Support Size Estimation: The Power of Conditioning

Diptarka Chakraborty
National University of Singapore, Singapore

Gunjan Kumar
National University of Singapore, Singapore

Kuldeep S. Meel
National University of Singapore, Singapore

Abstract

We consider the problem of estimating the support size of a distribution D. Our investigations are pursued through the lens of distribution testing and seek to understand the power of conditional sampling (denoted as COND), wherein one is allowed to query the given distribution conditioned on an arbitrary subset S. The primary contribution of this work is to introduce a new approach to lower bounds for the COND model that relies on using powerful tools from information theory and communication complexity.

Our approach allows us to obtain surprisingly strong lower bounds for the COND model and its extensions.

- We bridge the longstanding gap between the upper bound $O\left(\log\log n + \frac{1}{\epsilon^2}\right)$ and the lower bound $\Omega\left(\sqrt{\log\log n}\right)$ for the COND model by providing a nearly matching lower bound. Surprisingly, we show that even if we get to know the actual probabilities along with COND samples, still $\Omega\left(\log\log n + \frac{1}{\epsilon^2\log(1/\epsilon)}\right)$ queries are necessary.
- We obtain the first non-trivial lower bound for the COND equipped with an additional oracle that reveals the actual as well as the conditional probabilities of the samples (to the best of our knowledge, this subsumes all of the models previously studied): in particular, we demonstrate that $\Omega\left(\log\log\log n + \frac{1}{\epsilon^2\log(1/\epsilon)}\right)$ queries are necessary.

2012 ACM Subject Classification Theory of computation → Streaming, sublinear and near linear time algorithms

Keywords and phrases Support-size estimation, Distribution testing, Conditional sampling, Lower bound

Digital Object Identifier 10.4230/LIPIcs.MFCS.2023.33

Related Version *Full Version*: `https://arxiv.org/abs/2211.11967`

Funding *Diptarka Chakraborty*: Supported in part by an MoE AcRF Tier 2 grant (MOE-T2EP20221-0009).
Gunjan Kumar: Supported in part by National Research Foundation Singapore under its NRF Fellowship Programme[NRF-NRFFAI1-2019-0004].
Kuldeep S. Meel: Supported in part by National Research Foundation Singapore under its NRF Fellowship Programme[NRF-NRFFAI1-2019-0004] , Ministry of Education Singapore Tier 2 grant MOE-T2EP20121-0011, and Ministry of Education Singapore Tier 1 Grant [R-252-000-B59-114].

Acknowledgements The authors would like to thank anonymous reviewers for their useful suggestions and comments on an earlier version of this paper.

1 Introduction

We consider the problem of estimating the support size of a distribution D over a domain Ω (of size n), which is defined as follows:

$$\textsc{supp}(D) := \{x \mid D(x) > 0\}.$$

We are interested in (ϵ, δ)-approximation[1] of the size of $\textsc{supp}(D)$ (i.e., $|\textsc{supp}(D)|$). For simplicity in exposition, throughout this paper, we consider δ to be a small constant (more specifically, 1/3, which can be reduced to any arbitrary small constant.). The support size estimation is a fundamental problem in data science and finds a myriad of applications ranging from database management, biology, ecology, genetics, linguistics, neuroscience, and physics (see [30] and the references therein). Naturally, the distribution is not specified explicitly, and therefore, the complexity of the problem depends on the queries that one is allowed to the distribution. As such, the primary objective is to minimize the number of queries (aka *query complexity*).

Along with support size estimation, several other properties of distributions have attracted investigations over the past three decades (see [9]). As such, several query models have been considered by the research community. The simplest model SAMP only allows drawing independent and identically distributed samples from D. Valiant and Valiant [30] showed that to get an estimation up to an additive factor of ϵn (for any $\epsilon > 0$), $O(n/\epsilon^2 \log n)$ samples suffice, which was subsequently improved to $O(\frac{n}{\log n} \log^2(1/\epsilon))$ by Wu and Yang [31]. Further, Wu and Yang proved that $\Omega(\frac{n}{\log n} \log^2(1/\epsilon))$ samples are also necessary to get an estimate up to an additive error of ϵn. A natural extension to SAMP is called *probability-revealing sample* or PR-SAMP, due to Onak and Sun [27], wherein instead of just returning an independent sample x from D (as in SAMP), the oracle provides a pair $(x, D(x))$ (i.e., a sample along with the probability assigned on it by D). Onak and Sun showed that to estimate the support size up to an additive error of ϵn, $\Theta(1/\epsilon^2)$ samples are necessary and sufficient in the PR-SAMP model. The same upper bound for the PR-SAMP model was also implicit in the work by Canonne and Rubinfeld [8].

