


Distance Estimation for High-Dimensional Discrete Distributions

Kumar, Gunjan 
IIT-Kanpur

Kuldeep S. Meel 
Georgia Institute of Technology

Yash Pote
National University of Singapore
CREATE

Abstract

Given two distributions \mathcal{P} and \mathcal{Q} over a high-dimensional domain $\{0, 1\}^n$, and a parameter ε , the goal of distance estimation is to determine the statistical distance between \mathcal{P} and \mathcal{Q} , up to an additive tolerance $\pm\varepsilon$. Since exponential lower bounds (in n) are known for the problem in the standard sampling model, research has focused on richer query models where one can draw conditional samples. This paper presents the first polynomial query distance estimator in the conditional sampling model (COND).

We base our algorithm on the relatively weaker *subcube conditional* sampling (SUBCOND) oracle, which draws samples from the distribution conditioned on some of the dimensions. SUBCOND is a promising model for widespread practical use because it captures the natural behavior of discrete samplers. Our algorithm makes $\tilde{O}(n^3/\varepsilon^5)$ queries to SUBCOND.

1 INTRODUCTION

Given two discrete distributions \mathcal{P} and \mathcal{Q} over $\{0, 1\}^n$, the total variation (TV) distance between \mathcal{P} and \mathcal{Q} , denoted by $d_{TV}(\mathcal{P}, \mathcal{Q})$, is defined as:

$$d_{TV}(\mathcal{P}, \mathcal{Q}) = \frac{1}{2} \sum_{\sigma \in \{0, 1\}^n} |\mathcal{P}(\sigma) - \mathcal{Q}(\sigma)|$$

In this paper, we are interested in the computation of (ε, δ) -approximation of $d_{TV}(\mathcal{P}, \mathcal{Q})$: i.e., we would like to compute an estimate **est** such that $\Pr[d_{TV}(\mathcal{P}, \mathcal{Q}) - \varepsilon \leq \mathbf{est} \leq d_{TV}(\mathcal{P}, \mathcal{Q}) + \varepsilon] \geq 1 - \delta$. TV distance is a

fundamental notion in probability and finds applications in diverse domains of computer science such as generative models (Goodfellow et al., 2014; Ji et al., 2023), MCMC algorithms (Andrieu et al., 2003; Boyd et al., 2004; Brooks et al., 2011), and probabilistic programming (Aguirre et al., 2021; Pote and Meel, 2022).

Theoretical investigations into the problem of TV distance computation have revealed the intractability of exact computation: In particular, the problem is $\#P$ -hard even when \mathcal{P} and \mathcal{Q} are represented as product distributions (Bhattacharyya et al., 2023a). As a consequence, the focus has been on designing approximation techniques. Randomized polynomial-time approximation schemes are known for some classes of distributions when \mathcal{P} and \mathcal{Q} are specified explicitly. An example is Bayesian networks with bounded treewidth (Bhattacharyya et al., 2023b). Not every practical application allows explicit representation of probability distributions, and often, the output of some underlying process defines probability distributions. Accordingly, the field of distribution testing is concerned with the design of algorithmic techniques for different models of access to the underlying processes. Furthermore, in addition to the classical notion of time complexity, we are also concerned with the *query complexity*: how many queries do we make to a given access model?

The earliest investigations focused on the classical model of access where one is only allowed to access samples from \mathcal{P} and \mathcal{Q} (Paninski, 2008; Valiant and Valiant, 2011); however, a lower bound of $\Omega(2^n/n)$ (Valiant and Valiant, 2010, 2011) restricts the applicability of these estimators in practical scenarios. This motivates the need to focus on more powerful models. In this work, we will focus on the SUBCOND access model owing to its ability to capture the behavior of probabilistic processes in diverse settings (Jerum et al., 1986; Chaudhuri et al., 1999; Zhao et al., 2018). For example, SUBCOND access perfectly models autoregressive sampling as employed in state-of-the-art LLMs and image models (Van Den Oord et al., 2016; Kalchbrenner, 2016).

Formally, the SUBCOND oracle for a distribution \mathcal{P} takes in a query string $\rho \in \{0, 1, *\}^n$, constructs the conditioning set $S_\rho = \{\sigma \in \{0, 1\}^n \mid (\rho_i = *) \vee (\rho_i = \sigma_i)\}$ and returns $\sigma \in S_\rho$ with probability $\frac{\mathcal{P}(\sigma)}{\sum_{\pi \in S_\rho} \mathcal{P}(\pi)}$. It is worth remarking that while we use the name SUBCOND to be consistent with recent literature (Bhattacharyya and Chakraborty, 2018), there have been algorithmic frameworks since the late 1980s that have relied on the underlying query model (Jerrum et al., 1986).

The starting point of our investigation is the observation that, on the one hand, practical applications of distance estimation rely on heuristic methods and hence don't provide any guarantees. On the other hand, no known algorithm, even when given access to the SUBCOND oracle, makes less than $O(2^n/n)$ queries. The primary contribution of our work is to address the mentioned gap: we design the first algorithm that computes (ε, δ) -approximation of TV distance and makes only polynomially many queries to SUBCOND oracle. Formally,

Theorem 1.1. *Given two distributions \mathcal{P} and \mathcal{Q} over $\{0, 1\}^n$, along with parameters $\varepsilon \in (0, 1)$, and $\delta \in (0, 1)$, the algorithm $\text{DistEstimate}(\mathcal{P}, \mathcal{Q}, \varepsilon, \delta)$ returns estimate κ such that*

$$\Pr[\kappa \in (d_{TV}(\mathcal{P}, \mathcal{Q}) \pm \varepsilon)] \geq 1 - \delta$$

DistEstimate makes $\tilde{O}(n^3 \log(1/\delta)/\varepsilon^4)$ queries to the SUBCOND oracle.

We now provide a high-level overview of DistEstimate : From the fact that,

$$d_{TV}(\mathcal{P}, \mathcal{Q}) = \mathbb{E}_{\sigma \sim \mathcal{Q}} \left[\max \left(1 - \frac{\mathcal{P}(\sigma)}{\mathcal{Q}(\sigma)}, 0 \right) \right]$$

we can use the standard approach of sampling σ from \mathcal{Q} , estimating $\mathcal{P}(\sigma)$ and $\mathcal{Q}(\sigma)$ up to some multiplicative factor, and then setting the value of the random variable to be $\max(1 - \mathcal{P}(\sigma)/\mathcal{Q}(\sigma), 0)$. This approach requires a constant number of samples from \mathcal{Q} to compute an approximation of $d_{TV}(\mathcal{P}, \mathcal{Q})$. The main issue is that it is not possible to approximate the value of $\mathcal{Q}(\sigma)$ for arbitrary σ with only polynomially many queries to SUBCOND since $\mathcal{Q}(\sigma)$ can be arbitrarily small and the query complexity scales inversely with $\mathcal{Q}(\sigma)$. The key technical contribution lies in showing that using polynomially many SUBCOND oracle calls, we can still compute estimates for $\mathcal{P}(\sigma)$ and $\mathcal{Q}(\sigma)$ at sufficiently many points to find a theoretically guaranteed estimate.

We are interested in designing distance estimation techniques for the SUBCOND model because it effectively captures the behavior of probabilistic processes in practice. Towards this goal, we compute

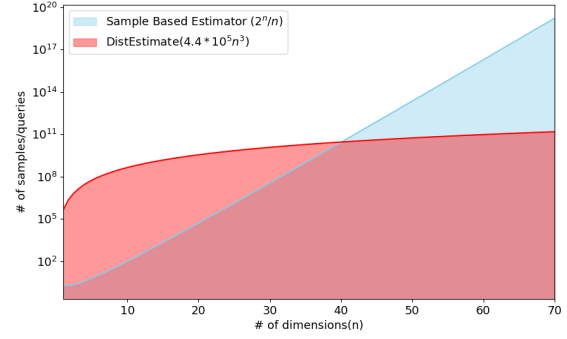


Figure 1: A plot comparing the sample/query complexity of the baseline non-conditional estimator vs. our estimator DistEstimate as a function of the number of dimensions n , for $\varepsilon = 0.3$. Note that the vertical axis is in the log scale.

the precise number of queries one would need to the test, and we find that DistEstimate offers a 10^7 factor speedup on problems of dimensionality $n = 70$, for which the baseline sample-based estimator would require $\simeq 10^{18}$ queries – a prohibitively large number. The result is presented in the Figure 1. Therefore, we demonstrate the application of DistEstimate in a real-world setting. Sampling from discrete domains such as $\{0, 1\}^n$ under combinatorial constraints is a challenging problem; therefore, several heuristic-based samplers have been proposed over the years. We can view a sampler as a probabilistic process, and consequently, one is interested in measuring how far the distribution of a given sampler is from the ideal distribution. Our experiments focus on combinatorial samplers, and SUBCOND is particularly well suited for this problem. We use a prototype of DistEstimate to evaluate the quality of two samplers for different benchmarks. Our empirical evaluation demonstrates the promise of scalability: in particular, DistEstimate offers a 10^7 factor speedup on problems of dimensionality $n = 70$.

Organization We start with a short background on related threads of investigation in Section 2. Then in Section 3 we define the notation we use in most of the paper. We present the paper's main contribution, the estimator DistEstimate , along with its proof of correctness in Section 4. In Section 5, we present the result of the evaluation of our implementation of DistEstimate . Finally, we conclude in Section 6 and discuss some open problems. In the interest of exposition, we defer some proofs to the Appendix.

