

Statistical Guarantees for Probabilistic Circuit Parameter Learning

John Leland¹ YooJung Choi¹

¹Arizona State University

Motivation

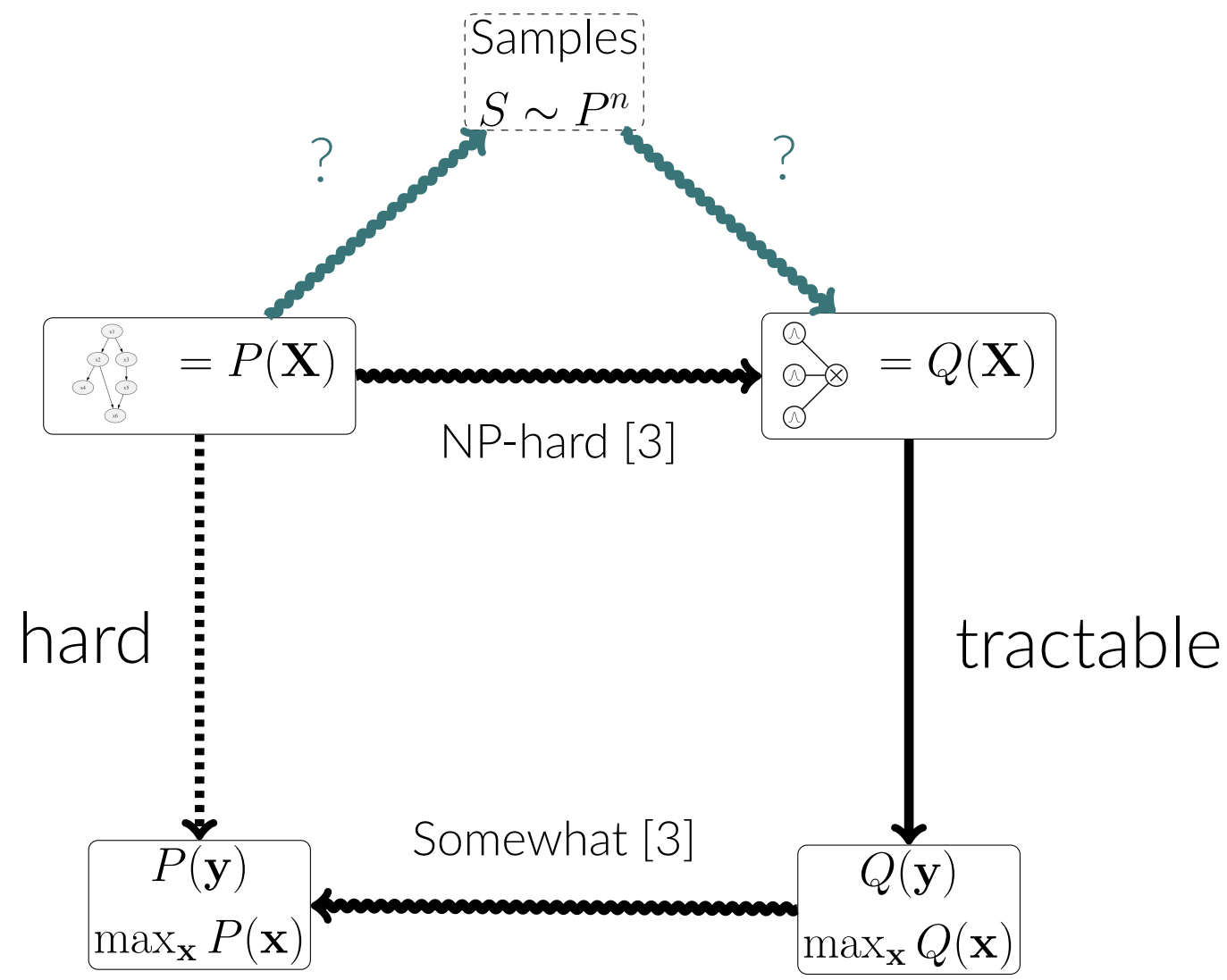


Figure: The current state of PC Approximation.

- Theoretical expressivity of probabilistic circuits (PCs) in exact and approximate case has been studied (to varying degrees).
- However, these results not address the setting of learning PCs from data.
- Can we learn to approximate (closely) with high probability?