As we seek to explore more powerful models than PR-SAMP, a model of interest is DUAL [2, 22, 8] wherein we have access to two oracles: One is SAMP that provides a sample from D, and another is EVAL that given any $x \in \Omega$, outputs the value of $D(x)$. In the DUAL model, for any $\epsilon_1, \epsilon_2 \in (0, 1]$, distinguishing between whether the support size of D is at most $\epsilon_1 n$ or at least $\epsilon_2 n$ requires $\Theta(1/(\epsilon_2 - \epsilon_1)^2)$ queries [8]. An extension of EVAL is CEVAL wherein for totally ordered domains, given x, CEVAL outputs $\sum_{y \preccurlyeq x} D(x)$. Similarly, CDUAL is an extension of DUAL where we have access to oracles SAMP and CEVAL. Caferov, Kaya, O'Donnell, and Say [7] showed that $\Omega(\frac{1}{\epsilon^2})$ queries are needed in the CDUAL model to estimate the support size up to an additive factor of ϵn. However, to the best of our knowledge, no non-trivial result is known for the support size estimation problem with $(1 + \epsilon)$ multiplicative error in the above models.

While SAMP, PR-SAMP, and DUAL are natural models, they are limiting in theory and practice as they fail to capture several scenarios wherein one is allowed more powerful access to the distribution under consideration. Accordingly, CFGM [13] and CRS [11]

[1] We want to estimate $|\textsc{supp}(D)|$ by $\hat{s}$ such that $\frac{|\textsc{supp}(D)|}{(1+\epsilon)} \leq \hat{s} \leq (1 + \epsilon)|\textsc{supp}(D)|$ with the success probability at least $1 - \delta$. This version is also referred to as $(1 + \epsilon)$-multiplicative factor estimation. Another interesting version to consider is the additive ϵn-estimation (where n denotes the size of the domain) which asks to output a $\hat{s}$ such that $|\textsc{supp}(D)| - \epsilon n \leq \hat{s} \leq |\textsc{supp}(D)| + \epsilon n$ with the success probability at least $1 - \delta$. Unless otherwise stated explicitly, we consider the multiplicative variant.

initiated the study of a more general sampling model COND, where we are allowed to draw samples conditioning on any arbitrary subsets of the domain Ω. More specifically, the sampling oracle takes a subset $S \subseteq \Omega$ chosen by the algorithm and returns an element $x \in S$ with probability $D(x)/D(S)$ if $D(S) > 0$. The models proposed by CFGM and CRS differ in their behavior for the case when $D(S) = 0$. The model proposed by CFGM [13] allows the oracle to return a uniformly random element from S when $D(S) = 0$. On the other hand, the COND model defined in CRS [11] assumes that the oracle (and hence the algorithm) returns "failure" and terminates if $D(S) = 0$[2]. Note that the COND model of CRS is more powerful than that of CFGM since, when $D(S) = 0$, in the former case, we get to know that $D(S) = 0$, whereas in the latter case, we get a uniformly random element of S.

The relative power of the COND model of CRS over that of CFGM is also exhibited in the context of support size estimation. Acharya, Canonne, and Kamath [1] designed an algorithm with query complexity $\tilde{O}(\log\log n/\epsilon^3)$ in the COND model of CFGM to estimate the support size up to $(1+\epsilon)$ multiplicative factor under the assumption that the probability of each element is at least $\Omega(1/n)$. They also note that the assumption of a lower bound on the probability of each element is required for their techniques to work. It is worth highlighting that for the problem of support size estimation in the SAMP model, one needs a lower bound of $\Omega(1/n)$ on the minimum probability assigned to any element in the support. Otherwise, one can assign a negligible probability to certain elements which will never be observed. In the COND model, such an assumption is not necessary. Indeed, a result of Falahatgar, Jafarpour, Orlitsky, Pichapati, and Suresh [18] implies that $O\left(\log\log n + \frac{1}{\epsilon^2}\right)$ queries are sufficient in the COND model of CRS for any arbitrary probability distribution, i.e., there is no requirement for the assumption of lower bound on the probability of each element. The key idea behind this result is that it is possible to decide whether the support size is $> 2t$ or $\leq t$ for any integer $t \geq 0$ (for any arbitrary probability distributions) with high probability using only $O(1)$ queries, even with weaker oracle access in which, given any $S \subseteq \Omega$, it can be determined whether $S \cap \textsc{supp}(D) = \emptyset$ or not. On the lower bound side, we only know that at least $\Omega(\sqrt{\log\log n})$ COND queries are necessary [13].

The two models were introduced in the context of uniformity testing, wherein the choice of how to handle the case of $D(S) = 0$ did not make any significant differences. We would like to emphasize that CRS's model is more powerful than that of CFGM, and thus any lower bound shown in the first one also provides the same in the latter one. Moreover, the model of CRS closely approximates the behavior of modern probabilistic programming systems [21]. Therefore, throughout this paper, we consider the COND model of CRS.