2 RELATED WORK

Distance estimation is one of the many problems in the broader area of distribution testing. Apart from estimation, there is extensive literature on the problems of identity and equivalence testing. The problem of identity testing involves returning **Accept** if $d_{TV}(\mathcal{P}, \mathcal{P}^*) = 0$ and returning **Reject** if $d_{TV}(\mathcal{P}, \mathcal{P}^*) > \varepsilon$, where \mathcal{P} is an unknown distribution and \mathcal{P}^* is known, i.e. you have a full description of \mathcal{P}^* . Equivalence testing is the generalization of identity testing. It is the problem of deciding between $d_{TV}(\mathcal{P}, \mathcal{Q}) = 0$ and $d_{TV}(\mathcal{P}, \mathcal{Q}) > \varepsilon$ where both \mathcal{P} and \mathcal{Q} are unknown. It is worth emphasizing that for both identity and equivalence testing problems, any answer from the tester (**Accept** or **Reject**) is considered valid if $0 < d_{TV}(\mathcal{P}, \mathcal{Q}) \leq \varepsilon$. Provided only sample access, the sample complexity of identity testing is $\Theta(2^{n/2}/\varepsilon^2)$ (Paninski, 2008; Valiant and Valiant, 2017) and of equivalence testing is $\max(2^{2n/3}\varepsilon^{-4/3}, 2^{n/2}\varepsilon^{-2})$ (Chan et al. (2014); Valiant and Valiant (2017)). While testing is of theoretical interest, its practical application faces significant limitations primarily because testers must accept only when two given distributions are identical. In real-world scenarios, distributions are rarely identical but often exhibit close similarity. Consequently, a simplistic tester that consistently returns **Reject** can meet the specifications. A more rigorous definition of a tester is required to address this limitation, including estimating the distance between the two distributions. Unfortunately, this introduces a considerable challenge. Valiant and Valiant (2011) demonstrate that in the classical sampling model, the necessary number of queries increases to $2^n/n$, a significant jump from the previous $2^{2n/3}$.

To sidestep the exponential lower bounds on testing, the conditional sampling model, or **COND**, was introduced independently by Chakraborty et al. and Canonne et al., and has been successfully applied to various problems, including identity and equivalence testing. In this model, the sample complexity of identity testing is $\Theta(\varepsilon^{-2})$ (independent of n), while for equivalence testing the best-known upper and lower bounds are $O((\log n)/\varepsilon^5)$ (Falahatgar et al., 2015), and $\Omega(\sqrt{\log n})$ (Acharya et al., 2014) respectively. A survey by Canonne (2020a) provides a detailed view of testing and related problems in various sampling models.

Our work investigates the distance estimation problem using the **SUBCOND** model, a restriction of **COND**. Unlike **COND**, which allows conditioning on arbitrary sets, the **SUBCOND** model allows conditioning only on sets that are subcubes of the domain. While **COND** significantly improves the sample complexity, it is not easily implementable in practice, as arbitrary subsets

are not efficiently represented and sampled from. With a view towards plausible conditional models, Canonne et al. (2015); Bhattacharyya and Chakraborty (2018) came up with the **SUBCOND** model, which is particularly suited to the Boolean hypercube $\{0, 1\}^n$. Canonne et al. (2021) used the **SUBCOND** model to construct a nearly-optimal $\Theta(\sqrt{n})$ uniformity testing algorithm for $\{0, 1\}^n$, demonstrating its natural applicability for high-dimensional distributions. Then Chen et al. (2021) used **SUBCOND** to study the problems of learning and testing junta distributions supported on $\{0, 1\}^n$. Bhattacharyya and Chakraborty (2018) developed a test for equivalence in the **SUBCOND** model, with query complexity of $O(n^2/\varepsilon^2)$. However, before this work, there was no distance estimation algorithm in the **SUBCOND** oracle model, and indeed even in the general **COND** model.

Lower Bound The problem of testing with **SUBCOND** access has a query complexity lower bound of $\Omega(n/\log(n))$ as a direct consequence of Theorem 11 of Canonne et al. (2020). For completeness, we formally prove the lower bound in Appendix A.2.

3 NOTATIONS AND PRELIMINARIES

We will focus on probability distributions over $\{0, 1\}^n$. For any distribution \mathcal{D} on $\{0, 1\}^n$ and an element $\sigma \in \{0, 1\}^n$, $\mathcal{D}(\sigma)$ is the probability of σ in distribution \mathcal{D} . Further, $\sigma \sim \mathcal{D}$ represents that σ is sampled from \mathcal{D} . The total variation (TV) distance of two probability distributions \mathcal{P} and \mathcal{Q} is defined as: $d_{TV}(\mathcal{P}, \mathcal{Q}) = \frac{1}{2} \sum_{\sigma \in \{0, 1\}^n} |\mathcal{P}(\sigma) - \mathcal{Q}(\sigma)|$. For a random variable v , the expectation is denoted as $\mathbb{E}[v]$, and the variance as $\mathbb{V}[v]$.

For clarity of exposition, we will hide the use of the ceiling operator $\lceil x \rceil$ wherever integral values are required, such as the number of samples or the number of iterations of a loop. We use $[n]$ to represent the set $\{1, 2, \dots, n\}$.

Consider a discrete r.v. that takes the value v with probability p . The count of trials required to observe k instances of v follows a *negative binomial* distribution, denoted as $\text{NB}(k, p)$. The expected value $\mathbb{E}[\text{NB}(k, p)]$ is k/p , and its variance $\mathbb{V}[\text{NB}(k, p)]$ is $k(1-p)/p^2$. We state the Chernoff and Chebyshev concentration bounds in Appendix A.1 for completeness. Further, we also make use of the following tail bound for negative binomials:

Proposition 3.1 (Brown (2011)). *For $\gamma > 1$, $\Pr[\text{NB}(k, p) > \gamma \mathbb{E}[\text{NB}(k, p)]] \leq \exp\left(-\frac{\gamma k(1-1/\gamma)^2}{2}\right)$*

If σ is a string of length $n > 0$, then σ_i denotes the

i^{th} element of σ , and for $1 \leq j \leq n$, $\sigma_{<i}$ denotes the substring of σ from 1 to $i-1$, $\sigma_{<i} = \sigma_1 \cdots \sigma_{i-1}$; similarly $\sigma_{\leq i} = \sigma_1 \cdots \sigma_i$, and $\sigma_{<1}$ denotes the empty (length 0) string, also denoted as \perp .

For any distribution \mathcal{D} and string ρ , such that $0 \leq |\rho| < n$, the distribution \mathcal{D}_ρ denotes the marginal distribution of \mathcal{D} in the $|\rho| + 1^{th}$ dimension, conditioned on the string ρ , i.e., $\mathcal{D}_\rho(b) = \frac{\Pr_{\sigma \sim \mathcal{D}}[(\sigma_{|\rho|+1}=b) \wedge (\sigma_{\leq |\rho|}=\rho)]}{\Pr_{\sigma \sim \mathcal{D}}[\sigma_{\leq |\rho|}=\rho]}$.

Definition 3.2. A sampling oracle $\text{SAMP}(\mathcal{D})$ takes in a distribution \mathcal{D} , and returns a sample $\sigma \in \{0, 1\}^n$ such that $\Pr[\text{SAMP}(\mathcal{D}) = \sigma] = \mathcal{D}(\sigma)$.

Definition 3.3. A subcube conditioning oracle $\text{SUBCOND}(\mathcal{D}, \rho)$ takes in a distribution \mathcal{D} , and a query string ρ with $0 \leq |\rho| < n$, and returns a sample $\sigma \in \{0, 1\}^n$ such that $\Pr[\text{SUBCOND}(\mathcal{D}, \rho) = \sigma] = 1_{(\sigma_{\leq |\rho|}=\rho)} \prod_{i=|\rho|+1}^n \mathcal{D}_{\sigma_{<i}}(\sigma_i)$.

Definition 3.4. A conditional marginal oracle $\text{CM}(\mathcal{D}, \rho)$ takes in a distribution \mathcal{D} , and a query string ρ with $0 \leq |\rho| < n$, and returns a sample $b \in \{0, 1\}$ such that $\Pr[\text{CM}(\mathcal{D}, \rho) = b] = \mathcal{D}_\rho(b)$.

Note that the chain rule implies that $\text{SUBCOND}(\mathcal{D}, \perp)$ is the same as $\text{SAMP}(\mathcal{D})$.

3.1 Distance Approximation

We adapt the distance approximation algorithm of [Bhattacharyya et al. \(2020\)](#), that takes as input two distributions \mathcal{P} and \mathcal{Q} , and provides an (η, δ) estimate of $d_{TV}(\mathcal{P}, \mathcal{Q})$. The proof is deferred to Appendix A.3.

Lemma 3.5. (Theorem 3.1 in [\(Bhattacharyya et al., 2020\)](#)) For distributions \mathcal{P} and \mathcal{Q} over $\{0, 1\}^n$, and $\sigma \in \{0, 1\}^n$, let p_σ and q_σ be functions such that $p_\sigma \in (1 \pm \eta)\mathcal{P}(\sigma)$, and $q_\sigma \in (1 \pm \eta)\mathcal{Q}(\sigma)$. Given a set of samples S from \mathcal{Q} , and $\eta \in (0, 1)$ along with the p_σ and q_σ for each $\sigma \in S$, let $\text{est} = \frac{1}{|S|} \sum_{i \in S} 1_{q_\sigma > p_\sigma} \left(1 - \frac{p_\sigma}{q_\sigma}\right)$.

$$\Pr \left[\text{est} \notin \left(d_{TV}(\mathcal{P}, \mathcal{Q}) \pm \frac{3\eta}{1-\eta} \right) \right] \leq 2 \exp \left(-2|S| \left(\frac{\eta}{1-\eta} \right)^2 \right)$$

3.2 Taming Distributions

Given a distribution \mathcal{D} , we will define and construct a new distribution \mathcal{D}' that has desirable properties critical for DistEstimate .

Definition 3.6. A distribution \mathcal{D}' is θ -tamed, if

$$\forall \sigma \in \{0, 1\}^n, \forall \ell \in [n] \quad \mathcal{D}'_{\sigma_{<\ell}}(\sigma_\ell) \in [\theta, 1 - \theta]$$

Definition 3.7. For a given distribution \mathcal{D} , and parameter $\theta \in [0, 1/n)$, distribution \mathcal{D}' is the θ -tamed sibling of \mathcal{D} , if \mathcal{D}' is θ -tamed and $d_{TV}(\mathcal{D}, \mathcal{D}') \leq \theta n$.

Henceforth, we will use \mathcal{D}' as shorthand to refer to the θ -tamed sibling of \mathcal{D} and omit mentioning θ whenever θ is evident from the context. We will now show in the following lemma that given SUBCOND query access to distribution \mathcal{D} , CM , and SAMP access to \mathcal{D}' can be simulated efficiently. We defer the proof to Appendix B.

