

---

# On the Hardness of Approximating Distributions with Tractable Probabilistic Models

---

**John Leland**    **YooJung Choi**  
School of Computing and Augmented Intelligence  
Arizona State University  
jslelan1@asu.edu, yj.choi@asu.edu

## Abstract

A fundamental challenge in probabilistic modeling is to balance expressivity and inference efficiency. Tractable probabilistic models (TPMs) aim to directly address this tradeoff by imposing constraints that guarantee efficient inference of certain queries while maintaining expressivity. In particular, probabilistic circuits (PCs) provide a unifying framework for many TPMs, by characterizing families of models as circuits satisfying different structural properties. Because the complexity of inference on PCs is a function of the circuit size, understanding the size requirements of different families of PCs is fundamental in mapping the trade-off between tractability and expressive efficiency. However, the study of expressive efficiency of circuits are often concerned with exact representations, which may not align with model learning, where we look to approximate the underlying data distribution closely by some distance measure. Moreover, due to hardness of inference tasks, exactly representing distributions while supporting tractable inference often incurs exponential size blow-ups. In this paper, we consider a natural, yet so far under-explored, question: *can we avoid such size blow-up by allowing for some small approximation error?* We study approximating distributions with probabilistic circuits with guarantees based on  $f$ -divergences, and analyze which inference queries remain well-approximated under this framework. We show that approximating an arbitrary distribution with bounded  $f$ -divergence is NP-hard for any model that can tractably compute marginals. In addition, we prove an exponential size gap for approximation between the class of decomposable PCs and that of decomposable and deterministic PCs.

## 1 Introduction

The expressive power of probabilistic and generative models has increased rapidly in recent years: from Bayesian networks [18], GANs [26], VAEs [29], and normalizing flows [35], to diffusion models [27] and transformers [46] have demonstrated remarkable success in capturing complex distributions. Yet, despite their expressivity, many of these models do not support efficient computation of fundamental probabilistic queries, such as marginals and conditionals, which are critical for inference in domains such as healthcare [40], neuro-symbolic AI [43], environmental science [5], and algorithmic fairness [9].

Tractable probabilistic models—such as probabilistic circuits [8], probabilistic generating circuits [50], determinantal point processes [1], and more—address the need for probabilistic queries by balancing expressivity and tractable inference, achieved through enforcing constraints on the models. To more compactly describe the constraints enforced, we focus on probabilistic circuits, which also have extensive literature on the conditions of tractability [8]. While these structural constraints enable efficient inference, they introduce a tradeoff by potentially affecting the models’ ability to compactly represent distributions (i.e., their expressive power). Naturally, there have been many

works characterizing the expressive efficiency of different circuit classes [19, 2, 8, 48, 49, 20]—i.e., their ability to compactly and *exactly* represent certain classes of functions or distributions. However, the approximate case—how structural constraints affect the ability to *approximately represent* distributions, remains comparatively underexplored. This motivates a shift in focus from exact to approximate modeling: representing a distribution approximately within a small distance under some metric.

This shift raises a fundamental question: *does allowing a small approximation error alleviate the exponential separation between circuit classes observed in exact modeling, or does hardness persist even in the approximate setting?* Our motivation for this study is two-fold. (1) In learning probabilistic circuits from finite data, often the goal is not necessarily to exactly represent some known distribution but rather to approximate it as closely as possible with a PC of reasonable size. Thus, showing that certain distributions cannot be approximated within a bounded distance by a compact PC satisfying some structural properties implies that any learning algorithm whose hypothesis space is that family of PCs would fail to learn the distribution with a bounded approximation error. (2) Moreover, probabilistic circuits can also be used to perform inference on other probabilistic models (such as Bayesian networks or probabilistic programs) by compiling them into PCs then running efficient inference on the compiled circuits [4, 28, 16, 7, 25]. This suggests the following approximate inference scheme: *approximately* compile a probabilistic model into a PC then run efficient *exact* inference on the approximately compiled PC. Moreover, if we could bound the distance between the target distribution and approximate model, we can hope to provide guarantees on the approximate inference results as well.

Our main contributions are as follows: (1) we prove that it is NP-hard to approximate distributions within a bounded  $f$ -divergence using *any model that supports tractable marginals*, with proof via a reduction from SAT (Theorem 3.4 and 3.5); (2) we derive an unconditional, exponential separation between decomposable PCs and decomposable & deterministic PCs for approximate modeling (Theorem 4.1); (3) we study the relationship between bounds on divergence measures for approximate modeling and approximation errors for marginal and maximum-a-posteriori (MAP) inference, characterizing when one is or is not sufficient to guarantee the other (Sections 3.1 and 5).

## 2 Preliminaries

**Notations** We use uppercase letters ( $X$ ) to denote random variables and lowercase letters ( $x$ ) to denote assignments to these random variables. Sets of random variables and assignments are denoted using bold letters ( $\mathbf{X}$  and  $\mathbf{x}$ ). The *accepting models* (i.e., satisfying assignments) of a Boolean function  $f : \{0, 1\}^n \rightarrow \{0, 1\}$  over  $n$  variables is denoted by  $f^{-1}(1)$ . The number of accepting models of  $f$  is referred to as  $\text{MC}(f)$ , a shorthand for its *model count*. Moreover, a Boolean function  $f$  is the *support* of a distribution  $P$ , if  $P$  is non-zero only over the models of  $f$ .

### 2.1 Probabilistic Circuits

Probabilistic circuits (PCs) [8] provide a unifying framework for a wide class of tractable probabilistic models, including arithmetic circuits [15], sum-product Networks [37], cutset networks [38], probabilistic sentential decision diagrams [30], and bounded-treewidth graphical models [24, 11].

**Definition 2.1** (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  $\theta$ . 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  $\text{in}(n)$ . 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  $n$ ,  $\phi(n) = \cup_{c \in \text{in}(n)} \phi(c)$ , and the scope of a leaf node is a single variable in  $\mathbf{X}$ . Naturally, the scope of the root node is  $\mathbf{X}$ . Each node  $n$  of a PC is then recursively defined as:

$$p_n(\mathbf{x}) := \begin{cases} l(x), & \text{if } n \text{ is a leaf} \\ \prod_{c \in \text{in}(n)} p_c(\mathbf{x}) & \text{if } n \text{ is a } \otimes \\ \sum_{c \in \text{in}(n)} \theta_{n,c} p_c(\mathbf{x}) & \text{if } n \text{ is a } \oplus \end{cases} \quad (1)$$



**Definition 2.4** (*f*-divergence [36]). Let  $f : (0, \infty) \rightarrow \mathbb{R}$  be a convex function with  $f(1) = 0$ , and  $P, Q$  be two probability distributions over a set of Boolean variables  $\mathbf{X}$ . If  $Q > 0$  wherever  $P > 0$ , the *f*-divergence between  $P$  and  $Q$  is defined as  $D_f(P||Q) = \sum_{\mathbf{x}} Q(\mathbf{x})f(\frac{P(\mathbf{x})}{Q(\mathbf{x})})$ .

Commonly used *f*-divergences include the Kullback-Leibler divergence,  $\chi^2$ -divergence, and total variation distance. The total variation distance is especially relevant to our results.

**Definition 2.5** (Total variation distance). The *total variation distance (TVD)* between two probability distributions  $P$  and  $Q$  over a set of  $n$  Boolean variables  $\mathbf{X}$  is defined as  $D_{\text{TV}}(P||Q) = \frac{1}{2} \sum_{\mathbf{x} \in \mathbf{X}} |P(\mathbf{x}) - Q(\mathbf{x})|$ , or equivalently  $D_{\text{TV}}(P||Q) = \max_{S \subseteq \{0,1\}^n} |P(S) - Q(S)|$ .

We introduce the following notion to describe probabilistic models that approximate distributions within some bounded distance.

**Definition 2.6** ( $\epsilon$ -*D*-Approximation). Let  $P, Q$  be two probability distributions and  $D$  be a distance measure between distributions. We say that  $Q$  is an  $\epsilon$ -*D*-approximator of  $P$  if  $D(P||Q) < \epsilon$  for some  $\epsilon > 0$ .

For instance, we refer to a probabilistic circuit  $Q$  that approximates our target distribution  $P$  such that  $D_{\text{TV}}(P||Q) < \epsilon$  a  $\epsilon$ - $D_{\text{TV}}$ -approximator. The majority of our results are derived using properties of the total variation distance, due to its nice properties as a distance metric. To extend our results to other *f*-divergences, we utilize the following class, which provides an upper bound on the total variation distance.

**Definition 2.7** (*k*-convex *f*-divergence [34]). A  $\mathbb{R} \cup \{\infty\}$ -valued function  $f$  on a convex set  $K \subseteq \mathbb{R}$  is *k*-convex if  $x, y \in K$  and  $t \in [0, 1]$  implies  $f((1-t)x+ty) \leq (1-t)f(x)+tf(y)-kt(1-t)\frac{(x-y)^2}{2}$ . When  $f$  is twice differentiable, this is equivalent to  $f''(x) \geq k$  for all  $x \in K$ . In the case that  $k = 0$  this reduces to the normal notion of convexity. An *f*-divergence  $D_f$  is *k*-convex over an interval  $K$  for  $k \geq 0$  if the function  $f$  is *k*-convex on  $K$ .