Since its introduction, the COND model has attained significant attention both in theory and practice. From a theoretical perspective, various other distribution testing problems have been studied under the COND model [17, 24, 26] and its variant like *subcube conditioning model* [3, 10, 15]. Apart from that, the COND model and its variants find real-world applications in the areas like formal methods and machine learning (e.g., [14, 25, 20]). Also, the modern probabilistic programming systems extend classical programs with the addition of sampling and *observe*, where the semantics of the observe match that of CRS's COND model [21].

[2] Note that analyzing each step of the algorithm makes it possible to determine the set, conditioning on which caused the algorithm to output "failure" and terminate. The rest of the algorithm can then execute with the information that $D(S) = 0$ for the above set S. Therefore, for the simplicity of exposition, we will assume that the COND model defined in CRS returns "failure" when $D(S) = 0$, but the algorithm does not terminate.

It is worth remarking that the COND model is incomparable with PR-SAMP. Therefore, it is quite natural, both from a theory and practical perspective, to consider a sampling model that inherits power from both COND and PR-SAMP. We consider a model where we are allowed to condition on any arbitrary subset $S \subseteq \Omega$, and if $D(S) > 0$, we receive a sample $x \in S$ with probability $D(x)/D(S)$ (as in COND) along with the probability assigned on it by D (i.e., $D(x)$); "failure" otherwise. We refer to this model as *probability-revealing conditional sample*[3] or in short COND-PR. To the best of our knowledge, Golia, Juba, and Meel [20] were the first ones to initiate the study of the COND-PR model. Their work focused on the multiplicative estimation of entropy on the COND-PR model. Golia et al. were primarily motivated to investigate the COND-PR model upon the observation that the usage of the model counter and a sampler can simulate the COND-PR model wherein a circuit specifies the distribution. Also, implicit in their study is that the availability of model counter and samplers [16] allows one to simulate generalization of COND-PR model wherein for a given input D and S in addition to $D(x)$ for a sampled item $x \in S$, the oracle also returns the value of $D(x)/D(S)$ (the conditional probability of x given S). We refer to this model as *conditional sampling evaluation model*[4], or in short COND-EVAL. To the best of our knowledge, the COND-EVAL subsumes all the previously studied variants of the COND model (see Figure 1).

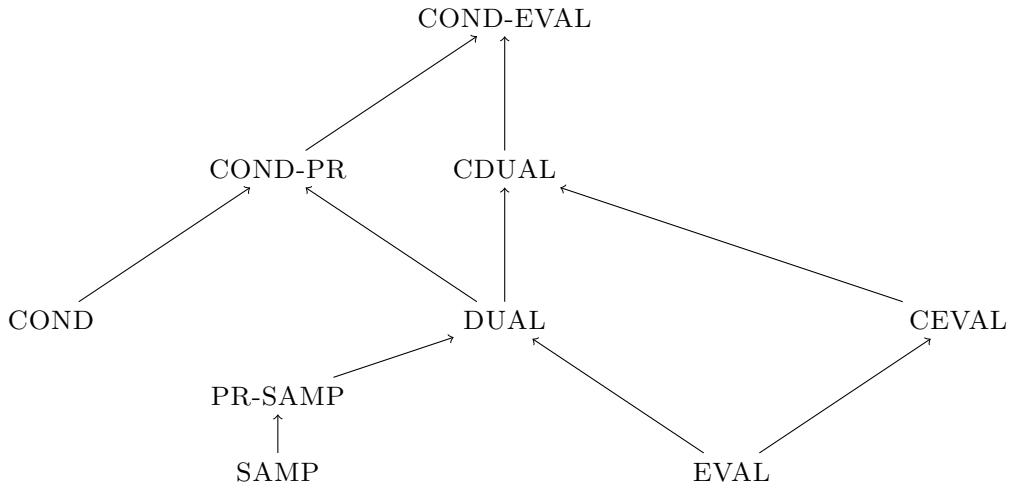

Figure 1 Relative power of different models: An edge $u \to v$ means the model v is more powerful than the model u.

To summarize, there has been a long line of research that has relied on the usage of the COND model and its variants, resulting in significant improvements in the query complexity for several problems in distribution testing. While there has been a multitude of techniques for obtaining upper bounds for the COND model and its variants, such has not been the case for lower bounds. In particular, the prior techniques developed in the context of support size estimation for COND model have primarily relied on the observation that an algorithm $\mathcal{A}$ that makes q COND queries can be simulated by a decision tree of $O(q2^{q^2})$ nodes. Accordingly, the foregoing observation allows one to obtain $\Omega(\sqrt{\log \log n})$ lower bound for

[3] The name is motivated from the PR-SAMP model [27].
[4] The name is motivated from the standard *evaluation* model EVAL [28] where given any $x \in \Omega$, we get the value of the probability density function of D at x.

COND, which leaves open a major gap with respect to the upper bound of $O(\log\log n + \frac{1}{\epsilon^2})$. The situation is even direr when considering models that augment COND with powerful oracles such as EVAL since the approach based on a decision tree fails due to additional information supplied by oracles such as EVAL, and accordingly, no non-trivial lower bounds are known for models such as CDUAL, COND-EVAL, and the like. Therefore, there is a desperate need for new lower bound techniques to fully understand the power of the COND model and its natural extensions.

