$ indicates the error goes to 0 as $m \to \infty$. $\mapsto$ indicates the error depends on the structure.

Dataset	# Vars	m	S	$g_\epsilon^\downarrow$	$g \notin \mathbf{M}^\downarrow$	$\hat{P} \notin \mathbf{M}^\mapsto$	$D_{\mathsf{TV}}(g\|\hat{P})^\mapsto$	Structural Error	TVD Bound
accidents	111	2551	1612	0.162109	0.500000	0.500000	0.500000	0.999999	1.662108
ad	1556	491	21332	0.736581	0.344196	0.429628	0.129755	0.559383	1.640160
baudio	100	3000	1748	0.151299	0.499667	0.499969	0.000664	0.500633	1.151599
bbc	1058	330	15332	0.440394	0.490909	0.500000	0.018209	0.518209	1.449512
bnetflix	100	3000	1652	0.150700	0.500000	0.500000	0.500000	1.000000	1.650700
book	500	1739	7172	0.208110	0.491662	0.498320	0.014698	0.513018	1.212790
c20ng	910	3764	12052	0.137944	0.495882	0.500000	0.004651	0.504651	1.138477
connect4	126	47557	1780	0.037106	0.500000	0.489770	0.489770	0.979539	1.516645
cr52	889	1540	11612	0.222104	0.479545	0.499998	0.023390	0.523388	1.225037
cwebkb	839	838	11900	0.323785	0.494033	0.500000	0.500000	1.000000	1.817818
dna	180	1186	3268	0.253224	0.476813	0.500000	0.029536	0.529536	1.259572
jester	100	4116	1636	0.131619	0.497206	0.499981	0.003166	0.503147	1.131972
kdd	64	34955	1108	0.258365	0.022271	0.024343	0.058497	0.082840	**0.363477**
kosarek	190	6675	2580	0.531244	0.181873	0.196713	0.108128	0.304841	1.017958
moviereview	1001	250	15020	0.517577	0.500000	0.500000	0.500000	1.000000	2.017577
msnbc	17	58265	236	0.624807	0.018905	0.102597	0.259012	0.361609	1.005321
mushrooms	112	5624	1652	0.108739	0.500000	0.495189	0.495189	0.990377	1.599116
nips	500	1240	8212	0.232223	0.499194	0.500000	0.001614	0.501614	1.233030
nltcs	16	3236	260	0.721158	0.108004	0.118302	0.183459	0.301761	1.130923
ocrletters	128	10000	2820	0.103859	0.479000	0.499998	0.024104	0.524102	1.106961
plants	69	3482	1324	0.515260	0.264647	0.433226	0.208842	0.642068	1.421975
pumsbstar	163	2452	2396	0.169776	0.496737	0.499852	0.006444	0.506297	1.172810
rcv1	150	150000	1972	0.055217	0.414840	0.499675	0.085301	0.584977	1.055033
tmovie	500	591	7876	0.436189	0.445854	0.467218	0.063813	0.531031	1.413074
tretail	135	4408	1948	0.422726	0.261230	0.268193	0.101758	0.369951	1.053906
voting	1359	350	16572	0.683729	0.372857	0.500000	0.500000	1.000000	2.056586

Table 2: Upper bounds w.r.t. the **test** set for PCs learned on density estimation benchmark datasets. $\downarrow$ indicates the error goes to 0 as $m \to \infty$. $\mapsto$ indicates the error depends on the structure.