Question 1

How does the number of samples needed to learn a PC relate to its size?

Setup

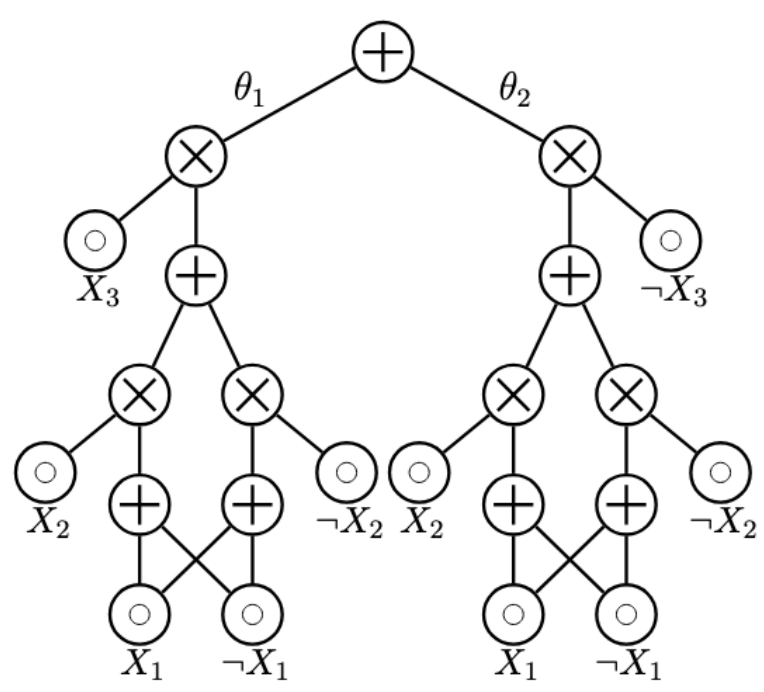


Figure: A smooth, decomposable, and deterministic PC

- Probabilistic Circuits (PCs) are directed acyclic graphs that recursively define probability distributions by the composition of sum $\oplus$, product $\otimes$, and leaf nodes. [1].
- Circuits impose structural constraints to access to more queries. Two powerful constraints are *decomposability* and *determinism*.
 - Enforcing decomposability guarantees linear-time marginal inference in PCs.
 - Enforcing both decomposability and determinism guarantees linear-time maximum-a-posteriori (MAP) inference.
- Sample Complexity Background
 - Given samples from a true distribution P
 - Want to learn a PC, $\hat{P}$, from these samples.
 - With enough samples can we guarantee $\Pr[D(\hat{P}||P) \leq \epsilon] > 1 - \delta$?

Upper Bounds: **There exists** a algorithm such that given at least M samples guarantees

$$\Pr[D(\hat{P}||P) \leq \epsilon] > 1 - \delta.$$

Lower Bounds: **Any algorithm** must have at least M samples to guarantee

$$\Pr[D(\hat{P}||P) \leq \epsilon] > 1 - \delta.$$

Lower Bounds for Learning PCs

Theorem 1 (Shortened Version)

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 $D_H(\hat{P}||P) \leq \epsilon$ with probability $> 1 - \delta$ requires $M = \Omega\left(\frac{S + \log(1/\delta)}{\epsilon^2}\right)$.

- This holds **even if** P is in a family of **deterministic PCs of a fixed structure $\mathcal{G}$** with S parameters, **and $\mathcal{G}$ is known to A .**
- The proof also shows $D_{TV}(\hat{P}||P) \leq \epsilon$ or $D_{KL}(\hat{P}||P) \leq \epsilon$ with prob. $1 - \delta$ and requires, $M = \Omega\left(\frac{S + \log(1/\delta)}{\epsilon}\right)$.

Proof Intuition 1

- ♦ Construct a family of deterministic PCs by making many different “choices”, by perturbing an initial distribution slightly with each choice.
- ♦ This family will include a large set of distributions that are difficult to tell apart by any learning algorithm without using many samples. $M = \Omega\left(\frac{S}{\epsilon^2}\right)$ (Assouad's Lemma)
- ♦ Suppose we have two distributions in this family that differ by only one perturbation.
- ♦ This perturbation can be mapped to the task of learning the bias a fair coin. $M = \Omega\left(\frac{\log(1/\delta)}{\epsilon^2}\right)$.