1.1 Our Contribution

One of our primary contributions is to provide a seemingly new approach to proving lower bounds in the COND model and its more powerful variants. Our approach is based on information theory and reductions to problems in communication complexity. We note that the communication complexity-based approaches to lower bounds have been explored in prior work; such approaches are only limited to weaker models such as the SAMP and the PAIRCOND models [6, 5]. While we demonstrate the application of our approach in the context of support size estimation for different variants of COND, we believe our approach is of general interest and can be applied to other distribution testing problems.

For ease of exposition, here we situate the discussion in the context of the most general model, COND-EVAL. One of the inherent difficulties in proving any non-trivial lower bound for the COND-EVAL model arises from the fact that the different sets for conditioning can overlap in an arbitrary manner (and unlike in PAIRCOND model, these sets are of arbitrary size) and further be chosen in an adaptive way. The adaptivity and arbitrary size of sets make it extremely difficult to upper bound the conditional entropy at any step of the algorithm. Furthermore, the power of revealing the probability mass (on any set) by a COND-EVAL query risks licking "a lot of" information which makes it even more challenging. The key departure from earlier work is the choice of an infinite family of distributions, so the range of outcomes of an algorithm is continuous and so cannot be encoded with any finite (or even infinite) length message. To this end, we rely on Fano's inequality, a fundamental tool in information theory, to show lower bounds for statistical estimation. In order to apply Fano's inequality, we need to upper bound the information gain at every step of the algorithm. Our approach proceeds by relying on a restricted model of conditioning where the queried sets are *laminar*, i.e., either they do not intersect or are subsets/supersets of each other. Accordingly, we first obtain lower bounds for the restricted model and lift to the lower bounds for the COND-EVAL model.

Our approach is compelling enough to provide non-trivial lower bounds in the most potent COND-EVAL model, for which no lower bound was known before. However, before providing the usefulness of the general framework, let us demonstrate how a special instantiation of our approach can be applied to obtain strong lower bounds in the context of support size estimation for the COND model and its (simpler) variants. Our first result bridges the long-standing gap between the upper bound of $O(\log\log n + \frac{1}{\epsilon^2})$ and the lower bound of $\Omega(\sqrt{\log\log n})$ in case of COND model. In particular, we obtain an $\Omega(\log\log n + \frac{1}{\epsilon^2\log(1/\epsilon)})$ lower bound on the query complexity in the COND-PR model, which in turn provides the same lower bound for the PR-SAMP, DUAL, and COND model.

▶ **Theorem 1.** *Every algorithm that, given COND-PR access to a distribution D on $[n]$ and $\epsilon \in (0,1]$, estimates the support size $|\textsc{supp}(D)|$ within a multiplicative $(1+\epsilon)$-factor with probability at least $\frac{2}{3}$, must make $\Omega\left(\log\log n + \frac{1}{\epsilon^2\log(1/\epsilon)}\right)$ queries to the COND-PR oracle.*

Recall the best-known upper bound for the support size estimation in COND is $O(\log\log n + \frac{1}{\epsilon^2})$ and therefore, the above theorem achieves a near-matching lower bound in the context of COND and COND-PR. It is rather surprising that the combination of PR-SAMP and COND does not yield power in the context of the support size estimation. It is worth mentioning that we already know of a lower bound of $\Omega(1/\epsilon^2)$ for the PR-SAMP and DUAL model due to [8]. Hence, our result along with [8] provides a lower bound of $\Omega(\log\log n + \frac{1}{\epsilon^2})$ for these two models (i.e., we can get rid of the annoying $\log(1/\epsilon)$ factor from the lower bound term of Theorem 1).

Our primary result is to establish the first non-trivial lower bound in the context of COND-EVAL, which in turn provides the first-known lower bound for many other previously studied models, such as CDUAL.

▶ **Theorem 2.** *Any algorithm that, given COND-EVAL access to a distribution D on $[n]$ approximates the support size $|\textsc{supp}(D)|$ within a multiplicative $(1+\epsilon)$-factor with probability at least $2/3$, must make $\Omega\left(\log\log\log n + \frac{1}{\epsilon^2\log(1/\epsilon)}\right)$ queries.*

Since COND-EVAL subsumes COND-PR, we have the upper bound of $O(\log\log n + \frac{1}{\epsilon^2})$, and the ensuing gap leaves open an interesting question. It is worth remarking that it is not hard to see (by extending the proof of Theorem 1) that the upper bound is nearly tight if one were to replace COND-EVAL with approximate-COND-EVAL wherein, for a given D and S, the oracle essentially provides an estimate of $D(S)$ up to a small multiplicative error (see the discussion in the full version). We conjecture that the upper bound is tight for COND-EVAL.