We provide a table in Appendix A.1 summarizing which *f*-divergence measures are *k*-convex and for which value of  $k$ . Throughout this paper, we express approximation bounds using *k*-convex *f*-divergences as they naturally encapsulate bounds on many common distance measures. For instance, for any *k*-convex *f*-divergence between  $P$  and  $Q$ , we have that  $D_{\text{TV}}(P||Q)^2 < D_f(P||Q)/k$  [34]. As we will see later, the bounds on the TV distance can naturally be connected to guarantees for approximation inference. For KL-divergence, which is the most commonly used objective for learning probabilistic models, we can use Pinsker’s inequality [45] to obtain  $D_{\text{TV}}(P||Q) < \sqrt{\frac{1}{2}D_{\text{KL}}(P||Q)}$ .

### 3 Approximate Modeling with Tractable Marginals is NP-hard

Most works characterizing the expressive efficiency of different circuit classes have been concerned with *exact* representations [19, 2, 8, 48, 49, 20]. While Chubarian and Turán [12] and De Colnet and Mengel [21] have recently studied the ability (and hardness) of logical circuit classes to compactly *approximate* Boolean functions, to the best of our knowledge, our results are the first to show hardness of *compactly approximating probability distributions* using different families of tractable PCs.

As discussed previously, the complexity of approximately modeling distributions with PCs is valuable for understanding: (1) potential limitations in the hypothesis space of PC learning algorithms, and (2) the feasibility of approximate inference with guarantees through approximate compilation. This section aims to answer this focusing on probabilistic models that are tractable for marginal queries. We first show that a form of approximate marginal inference using this scheme requires a non-trivial bound on the total variation distance between the target distribution and the approximate model, and next prove that finding such an approximator is NP-hard.

#### 3.1 Relative Approximation of Marginals

We consider *relative approximation*<sup>4</sup> of marginal queries. Let  $P(\mathbf{X})$  be a probability distribution over a set of variables  $\mathbf{X}$ . Then we say another distribution  $Q(\mathbf{X})$  is a *relative approximator* of

<sup>4</sup>Also called multiplicative approximation or approximation within a factor.

marginals of  $P$  w.r.t.  $0 \leq \epsilon \leq 1$  if:  $\frac{1}{1+\epsilon} \leq \frac{P(\mathbf{y})}{Q(\mathbf{y})} \leq 1 + \epsilon$  for every assignment  $\mathbf{y}$  to subset of variables  $Y \subseteq \mathbf{X}$ . Relative approximation is often considered for approximate inference of graphical models and the closely related approximate (weighted) model counting [23, 3]. We first show that relative approximation for all marginal queries implies a non-trivially bounded total variation distance.

**Theorem 3.1** (Relative approximation implies bounded  $D_{\text{TV}}(P\|Q)$ ). *Let  $\epsilon > 0$  and  $P, Q$  be two probability distributions over  $\mathbf{X}$ . If  $Q$  is a relative approximator of marginals for  $P$ , then  $D_{\text{TV}}(P\|Q) \leq \frac{\epsilon}{2}$ .*

*Proof.* As  $Q$  is a relative approximator of  $P$ , for all assignment  $\mathbf{x}$  we have that  $\frac{1}{1+\epsilon} \leq \frac{P(\mathbf{x})}{Q(\mathbf{x})} \leq 1 + \epsilon$  which implies  $|P(\mathbf{x}) - Q(\mathbf{x})| \leq \epsilon \min(P(\mathbf{x}), Q(\mathbf{x}))$ . Therefore,  $D_{\text{TV}}(P\|Q) = \frac{1}{2} \sum_{\mathbf{x}} |P(\mathbf{x}) - Q(\mathbf{x})| \leq \frac{1}{2} \sum_{\mathbf{x}} \epsilon \min(P(\mathbf{x}), Q(\mathbf{x})) \leq \frac{\epsilon}{2}$ .  $\square$

In other words,  $D_{\text{TV}}(P\|Q) \leq \epsilon/2$  is a necessary condition for  $Q$  to be a relative approximator of marginals of  $P$  w.r.t.  $\epsilon$ . However, it is still not a sufficient condition as shown below.

**Proposition 3.2** (Bounded  $D_{\text{TV}}(P\|Q)$  does not imply relative approximation). *There exists a family of distributions  $P$  that have  $\epsilon$ - $D_{\text{TV}}$ -approximators, yet for any such approximator  $Q$ , the relative approximation error of marginals between  $P$  and  $Q$  can be arbitrarily large.*

We prove the above proposition by explicitly constructing a family of distributions  $\mathcal{Q}$  such that every  $Q \in \mathcal{Q}$  is an  $\epsilon$ - $D_f$ -approximator for any arbitrary  $\epsilon > 0$  and distribution  $P$  yet  $P(\mathbf{x})/Q(\mathbf{x})$  can be arbitrarily large for some  $\mathbf{x}$ . See Appendix A.3.1 for the full construction.

It is known that relative approximation of marginals is NP-hard for Bayesian networks [14].<sup>5</sup> Thus, it immediately follows that approximately representing arbitrary distributions using polynomial-sized PCs tractable for marginals (e.g., decomposable PCs) such that the PC is a relative approximator of all marginals is also NP-hard. However, approximating *all* marginal queries is quite a strong condition, and we may still want to closely approximate distributions as they could be useful in approximating *some* marginal queries. In particular, because approximating a distribution  $P$  with a bounded TV distance is a necessary but not sufficient condition for the NP-hard problem of relative approximation of marginals, this raises the question whether it is still possible to efficiently approximate the distribution  $P$  with a compact PC  $Q$  that is tractable for marginals. Unfortunately, we next answer this in the negative.

### 3.2 Hardness of Approximating Distributions using Tractable Models for Marginals

We consider approximating potentially *unnormalized* distributions, which can be considered a generalization of probability distributions by omission of the normalizing constant.

**Definition 3.3** (Unnormalized Distributions). Any (unnormalized) distribution  $\hat{P} : \text{val}(\mathbf{X}) \rightarrow \mathbb{R}$  must satisfy the following: (1)  $\hat{P}(\mathbf{x}) \geq 0$  for any  $\mathbf{x}$ ; (2) The normalization constant  $Z = \sum_{\mathbf{x} \in \mathbf{X}} \hat{P}(\mathbf{x})$  is well-defined and finite.

See from this definition that an unnormalized distribution can easily be converted to a probability distribution  $P$  if  $Z$  is computable in polynomial time:  $P(\mathbf{x}) = \hat{P}(\mathbf{x})/Z$ . Unnormalized distributions are relevant to the goal of tractable approximation, as many probabilistic models that we may want to approximate—including factor graphs [32] and energy-based models [44]—represent unnormalized distributions.

**Theorem 3.4** (Hardness of  $D_f$ -approximation). *Given a (potentially unnormalized) probability distribution  $\hat{P}$  and a  $k$ -convex  $f$ -divergence  $D_f$ , for any  $0 < \epsilon < \frac{1}{4}$ , it is NP-hard to represent the  $k\epsilon^2$ - $D_f$ -approximation of its normalized distribution  $P$  as a model that can tractably compute marginals.*

*Proof.* We will prove the above using a reduction from SAT. Let  $\hat{P}$  be a Boolean formula over  $\mathbf{X} = \{X_1, \dots, X_n\}$  and  $\epsilon < \frac{1}{4}$ . See that  $\hat{P}$  satisfies all requirements to be considered an unnormalized

<sup>5</sup>In fact, there also exists no randomized polynomial time algorithm for relative approximation of marginals unless  $\text{RP} = \text{NP}$  [14].

probability distribution. We then define a new Boolean formula  $\hat{P}'$  over  $\mathbf{X}$  and an auxiliary variable  $Y$ :  $\hat{P}' = (Y \wedge \hat{P}) \vee (\neg Y \wedge X_1 \wedge \dots \wedge X_n)$ . Clearly  $\hat{P}'$  has  $\text{MC}(\hat{P}) + 1$  models. Let us now define a uniform distribution  $P$  over these models of  $\hat{P}'$  (i.e., by normalizing  $\hat{P}'$ ). Suppose that we can efficiently obtain a probability distribution  $Q$  such that  $D_f(P||Q) < k\epsilon^2$ , which in turn implies that  $D_{\text{TV}}(P||Q) < \epsilon$ . From the definition of total variation distance,  $|P(Y = 1) - Q(Y = 1)| < \epsilon < \frac{1}{4}$ .