Lemma 3.8. Given a distribution \mathcal{D} and parameter $\theta \in [0, 1/n)$, every CM query to \mathcal{D}' can be simulated by making one SUBCOND query to \mathcal{D} , and every SAMP query to \mathcal{D}' can be simulated by making n SUBCOND queries to \mathcal{D} .

4 DistEstimate: A DISTANCE ESTIMATION ALGORITHM

We now present the pseudocode of our algorithm DistEstimate , and the SubToEval and DistEstimateCore subroutines. The following subsection will provide a high-level overview of all our algorithms and formal analysis.

Algorithm 1: $\text{DistEstimate}(\mathcal{P}, \mathcal{Q}, \varepsilon, \delta)$

```

1 All  $j = 1$  to  $4.5 \log(2/\delta)$ 
    $r_j \leftarrow \text{DistEstimateCore}(\mathcal{P}, \mathcal{Q}, \varepsilon)$ 
    $\kappa \leftarrow \text{Median}_j(r_j)$ 
2 return  $\kappa$ 
```

Algorithm 2: $\text{DistEstimateCore}(\mathcal{P}, \mathcal{Q}, \varepsilon)$

```

▷  $\mathcal{P}'$  and  $\mathcal{Q}'$  are  $\varepsilon/8n$ -tamed siblings of  $\mathcal{P}$  and  $\mathcal{Q}$  resp.
1  $\eta \leftarrow \varepsilon/(\varepsilon + 4)$ 
2  $m_{out} \leftarrow \frac{\log(24)}{2} \left( \frac{1-\eta}{\eta} \right)^2$ 
3  $m_{in} \leftarrow 32 \log(48m_{out})$ 
4  $\text{est} \leftarrow 0$ 
5 All  $i = 1$  to  $m_{out}$ 
6  $\sigma \leftarrow \text{SAMP}(\mathcal{Q}')$ 
7 All  $j = 1$  to  $m_{in}$   $p_j \leftarrow \text{SubToEval}(\mathcal{P}', \sigma, \eta)$ 
8  $q_j \leftarrow \text{SubToEval}(\mathcal{Q}', \sigma, \eta)$ 
9  $\hat{p} \leftarrow \text{Median}_j(p_j)$ 
10  $\hat{q} \leftarrow \text{Median}_j(q_j)$ 
11 if  $\hat{q} > \hat{p}$  then
12 |  $\text{est} \leftarrow \text{est} + 1 - \hat{p}/\hat{q}$ 
13 return  $\text{est}/m_{out}$ 
```

Algorithm 3: SubToEval($\mathcal{D}', \sigma, \eta$)

```
▷  $\mathcal{D}'$  is  $\varepsilon/8n$ -tamed sibling of  $\mathcal{D}$ 
1  $t \leftarrow 0$ 
2  $k \leftarrow 4n\eta^{-2}(1 + \eta^2)$ 
3 forall  $i = 1$  to  $n$  do
4    $x_i \leftarrow 0$ 
5    $f \leftarrow 0$ 
6   while  $f < k$  do
7      $\alpha \leftarrow \text{CM}(\mathcal{D}', \sigma_{<i})$ 
8      $x_i \leftarrow x_i + 1$ 
9      $t \leftarrow t + 1$ 
10    if  $t = 64n^3\eta^{-2}(1 + \eta)^2\varepsilon^{-1}$  then
11      return 0
12    if  $\alpha = \sigma_i$  then  $f \leftarrow f + 1$ 
13  $d \leftarrow \prod_{i=1}^n k/x_i$ 
return  $d$ 
```

4.1 High-Level Overview

In Section 4.1.1, we introduce the main ideas of our algorithms, DistEstimate and DistEstimateCore. Then, in Section 4.1.2, we explain the key concepts of the SubToEval subroutine.

4.1.1 Outline of the DistEstimate and DistEstimateCore routines

The pseudocode of DistEstimate and DistEstimateCore is given in Alg. 1 and 2 respectively. DistEstimate takes as input two distributions \mathcal{P} and \mathcal{Q} defined over the support $\{0, 1\}^n$, along with the parameter ε for tolerance and the parameter δ for confidence, and returns an ε -additive estimate of $d_{TV}(\mathcal{P}, \mathcal{Q})$ with probability at least $1 - \delta$.

DistEstimateCore returns a constant-error estimate of $d_{TV}(\mathcal{P}, \mathcal{Q})$. Specifically, DistEstimateCore returns an estimate r_j of $d_{TV}(\mathcal{P}, \mathcal{Q})$ such that $\Pr[r_j \in (d_{TV}(\mathcal{P}, \mathcal{Q}) \pm \varepsilon)] \geq 2/3$. To drive the error down to the required δ , we use the standard median trick, where DistEstimate makes $O(\log(1/\delta))$ independent calls to DistEstimateCore, and returns the median of the estimates.

DistEstimateCore starts by creating the $\varepsilon/8n$ -tamed siblings \mathcal{P}' and \mathcal{Q}' that are $\varepsilon/8$ close to \mathcal{P} and \mathcal{Q} in TV distance, and have the property that all of their marginal probabilities are lower bounded by $\Omega(\varepsilon/n)$. The bounded marginal property of \mathcal{P}' and \mathcal{Q}' is crucial for the polynomial query complexity of DistEstimateCore. The construction of \mathcal{P}' and \mathcal{Q}' , and the claimed guarantees, are discussed in Section 3.2.

DistEstimateCore then draws m_{out} samples $\sigma \sim \mathcal{Q}'$, and for each sample σ , calls SubToEval m_{in} times to

find the $(1 \pm \eta)$ estimates of $\mathcal{Q}'(\sigma)$ and $\mathcal{P}'(\sigma)$ (discussed below). The SubToEval subroutine puts an upper limit on the number of CM oracle calls, and the limit is set high enough to ensure that the estimates, \hat{p} and \hat{q} , are correct with the required confidence. DistEstimateCore then computes the distance using these estimates as given in Lemma 3.5.

4.1.2 Outline of the SubToEval subroutine

The SubToEval subroutine takes as input an element $\sigma \in \{0, 1\}^n$, a distribution \mathcal{D} over $\{0, 1\}^n$, and a parameter η . SubToEval outputs an η -multiplicative estimate of $\mathcal{D}(\sigma)$. The probability $\mathcal{D}(\sigma)$ can be expressed as a product of marginals, $\mathcal{D}(\sigma) = \prod_{i=1}^n \mathcal{D}_{\sigma_{<i}}(\sigma_i)$, by applying the chain rule. Essentially, the subroutine approximates each marginal $\mathcal{D}_{\sigma_{<i}}(\sigma_i)$ by k/x_i for each $i \in [n]$, using the CM oracle. The product $\prod_{i=1}^n k/x_i$ is then employed as the final estimate for $\mathcal{D}(\sigma)$.

In this context, the variable x_i represents the total count of $\text{CM}(\mathcal{D}, \sigma_{<i})$ queries executed until k occurrences of σ_i are observed. Given that $\mathcal{D}_\rho(b) = \Pr_{w \sim \text{CM}(\mathcal{D}, \rho)}[w = b]$ for any ρ (as discussed in Section 3), the ratio k/x_i is an intuitive choice as an estimator for $\mathcal{D}_{\sigma_{<i}}(\sigma_i)$. Moreover, to ensure the subroutine terminates, a total number of calls to the CM oracle are monitored, and if they ever exceed the threshold $64n^3\eta^{-2}(1 + \eta)^2\varepsilon^{-1}$, the subroutine terminates and returns 0.

We now discuss our technical contribution - showing the correctness of SubToEval when the threshold is set to $O(n^3)$ (for this discussion, we will set aside the dependency on η). To estimate $\mathcal{D}(\sigma)$, it is essential to estimate each of the n marginals, $\mathcal{D}_{\sigma_{<i}}(\sigma_i)$, to within an error margin of approximately $1 + 1/n$. This would require at least $n^2/\mathcal{D}_{\sigma_{<i}}(\sigma_i)$ queries for each marginal. Consequently, the total query complexity would sum up to $\sum_{i=1}^n n^2/\mathcal{D}_{\sigma_{<i}}(\sigma_i)$. This quantity is at least $\Omega(n^2)$, but it could potentially be unbounded as $\mathcal{D}_{\sigma_{<i}}(\sigma_i)$ can take arbitrarily small values. In the forthcoming section, we reduce this complexity to $O(n^3)$ through a more nuanced analysis.

4.2 Theoretical Analysis

In this section, we will prove our main Theorem 1.1. The proof of Theorem 1.1 relies on Lemma 4.1, which claims the correctness of the SubToEval subroutine and upper bound its query complexity. We will prove the lemma later.

Lemma 4.1. SubToEval($\mathcal{D}', \sigma, \eta$) takes as input distribution \mathcal{D}' , $\sigma \in \{0, 1\}^n$, $\eta \in (0, 1/5)$ and returns d , then

$$\Pr[d \in (1 \pm \eta)\mathcal{D}'(\sigma)] \geq 5/8$$

SubToEval makes $O(n^3/\eta^2)$ CM queries to \mathcal{D}' .

Theorem 1.1. *Given two distributions \mathcal{P} and \mathcal{Q} over $\{0,1\}^n$, along with parameters $\varepsilon \in (0,1)$, and $\delta \in (0,1)$, the algorithm $\text{DistEstimate}(\mathcal{P}, \mathcal{Q}, \varepsilon, \delta)$ returns estimate κ such that*

$$\Pr[\kappa \in (d_{TV}(\mathcal{P}, \mathcal{Q}) \pm \varepsilon)] \geq 1 - \delta$$

DistEstimate makes $\tilde{O}(n^3 \log(1/\delta)/\varepsilon^4)$ queries to the SUBCOND oracle.