▶ **Conjecture 3.** *Any algorithm that, given COND-EVAL access to a distribution D on $[n]$ approximates the support size $|\textsc{supp}(D)|$ within a multiplicative $(1+\epsilon)$-factor with probability at least $2/3$, must make $\Omega\left(\log\log n + \frac{1}{\epsilon^2\log(1/\epsilon)}\right)$ queries.*

The validity of the above conjecture would establish the significant power of the COND model in the context of support size estimation as COND-EVAL and COND-PR, despite being augmented with powerful oracles in addition to conditioning, do not yield better algorithms.

2 Preliminaries

Notations

We use the notation $[n]$ to denote the set of integers $\{1, 2, \cdots, n\}$. For any probability distribution D defined over $[n]$, for any $i \in [n]$, let $D(i)$ denote the probability of choosing i when sampling according to D. For any subset $S \subseteq [n]$, we use $D(S)$ to denote the probability mass assigned on S by the distribution D, i.e., $D(S) := \sum_{i\in S} D(i)$.

Different access models

Let D be a distribution over $[n]$. Below we formally define the query models that we consider in this paper.

▶ **Definition 4** (COND Query Model)**.** *A* conditional *(in short, COND) oracle for D takes as input a set $S \subseteq [n]$, and if $D(S) > 0$, returns an element $j \in S$ with probability $D(j)/D(S)$. If $D(S) = 0$, then the oracle returns "failure".*

▶ **Definition 5** (COND-PR Query Model). *A* probability-revealing conditional sampling *(in short, COND-PR) oracle for D takes as input a set $S \subseteq [n]$, and if $D(S) > 0$, returns a pair $(j, D(j))$ (where $j \in S$) with probability $D(j)/D(S)$. If $D(S) = 0$, then the oracle returns "failure".*

▶ **Definition 6** (COND-EVAL Query Model). *A* conditional evaluation *(in short, COND-EVAL) oracle for D takes as input a set $S \subseteq [n]$, and if $D(S) > 0$, returns a tuple $(j, D(j), D(j)/D(S))$ (where $j \in S$) with probability $D(j)/D(S)$. If $D(S) = 0$, then the oracle returns "failure".*

▶ **Definition 7** (SET-EVAL Query Model). *A* set evaluation *(in short, SET-EVAL) oracle for D takes as input a set $S \subseteq [n]$, and returns the value $D(S)$.*

It is straightforward to observe that the COND-EVAL is at least as powerful as the COND-PR oracle which in turn is at least as powerful as the COND oracle. Further, the COND-EVAL oracle is at least as powerful as the SET-EVAL oracle.[5]

Shannon entropy and source coding theorem

The *entropy* of a discrete random variable X taking values in $\mathcal{X}$ is defined as $H(X) := \sum_{x \in \mathcal{X}} p(x) \log \frac{1}{p(x)}$ where $p(x) = \Pr[X = x]$.

The seminal work of Shannon [29] establishes a connection between the entropy and the expected length of an optimal code that encodes a random variable.

▶ **Theorem 8** (Shannon's Source Coding Theorem [29]). *Let X be a discrete random variable over domain $\mathcal{X}$. Then for every uniquely decodable code $C : \mathcal{X} \to \{0,1\}^*$, $\mathbb{E}(|C(X)|) \geq H(X)$. Moreover, there exists a uniquely decodable code $C : \mathcal{X} \to \{0,1\}^*$ such that $\mathbb{E}(|C(X)|) \leq H(X) + 1$.*

3 Technical Overview

Lower bound for COND-PR

We start with deriving the lower bound of Theorem 1. It consists of two parts – an $\Omega(\log \log n)$ lower bound for a multiplicative 4/3-factor estimation algorithm and an $\Omega(\frac{1}{\epsilon^2 \log \frac{1}{\epsilon}})$ lower bound for an additive ϵn-factor algorithm. We first show an $\Omega(\log \log n)$ lower bound for a multiplicative 4/3-factor algorithm. For that purpose, we consider an *integer-guessing* game between Alice and Bob, where Alice uniformly at random chooses an integer $x \in [\log n]$. Then sends a message (binary string) to Bob. Upon receiving the message, Bob's task is to guess x correctly (with high probability). Since the entropy of the chosen integer is $\log \log n$, by Shannon's source coding theorem, the length of the message, on average, must be $\Omega(\log \log n)$. We show that if there exists an algorithm T that makes t COND-PR queries, then it suffices for Alice to send a message of length $O(t)$, and hence $t = \Omega(\log \log n)$.