Answer 1

Thus, learning within Hellinger distance ϵ with probability $1 - \delta$ requires at least linearly many samples as the number of PC parameters.

- Why does the lower bound construction only use deterministic PCs?
 1. The output of the learning algorithm is *unrestricted*.
 2. Sample complexity considers worst-case distributions, thus a lower bound on a subset is also a lower bound on the sample complexity of the larger family.
- This result applies to sum-product networks [6], arithmetic circuits [2], sum-of-squares circuits [4], InceptionPCs [8], probabilistic generating circuits [9], and more.

Tight Bounds for PAC Learning

Question 2

When can a parameter learning algorithm attain the above lower bound?

- Learning a PC seems like learning a categorical variable for each sum node with the latent variable interpretation of PCs [5].
- Can we then just take the sample complexity of each sum node individually?
- Unfortunately no, for two reasons:
 1. Sum nodes have different probabilities of being reached, and not every sample reaches every sum.
 2. Non-deterministic PCs route samples across potentially multiple different categories.

Theorem 2 (Shortened Version)

Let $\mathcal{C}_G$ be a family of distributions represented by PCs of a fixed structure $\mathcal{G}$ of size S (not necessarily deterministic). For $\epsilon > 0$ and $\delta \in (0, 1]$, the closed-form estimator given M samples from $P \in \mathcal{C}_G$ outputs $\hat{P}$ such that $D_{KL}(\hat{P}||P) \leq \epsilon$ with probability $> 1 - \delta$ if: $M = O\left(\frac{S + \log(1/\delta)}{\epsilon}\right)$.

- Also, $D_H(\hat{P}||P) \leq \epsilon$ and $D_{TV}(\hat{P}||P) \leq \epsilon$ with prob. $> 1 - \delta$ if: $M = O\left(\frac{S + \log(1/\delta)}{\epsilon^2}\right)$.
- 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 $\mathbf{X} \cup \mathbf{Z}$. Denote $\mathbf{Z}$ as the set of latent variables.
- This closed-form estimator is for non-deterministic PCs
 - Note this is **not guaranteed** to achieve MLE.
- Estimates can be collected in polytime.

Proof Intuition 2

- ♦ A PC is deterministic on an instantiation of $\mathbf{Z} = \mathbf{z}$
- ♦ P and $\hat{P}$ only differ in parameters, so the KL divergence between is a weighted sum depending on the KL divergence **between parameters**.
- ♦ Consider $D_{KL}(\hat{P}||P)$ as a random variable, and take its moment generating function (mgf).
- ♦ Use the law of total expectation to condition on each node getting m_n samples.
- ♦ This allows us to upper bound the mgf of each sum node and take a Chernoff bound.

Answer 2

Our necessary number of samples is in fact sufficient to PAC learn the parameters of a given PC structure for Hellinger and KLD.

Relaxing Structural Assumptions

Theorem 3

Let g be the GT 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 $1 - \delta$, $D_{TV}(P||\hat{P})$ is at most:

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

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

- What if the true distribution isn't a PC?
- Approximate P with the Good-Turing (GT) estimator [7].
- Bound is computable for a pair of dataset and a learned PC, with respective sizes M and S , in time $O(M \cdot S)$.
- Experimentally, the bound estimates are non-trivial given enough data. (See paper for more details)

References

- [1] Y Choi et al. Probabilistic circuits: A unifying framework for tractable probabilistic models. *UCLA*, 2020.
- [2] A Darwiche. A differential approach to inference in bayesian networks. *J. ACM*, 2003.
- [3] Leland et al. On the hardness of approximating distributions with tractable probabilistic models. *NeurIPS*, 2026.
- [4] L Loconte et al. Sum of squares circuits. In *AAAI*, 2025.
- [5] Robert Peharz et al. On the latent variable interpretation in sum-product networks. *IEEE*, 2016.
- [6] H Poon et al. Sum-product networks: A new deep architecture. In *ICCV Workshops*, 2011.
- [7] Valiant and Valiant. *Instance Optimal Distribution Testing and Learning*. 2021.
- [8] B Wang et al. On the relationship between monotone and squared probabilistic circuits. In *AAAI*, 2025.
- [9] H Zhang et al. Probabilistic generating circuits. In *ICML*, 2021.