By construction, if  $\hat{P}$  is unsatisfiable, there is no satisfying model of  $\hat{P}'$  such that  $Y = 1$ , and thus  $P(Y = 1) = 0$  and  $Q(Y = 1) < \frac{1}{4}$ . Otherwise, if  $\hat{P}$  is satisfiable, then there are  $\text{MC}(\hat{P})$  many satisfying model of  $\hat{P}'$  setting  $Y = 1$ , and thus we have  $\text{MC}(\hat{P}) \geq 1$  and  $P(Y = 1) = \frac{\text{MC}(\hat{P})}{1 + \text{MC}(\hat{P})}$ , implying

$$Q(Y = 1) > \frac{\text{MC}(\hat{P})}{1 + \text{MC}(\hat{P})} - \frac{1}{4} \geq \frac{1}{2} - \frac{1}{4} \geq \frac{1}{4}.$$

Therefore,  $\hat{P}$  is satisfiable if and only if  $Q(Y = 1) \geq \frac{1}{4}$ . In other words, we can decide SAT if we can efficiently compute an  $k\epsilon^2$ - $D_f$ -approximation as a model that supports tractable marginals.  $\square$

The following corollary immediately follows from the above proof.

**Corollary 3.5.** *Given a (potentially unnormalized) probability distribution  $\hat{P}$ , for  $0 < \epsilon < \frac{1}{4}$ , it is NP-hard to represent the  $\epsilon$ - $D_{\text{TV}}$ -approximation of its normalized distribution  $P$  as a model that can tractably compute marginals.*

Thus, the class of polynomial-sized probabilistic models supporting tractable marginals cannot contain  $k\epsilon^2$ - $D_f$ -approximations for all distributions unless  $\text{P} = \text{NP}$ . This includes decomposable PCs [8], sum of squares circuits [31], probabilistic generating circuits [50], determinantal point processes [1], Inception PCs [47], and positive unital circuits [22]. Furthermore, by Pinsker's inequality this suggests that there exists distributions for which obtaining a decomposable PC with bounded KL-divergence of  $1/8$  is NP-hard. While Martens and Medabalimi [33] previously showed a related result that there exists a function for which a sequence of decomposable PCs converging to approximate the function arbitrarily well requires an exponential size, our result applies more broadly to any class of models supporting tractable marginals as well as allowing for a bounded *but non-vanishing* approximation error. Thus, it is difficult not only to exactly represent functions or distributions as compact decomposable PCs, but also to approximate within some small distance. Consequently, this presents a major challenge to the idea of using approximate compilation of PCs for approximate inference if the desired error tolerance  $\epsilon$  is sufficiently small.

## 4 Large Deterministic & Decomposable PCs for Approximate Modeling

Continuing our characterization of the approximation power of PCs, we now turn to the family of deterministic and decomposable PCs. We take inspiration from related results for logical circuits. In particular, Bova et al. [2] proved an exponential separation between DNNFs and d-DNNFs: i.e., there is a family of Boolean functions that can be compactly represented by decomposable circuits but requires exponentially sized deterministic and decomposable circuits. Furthermore, De Colnet and Mengel [21] showed that there exist functions that require exponential size to approximate with d-DNNFs under two notions of approximation for Boolean functions.

Nevertheless, this does not immediately imply the same separation for probabilistic circuits due to two key reasons: (1) approximation for PCs is measured in terms of divergences between distributions rather than some probabilistic error between Boolean functions, and (2) our approximator can represent arbitrary distributions instead of being limited to a Boolean function (or a uniform distribution over it). This section presents our proof of exponential separation between the class of decomposable PCs and that of deterministic and decomposable ones, by constructing a family of distributions that can be represented by compact decomposable PCs, but any PC that is also deterministic and approximates it within a bounded TV distance must have an exponential size.

We consider the *Sauerhoff function* [41] which was used to show the separation between DNNFs and d-DNNFs for exact compilation [2]. Let  $g_n : \{0, 1\}^n \rightarrow \{0, 1\}$  be a function evaluating to 1 if and only if the sum of its inputs is divisible by 3. The *Sauerhoff function* is defined as  $S_n :$

$\{0, 1\}^{n^2} \rightarrow \{0, 1\}$  over the  $n \times n$  matrix  $X = (x_{ij})_{1 \leq i, j \leq n}$  such that  $S_n(X) = R_n(X) \vee C_n(X)$ , where  $R_n, C_n : \{0, 1\}^{n^2} \rightarrow \{0, 1\}$  are defined as  $R_n(X) = \bigoplus_{i=1}^n g_n(x_{i,1}, x_{i,2}, \dots, x_{i,n})$  and  $C_n(X) = R_n(X^T)$ . Here,  $\oplus$  represents addition modulo 2.

There exists a DNNF of size  $O(n^2)$  that exactly represents the Sauerhoff function  $S_n$ , constructed as a disjunction of two compact ordered binary decision diagrams (OBDDs)—a more restrictive kind of deterministic and decomposable circuits—that represent  $R_n$  and  $C_n$ , respectively [2, Proposition 7]. We then define our family of target distributions  $P_n$  as follows: let  $\mathcal{C}_n$  be a DNNF for  $S_n$  with size  $O(n^2)$ ; then  $P_n$  is a decomposable PC obtained by replacing the literals of  $\mathcal{C}_n$  with corresponding indicator functions,  $\wedge$  with  $\otimes$ , and  $\vee$  with  $\oplus$  nodes with uniform parameters, followed by smoothing the circuit.<sup>6</sup> Note that  $P_n$  outputs a positive value on an input  $\mathbf{x}$  if and only if  $S_n(\mathbf{x}) = 1$ . We will show that a deterministic and decomposable PC approximating  $P_n$  requires exponential size.

**Theorem 4.1** (Exponential-size deterministic PC). *A deterministic and decomposable PC that is a  $\epsilon$ - $D_{TV}$ -approximator of  $P_n$  for some  $\epsilon \leq \frac{1}{16} - \Omega(1/\text{Poly}(n^2))$  has size  $2^{\Omega(n)}$ .*

We will prove the above by first showing that approximation of  $P_n$  with a deterministic and decomposable PC implies a form of *weak approximation* [21] of  $S_n$  with a d-DNNF of the same size, and next proving that such d-DNNF must be exponentially large.

**Definition 4.2** (Weak approximation [21]). A Boolean formula  $g$  is a weak  $\epsilon$ -approximation of another Boolean formula  $f$  if  $\text{MC}(f \wedge \neg g) + \text{MC}(\neg f \wedge g) \leq \epsilon \cdot 2^n$ .

**Proposition 4.3** (Bounded  $D_{TV}$  implies weak approximation). *Let  $0 \leq \epsilon < \frac{1}{8}$  and  $P$  be a uniform distribution whose support is given by a Boolean function  $f$ . Suppose that  $Q$  is a deterministic and decomposable PC representing an  $\epsilon$ - $D_{TV}$ -approximator of  $P$ . Then there exists a d-DNNF  $g$  which has size polynomial in the size of  $Q$  that represents a  $4\epsilon$ -weak-approximator of  $f$ .*

*Proof.* Suppose that  $P$  is a uniform distribution over the support given by a Boolean function  $f$ , and  $Q$  a probability distribution such that  $D_{TV}(P||Q) < \epsilon < \frac{1}{8}$ . Then, we can construct an (unnormalized) deterministic and decomposable PC  $Q'$  by pruning the edges of  $Q$  such that any assignment  $\mathbf{x}$  is eliminated from the resulting support of  $Q'$  if and only if  $Q(\mathbf{x}) < \frac{1}{2^{n+1}}$ . We call this support  $g$ . We provide this pruning algorithm, which relies on determinism as a key property, in Appendix A.3.2. Clearly, the size of  $Q'$  is at most the size of  $Q$ . We will now briefly summarize how this pruning scheme induces a weak approximation and refer to Appendix A.3.3 for the full derivation. Intuitively, we know that the support of  $Q$  must cover most of the support of  $P$ , as  $P$  is a uniform distribution and  $D_{TV}(P||Q)$  is bounded. While it is possible for the support of  $Q$  to be much larger than the support of  $P$ , then the probability assigned by  $Q$  to assignments outside of the support of  $P$  must also be very small to maintain a small TV distance. Thus, pruning away these assignments with small probability allows us to retrieve  $Q'$  whose support has only a small number of non-overlapping assignments with models of  $f$ . More formally, we know that by our pruning scheme,  $Q(\mathbf{x}) \geq \frac{1}{2^{n+1}}$  on  $g$  and  $Q(\mathbf{x}) < \frac{1}{2^{n+1}}$  on  $\neg g$ . Using this fact, we can lower bound our original total variation distance by  $\frac{1}{2}(\text{MC}(f \wedge \neg g)/2^{n+1} + \text{MC}(\neg f \wedge g)/2^{n+1})$ , which implies that  $\text{MC}(f \wedge \neg g) + \text{MC}(\neg f \wedge g) < 4\epsilon \cdot 2^n$ . Thus,  $g$  is a  $4\epsilon$ -weak-approximator of  $f$ , and we can represent it as a polynomially sized d-DNNF by taking the deterministic and decomposable PC  $Q'$  and converting it to a logical circuit.  $\square$

**Proposition 4.4** (d-DNNF approximating  $S_n$  has exponential size). *A d-DNNF representing a  $(\frac{1}{4} - \Omega(1/\text{Poly}(n^2)))$ -weak-approximation of  $S_n$  has size  $2^{\Omega(n)}$ .*

*Proof.* Let  $\mathcal{C}$  be a d-DNNF such that it is a  $(\frac{1}{4} - \Omega(1/\text{Poly}(n^2)))$ -weak-approximation of  $S_n$ . Sauerhoff [41] showed that any “two-sided” rectangle approximation<sup>7</sup> (which matches the notion of weak approximation) of  $S_n$  within  $\frac{1}{4} - \Omega(1/\text{Poly}(n^2))$  must have size  $2^{\Omega(n)}$ . Bova et al. [2] further showed that a d-DNNF  $\mathcal{C}$  computing a function  $f$  is a balanced rectangle partition of  $f$  with size at most  $|\mathcal{C}|$ . Thus,  $\mathcal{C}$  must have size  $2^{\Omega(n)}$ .  $\square$

We are now ready to prove our main result about exponential size lower bound on deterministic and decomposable PCs as approximators.