To show the same, for each $x \in [\log n]$, Alice considers a distribution D_x with support $[2^x]$. The probability $D_x(j)$ of an element $j \in [2^x]$ in D_x decreases exponentially as j increases. Alice runs the algorithm T on the distribution D_x and would like to send an encoding of the run (i.e., the sampled element along with its probability for each step). Using a trivial

[5] Since on input S, the COND-EVAL oracle returns a tuple $(j, D(j), D(j)/D(S))$ where $j \in S$, one can compute the value of $D(S)$ whenever $D(S) > 0$; otherwise (when $D(S) = 0$) the COND-EVAL oracle returns "failure", from which one can infer that $D(S) = 0$.

encoding, even to send a sampled element, requires $\Theta(\log n)$ bits, which is already more than the Shannon entropy and thus would not give any lower bound. So Alice needs to use a slightly clever encoding. Roughly speaking, since the probabilities are exponentially decreasing, the conditional sampling from any set $S \subseteq [n]$ returns an element from the first "few" smallest elements of S (with high probability). Thus even though $|S|$ can be large, the sampled element (at each step) can be specified by only constantly many bits. Alice sends this encoding of the sampled element (along with its probability value which can again be encoded with constantly many bits due to the construction of D_x's) at each of t steps to Bob. Hence Bob knows the complete run and thus can determine the index x using the algorithm T. We provide detailed proof in the full version.

Next, we turn our attention to showing the dependency of ϵ in the lower bound. We show an $\Omega\left(\frac{1}{\epsilon^2 \log \frac{1}{\epsilon}}\right)$ lower bound for an additive ϵn-factor algorithm. Since a multiplicative $(1+\epsilon)$-factor algorithm also provides an additive ϵn-factor estimation, the above lower bound also works for $(1+\epsilon)$-factor algorithms. We prove this bound by showing a reduction from a well-studied communication complexity problem, namely the *Gap-Hamming distance* problem, and then applying the known lower bound for the Gap-Hamming distance [12]. The proof argument (and the dependence on ϵ) also holds true for COND-EVAL model.

Lower bound for COND-EVAL

The approach used to get an $\Omega(\log \log n)$ lower bound in the COND-PR model cannot be used to show the same for the COND-EVAL model. One of the powers of COND-EVAL model (over the COND-PR) comes from its ability to compute $D(S) := \sum_{j \in S} D(j)$ for any set $S \subseteq [n]$.

Recall the hard instance D_x's used in the $\Omega(\log \log n)$ lower bound proof for the COND-PR model. Let $X^* := \{2^x \mid x \in [\log n]\}$. It is easy to verify that $D_x(X^*) \neq D_{x'}(X^*)$ for any $x \neq x' \in [\log n]$. So, the value of x (and hence the support of D_x) can be determined using only one COND-EVAL query with the set X^*. Thus our lower bound argument fails in this model. For the sake of intuition about how we overcome the above issue, we want to point out that the above argument does not fail if we have an approximate COND-EVAL query instead of COND-EVAL query, i.e., if the oracle gives the estimate of $D(S)$ up to a small additive error say $\frac{3}{2^{n^{0.1}}}$. This is because for $x, x' \in [\frac{\log n}{10}, \log n]$, we have $|D_x(X^*) - D_{x'}(X^*)| \leq \frac{2}{2^{n^{0.1}}}$. Hence, given that $x, x' \in [\frac{\log n}{10}, \log n]$, we can not distinguish between x and x' as the estimate could be the same for both x and x'.

To mitigate the above issue (for COND-EVAL), we construct a new set of hard distributions. Our objective is that for any set X^*, if the value of $D_x(X^*)$ is in $(0,1)$, then value of $D_x(X^*)$ should not give information about x. One plausible approach could be to replace each distribution D_x with a finite set of distributions such that for any set $S \subseteq [n]$, there are many distributions in the instance with the same value of $D(S)$. Unfortunately, we do not know how to get such a set of distributions preserving other useful properties needed for our proof. Our key high-level idea is to replace the distribution D_x (for each $x \in [\log n]$) with a distribution over an infinite number of distributions. This way, the value of $D_x(X^*)$ cannot be used to determine the value of chosen x (as there can be infinite values of x having the same value $D(X^*)$). However, one immediate issue that arises is if our instance has an infinite domain (here distributions), then how do we even get a distribution over an infinite space? Further, like before, we still want the probabilities to exponentially decrease so that the sampled element is always among the first few smallest elements of the conditioning set S. Fortunately, in statistics and compositional data analysis, there has been a study of

probability distributions on the set of all (infinite) distributions. Based on requirements, our choice is the well-studied Dirichlet distribution that satisfies a strong independence property which is necessary for our analysis.