Proof. We will first show that the algorithm $\text{DistEstimateCore}(\mathcal{P}, \mathcal{Q}, \varepsilon)$ returns **est** such that

$$\Pr[\mathbf{est} \in (d_{TV}(\mathcal{P}, \mathcal{Q}) \pm \varepsilon)] \geq 5/6$$

Since DistEstimate returns the median of the independent estimates provided by DistEstimateCore , then applying the Chernoff bound, we have $\Pr[\kappa \in (d_{TV}(\mathcal{P}, \mathcal{Q}) \pm \varepsilon)] \geq 1 - \delta$.

We will now consider the events that could lead to an incorrect estimate. Recalling that \mathcal{P}' and \mathcal{Q}' are $\varepsilon/8n$ -tamed siblings of \mathcal{P} and \mathcal{Q} we define $\text{Bad}_i^{\hat{p}}$ and $\text{Bad}_i^{\hat{q}}$ to be the events that in the i^{th} iteration of DistEstimateCore , $\hat{p} \notin (1 \pm \eta)\mathcal{P}'(\sigma)$, and $\hat{q} \notin (1 \pm \eta)\mathcal{Q}'(\sigma)$, respectively. We bound the probability of $\text{Bad}_i^{\hat{p}}$ and $\text{Bad}_i^{\hat{q}}$ in the following claim, whose proof is deferred to Appendix.

Claim 4.2. $\Pr[\text{Bad}_i^{\hat{p}}] \leq 1/24m_{out}$ and $\Pr[\text{Bad}_i^{\hat{q}}] \leq 1/24m_{out}$.

Now we define $\text{Bad} = \bigcup_{i \in [m_{out}]} (\text{Bad}_i^{\hat{p}} \cup \text{Bad}_i^{\hat{q}})$, i.e., Bad captures the event that at least one of the estimates is incorrect. Then from Claim 4.2 and the union bound, $\Pr[\text{Bad}] =$

$$\Pr \left[\bigcup_{i \in [m_{out}]} (\text{Bad}_i^{\hat{p}} \cup \text{Bad}_i^{\hat{q}}) \right] \leq \sum_{i \in [m_{out}]} (\Pr[\text{Bad}_i^{\hat{p}}] + \Pr[\text{Bad}_i^{\hat{q}}]) \leq m_{out} \left(\frac{1}{24m_{out}} + \frac{1}{24m_{out}} \right) \leq \frac{1}{12}.$$

Now, let's assume the event $\overline{\text{Bad}}$. We have a set of m_{out} samples from \mathcal{Q}' , and for each sample σ we have \hat{p} and \hat{q} such that $\hat{p} \in (1 \pm \eta)\mathcal{P}'(\sigma)$ and $\hat{q} \in (1 \pm \eta)\mathcal{Q}'(\sigma)$. This fulfills the condition of Lemma 3.5, and hence substituting $|S| = m_{out}$ (Line 2 of Alg.2) we have,

$$\begin{aligned} & \Pr \left[\mathbf{est} \notin \left(d_{TV}(\mathcal{P}', \mathcal{Q}') \pm \frac{3\eta}{1-\eta} \right) \cap \overline{\text{Bad}} \right] \\ & \leq 2 \exp \left(-2m_{out} \left(\frac{\eta}{1-\eta} \right)^2 \right) \leq 2 \exp(-\log(24)) = \frac{1}{12} \end{aligned}$$

Substituting $\eta = \frac{\varepsilon}{\varepsilon+4}$ from Alg.2, we have,

$\Pr[\mathbf{est} \notin (d_{TV}(\mathcal{P}', \mathcal{Q}') \pm \frac{3\varepsilon}{4}) \cap \overline{\text{Bad}}] \leq \frac{1}{12}$. Then,

$$\begin{aligned} & \Pr \left[\mathbf{est} \notin \left(d_{TV}(\mathcal{P}', \mathcal{Q}') \pm \frac{3\varepsilon}{4} \right) \right] \\ & \leq \Pr \left[\mathbf{est} \notin \left(d_{TV}(\mathcal{P}', \mathcal{Q}') \pm \frac{3\varepsilon}{4} \right) \cap \overline{\text{Bad}} \right] + \Pr[\text{Bad}] \\ & \leq 1/12 + 1/12 = 1/6 \end{aligned}$$

Since \mathcal{P}' and \mathcal{Q}' are $\varepsilon/8n$ -tamed siblings of \mathcal{P} and \mathcal{Q} , from Lemma 3.8 we know that $d_{TV}(\mathcal{P}', \mathcal{P}) \leq \varepsilon/8$ and $d_{TV}(\mathcal{Q}', \mathcal{Q}) \leq \varepsilon/8$. Then, from the triangle inequality, we have the bounds on $d_{TV}(\mathcal{P}, \mathcal{Q})$:

$$\begin{aligned} d_{TV}(\mathcal{P}', \mathcal{Q}') & \in d_{TV}(\mathcal{P}, \mathcal{Q}) \pm (d_{TV}(\mathcal{P}', \mathcal{P}) + d_{TV}(\mathcal{Q}', \mathcal{Q})) \\ & \in d_{TV}(\mathcal{P}, \mathcal{Q}) \pm \varepsilon/4 \end{aligned}$$

Combining the two, we get that $\Pr[\mathbf{est} \notin (d_{TV}(\mathcal{P}, \mathcal{Q}) \pm \varepsilon)] \leq 1/6$, and hence we have our claim.

Now, we will complete the proof by showing an upper bound on the query complexity. The total number of CM queries made by $\text{SubToEval}(\mathcal{D}', \sigma, \eta)$ in a single invocation is $64n^3\eta^{-2}(1+\eta)^2\varepsilon^{-1} = O(n^3\varepsilon^{-3})$. Then $\text{DistEstimateCore}(\mathcal{P}, \mathcal{Q}, \varepsilon)$ makes $m_{in}m_{out} = O(\varepsilon^{-2} \log(\varepsilon^{-1}))$ many calls to SubToEval . Finally, DistEstimate calls DistEstimateCore $48 \log(1/\delta)$ many times. Thus the total number of queries to the CM oracle made by DistEstimate is $O(n^3 \log(1/\delta) \log(\varepsilon^{-1})/\varepsilon^5)$. \square

Proof of Lemma 4.1. Consider the subroutine $\text{SubToEval}_1(\mathcal{D}', \sigma, \eta)$ (Alg. 4), that is the same as $\text{SubToEval}(\mathcal{D}', \sigma, \eta)$ (Alg. 3) except in one critical aspect: the termination condition on Line 10 of SubToEval has been removed. This implies that while $\text{SubToEval}(\mathcal{D}', \sigma, \eta)$ terminates if the number of calls to the CM oracle exceeds the threshold $64n^3\eta^{-2}(1+\eta)^2\varepsilon^{-1}$, $\text{SubToEval}_1(\mathcal{D}', \sigma, \eta)$ does not enforce this restriction, thereby allowing an unlimited number of calls to the CM oracle. Note that we use variable names d_1 and t_1 in $\text{SubToEval}'$ to distinguish them from d of t of SubToEval . This modification is critical for our analysis as it leads to the variable x_i in $\text{SubToEval}_1(\mathcal{D}', \sigma, \eta)$ following the negative binomial distribution.

Remark 4.3. Henceforth we will use t_1 and x_i to denote the final values of t_1 and x_i , as on Line 10.

We will now show that the $\text{SubToEval}_1(\mathcal{D}', \sigma, \eta)$ correctly estimates $\mathcal{D}'(\sigma)$ with high probability (Lemma 4.6) and then we show that it makes fewer than $64n^3\eta^{-2}(1+\eta)^2\varepsilon^{-1}$ calls to CM oracle with high probability (Lemma 4.7). These results will help us establish analogous results for the subroutine $\text{SubToEval}(\mathcal{D}, \sigma, \eta)$ and in validating our Lemma 4.1.

Algorithm 4: SubToEval₁($\mathcal{D}', \sigma, \eta$)

```
1  $t_1 \leftarrow 0$ 
2  $k \leftarrow 4n\eta^{-2}(1 + \eta^2)$ 
3 for  $i = 1$  to  $n$   $x_i \leftarrow 0$ 
4  $f \leftarrow 0$ 
5 while  $f < k$  do
6    $\alpha \leftarrow \text{CM}(\mathcal{D}', \sigma_{<i})$ 
7    $x_i \leftarrow x_i + 1$ 
8    $t_1 \leftarrow t_1 + 1$ 
9   if  $\alpha = \sigma_i$  then  $f \leftarrow f + 1$ 
10  $d_1 \leftarrow \prod_{i=1}^n k/x_i$ 
11 return  $d_1$ 
```

Observation 4.4. Comparing SubToEval and SubToEval₁, we observe that SubToEval returns an incorrect estimate d in two cases. Either SubToEval returns incorrect d_1 , or else SubToEval₁ makes more than $64n^3\eta^{-2}(1 + \eta)^2\varepsilon^{-1}$ queries. Stated formally,

$$\begin{aligned} & \Pr[d \notin (1 \pm \eta)\mathcal{D}'(\sigma)] \\ & \leq \Pr[d_1 \notin (1 \pm \eta)\mathcal{D}'(\sigma)] + \Pr[t_1 \geq 64n^3\eta^{-2}(1 + \eta)^2\varepsilon^{-1}] \end{aligned}$$

Our proof will use the following prop. and lemmas.

Proposition 4.5. For $i \in [n]$, the value of x_i (in Alg. 4) is distributed as $\text{NB}(k, \mathcal{D}_{\sigma_{<i}}(\sigma_i))$

We prove the above proposition in Appendix C.

Lemma 4.6. $\Pr[d_1 \in (1 \pm \eta)\mathcal{D}'(\sigma)] \geq 2/3$.