<sup>6</sup>Smoothing a decomposable PC takes polynomial (worst-case quadratic) time [8].

<sup>7</sup>See Appendix A.2 for details on rectangle partitions.

*Proof of Theorem 4.1.* Suppose that we have a deterministic and decomposable PC  $Q$  that is an  $\epsilon$ - $D_{\text{TV}}$ -approximator of  $P_n$ , where  $\epsilon = (\frac{1}{16} - \Omega(1/\text{Poly}(n^2)))$ . Consider then the uniform distribution  $U$  over  $S_n$ . Then, by the triangle inequality,  $D_{\text{TV}}(U\|Q) \leq D_{\text{TV}}(U\|P_n) + D_{\text{TV}}(P_n\|Q) < D_{\text{TV}}(U\|P_n) + \epsilon$ . By Bova et al. [2, Proposition 7], we know that the DNNF constructed to represent the Sauerhoff function is a disjunction of two OBDDs, which respectively represent  $R_n, C_n$ . As each OBDD can easily be translated to a d-DNNF with a polynomial size increase, their PC counterparts (OR to  $\oplus$  and AND to  $\otimes$ ) will still represent the same Boolean functions [6]. Thus, the non-deterministic PC representing  $P_n$  based on this construction only has one non-deterministic sum node at the root, and can return values at most 2. This allows us to see that  $D_{\text{TV}}(U\|P) \leq \left| \frac{1}{|S_n|} - \frac{2}{|S_n|+1} \right| < \frac{1}{|S_n|} < \eta$  where  $\eta = \frac{1}{(1-1/\sqrt{2})2^{n^2}}$  based on the fact that  $|S_n| > (1-\beta)2^{n^2}$  for  $\beta < 1/\sqrt{2}$ , derived from the low 0-density property of  $S_n$  under the uniform distribution [41]. Therefore,  $D_{\text{TV}}(U\|Q) < \epsilon + \eta$ . Since, the  $\Omega(1/\text{Poly}(n^2))$  term in  $\epsilon$  subsumes  $\eta$ , we can more simply say  $D_{\text{TV}}(U\|Q) < \frac{1}{16} - \Omega(1/\text{Poly}(n^2))$ . Next, we construct a d-DNNF  $C'$  from  $Q$  by replacing indicators with literals,  $\oplus$  with  $\vee$ , and  $\otimes$  with  $\wedge$ . By Proposition 4.3,  $C'$  is a  $(\frac{1}{4} - \Omega(1/\text{Poly}(n^2)))$ -weak-approximation of  $S_n$ . Thus, by Proposition 4.4,  $|Q| = |C'| = 2^{\Omega(n)}$ .  $\square$

To sum up, we constructed a decomposable PC  $P_n$  that has size  $O(n^2)$  such that any deterministic and decomposable PC approximating it has size  $O(2^{\Omega(n)})$ , thereby showing an unconditional exponential gap for approximation between decomposable PCs and deterministic and decomposable PCs. This result highlights a fundamental limitation: approximate modeling does not grant us an additional flexibility to overcome exponential expressive efficiency gaps. Moreover, we next show that approximate modeling, even when somehow obtained, is unfortunately still not enough to use PCs for efficient approximate inference with guarantees in general.

## 5 Relationship between Approximate Modeling and Inference

We have shown that even if we allow some approximation error, it is hard to efficiently approximate distributions using tractable probabilistic circuits. Given that approximate modeling remains a hard task, one would hope that computing the approximators with bounded distance would allow us to approximate hard inference queries with bounded error. In this section, we study the relationship between approximate modeling and inference, in particular focusing on *relative* and *absolute* approximations of *marginal*, *conditional*, and *maximum-a-posteriori (MAP)* queries.

In Section 3.1, we showed that bounded total variation distance is a necessary but not sufficient condition for relative approximation of marginals. We now consider a slightly weaker notion of approximation called *absolute approximation*. Let  $P(\mathbf{X})$  be a probability distribution over a set of variables  $\mathbf{X}$ . Then we say another distribution  $Q(\mathbf{X})$  is an *absolute approximator* of marginals of  $P$  with respect to  $0 \leq \epsilon \leq 1$  if:  $|P(\mathbf{y}) - Q(\mathbf{y})| \leq \epsilon$  for every assignment  $\mathbf{y}$  to a subset of variables  $Y \subseteq \mathbf{X}$ . We show that any model that is a  $k\epsilon^2$ - $D_f$ -approximator of  $P$  must also be an absolute approximator of marginals of  $P$  with respect to  $\epsilon$ .

**Theorem 5.1** (Bounded  $D_f$  implies absolute approximation of marginals). *Given two distributions  $P$  and  $Q$  over a set of variables  $\mathbf{X}$  and  $0 \leq \epsilon \leq 1$ , if  $D_f(P\|Q) < k\epsilon^2$  then for all assignments  $\mathbf{y}$  to a subset  $\mathbf{Y} \subseteq \mathbf{X}$ , we have  $|P(\mathbf{y}) - Q(\mathbf{y})| < \epsilon$ .*

*Proof.* Note that while the absolute error of marginals is symmetric between  $P$  and  $Q$ ,  $f$ -divergence between  $P$  and  $Q$ , such as the KL-divergence, is not symmetric. Therefore, we utilize the implications derived in [34], that  $D_f(P\|Q) < k\epsilon^2$  then  $D_{\text{TV}}(P\|Q) < \epsilon$ . Moreover, given that the total variation distance is an  $f$ -divergence, we know that by the monotonicity property [36]  $D_f(P(\mathbf{Y}), Q(\mathbf{Y})) \leq D_f(P(\mathbf{X}), Q(\mathbf{X}))$  for any  $\mathbf{Y} \subseteq \mathbf{X}$ . By definition,  $\max_{S \subseteq \{0,1\}^n} |P(S) - Q(S)| < \epsilon$ , and thus  $\forall \mathbf{y} : |P(\mathbf{y}) - Q(\mathbf{y})| < \epsilon$ .  $\square$

From the above proof, we also immediately derive the following corollary.

**Corollary 5.2** (Bounded  $D_{\text{TV}}$  implies absolute approximation of marginals). *Given two distributions  $P$  and  $Q$  over a set of variables  $\mathbf{X}$  and  $0 \leq \epsilon \leq 1$ , if  $D_{\text{TV}}(P\|Q) < \epsilon$ , then for all assignments  $\mathbf{y}$  to  $\mathbf{Y} \subseteq \mathbf{X}$ , we have  $|P(\mathbf{y}) - Q(\mathbf{y})| < \epsilon$ .*



Since marginals are tractable for decomposable PCs, approximating a target distribution with bounded  $f$ -divergence using a decomposable PC implies that marginals can be approximated in polynomial-time with bounded absolute error. This aligns with Dagum and Luby [14], who showed that there exists a randomized polynomial-time algorithm for the absolute approximation of marginals for Bayesian networks.

We next study approximate inference implications for MAP inference. Let  $P(\mathbf{X})$  be a probability distribution over a set of variables  $\mathbf{X}$ . We say another distribution  $Q(\mathbf{X})$  is an absolute approximator of the *maximum-a-posteriori* of  $P$  with respect to  $0 \leq \epsilon \leq 1$  if: for every assignment  $\mathbf{e}$  (called the evidence) to a subset  $\mathbf{E} \subseteq \mathbf{X}$ ,  $|\max_{\mathbf{y}} P(\mathbf{y}, \mathbf{e}) - \max_{\mathbf{y}} Q(\mathbf{y}, \mathbf{e})| \leq \epsilon$  where  $\mathbf{Y} = \mathbf{X} \setminus \mathbf{E}$ . We next show that  $k\epsilon^2$ - $D_f$ -approximators are also absolute approximators of MAP with respect to  $\epsilon$ .