A Dirichlet distribution on support $[K]$ with parameters $\alpha_1, \cdots, \alpha_K > 0$ has a probability density function given by $f(p_1, \ldots, p_K) = \frac{\prod_{i\in[K]} p_i^{\alpha_i - 1}}{B(\alpha_1, \cdots, \alpha_K)}$ where $\{p_i\}_{i\in K}$ belongs to the standard $K-1$ simplex, i.e., $\sum_{i\in[K]} p_i = 1$ and $p_i \geq 0$ for all $i \in [K]$ and $B(\alpha_1, \cdots, \alpha_K)$ is a normalizing constant. When $\alpha_1 = \cdots = \alpha_K = 1$, the Dirichlet distribution is just the uniform distribution on $K-1$ simplex. The higher the value of parameter α_i, the higher the (expected) value of p_i. Since we want the probability value to be exponentially decreasing, we set the values of $\alpha_1, \cdots, \alpha_n$ exponentially decreasing. Then for each index x chosen uniformly at random from $[\log n]$, we sample a distribution D_x with support $[2^x]$ from the Dirichlet distribution with parameters $\alpha_1, \cdots, \alpha_{2^x}$. By the standard Yao's principle, it suffices to show a high error probability of any deterministic algorithm that correctly estimates the support size (and hence determines the index x) of the distribution sampled as above. Note that the entropy of the index is still $\log\log n$, but the previous communication framework (between Alice and Bob) will not be useful here. This is because the range of the outcomes of the algorithm (the actual and the conditional probabilities) is continuous, and so cannot be encoded with any finite (or even infinite) length message. Instead, we apply Fano's inequality, a tool from information theory.

Roughly speaking, we show that the information gain (about the index) by the query's outcome at every step is $O(1)$. Since the initial entropy of the index is $\log\log n$, at least $\log\log n$ steps of the algorithm are needed. The main technical challenge is to upper bound the information gain at every step. It is particularly challenging as it requires calculating the explicit density function (for the outcome) corresponding to each index. These density functions are conditioned on the previous outcomes and thus change at every step. Further, the set queried by the algorithms can be adaptive, which makes our task even more difficult. To ease our analysis, we first assume that the queried sets by the algorithm are *laminar*, i.e., either they do not intersect or are subsets/supersets of each other. Our $\Omega(\log\log n)$ lower bound holds for COND-EVAL model for all the algorithms satisfying this laminar condition. It is not hard to observe that any algorithm that makes t general queries can be simulated by an algorithm that queries the laminar family of sets and makes at most 2^t queries. This observation gives us $\Omega(\log\log\log n)$ lower bound for the general case. We believe that $\Omega(\log\log n)$ is the correct lower bound for the general case, but (perhaps it is an artifact of our analysis that) the laminar structure is necessary for applying the independence properties of Dirichlet distribution which leads to only an $\Omega(\log\log\log n)$ lower bound. For the sake of simplicity, we first prove the lower bound for a weaker model called, SET-EVAL, and then extend to the general COND-EVAL model. We refer to the oracle that, given any $S \subseteq [n]$, just outputs the value of $D(S)$, as SET-EVAL oracle. Since using a COND-EVAL query, we can simulate a SET-EVAL query, the COND-EVAL model is at least as powerful as SET-EVAL. We now describe the proof of the lower bound for the SET-EVAL model in more detail.

For an index x chosen uniformly at random from $[\log n]$, we sample a distribution D_x with support $[2^x]$ from the Dirichlet distribution with parameters $\alpha_1, \cdots, \alpha_{2^x}$ ($\alpha_j = \frac{1}{2^j}$ for all $j \in [x]$). By the standard Yao's principle, it suffices for us to show a high error probability of any deterministic algorithm that correctly estimates the support size (and hence determines the index x) of the distribution sampled as above. Let T be any such deterministic algorithm that queries a laminar family of sets. We need to show that T must make $\Omega(\log\log n)$ queries. The main technical ingredient in the proof is to show that the information gain (about

the index x) by the outcome of any query of the algorithm is $O(1)$. This implies that the total information gain is $O(t)$, where t is the number of queries. Then Fano's inequality immediately implies that $t \geq \Omega(\log \log n)$. Let the i-th query ($i \in [t]$) be denoted by set A_i, and the outcome of the i-th query (sum of probabilities of elements in the set A_i) be denoted by Z_i. The information gain (about the index x) by the i-th query (for any $i \in [t]$) is the conditional mutual information $I(X; Z_i|Z^{i-1})$ where Z_i is the random variable denoting the outcome of i-th query, $Z^{i-1} = (Z_1, \ldots, Z_{i-1})$ is the random variable denoting the vector of previous outcomes and X is the random variable denoting the uniformly chosen index x from $[\log n]$. By definition, the conditional mutual information $I(A; B|C)$ for random variables A, B, C is equal to the expectation (over C) of the KL divergence between the joint distribution $Q_{(A,B)|C}$ and the product distribution $Q_{A|C} \times Q_{B|C}$. The technical difficulty is to upper bound the conditional mutual information $I(X; Z_i|Z^{i-1})$ by $O(1)$. This is particularly challenging since the joint and the product distributions are not explicitly given, and queries are adaptive. To overcome this difficulty, at any step $i \in [t]$ (after $(i-1)$-th query), we first partition the $[\log n]$ into four groups denoted by $L_0^i, L_1^i, L_2^i, L_3^i$ such that for the first three groups, the outcome of the algorithm (i.e., Z_i) is (deterministically) determined by the previous outcomes of the algorithm (i.e., $Z^{i-1} = (Z_1, \ldots, Z^{i-1})$) whereas for any x in the fourth group L_3^i, the outcome is not fixed (given previous outcomes) but comes from a distribution (which we show to be also Dirichlet). Let this distribution be denoted by Q_x for $x \in L_3^i$. The information gain by the i-th query is $\log 3$ (because there are three groups for which outcome is deterministically determined) plus the information gain corresponding to the last group L_3^i. This term can be upper bounded by the maximum KL divergence between distributions Q_x and $Q_{x'}$ for any $x, x' \in L_3^i$. Thus our goal is to show that KL divergence between Q_x and $Q_{x'}$ for any $x, x' \in L_3^i$ is $O(1)$. Using the independence property of Dirichlet distributions and the laminar structure of query sets, we show that the KL divergence between the distributions Q_x and $Q_{x'}$ is equal to the KL divergence between two beta distributions with different parameters (beta distributions are a special case of Dirichlet distributions). The explicit formula for KL divergence between two beta distributions is well-known (e.g., see [23]), and we use this formula to upper bound the KL divergence by $O(1)$.