Proof. We use a variance reduction technique introduced by Dyer and Frieze (1991). x_i on Line 10 is distributed according to $\text{NB}(k, \mathcal{D}'_{\sigma_{<i}}(\sigma_i))$, so we have $\mathbb{E}[x_i] = k/\mathcal{D}'_{\sigma_{<i}}(\sigma_i)$, and hence, $k/\mathbb{E}[x_i] = \mathcal{D}'_{\sigma_{<i}}(\sigma_i)$. Now since $d_1 = \prod_{j=1}^n k/x_j$, we have $\mathbb{E}[1/d_1] = \mathbb{E}[\prod_{i=1}^n x_i/k] = \prod_{i=1}^n 1/\mathcal{D}'_{\sigma_{<i}}(\sigma_i)$.

$$\begin{aligned} \frac{\mathbb{V}[1/d_1]}{\mathbb{E}[1/d_1]^2} &= \frac{\mathbb{E}[1/d_1^2]}{\mathbb{E}[1/d_1]^2} - 1 = \prod_{i=1}^n \frac{\mathbb{E}[(x_i/k)^2]}{\mathbb{E}[x_i/k]^2} - 1 \\ &= \prod_{j=1}^n \left(1 + \frac{\mathbb{V}[x_i/k]}{\mathbb{E}[x_i/k]^2}\right) - 1 \end{aligned}$$

Using the fact that x_i is negative binomial, we substitute $\mathbb{V}[x_i/k]$ and $\mathbb{E}[x_i/k]^2$,

$$\begin{aligned} \frac{\mathbb{V}[1/d_1]}{\mathbb{E}[1/d_1]^2} &= \prod_{j=1}^n \left(1 + \frac{(1 - \mathcal{D}'_{\sigma_{<i}})/k\mathcal{D}'_{\sigma_{<i}}^2}{(1/\mathcal{D}'_{\sigma_{<i}})^2}\right) - 1 \\ &= \prod_{j=1}^n \left(1 + \frac{1 - \mathcal{D}'_{\sigma_{<i}}(\sigma_i)}{k}\right) - 1 \leq \prod_{j=1}^n \left(1 + \frac{1}{k}\right) - 1 \end{aligned}$$

Substituting the value of k from the algorithm, we have

$$\begin{aligned} \frac{\mathbb{V}[1/d_1]}{\mathbb{E}[1/d_1]^2} &\leq \left(1 + \frac{\eta^2}{4n(1 + \eta)^2}\right)^n - 1 \\ &\leq \exp\left(\frac{\eta^2}{4}\right) - 1 \leq \frac{\eta^2}{3(1 + \eta)^2} \end{aligned} \quad (1)$$

The last inequality comes from the fact that for $r \in (0, 1)$, $s > 1$, $\exp\left(\frac{r}{s+1}\right) \leq 1 + \frac{r}{s}$. Recall that from the chain rule we have $\mathcal{D}'(\sigma) = \prod_{j=1}^n \mathcal{D}'_{\sigma_{<j}}(\sigma_j)$, then $\mathbb{E}[1/d_1] = 1/\mathcal{D}'(\sigma)$.

$$\begin{aligned} & \Pr[d_1 \in (1 \pm \eta)\mathcal{D}'(\sigma)] \\ &= \Pr\left[\frac{1}{d_1} \in \left[\frac{1}{1 + \eta}, \frac{1}{1 - \eta}\right] \frac{1}{\mathcal{D}'(\sigma)}\right] \\ &= \Pr\left[\frac{1}{d_1} - \mathbb{E}\left[\frac{1}{d_1}\right] \in \left[-\frac{\eta}{1 + \eta}, \frac{\eta}{1 - \eta}\right] \mathbb{E}\left[\frac{1}{d_1}\right]\right] \\ &\geq \Pr\left[\left|\mathbb{E}\left[\frac{1}{d_1}\right] - \frac{1}{d_1}\right| \leq \frac{\eta}{1 + \eta} \mathbb{E}\left[\frac{1}{d_1}\right]\right] \\ &\geq 1 - \frac{(1 + \eta)^2}{\eta^2} \frac{\mathbb{V}\left[\frac{1}{d_1}\right]}{\mathbb{E}\left[\frac{1}{d_1}\right]^2} \geq 1 - \frac{1}{3} = \frac{2}{3} \end{aligned} \quad (2)$$

We use the Chebyshev bound to get the second to last inequality and then substitute (1). \square

Note that in every iteration, t_1 gets incremented by the value of x_i . In the following lemma, we claim that t_1 , the number of queries made by SubToEval₁, exceeds the threshold on Line 10 of SubToEval with low probability. We defer the proof to the Appendix C.

Lemma 4.7. $\Pr[t_1 \geq 64n^3\eta^{-2}(1 + \eta)^2\varepsilon^{-1}] \leq 1/24$

Putting together lemmas 4.6 and 4.7 along with the observation 4.4, we complete the proof:

$$\begin{aligned} & \Pr[d \notin (1 \pm \eta)\mathcal{D}'(\sigma)] \\ &\leq \Pr[d_1 \notin (1 \pm \eta)\mathcal{D}'(\sigma)] + \Pr\left[t_1 \geq 64 \frac{n^3\eta^{-2}(1 + \eta)^2}{\varepsilon^{-1}}\right] \\ &\leq \frac{1}{3} + \frac{1}{24} = \frac{3}{8} \end{aligned}$$

\square

4.3 The Discrete Hypergrid Σ^n

This section extends our results beyond the hypercube $\{0, 1\}^n$ to the hypergrid Σ^n , where Σ is any discrete set. This line of investigation is motivated by the fact that in modern ML, distributions models are frequently described over hypergrids. For instance, language models are defined to be distributions over Σ^n where Σ is the set of tokens, and n the length of the generated string. A prompt of length k (itself a

Table 1: The sample complexity and runtime performance of **DistEstimate** on real-world instances.

Benchmark	Dimensions	STS		CMSGen	
		# of samples	time (in s)	# of samples	time (in s)
s1196a_3_2	33	1.8e+09	4.1e+05	1.9e+09	5.3e+05
53.sk_4_32	33	1.7e+09	2.5e+05	1.9e+09	1.6e+06
27.sk_3_32	33	1.7e+09	1.9e+05	1.9e+09	1.0e+06
s1196a_7_4	33	1.8e+09	4.6e+05	1.9e+09	5.5e+05
s420_15_7	35	2.1e+09	4.2e+05	2.3e+09	4.0e+05
111.sk_2_36	37	2.2e+09	3.5e+05	8.3e+08	6.6e+05

string from Σ^k) fixes k dimensions and specifies a distribution over the subcube of $n - k$ dimensions. After that, the LLM generates strings from the specified subcube distribution. More generally, we believe that the SUBCOND oracle is particularly suitable for use in ML applications as it models autoregressive generation.

The SUBCOND oracle for \mathcal{D} supported on Σ^n , takes a query string $\rho \in \{\Sigma \cup *\}^n$ and draws samples from the set of strings that match all the non- $*$ characters of ρ . As noted in (Chen and Marcussen, 2023), algorithms for $\{0, 1\}^n$ do not immediately translate into algorithms for Σ^n , because the SUBCOND oracle does not work with the natural reduction of replacing elements $c \in \Sigma$ with their binary encoding. Nevertheless, **DistEstimateCore** can be extended to distributions over Σ^n , incurring a linear dependence on $|\Sigma|$.

We will now restate our result adapted to the new setting:

Theorem 4.8. *Given two distributions \mathcal{P} and \mathcal{Q} over Σ^n , along with parameters $\varepsilon \in (0, 1)$, $\delta \in (0, 1/2)$, the algorithm **DistEstimate**($\mathcal{P}, \mathcal{Q}, \varepsilon, \delta$), and with probability at least $1 - \delta$ returns κ , s.t.*

$$\Pr[\kappa \in (d_{TV}(\mathcal{P}, \mathcal{Q}) \pm \varepsilon)] \geq 1 - \delta$$

DistEstimate($\mathcal{P}, \mathcal{Q}, \varepsilon, \delta$) makes $\tilde{O}(n^3 |\Sigma| \log(1/\delta)/\varepsilon^5)$ SUBCOND queries.

The only change required in **DistEstimate** to make it work for distributions over the Σ^n is in the construction of the tamed siblings \mathcal{P}' , and \mathcal{Q}' . We update the taming parameter from $\varepsilon/8n$ to $\varepsilon/8n|\Sigma|$. Since the query complexity is proportional to $1/\theta$, we observe a linear dependence on $|\Sigma|$.

5 EXPERIMENTS

We implemented **DistEstimate** in Python. We focus on distributions generated by state-of-the-art combinatorial samplers STS (Ermon et al., 2012) and CMSGen (Golia et al., 2021). Our assessment included two datasets: (1) **scalable** comprising random Boolean

functions over n variables, with n ranging from 30 to 70, and (2) **real-world**, containing instances from the ISCAS89 dataset, a standard in combinatorial testing and sampling evaluations (Meel, 2020). To determine the ground truth TV distance for the above instances, we implement a learning-based distance estimator Canonne (2020b).

For our experiments, we set the tolerance $\varepsilon = 0.3$ and confidence $\delta = 0.4$ as the default throughout the evaluation. These parameters indicate that the estimate returned by **DistEstimate** is expected to be within ± 0.3 of the ground truth, with a probability of at least 0.6.

The experiments were conducted on a cluster with AMD EPYC 7713 CPU cores. We use 32 cores with 4GB of memory for each benchmark and a 24-hour timeout per instance.

Our aim was to answer the question: To what extent does **DistEstimate** scale, i.e., how many dimensions can the estimator handle while providing guarantees?

We found that **DistEstimate** scales to $n = 70$ dimensional problems, a regime where the baseline sample-based estimators would require $10^7 \times$ more samples. Empirically we found the mean absolute difference between the predicted TV distance and ground truth to be 0.09, which is smaller than the allowed tolerance(ε) of 0.3. The prediction was within the tolerance band in all cases, whereas our confidence threshold was 0.6.

Table 1 details the performance of **DistEstimate** on 6 **real-world** benchmarks. The algorithm successfully finished on all benchmarks with dimensionality up to $n = 37$. The table specifies the benchmark name, dimensionality, sample count, and processing time for both STS and CMSGen.

The sample complexity of **DistEstimate** relative to a baseline sample-based estimator is illustrated in Figure 1 (in Section 1). For this, we use **scalable** benchmarks. Remarkably, for the largest instance handled ($n = 70$ dimensions), **DistEstimate** outperformed the baseline by a factor greater than 10^7 .