**Theorem 5.3** (Bounded  $D_f$  implies absolute approximation of MAP). *Given two distributions  $P$  and  $Q$  over a set of variables  $\mathbf{X}$  and  $0 \leq \epsilon \leq 1$ , if  $D_f(P||Q) < k\epsilon^2$  then for every assignment  $\mathbf{e}$  to a subset  $\mathbf{E} \subseteq \mathbf{X}$ , we have  $|\max_{\mathbf{y} \in \mathbf{Y}} P(\mathbf{y}, \mathbf{e}) - \max_{\mathbf{y} \in \mathbf{Y}} Q(\mathbf{y}, \mathbf{e})| < \epsilon$  where  $\mathbf{Y} = \mathbf{X} \setminus \mathbf{E}$ .*

*Proof.* Analogous to Theorem 5.1, we utilize the fact that if we have  $D_f(P||Q) < k\epsilon^2$ , we know  $D_{\text{TV}}(P||Q) < \epsilon$ . Using  $D_{\text{TV}}(P||Q) < \epsilon$ , we have  $\max_{\mathbf{x}} |P(\mathbf{x}) - Q(\mathbf{x})| < \epsilon$  by definition. Then for all  $\mathbf{x}$ ,  $Q(\mathbf{x}) - \epsilon < P(\mathbf{x}) < Q(\mathbf{x}) + \epsilon$ . W.l.o.g., suppose  $\max P(\mathbf{x}) > \max Q(\mathbf{x})$ . Then  $|\max P(\mathbf{x}) - \max Q(\mathbf{x})| < (\max Q(\mathbf{x}) + \epsilon) - \max Q(\mathbf{x}) = \epsilon$ . Thus, as this holds for all  $\mathbf{x}$ , we can extend this to  $|\max_{\mathbf{y} \in \mathbf{Y}} P(\mathbf{y}, \mathbf{e}) - \max_{\mathbf{y} \in \mathbf{Y}} Q(\mathbf{y}, \mathbf{e})| < \epsilon$  for every assignment  $\mathbf{e}$  to a subset  $\mathbf{E} \subseteq \mathbf{X}$  and  $\mathbf{Y} = \mathbf{X} \setminus \mathbf{E}$ . Thus, approximating with bounded  $f$ -divergence by a deterministic and decomposable PC implies polynomial-time approximation of MAP with bounded absolute error.  $\square$

Again, we can restrict this to the special case of total variation distance via the above.

**Corollary 5.4** (Bounded  $D_{\text{TV}}$  implies absolute approximation of MAP). *Given two distributions  $P$  and  $Q$  over a set of variables  $\mathbf{X}$  and  $0 \leq \epsilon \leq 1$ , if  $D_{\text{TV}}(P||Q) < \epsilon$  then for every assignment  $\mathbf{e}$  to a subset  $\mathbf{E} \subseteq \mathbf{X}$ , we have  $|\max_{\mathbf{y} \in \mathbf{Y}} P(\mathbf{y}, \mathbf{e}) - \max_{\mathbf{y} \in \mathbf{Y}} Q(\mathbf{y}, \mathbf{e})| < \epsilon$  where  $\mathbf{Y} = \mathbf{X} \setminus \mathbf{E}$ .*

Thus, a deterministic and decomposable PC that is an  $\epsilon$ - $D_{\text{TV}}$ -approximator of a distribution  $P$  would imply that exact MAP inference w.r.t. this PC grants us tractable approximate MAP inference w.r.t. the original distribution  $P$ . However, the converse does not hold: a PC that can be used for approximate MAP inference is not necessarily a good approximation of the full distribution.

**Counterexample 1.** Consider a family of distributions  $P(\mathbf{x})$  such that  $\max_{\mathbf{x}} P(\mathbf{x}) < \epsilon$ . Then, we construct a distribution  $P'(\mathbf{X}, Z)$  such that  $P(\mathbf{x}, Z = 1) = P(\mathbf{x})$  and  $P(\mathbf{x}, Z = 0) = 0$ . Similarly, let  $Q(\mathbf{X}, Z)$  be such that  $Q(\mathbf{x}, Z = 0) = P(\mathbf{x})$  and  $Q(\mathbf{x}, Z = 1) = 0$ . Thus, for any assignment  $\mathbf{e}$  to  $\mathbf{E} \subseteq \mathbf{X} \cup \{Z\}$ ,  $|\max_{\mathbf{y}} P(\mathbf{y}, \mathbf{e}) - \max_{\mathbf{y}} Q(\mathbf{y}, \mathbf{e})| \leq P(\mathbf{x}) < \epsilon$ , so  $Q$  is an absolute approximator of the MAP of  $P$ . However,  $D_{\text{TV}}(P||Q) = 1$  as  $P$  and  $Q$  have disjoint supports.

Lastly, not all tractable queries for PCs are guaranteed to admit absolute approximation even under this framework of approximate modeling with bounded distance.

**Theorem 5.5** (Bounded  $D_{\text{TV}}$  does not imply absolute approx. of conditionals/conditional MAP). *There exists a family of distributions  $P$  that have  $\epsilon$ - $D_{\text{TV}}$ -approximators, yet the absolute approximation for conditional marginals and conditional MAP can be arbitrarily large.*

*Proof.* Let  $P$  be a probability distribution over  $\mathbf{X}$  such that  $P(\mathbf{e}) < 1/k$  for some assignment  $\mathbf{e}$  to  $\mathbf{E} \subseteq \mathbf{X}$ . Let  $\mathbf{Y} = \mathbf{X} \setminus \mathbf{E}$ . We construct another distribution  $Q$  such that:  $Q(\mathbf{y}^*, \mathbf{e}) = P(\mathbf{e})P(\mathbf{y}^*|\mathbf{e}) + k\epsilon P(\mathbf{e})$  where  $\mathbf{y}^*$  maximizes  $P(\mathbf{y}|\mathbf{e})$ ;  $Q(\mathbf{y}_1, \mathbf{e}) = P(\mathbf{e})P(\mathbf{y}_1|\mathbf{e}) - k\epsilon P(\mathbf{e})$  for another assignment  $\mathbf{y}_1$ ; and  $Q(\mathbf{x}) = P(\mathbf{x})$  for all other assignments  $\mathbf{x}$ . Note that  $Q$  is normalized by construction. Then the total variation distance between  $P$  and  $Q$  is:  $D_{\text{TV}}(P||Q) = \frac{1}{2}(|P(\mathbf{y}^*, \mathbf{e}) - Q(\mathbf{y}^*, \mathbf{e})| + |P(\mathbf{y}_1, \mathbf{e}) - Q(\mathbf{y}_1, \mathbf{e})|) = k\epsilon P(\mathbf{e}) < \epsilon$ . On the other hand, the absolute approximation error of conditional MAP can grow arbitrarily by increasing  $k$ :  $|\max_{\mathbf{y}} P(\mathbf{y}|\mathbf{e}) - \max_{\mathbf{y}} Q(\mathbf{y}|\mathbf{e})| = |P(\mathbf{y}^*|\mathbf{e}) - P(\mathbf{y}^*|\mathbf{e}) - k\epsilon| = k\epsilon$ . Note that the same also holds for absolute approximation of conditionals.  $\square$

Moreover, because relative approximation of conditionals imply their absolute approximation [14], bounded  $D_{\text{TV}}$  also does not imply relative approximation of conditionals. It is well known that absolute and relative approximation of conditionals is NP-hard in Bayesian networks [14]. Even though approximate modeling with a bounded  $D_{\text{TV}}$  is also an NP-hard task, solving it still does not

guarantee a polynomial-time algorithm for approximating conditional queries. This highlights a key limitation: while tractability of queries is guaranteed by the structural properties of our learned PCs, some queries do not yield “good” approximations for all assignments even after learning within bounded distance.

## 6 Conclusion and Discussions

We established the hardness of approximating distributions with tractable probabilistic models such that the  $f$ -divergence is small. First, we showed that this task is NP-hard for any model supporting tractable marginal inference, including decomposable PCs. Then, we used the Sauerhoff function to demonstrate an exponential size gap between the class of decomposable PCs and that of deterministic and decomposable PCs when allowing for a bounded approximation error. This proves that the expressive efficiency gap that exists in exact compilation persists even under the relaxed approximation conditions. Finally, we characterized which queries remain well-approximated under the framework of approximate compilation.

These results highlight key challenges in learning compact and expressive PCs while maintaining tractable inference. In light of this, we ask: can a polynomial-time algorithm enable learning an  $\epsilon$ -approximator for a broad family of distributions with a more relaxed  $\epsilon$ ? While this is trivial when total variation is near 1, investigating whether structurally constrained PCs remain expressive under weaker approximation could reveal key limits of learnability. Furthermore, does there exist modeling conditions that are sufficient to guarantee relative approximation of various queries? Lastly, we see this work as a first step to encourage further theoretical studies on approximate modeling and inference with guarantees using tractable models. In this paper, we focused on PCs that are tractable for marginal and MAP queries, but there are large classes of tractable models whose efficient approximation power remain largely unknown.

## Acknowledgments and Disclosure of Funding

We thank the anonymous reviewers for their feedback towards improving this paper. We also thank Adrian Ciotinga for helpful feedback and discussion. This work was supported in part by the AFOSR grant FA9550-25-1-0320.