6 CONCLUSION

This paper focused on the distance estimation problem in the SUBCOND model. We sought to alleviate the significant weakness of the prior state of the art: the estimators required an exponentially large number of queries. Our primary contribution, **DistEstimate**, enables distance estimation in $\mathcal{O}(n^3/\varepsilon^5)$ queries. We also implemented **DistEstimate** and tested it on distributions generated by combinatorial samplers, showing the scalability of our approach. The problem of closing the gap between the $\mathcal{O}(n^3/\varepsilon^5)$ upper bound and the $\Omega(n/\log(n))$ lower bound, remains open in all COND models.

7 ACKNOWLEDGEMENTS

This research is part of the programme DesCartes and is supported by the National Research Foundation, Prime Minister's Office, Singapore, under its Campus for Research Excellence and Technological Enterprise (CREATE) programme. The computational works of this article were performed on the resources of the National Supercomputing Centre, Singapore(www.nscc.sg).

The authors decided to forgo the old convention of alphabetical ordering of authors in favor of a randomized ordering, denoted by \textcircled{R} .

References

- Jayadev Acharya, Clément L. Canonne, and Gautam Kamath. A chasm between identity and equivalence testing with conditional queries. *Electron. Colloquium Comput. Complex.*, 2014.
- Alejandro Aguirre, Gilles Barthe, Justin Hsu, Benjamin Lucien Kaminski, Joost-Pieter Katoen, and Christoph Matheja. A pre-expectation calculus for probabilistic sensitivity. *Programming Languages*, (POPL), 2021.
- Christophe Andrieu, Nando De Freitas, Arnaud Doucet, and Michael I Jordan. An introduction to mcmc for machine learning. *Machine learning*, 2003.
- Arnab Bhattacharyya, Sutanu Gayen, Kuldeep S. Meel, and N. V. Vinodchandran. Efficient distance approximation for structured high-dimensional distributions via learning. *ArXiv*, abs/2002.05378, 2020.
- Arnab Bhattacharyya, Sutanu Gayen, Kuldeep S. Meel, Dimitrios Myrisiotis, A. Pavan, and N. V. Vinodchandran. On approximating total variation distance. In *International Joint Conference on Artificial Intelligence, IJCAI 2023*, 2023a.
- Arnab Bhattacharyya, Sutanu Gayen, Kuldeep S. Meel, Dimitrios Myrisiotis, A. Pavan, and N. V. Vinodchandran. Total variation distance estimation is as easy as probabilistic inference, 2023b.
- Rishiraj Bhattacharyya and Sourav Chakraborty. Property testing of joint distributions using conditional samples. *ACM Transactions on Computation Theory (TOCT)*, 2018.
- Stephen Boyd, Persi Diaconis, and Lin Xiao. Fastest mixing markov chain on a graph. *SIAM review*, 2004.
- Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng. *Handbook of markov chain monte carlo*. CRC press, 2011.
- Daniel G Brown. How i wasted too long finding a concentration inequality for sums of geometric variables. 2011.
- Clément L Canonne. A survey on distribution testing: Your data is big. but is it blue? *Theory of Computing*, 2020a.
- Clément L Canonne. A short note on learning discrete distributions. *arXiv preprint arXiv:2002.11457*, 2020b.
- Clément L Canonne, Dana Ron, and Rocco A Sereddio. Testing probability distributions using conditional samples. *SIAM Journal on Computing*, 2015.
- Clément L Canonne, Xi Chen, Gautam Kamath, Amit Levi, and Erik Waingarten. Random restrictions of high dimensional distributions and uniformity testing with subcube conditioning. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*. SIAM, 2021.
- Clément L. Canonne, Ilias Diakonikolas, Daniel M. Kane, and Alistair Stewart. Testing bayesian networks. *IEEE Transactions on Information Theory*, 2020.
- Sourav Chakraborty, Eldar Fischer, Yonatan Goldhirsh, and Arie Matsliah. On the power of conditional samples in distribution testing. *SIAM Journal on Computing*, 2016.
- Siu-On Chan, Ilias Diakonikolas, Paul Valiant, and Gregory Valiant. Optimal algorithms for testing closeness of discrete distributions. In *Symposium on Discrete Algorithms*. SIAM, 2014.
- Surajit Chaudhuri, Rajeev Motwani, and Vivek Narasayya. On random sampling over joins. *ACM SIGMOD Record*, 1999.
- Xi Chen and Cassandra Marcussen. Uniformity testing over hypergrids with subcube conditioning. *arXiv preprint arXiv:2302.09013*, 2023.

-
- Xi Chen, Rajesh Jayaram, Amit Levi, and Erik Waingarten. Learning and testing junta distributions with sub cube conditioning. In *Conference on Learning Theory*. PMLR, 2021.
- Martin Dyer and Alan Frieze. Computing the volume of convex bodies: a case where randomness provably helps. *Probabilistic combinatorics and its applications*, 1991.
- Stefano Ermon, Carla P. Gomes, and Bart Selman. Uniform solution sampling using a constraint solver as an oracle. In *UAI*, 2012.
- Moein Falahatgar, Ashkan Jafarpour, Alon Orlitsky, Venkatadheeraj Pichapati, and Ananda Theertha Suresh. Faster algorithms for testing under conditional sampling. In *Conference on Learning Theory*. PMLR, 2015.
- Priyanka Golia, Mate Soos, Sourav Chakraborty, and Kuldeep S. Meel. Designing samplers is easy: The boon of testers. In *Proceedings of Formal Methods in Computer-Aided Design (FMCAD)*, 8 2021.
- Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.
- Mark R Jerrum, Leslie G Valiant, and Vijay V Vazirani. Random generation of combinatorial structures from a uniform distribution. *Theoretical computer science*, 1986.
- Haozhe Ji, Pei Ke, Zhipeng Hu, Rongsheng Zhang, and Minlie Huang. Tailoring language generation models under total variation distance. *ArXiv*, abs/2302.13344, 2023.
- N Kalchbrenner. Neural machine translation in linear time. *arXiv preprint arXiv:1610.10099*, 2016.
- Kuldeep S. Meel. Model counting and uniform sampling instances, 2020. URL <https://zenodo.org/record/3793090>.
- Liam Paninski. A coincidence-based test for uniformity given very sparsely sampled discrete data. *IEEE Transactions on Information Theory*, 54(10): 4750–4755, 2008.
- Yash Pote and Kuldeep S Meel. On scalable testing of samplers. In *Advances in Neural Information Processing Systems*, volume 35, 2022.
- Gregory Valiant and Paul Valiant. Estimating the unseen: A sublinear-sample canonical estimator of distributions. In *Electronic Colloquium on Computational Complexity (ECCC)*, volume 17, page 9, 2010.
- Gregory Valiant and Paul Valiant. The power of linear estimators. In *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, 2011.
- Gregory Valiant and Paul Valiant. An automatic inequality prover and instance optimal identity testing. *SIAM Journal on Computing*, 46(1):429–455, 2017.
- Aäron Van Den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *International conference on machine learning*, pages 1747–1756. PMLR, 2016.
- Zhuoyue Zhao, Robert Christensen, Feifei Li, Xiao Hu, and Ke Yi. Random sampling over joins revisited. In *Proceedings of the 2018 International Conference on Management of Data*, 2018.

Checklist

1. For all models and algorithms presented, check if you include:
 - (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
 - (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
 - (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [No]
2. For any theoretical claim, check if you include:
 - (a) Statements of the full set of assumptions of all theoretical results. [Yes]
 - (b) Complete proofs of all theoretical results. [Yes]
 - (c) Clear explanations of any assumptions. [Yes]
3. For all figures and tables that present empirical results, check if you include:
 - (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Not Applicable]
 - (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
 - (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Not Applicable]
 - (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
 - (a) Citations of the creator If your work uses existing assets. [Yes]
 - (b) The license information of the assets, if applicable. [Not Applicable]
 - (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
 - (d) Information about consent from data providers/curators. [Not Applicable]
 - (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]
5. If you used crowdsourcing or conducted research with human subjects, check if you include:
 - (a) The full text of instructions given to participants and screenshots. [Not Applicable]
 - (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
 - (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

A Appendix

A.1 Useful Inequalities

Lemma A.1 (Chernoff). *For any $\gamma, \delta \in (0, 0.5]$, let $n \geq \log(2/\delta)/2\gamma^2$, and let x_1, x_2, \dots, x_n be i.i.d. variables taking value in $(0, 1]$, with mean $\mathbb{E}[x]$, then*

$$\Pr \left[\left| \frac{1}{n} \sum_{i=1}^n x_i - \mathbb{E}[x] \right| < \gamma \right] \geq 1 - \delta$$

Lemma A.2 (Chebyshev). *Let X be a random variable with $\mathbb{E}[X^2] < \infty$. For any $t > 0$, we have $\Pr[|\mathbb{E}[X] - X| \leq t] \leq \mathbb{V}[X]/t^2$*

A.2 Lower Bound

To complement the upper bound shown in the main paper, we show that the best-known lower bound for the problem in Theorem 1.1, is $\Omega(n/\log(n))$. This bound is from [Canonne et al. \(2020\)](#).

Theorem A.3 (Theorem 11 in [\(Canonne et al., 2020\)](#)). *An absolute constant $\varepsilon_0 < 1$ exists, such as the following holds. Any algorithm that, given a parameter $\varepsilon \in (0, \varepsilon_0]$, and sample access to product distributions \mathcal{P}, \mathcal{Q} over $\{0, 1\}^n$, distinguishes between $d_{TV}(\mathcal{P}, \mathcal{Q}) < \varepsilon$ and $d_{TV}(\mathcal{P}, \mathcal{Q}) > 2\varepsilon$, with probability at least $2/3$, requires $\Omega(n/\log(n))$ samples. Moreover, the lower bound still holds in the case where \mathcal{Q} is known, and provided as an explicit parameter.*

The lower bound is shown for the case where the tester has access to samples from a product distribution \mathcal{P} and \mathcal{Q} (over $\{0, 1\}^n$). As observed by [Bhattacharyya and Chakraborty \(2018\)](#), SUBCOND access is no stronger than SAMP when it comes to product distributions. Thus we have the following lower bound:

Corollary 1. Let $\mathcal{S}(\varepsilon_1, \varepsilon_2, \mathcal{P}, \mathcal{Q})$ be any algorithm that has SUBCOND access to distribution \mathcal{P} , and explicit knowledge of \mathcal{Q} (defined over $\{0, 1\}^n$), and distinguishes between $d_{TV}(\mathcal{P}, \mathcal{Q}) \leq \varepsilon_1$ and $d_{TV}(\mathcal{P}, \mathcal{Q}) > \varepsilon_2$ with probability $> 2/3$. Then, \mathcal{S} makes $\Omega(n/\log(n))$ SUBCOND queries.