## References

- [1] Alexei Borodin. Determinantal point processes. *arXiv preprint arXiv:0911.1153*, 2009.
- [2] Simone Bova, Florent Capelli, Stefan Mengel, and Friedrich Slivovsky. Knowledge compilation meets communication complexity. In *IJCAI*, volume 16, pages 1008–1014, 2016.
- [3] Supratik Chakraborty, Kuldeep Meel, Rakesh Mistry, and Moshe Vardi. Approximate probabilistic inference via word-level counting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016.
- [4] Hei Chan and Adnan Darwiche. Reasoning about bayesian network classifiers. In *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*, pages 107–115, 2002.
- [5] Felicia Chiang, Peter Greve, Omid Mazdiyasn, Yoshihide Wada, and Amir AghaKouchak. A multivariate conditional probability ratio framework for the detection and attribution of compound climate extremes. *Geophysical Research Letters*, 48(15):e2021GL094361, 2021.
- [6] Arthur Choi and Adnan Darwiche. On relaxing determinism in arithmetic circuits. In *International Conference on Machine Learning*, pages 825–833. PMLR, 2017.
- [7] YooJung Choi, Adnan Darwiche, and Guy Van den Broeck. Optimal feature selection for decision robustness in bayesian networks. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, August 2017.
- [8] 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.

- [9] YooJung Choi, Meihua Dang, and Guy Van den Broeck. Group fairness by probabilistic modeling with latent fair decisions. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, Feb 2021.
- [10] YooJung Choi, Tal Friedman, and Guy Van den Broeck. Solving marginal map exactly by probabilistic circuit transformations. In *International Conference on Artificial Intelligence and Statistics*, pages 10196–10208. PMLR, 2022.
- [11] C. Chow and C. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14(3):462–467, 1968. doi: 10.1109/TIT.1968.1054142.
- [12] Karine Chubarian and György Turán. Interpretability of bayesian network classifiers: Obdd approximation and polynomial threshold functions. In *ISAIM*, 2020.
- [13] Gregory F Cooper. The computational complexity of probabilistic inference using bayesian belief networks. *Artificial intelligence*, 42(2-3):393–405, 1990.
- [14] Paul Dagum and Michael Luby. Approximating probabilistic inference in bayesian belief networks is np-hard. *Artificial intelligence*, 60(1):141–153, 1993.
- [15] Adnan Darwiche. A logical approach to factoring belief networks. *KR*, 2:409–420, 2002.
- [16] Adnan Darwiche. A differential approach to inference in bayesian networks. *Journal of the ACM (JACM)*, 50(3):280–305, 2003.
- [17] 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>.
- [18] Adnan Darwiche. Bayesian networks. *Foundations of Artificial Intelligence*, 3:467–509, 2008.
- [19] Adnan Darwiche and Pierre Marquis. A knowledge compilation map. *Journal of Artificial Intelligence Research*, 17:229–264, 2002.
- [20] Alexis de Colnet and Stefan Mengel. A compilation of succinctness results for arithmetic circuits. In *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning*, volume 18, pages 205–215, 2021.
- [21] Alexis De Colnet and Stefan Mengel. Lower bounds for approximate knowledge compilation. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 1834–1840, 2021.
- [22] Pedro Zuidberg Dos Martires. A quantum information theoretic approach to tractable probabilistic models. In *The 41st Conference on Uncertainty in Artificial Intelligence*.
- [23] Alexandre Dubray, Pierre Schaus, and Siegfried Nijssen. Anytime weighted model counting with approximation guarantees for probabilistic inference. In *30th International Conference on Principles and Practice of Constraint Programming (CP 2024)*, pages 10–1. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2024.
- [24] Gal Elidan and Stephen Gould. Learning bounded treewidth bayesian networks. *Advances in neural information processing systems*, 21, 2008.
- [25] Daan Fierens, Guy Van den Broeck, Joris Renkens, Dimitar Shterionov, Bernd Gutmann, Ingo Thon, Gerda Janssens, and Luc De Raedt. Inference and learning in probabilistic logic programs using weighted boolean formulas. *Theory and Practice of Logic Programming*, 15(3):358–401, 2015.
- [26] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Commun. ACM*, 63(11):139–144, October 2020. ISSN 0001-0782. doi: 10.1145/3422622. URL <https://doi.org/10.1145/3422622>.

- [27] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.
- [28] Steven Holtzen, Guy Van den Broeck, and Todd Millstein. Scaling exact inference for discrete probabilistic programs. *Proceedings of the ACM on Programming Languages*, 4(OOPSLA): 1–31, 2020.
- [29] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022. URL <https://arxiv.org/abs/1312.6114>.
- [30] Doga Kisa, Guy Van den Broeck, Arthur Choi, and Adnan Darwiche. Probabilistic sentential decision diagrams. In *KR*, 2014.
- [31] 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.
- [32] H.-A. Loeliger. An introduction to factor graphs. *IEEE Signal Processing Magazine*, 21(1): 28–41, 2004. doi: 10.1109/MSP.2004.1267047.
- [33] James Martens and Venkatesh Medabalimi. On the expressive efficiency of sum product networks. *arXiv preprint arXiv:1411.7717*, 2014.
- [34] James Melbourne. Strongly convex divergences. *Entropy*, 22(11):1327, 2020.
- [35] George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021.
- [36] Yury Polyanskiy. Lecture notes on f-divergence, 2014. URL [https://people.lids.mit.edu/yp/homepage/data/LN\\_fdiv.pdf](https://people.lids.mit.edu/yp/homepage/data/LN_fdiv.pdf). Accessed: 2025-02-10.
- [37] 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.
- [38] Tahrira Rahman, Prasanna Kothalkar, and Vibhav Gogate. Cutset networks: A simple, tractable, and scalable approach for improving the accuracy of chow-liu trees. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2014, Nancy, France, September 15-19, 2014. Proceedings, Part II 14*, pages 630–645. Springer, 2014.
- [39] Dan Roth. On the hardness of approximate reasoning. *Artificial intelligence*, 82(1-2):273–302, 1996.
- [40] Paul G Sanfilippo, Alex W Hewitt, and David A Mackey. The importance of conditional probability in diagnostic reasoning and clinical decision making: A primer for the eye care practitioner. *Ophthalmic epidemiology*, 24(2):81–89, 2017.
- [41] Martin Sauerhoff. Approximation of boolean functions by combinatorial rectangles. *Theoretical computer science*, 301(1-3):45–78, 2003.
- [42] Solomon Eyal Shimony. Finding maps for belief networks is np-hard. *Artificial intelligence*, 68(2):399–410, 1994.
- [43] Arseny Skryagin, Daniel Ochs, Devendra Singh Dhami, and Kristian Kersting. Scalable neural-probabilistic answer set programming. *J. Artif. Int. Res.*, 78, January 2024. ISSN 1076-9757. doi: 10.1613/jair.1.15027. URL <https://doi.org/10.1613/jair.1.15027>.
- [44] Yee Whye Teh, Max Welling, Simon Osindero, and Geoffrey E Hinton. Energy-based models for sparse overcomplete representations. *Journal of Machine Learning Research*, 4(Dec): 1235–1260, 2003.
- [45] Alexandre B Tsybakov. *Introduction to Nonparametric Estimation*. Springer series in statistics. Springer, 2009. doi: 10.1007/b13794.

- [46] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [47] 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.
- [48] Lang Yin and Han Zhao. On the expressive power of tree-structured probabilistic circuits. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL <https://openreview.net/forum?id=suYAA0I5bd>.
- [49] Honghua Zhang, Steven Holtzen, and Guy Broeck. On the relationship between probabilistic circuits and determinantal point processes. In *Conference on Uncertainty in Artificial Intelligence*, pages 1188–1197. PMLR, 2020.
- [50] Honghua Zhang, Brendan Juba, and Guy Van den Broeck. Probabilistic generating circuits. In *International Conference on Machine Learning*, pages 12447–12457. PMLR, 2021.

## A Appendix

### A.1 $k$ -convex $f$ -divergences

Table 1: Examples of  $k$ -convex  $f$ -divergences [34]

Divergence	$f$	$k$	Domain
Kullback-Leibler	$t \log t$	$\frac{1}{M}$	$(0, M]$
Total variation	$\frac{ t-1 }{2}$	0	$(0, \infty)$
Pearson's $\chi^2$	$(t-1)^2$	2	$(0, \infty)$
Squared Hellinger	$2(1 - \sqrt{t})$	$M^{-\frac{3}{2}}/2$	$(0, M]$
Reverse KL	$-\log t$	$\frac{1}{M^2}$	$(0, M]$
Vincze-Le Cam	$\frac{(t-1)^2}{t+1}$	$\frac{8}{(M+1)^3}$	$(0, M]$
Jensen-Shannon	$(t+1) \log \frac{2}{t+1} + t \log t$	$\frac{1}{M(M+1)}$	$(0, M]$
Neyman's $\chi^2$	$\frac{1}{t} - 1$	$\frac{2}{M^3}$	$(0, M]$
Sason's $s$	$\log(s+t)^{(s+t)^2} - \log(s+1)^{(s+1)^2}$	$2 \log(s+M) + 3$	$[M, \infty), s > e^{-3/2}$
$\alpha$ -divergence	$\frac{4 \left(1 - t^{\frac{1+\alpha}{2}}\right)}{1 - \alpha^2}, \quad \alpha \neq \pm 1$	$M^{\frac{\alpha-3}{2}}$	$\begin{cases} [M, \infty), & \alpha > 3 \\ (0, M], & \alpha < 3 \end{cases}$

### A.2 Rectangle Partitions

Rectangle partitions are a powerful tool used in communication complexity to analyze the size of communication protocols. The main idea is to represent the communication protocol for a function  $f : \{0, 1\}^n \rightarrow \{0, 1\}$  into a  $2^n \times 2^n$  matrix  $M_f$  where  $M_f[\mathbf{x}, \mathbf{y}] = f(\mathbf{x}, \mathbf{y})$ , then partition  $M_f$  into a set of monochromatic rectangles which cover the input space of all possible pairs. Here, monochromatic means that a given rectangle covers only the outputs equal to 0 or 1, but not both. This allows us to derive lower bounds on the communication complexity of a function  $f$ . Furthermore, Bova et al. [2] showed the relation between rectangle covers and partitions to the size of DNNF and d-DNNF formulas. For all definitions below, assume that  $\mathbf{X}$  is a finite set of variables.

We begin by describing *partitions* of  $\mathbf{X}$ , corresponding to our partition of  $M_f$ .

**Definition A.1** (Partition [2]). A partition of  $\mathbf{X}$  is a sequence of pairwise disjoint subsets  $\mathbf{X}_i$  of  $\mathbf{X}$  such that  $\bigcup_i \mathbf{X}_i = \mathbf{X}$ . A partition of two blocks  $(\mathbf{X}_1, \mathbf{X}_2)$  is *balanced* if  $|\mathbf{X}|/3 \leq \min(|\mathbf{X}_1|, |\mathbf{X}_2|)$ .

We can now define *rectangles*:

**Definition A.2** (Combinatorial Rectangle [2]). A rectangle over  $\mathbf{X}$  is a function  $r : \{0, 1\}^{|\mathbf{X}|} \rightarrow \{0, 1\}$  such that there exists an underlying partition of  $\mathbf{X}$ , called  $(\mathbf{X}_1, \mathbf{X}_2)$  and functions  $r_i : \{0, 1\}^{|\mathbf{X}|} \rightarrow \{0, 1\}$  for  $i = 1, 2$  such that  $r^{-1}(1) = r_1^{-1}(1) \times r_2^{-1}(1)$ . A rectangle is *balanced* if the underlying partition is balanced.

Combining many of these rectangles together allows us to cover  $M_f$ , effectively covering the function  $f$ . The size of these covers provides lower bounds on the communication complexity of  $f$ .

**Definition A.3** (Rectangular Cover [2]). Let  $f : \{0, 1\}^{|\mathbf{X}|} \rightarrow \{0, 1\}$  be a function. A finite set of rectangles  $\{r_i\}$  over  $\mathbf{X}$  is called a *rectangle cover* if

$$f^{-1}(1) = \bigcup_i r_i^{-1}(1).$$

The rectangle cover is referred to as a *rectangle partition* if the above union is disjoint. A rectangle cover is *balanced* if each rectangle in the cover is balanced.

To understand how these rectangle partitions relate to d-DNNFs and DNNFs, we utilize the following notions of *certificates* and *elimination*.

**Definition A.4** (Certificate [2]). Let  $\mathcal{C}$  be a DNNF on  $\mathbf{X}$ . A *certificate* of  $\mathcal{C}$  is a DNNF  $T$  on  $\mathbf{X}$  such that:  $T$  contains the output gate of  $\mathcal{C}$ ; if  $T$  contains an  $\wedge$ -gate,  $v$ , of  $\mathcal{C}$  then  $T$  also contains every gate of  $\mathcal{C}$  having an output wire to  $v$ ; if  $T$  contains an  $\vee$ -gate of  $\mathcal{C}$ , then  $T$  also contains exactly one gate of  $\mathcal{C}$  having an output wire to  $v$ . The output gate of  $T$  coincides with the output gate of  $\mathcal{C}$ , and the gates of  $T$  inherit their labels and wires from  $\mathcal{C}$ . We let  $\text{cert}(\mathcal{C})$  denote the certificates of  $\mathcal{C}$ .

See from the above definition that

$$\mathcal{C}^{-1}(1) = \bigcup_{T \in \text{cert}(\mathcal{C})} T^{-1}(1).$$

This is useful tool in relation to rectangle partitions due to the fact that given a DNNF  $\mathcal{C}$ ,  $T \in \text{cert}(\mathcal{C})$  and gate  $g$ , then  $\mathcal{C}_g^{-1}(1) = \bigcup_{T \in \text{cert}(\mathcal{C}_g)} T^{-1}(1)$  where  $\mathcal{C}_g$  represents the sub-circuit  $\mathcal{C}$  rooted at gate  $g$ . Then we can represent  $\mathcal{C}_g^{-1}(1)$  as a rectangle which separates the variables in the sub-circuit  $\mathcal{C}$  rooted at  $g$ . Using this in conjunction with the *elimination* operation gives us the ability to compute the size of our circuit using rectangles.

**Definition A.5** (Elimination [2]). Let  $\mathcal{C}$  be a DNNF and  $g$  be a non-input gate. Then,

$$\mathcal{C}_{\neg g}^{-1}(1) = \bigcup_{T \in \text{cert}(\mathcal{C}) \setminus \text{cert}(\mathcal{C}_g)} T^{-1}(1)$$

In the case of a d-DNNF, by determinism we can write  $\mathcal{C}_{\neg g}^{-1}(1) = \mathcal{C}^{-1}(1) \setminus \mathcal{C}_g^{-1}(1)$ .

Next we provide a short description on the relationship between the size of rectangle covers and Boolean circuits; for the full detailed proofs see [2]. Effectively, start with a d-DNNF  $\mathcal{C}$  over variables  $\mathbf{X}$  which computes a function  $f$ . Then, construct  $\mathcal{C}^{i+1} = \mathcal{C}_{\neg g_i}^i$  by eliminating  $g_i \in \mathcal{C}^i$  until we hit  $l \leq |\mathcal{C}|$  such that  $\mathcal{C}^l \equiv 0$ . It can be shown that  $R_i = \mathcal{C}_{g_i}^{i-1}(1)$  is a balanced rectangle over  $\mathbf{X}$ . The set  $\{R_i | i = 0, \dots, l-1\}$  is then a balanced rectangle partition of  $\mathcal{C}$  since  $\mathcal{C}_{\neg g_i}^{i+1-1}(1) = \emptyset$ . Therefore, we can represent the size of d-DNNFs representing functions as the size of a balanced rectangle partition over said function. This implies that an exponential size rectangle partition implies exponentially large d-DNNF.

### A.3 Complete Proofs

#### A.3.1 Distribution Construction for Proposition 3.2

*Proof.* Let  $A$  be an event such that  $P(A) = \delta$  and for some  $K > 0$ ,

$$Q(\mathbf{x}) = \frac{P(\mathbf{x})}{K}, \quad \forall \mathbf{x} \in A.$$

Then, let  $Q(\mathbf{x}) = \lambda P(\mathbf{x})$ ,  $\forall \mathbf{x} \in A^c$  for some constant  $\lambda$ . To ensure that  $Q$  is normalized, see that we must have  $\sum_{\mathbf{x}} Q(\mathbf{x}) = \sum_{\mathbf{x} \in A} Q(\mathbf{x}) + \sum_{\mathbf{x} \in A^c} Q(\mathbf{x}) = 1$ . Therefore, we must have:

$$\begin{aligned} 1 &= \sum_{\mathbf{x} \in A} Q(\mathbf{x}) + \sum_{\mathbf{x} \in A^c} Q(\mathbf{x}) = \sum_{\mathbf{x} \in A} \frac{P(\mathbf{x})}{K} + \sum_{\mathbf{x} \in A^c} \lambda P(\mathbf{x}) = \frac{\delta}{K} + \lambda(1 - \sum_{\mathbf{x} \in A} P(\mathbf{x})) \\ &= \frac{\delta}{K} + \lambda(1 - \delta) \end{aligned}$$

Hence  $\lambda = \frac{1-\delta/K}{1-\delta}$  and thus we can define

$$Q(\mathbf{x}) = \begin{cases} \frac{P(\mathbf{x})}{K}, & \mathbf{x} \in A \\ \frac{1-\delta/K}{1-\delta} P(\mathbf{x}), & \mathbf{x} \in A^c \end{cases}$$

Therefore, we just need to check then that the  $f$ -divergence between  $P$  and  $Q$  must be bounded.

$$\begin{aligned}
\sum_{\mathbf{x}} Q(\mathbf{x}) f\left(\frac{P(\mathbf{x})}{Q(\mathbf{x})}\right) &= \sum_A Q(\mathbf{x}) f\left(\frac{P(\mathbf{x})}{Q(\mathbf{x})}\right) + \sum_{A^c} Q(\mathbf{x}) f\left(\frac{P(\mathbf{x})}{Q(\mathbf{x})}\right) \\
&= \sum_A \frac{P(\mathbf{x})}{K} f\left(\frac{P(\mathbf{x})}{P(\mathbf{x})/K}\right) + \sum_{A^c} \frac{1-\delta/K}{1-\delta} P(\mathbf{x}) f\left(\frac{1-\delta}{1-\delta/K}\right) \\
&= \frac{f(K)}{K} \sum_A P(\mathbf{x}) + \frac{1-\delta/K}{1-\delta} f\left(\frac{1-\delta}{1-\delta/K}\right) \sum_{A^c} P(\mathbf{x}) \\
&= \frac{\delta f(K)}{K} + \frac{1-\delta/K}{1-\delta} f\left(\frac{1-\delta}{1-\delta/K}\right) (1-\delta) \\
&= \frac{\delta f(K)}{K} + \left(1 - \frac{\delta}{K}\right) f\left(\frac{1-\delta}{1-\delta/K}\right)
\end{aligned}$$

See that as  $\delta \rightarrow 0$ , the above approaches  $0 + f(1)$ .