A.3 Proof of Lemma 3.5

Lemma 3.5. (Theorem 3.1 in (Bhattacharyya et al., 2020)) For distributions \mathcal{P} and \mathcal{Q} over $\{0,1\}^n$, and $\sigma \in \{0,1\}^n$, let p_σ and q_σ be functions such that $p_\sigma \in (1 \pm \eta)\mathcal{P}(\sigma)$, and $q_\sigma \in (1 \pm \eta)\mathcal{Q}(\sigma)$. Given a set of samples S from \mathcal{Q} , and $\eta \in (0,1)$ along with the p_σ and q_σ for each $\sigma \in S$, let $\mathbf{est} = \frac{1}{|S|} \sum_{i \in S} \mathbb{1}_{q_\sigma > p_\sigma} \left(1 - \frac{p_\sigma}{q_\sigma}\right)$.

$$\begin{aligned} & \Pr \left[\mathbf{est} \notin \left(d_{TV}(\mathcal{P}, \mathcal{Q}) \pm \frac{3\eta}{1-\eta} \right) \right] \\ & \leq 2 \exp \left(-2|S| \left(\frac{\eta}{1-\eta} \right)^2 \right) \end{aligned}$$

Proof. Recall that $p_\sigma \in (1 \pm \eta)\mathcal{P}(\sigma)$ and $q_\sigma \in (1 \pm \eta)\mathcal{Q}(\sigma)$ then, using the definition of $d_{TV}(\mathcal{P}, \mathcal{Q})$,

$$\begin{aligned} d_{TV}(\mathcal{P}, \mathcal{Q}) &= \sum_{\sigma \in \{0,1\}^n} \mathbb{1}_{\mathcal{Q}(\sigma) > \mathcal{P}(\sigma)} \left(1 - \frac{\mathcal{P}(\sigma)}{\mathcal{Q}(\sigma)} \right) \mathcal{Q}(\sigma) \\ &= \sum_{\sigma \in \{0,1\}^n} \mathbb{1}_{q_\sigma > p_\sigma} \left(1 - \frac{p_\sigma}{q_\sigma} \right) \mathcal{Q}(\sigma) \\ &+ \underbrace{\sum_{\sigma \in \{0,1\}^n} \left(\mathbb{1}_{\mathcal{Q}(\sigma) > \mathcal{P}(\sigma)} \left(1 - \frac{\mathcal{P}(\sigma)}{\mathcal{Q}(\sigma)} \right) \mathcal{Q}(\sigma) - \mathbb{1}_{q_\sigma > p_\sigma} \left(1 - \frac{p_\sigma}{q_\sigma} \right) \mathcal{Q}(\sigma) \right)}_A \end{aligned} \quad (3)$$

The first summand of (3) can be written as $\mathbb{E}_{\sigma \sim \mathcal{Q}} \left[\mathbb{1}_{q_\sigma > p_\sigma} \left(1 - \frac{p_\sigma}{q_\sigma} \right) \right]$.

To bound $|A|$, we will split the domain into three sets, $B_1 = \{x : \mathbb{1}_{\mathcal{Q}(\sigma) > \mathcal{P}(\sigma)} = \mathbb{1}_{q_\sigma > p_\sigma}\}$, $B_2 = \{x : \mathbb{1}_{\mathcal{Q}(\sigma) > \mathcal{P}(\sigma)} > \mathbb{1}_{q_\sigma > p_\sigma}\}$ and $B_3 = \{x : \mathbb{1}_{\mathcal{Q}(\sigma) > \mathcal{P}(\sigma)} < \mathbb{1}_{q_\sigma > p_\sigma}\}$.

$$\begin{aligned} |A| &= \left| \sum_{\sigma \in \{0,1\}^n} \left(\mathbb{1}_{\mathcal{Q}(\sigma) > \mathcal{P}(\sigma)} \left(1 - \frac{\mathcal{P}(\sigma)}{\mathcal{Q}(\sigma)} \right) \mathcal{Q}(\sigma) - \mathbb{1}_{q_\sigma > p_\sigma} \left(1 - \frac{p_\sigma}{q_\sigma} \right) \mathcal{Q}(\sigma) \right) \right| \\ &\leq \sum_{\sigma \in \{0,1\}^n} \left| \left(\mathbb{1}_{\mathcal{Q}(\sigma) > \mathcal{P}(\sigma)} \left(1 - \frac{\mathcal{P}(\sigma)}{\mathcal{Q}(\sigma)} \right) \mathcal{Q}(\sigma) - \mathbb{1}_{q_\sigma > p_\sigma} \left(1 - \frac{p_\sigma}{q_\sigma} \right) \mathcal{Q}(\sigma) \right) \right| \\ &= \sum_{\sigma \in B_1} \mathbb{1}_{\mathcal{Q}(\sigma) > \mathcal{P}(\sigma)} \left| \frac{\mathcal{P}(\sigma)}{\mathcal{Q}(\sigma)} - \frac{p_\sigma}{q_\sigma} \right| \mathcal{Q}(\sigma) + \sum_{\sigma \in B_2} \mathbb{1}_{\mathcal{Q}(\sigma) > \mathcal{P}(\sigma)} \left(1 - \frac{\mathcal{P}(\sigma)}{\mathcal{Q}(\sigma)} \right) \mathcal{Q}(\sigma) \\ &+ \sum_{\sigma \in B_3} \mathbb{1}_{q_\sigma > p_\sigma} \left(1 - \frac{p_\sigma}{q_\sigma} \right) \mathcal{Q}(\sigma) \end{aligned}$$

For $\sigma \in B_1$, $\left| \frac{\mathcal{P}(\sigma)}{\mathcal{Q}(\sigma)} - \frac{p_\sigma}{q_\sigma} \right| \leq \frac{2\eta}{1-\eta} \frac{\mathcal{P}(\sigma)}{\mathcal{Q}(\sigma)} \leq \frac{2\eta}{1-\eta}$. For $\sigma \in B_2$, $1 - \frac{\mathcal{P}(\sigma)}{\mathcal{Q}(\sigma)} \leq 1 - \frac{1-\eta}{1+\eta} = \frac{2\eta}{1+\eta}$, and for $\sigma \in B_3$, $1 - \frac{p_\sigma}{q_\sigma} \leq 1 - \frac{1-\eta}{1+\eta} = \frac{2\eta}{1+\eta}$. Thus, $|A| \leq \sum_{\sigma \in B_1} \frac{2\eta}{1-\eta} \mathcal{Q}(\sigma) + \sum_{\sigma \in B_2} \frac{2\eta}{1+\eta} \mathcal{Q}(\sigma) + \sum_{\sigma \in B_3} \frac{2\eta}{1+\eta} \mathcal{Q}(\sigma) \leq \frac{2\eta}{1-\eta}$. Plugging the bounds on $|A|$ back into (3), we get

$$\left| d_{TV}(\mathcal{P}, \mathcal{Q}) - \mathbb{E} \left[\mathbb{1}_{q_\sigma > p_\sigma} \left(1 - \frac{p_\sigma}{q_\sigma} \right) \right] \right| \leq \frac{2\eta}{1-\eta} \quad (4)$$

And hence, $\mathbb{E} \left[\mathbb{1}_{q_\sigma > p_\sigma} \left(1 - \frac{p_\sigma}{q_\sigma} \right) \right] - \frac{2\eta}{1-\eta} \leq d_{TV}(\mathcal{P}, \mathcal{Q}) \leq \mathbb{E} \left[\mathbb{1}_{q_\sigma > p_\sigma} \left(1 - \frac{p_\sigma}{q_\sigma} \right) \right] + \frac{2\eta}{1-\eta}$. The distance estimation algorithm draws $|S|$ samples to estimate $\mathbb{E} \left[\mathbb{1}_{q_\sigma > p_\sigma} \left(1 - \frac{p_\sigma}{q_\sigma} \right) \right]$. We will use \mathbf{est} to denote the empirical estimate of $\mathbb{E} \left[\mathbb{1}_{q_\sigma > p_\sigma} \left(1 - \frac{p_\sigma}{q_\sigma} \right) \right]$. Since each sample σ is drawn independently, and $\mathbb{1}_{q_\sigma > p_\sigma} \left(1 - \frac{p_\sigma}{q_\sigma} \right)$ is bounded in $[0, 1]$,

we can use the Hoeffding bound as follows,

$$\Pr \left[\left| \mathbf{est} - \mathbb{E} \left[\mathbb{1}_{q_\sigma > p_\sigma} \left(1 - \frac{p_\sigma}{q_\sigma} \right) \right] \right| \geq \frac{\eta}{1 - \eta} \right] \leq 1 - 2 \exp \left(-2|S| \left(\frac{\eta}{1 - \eta} \right)^2 \right) \quad (5)$$

Plugging (4) into (5), we complete the proof:

$$\Pr \left[|\mathbf{est} - d_{TV}(\mathcal{P}, \mathcal{Q})| \geq \frac{3\eta}{1 - \eta} \right] = \Pr \left[d_{TV}(\mathcal{P}, \mathcal{Q}) \notin \left(\mathbf{est} \pm \frac{3\eta}{1 - \eta} \right) \right] \leq 2 \exp \left(-2|S| \left(\frac{\eta}{1 - \eta} \right)^2 \right)$$

□

B Proof of Lemma 3.8

Lemma 3.8. *Given a distribution \mathcal{D} and parameter $\theta \in [0, 1/n)$, every CM query to \mathcal{D}' can be simulated by making one SUBCOND query to \mathcal{D} , and every SAMP query to \mathcal{D}' can be simulated by making n SUBCOND queries to \mathcal{D} .*

Proof. Our proof adapts the θ -balancing trick, devised for product distributions in [Canonne et al. \(2020, Thm. 6\)](#). To simulate the CM($\mathcal{D}', \sigma_{<\ell}$) query using SUBCOND access to \mathcal{D} , we use the following process: *all* $i \geq \ell$, given the substring $\sigma_{<i}$, set $\sigma_i = 0$ with probability $(1-2\theta)\mathcal{D}_{\sigma_{<i}}(0) + \theta$ and $\sigma_i = 1$ with probability $(1-2\theta)\mathcal{D}_{\sigma_{<i}}(1) + \theta$. To implement the above, with probability $1-2\theta$, draw $\rho \sim \text{SUBCOND}(\mathcal{D}, \sigma_{<i})$ and return ρ_i , else with probability 2θ draw a sample uniformly from $\{0, 1\}$.

Observe that *all* $\ell \in [n]$, $c \in \{0, 1\}$, and $\rho \in \{0, 1\}^{\ell-1}$, we have $\mathcal{D}'_\rho(c) = (1-2\theta)\mathcal{D}_\rho(c) + \theta$. Since $\theta \leq \mathcal{D}'_\rho(c) \leq 1-\theta$, we see that \mathcal{D}' is indeed θ -tamed. To simulate SAMP(\mathcal{D}'), we use the chain rule.

Now we will show that \mathcal{D}' is close to \mathcal{D} .

Claim B.1. *For distribution \mathcal{D} and its θ -tamed sibling \mathcal{D}' , we have $d_{TV}(\mathcal{D}, \mathcal{D}') \leq \theta n$*

Proof. Recall the definition of subcube $S_\rho = \{w \in \{0, 1\}^n : w_{\leq |\rho|} = \rho\}$. For any set $S \subseteq \{0, 1\}^n$, $\mathcal{D}(S)$ is the total probability of S in \mathcal{D} . For any distribution \mathcal{D} , string ρ (with $1 \leq |\rho| \leq n$) and $\omega \in \{0, 1\}^{n-|\rho|}$, the distribution \mathcal{D}^ρ denotes the marginal distribution of SUBCOND(\mathcal{D}, ρ) in the remaining dimensions, i.e. for any $\omega \in \{0, 1\}^{n-|\rho|}$, $\mathcal{D}^\rho(\omega) = \Pr_{w \sim \text{SUBCOND}(\mathcal{D}, \rho)}[w = \rho\omega]$.

Consider the induction hypothesis that $d_{TV}(\mathcal{D}, \mathcal{D}') \leq \theta i$ if \mathcal{D} is supported on $\{0, 1\}^i$. To verify the hypothesis for $i = 1$, wlog assume that $\mathcal{D}(0) \leq \mathcal{D}(1)$, then $d_{TV}(\mathcal{D}, \mathcal{D}') = \mathcal{D}(1) - \mathcal{D}'(1) = 2\theta\mathcal{D}(1) - \theta \leq \theta$. Assume the hypothesis holds for all $i \in [n-1]$. Now, we show the hypothesis is true for $i = n$.

Consider a distribution \mathcal{D} over $\{0, 1\}^n$ and its θ -tamed sibling \mathcal{D}' , then:

$$\begin{aligned}
d_{TV}(\mathcal{D}, \mathcal{D}') &= \frac{1}{2} \sum_{\sigma \in \{0, 1\}^n} |\mathcal{D}(\sigma) - \mathcal{D}'(\sigma)| = \frac{1}{2} \sum_{\rho \in \{0, 1\}} \sum_{\omega \in \{0, 1\}^{n-1}} |\mathcal{D}(\rho\omega) - \mathcal{D}'(\rho\omega)| \\
&= \frac{1}{2} \sum_{\rho \in \{0, 1\}} \sum_{\omega \in \{0, 1\}^{n-1}} |\mathcal{D}(S_\rho)\mathcal{D}^\rho(\omega) - \mathcal{D}'(S_\rho)\mathcal{D}'^\rho(\omega)| \\
&= \frac{1}{2} \sum_{\rho \in \{0, 1\}} \sum_{\omega \in \{0, 1\}^{n-1}} |\mathcal{D}(S_\rho)\mathcal{D}^\rho(\omega) - \mathcal{D}(S_\rho)\mathcal{D}'^\rho(\omega) + \mathcal{D}(S_\rho)\mathcal{D}'^\rho(\omega) - \mathcal{D}'(S_\rho)\mathcal{D}'^\rho(\omega)| \\
&\leq \frac{1}{2} \sum_{\rho \in \{0, 1\}} \sum_{\omega \in \{0, 1\}^{n-1}} |\mathcal{D}(S_\rho)\mathcal{D}^\rho(\omega) - \mathcal{D}(S_\rho)\mathcal{D}'^\rho(\omega)| + |\mathcal{D}(S_\rho)\mathcal{D}'^\rho(\omega) - \mathcal{D}'(S_\rho)\mathcal{D}'^\rho(\omega)| \\
&= \frac{1}{2} \sum_{\rho \in \{0, 1\}} \sum_{\omega \in \{0, 1\}^{n-1}} \mathcal{D}(S_\rho)|\mathcal{D}^\rho(\omega) - \mathcal{D}'^\rho(\omega)| + \mathcal{D}'_\rho(\omega)|\mathcal{D}'(S_\rho) - \mathcal{D}(S_\rho)| \\
&= \frac{1}{2} \sum_{\rho \in \{0, 1\}} (\mathcal{D}(S_\rho)2d_{TV}(\mathcal{D}^\rho, \mathcal{D}'^\rho)) + \frac{1}{2} \sum_{\rho \in \{0, 1\}} |\mathcal{D}'(S_\rho) - \mathcal{D}(S_\rho)| \\
&\leq \sum_{\rho \in \{0, 1\}} (\mathcal{D}(S_\rho)\theta(n-1)) + \theta = \theta n
\end{aligned}$$

We use $|a+b| \leq |a|+|b|$ in the first inequality. In the second, we use the induction hypothesis to bound the first summand, and for the second, we observe that for $c \in \{0, 1\}$, $|\mathcal{D}'(c) - \mathcal{D}(c)| \leq \theta$. \square

\square

C Proof of Claim 4.2, Proposition 4.5 and Lemma 4.7

Claim 4.2. $\Pr[\text{Bad}_i^{\hat{p}}] \leq 1/24m_{out}$ and $\Pr[\text{Bad}_i^{\hat{q}}] \leq 1/24m_{out}$.

Proof. For a fixed iteration j , applying Lemma 4.1 we have $\Pr[p_j \in (1 \pm \eta) \mathcal{P}'(\sigma)] \geq 5/8$. Since \hat{p} is the median of independent observations $p_j \in [0, 1]$, over $j \in [m_{in}]$, we can use the Chernoff bound to derive the claimed bound, $\Pr[\text{Bad}_i^{\hat{p}}] \leq 1/24m_{out}$. The proof for the claim $\Pr[\text{Bad}_i^{\hat{q}}] \leq 1/24m_{out}$ proceeds identically. \square

Proposition 4.5. For $i \in [n]$, the value of x_i (in Alg. 4) is distributed as $\text{NB}(k, \mathcal{D}_{\sigma_{< i}}(\sigma_i))$

Proof. Fix any $i \in [n]$. In Alg. 4, the r.v α takes the value σ_i with probability $\mathcal{D}_{\sigma_{< i}}(\sigma_i)$. Note that while the value of x_i increments by one in every iteration of the loop (lines 6-11), while the value of f increases by one only when $\alpha = \sigma_i$. Since the loop runs until the value of f is k , the distribution of x_i is $\text{NB}(k, \mathcal{D}_{\sigma_{< i}}(\sigma_i))$. \square

Lemma 4.7. $\Pr[t_1 \geq 64n^3\eta^{-2}(1 + \eta)^2\varepsilon^{-1}] \leq 1/24$

Proof. The number of CM calls made by SubToEval_1 in the i^{th} iteration is captured by x_i . Recall from Prop 4.5 that x_i is drawn from $\text{NB}(k, \mathcal{D}'_{\sigma_{< i}}(\sigma_i))$, and therefore we have,

$$\mathbb{E}[x_i] = k/\mathcal{D}'_{\sigma_{< i}}(\sigma_i) = 4n\eta^{-2}(1 + \eta)^2/\mathcal{D}'_{\sigma_{< i}}(\sigma_i) \quad (\text{Using } k \text{ from Line 2 of SubToEval}_1)$$

From the fact that the distribution is $\varepsilon/8n$ -tamed, we know that $\mathcal{D}'_{\sigma_{< i}}(\sigma_i) \geq \varepsilon/8n$. Hence we have $\mathbb{E}[x_i] \leq 32n^2\eta^{-2}(1 + \eta)^2\varepsilon^{-1}$. Since $t_1 = \sum_{i \in [n]} x_i$, we have that $\mathbb{E}[t_1] = \mathbb{E}[\sum_{i \in [n]} x_i] = n\mathbb{E}[x_i] \leq 32n^3\eta^{-2}(1 + \eta)^2\varepsilon^{-1}$. Thus,

$$\begin{aligned} \Pr[t_1 \geq 64n^3\eta^{-2}(1 + \eta)^2\varepsilon^{-1}] &= \Pr[t_1 \geq 2\mathbb{E}[t_1]] \leq \Pr\left[\sum_{i \in [n]} x_i \geq 2\mathbb{E}\left[\sum_{i \in [n]} x_i\right]\right] \\ &\leq \sum_{i \in [n]} \Pr[x_i \geq 2\mathbb{E}[x_i]] \\ (\text{Prop. 3.1}) \quad &\leq \sum_{i \in [n]} \exp(-2k(1 - 1/2)^2/2) = n \exp(-k/4) \\ (\text{Substituting } k \text{ and } \eta \leq 1/5, \varepsilon < 1) \quad &\leq n \exp(-n\eta^{-2}(1 + \eta)^2\varepsilon^{-1}) \leq n \exp(-9n) \leq 1/24 \end{aligned}$$

In the last inequality we used the fact that for $s > 0$, $xe^{-sx} \leq 1/es$. \square