By definition of  $f$ -divergence,  $f(1) = 0$ . Thus, the  $f$ -divergence—including the total variation distance—between  $P$  and  $Q$  can be very small, approaching 0, while the relative approximation error stays at a constant factor  $K$ .  $\square$

### A.3.2 Pruning Deterministic PCs for the Proof of Proposition 4.3

Suppose that  $Q$  is a deterministic, decomposable and smooth probabilistic circuit. Given  $Q$ , wish to prune its edges such that in the resulting (unnormalized) PC  $Q'$ ,  $\mathbf{x}$  is in the support of  $Q'$  if and only if  $Q(\mathbf{x}) < \frac{1}{2^{n+1}}$ . We describe our pruning algorithm below.

First, we collect the an upper bound on each edge  $(n, c)$  that is the largest probability obtainable by any assignment  $\mathbf{x}$  that uses that edge (propagates non-zero value through the edge in the forward pass for  $Q(\mathbf{x})$ ). We denote this  $EB(n, c)$ , which stands for the Edge-Bound. This can be done in linear time in the size of the circuit using the Edge-Bounds algorithm [10]. This allows us to safely prune any edge whose Edge-Bound falls below a given threshold; i.e., prune edge  $(n, c)$  if  $EB(n, c) < \frac{1}{2^{n+1}}$ .

Note that pruning some edges may cause the edge bounds for remaining edges to be tightened. Thus, we will repeat this process until all  $Q(\mathbf{x}) < \frac{1}{2^{n+1}}$  are pruned away. Upon completion of this process, we return back the new pruned circuit  $Q'$ .

We know that this algorithm halts as there can only be a finite number of  $\mathbf{x}$  such that  $Q(\mathbf{x}) < \frac{1}{2^{n+1}}$ . Moreover, given that we have only deleted edges from a  $Q$ , our circuit  $Q'$  is still a deterministic, decomposable and smooth probabilistic circuit and has size polynomial in the size of  $Q$ . We are also assured by determinism that if we prune a path  $Q(\mathbf{x})$ , there exists no other path that can evaluate  $Q(\mathbf{x})$  [6]; thus all  $Q(\mathbf{x}) < \frac{1}{2^{n+1}}$  are deleted. Furthermore, by the property that there is only one accepting path per assignment  $\mathbf{x}$ , we know that we do not unintentionally delete any  $Q(\mathbf{x}) \geq \frac{1}{2^{n+1}}$ .

### A.3.3 Connecting Total Variation Distance and Weak Approximation for the Proof of Proposition 4.3

Suppose that  $P$  is a uniform distribution over the support given by a Boolean function  $f$ , and  $Q$  a probability distributions over the support given by a Boolean function  $h$ . We look to analyze the total variation distance with respect to support  $g$ , which is taken from  $Q'$ .

$$\begin{aligned}
D_{\text{TV}}(P||Q) &= \frac{1}{2} \left( \sum_{x=f \wedge h} |P(\mathbf{x}) - Q(\mathbf{x})| + \sum_{x=f \wedge \neg h} P(\mathbf{x}) + \sum_{x=\neg f \wedge h} Q(\mathbf{x}) \right) < \epsilon \\
\implies \sum_{x=f \wedge h} |P(\mathbf{x}) - Q(\mathbf{x})| + \sum_{x=f \wedge \neg h} P(\mathbf{x}) + \sum_{x=\neg f \wedge h} Q(\mathbf{x}) &< 2\epsilon
\end{aligned}$$

We partition  $h$  into the disjoint sets  $g$  and  $\neg g \wedge h$  (note that every model of  $g$  is already a model of  $h$ ).

$$\sum_{x=f \wedge g} \left| \frac{1}{\text{MC}(f)} - Q(\mathbf{x}) \right| + \sum_{x=f \wedge (\neg g \wedge h)} \left| \frac{1}{\text{MC}(f)} - Q(\mathbf{x}) \right| + \frac{\text{MC}(f \wedge \neg h)}{\text{MC}(f)} + \sum_{x=\neg f \wedge h} Q(\mathbf{x}) < 2\epsilon$$



The LHS of above inequality is again lower bounded by:

$$\begin{aligned}
& \sum_{\mathbf{x} \models f \wedge (\neg g \wedge h)} \left| \frac{1}{\text{MC}(f)} - Q(\mathbf{x}) \right| + \frac{\text{MC}(f \wedge \neg h)}{\text{MC}(f)} + \sum_{\mathbf{x} \models \neg f \wedge h} Q(\mathbf{x}) \\
& > \sum_{\mathbf{x} \models f \wedge (\neg g \wedge h)} \left| \frac{1}{2^n} - \frac{1}{2^{n+1}} \right| + \frac{\text{MC}(f \wedge \neg h)}{\text{MC}(f)} + \sum_{\mathbf{x} \models \neg f \wedge g} Q(\mathbf{x}) \\
& > \sum_{\mathbf{x} \models f \wedge (\neg g \wedge h)} \left| \frac{1}{2^n} - \frac{1}{2^{n+1}} \right| + \frac{\text{MC}(f \wedge \neg h)}{\text{MC}(f)} + \frac{\text{MC}(\neg f \wedge g)}{2^{n+1}}
\end{aligned}$$

as  $\text{MC}(f) \leq 2^n$  and  $Q(\mathbf{x}) < 1/2^{n+1}$  for every  $\mathbf{x} \models \neg g \wedge h$ . Thus,

$$\begin{aligned}
2\epsilon & > \sum_{\mathbf{x} \models f \wedge (\neg g \wedge h)} \left| \frac{1}{2^n} - \frac{1}{2^{n+1}} \right| + \frac{\text{MC}(f \wedge \neg h)}{\text{MC}(f)} + \frac{\text{MC}(\neg f \wedge g)}{2^{n+1}} \\
& > \frac{\text{MC}(f \wedge (\neg g \wedge h))}{2^{n+1}} + \frac{\text{MC}(f \wedge \neg h)}{\text{MC}(f)} + \frac{\text{MC}(\neg f \wedge g)}{2^{n+1}} \\
& > \frac{\text{MC}(f \wedge (\neg g \wedge h))}{2^{n+1}} + \frac{\text{MC}(f \wedge \neg h)}{2^{n+1}} + \frac{\text{MC}(\neg f \wedge g)}{2^{n+1}} = \frac{\text{MC}(f \wedge \neg g) + \text{MC}(\neg f \wedge g)}{2^{n+1}},
\end{aligned}$$

implying  $\text{MC}(f \wedge \neg g) + \text{MC}(\neg f \wedge g) < 2^{n+2}\epsilon = 2^n(4\epsilon)$ . Note that the last equality above is due to  $\neg h$  implying  $\neg g$ .

## NeurIPS Paper Checklist

### 1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [Yes]

Justification: We provide theorems and results for all claims made in the abstract and introduction. Specifically, each section builds a result described in the abstract and introduction.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

### 2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: We discuss all assumptions related to the hardness results provided. As well as the restriction to the form of  $f$ -divergence required for our results.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

### 3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We provide both proof sketches and full proofs in the appendix of our paper. The assumptions are clearly stated for each theoretical result.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

#### 4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [NA]

Justification: This paper does not include experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

#### 5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [NA]

Justification: This paper does not include experiments requiring code.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

## 6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [NA]

Justification: This paper does not include experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

## 7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [NA]

Justification: This paper does not include experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

## 8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [NA]

Justification: This paper does not include experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

## 9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics [https://neurips.cc/public/EthicsGuidelines?](https://neurips.cc/public/EthicsGuidelines)

Answer: [Yes]

Justification: This research as conducted in the paper conforms in every respect with the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

## 10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: There is no societal impact of the work performed.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

## 11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: This paper poses no such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

## 12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [NA]

Justification: This paper does not use existing assets.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, [paperswithcode.com/datasets](https://paperswithcode.com/datasets) has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset’s creators.

### 13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: This paper does not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

### 14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: This paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

### 15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: This paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

### 16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: LLMs were not used in the core method development of this paper, and does not impact the originality or scientific rigorousness of the research.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.