▶ Remark 9. Technically, the above lower bound for SET-EVAL (and COND-EVAL) does not hold if it is promised that the probabilities in the given distribution are rational numbers. This is because, in the lower bound instances above, the probability of an element can be any arbitrary real number in $(0, 1)$. However, we can use the following standard fact to show that the lower bound holds even with the rational probabilities.

A Polya urn is an urn containing α_i balls of color i, for each $i \in [K]$. The urn evolves at each discrete time step – a ball is sampled uniformly at random. The ball's color is observed, and two balls of the observed color are returned to the urn. Let $X_{i,m}$ be the number of balls of color i (for each $i \in [K]$) added after m time steps. Clearly $D_m = (\frac{X_{1,m}}{m}, \cdots, \frac{X_{K,m}}{m})$ is a probability distribution over $[K]$. It can be shown that the distributions $D_1, \cdots, D_m$ converges to a Dirichlet distribution with parameters $\alpha_1, \ldots, \alpha_K$ when m tends to ∞ [4].

Instead of using Dirichlet distribution in our lower bound proof, we can use the distribution D_m for sufficiently large m. Since for any m, the probabilities in D_m are rational numbers, we can get the lower bound even when the probabilities are rationals.

The power of COND-EVAL

We further demonstrate the power of the COND-EVAL model by showing an algorithm with constant query complexity for a number of distribution testing problems for which there are strong lower bounds known for the COND-PR and COND model. Our first

example is the well-studied *Equivalence testing* problem. Here, given two distributions D and D', the goal is to accept if $D = D'$ and reject if their *total variation distance* $||D - D'||_{TV} = \sum_{i\in[n]} |D(i) - D'(i)| > \epsilon$ (both with high probability). It is known that $\Omega(\sqrt{\log\log n})$ queries are necessary in the COND model [1]. On the other hand, $\Omega(1/\epsilon)$ queries are required in the COND-PR model. (Consider a uniform distribution U on $[n]$. Now randomly choose $i, j \in [n]$. We modify U to construct another distribution U' by setting $U'(i) = 2/n$, $U'(j) = 0$, and no changes in the probability mass of other elements. Note that $\epsilon = ||U - U'||_{TV} = 2/n$. It is easy to see that $\Omega(n) = \Omega(\frac{1}{\epsilon})$ queries are required to distinguish U and U' in the COND-PR model.) We show that Equivalence testing can be done in just two COND-EVAL queries. The above upper bound result extends to another unrelated problem for the COND-EVAL model – the problem of testing whether the given distribution is *m-grained*, i.e., the probability of each element is an integer multiple of $1/m$. Finally, we show that the multiplicative $(1 + \epsilon)$-approximation of square of the L_2 norm $(\sum_{j\in[n]} D(j)^2)$ of a distribution D can be computed using $O(\frac{1}{\epsilon^2})$ queries of COND-EVAL. To the best of our knowledge, this problem has been studied previously only in the SAMP model [19], wherein it was shown that $\Omega(\frac{\sqrt{n}}{\epsilon^2})$ queries are required.

The power of bounded-set conditioning

We further study the support size estimation problem when we allow SAMP oracle access and conditioning on sets of size at most k. We show a lower bound of $\Omega(n/k)$ and an upper bound of $O(\frac{n \log\log n}{k})$ for constant factor approximation in this model. The upper bound holds for the COND oracle model, while our lower bound holds for the stronger COND-EVAL oracle model.

Both the upper and lower bounds are not difficult to establish. Falahatgar et al. [18] showed that $O(\log\log n)$ queries are sufficient (with no restriction on the size of the set for conditioning) to get a constant approximation for oracle access which, given any $S \subseteq [n]$, returns whether $S \cap \text{SUPP}(D) = \emptyset$ or not. Oracle access to a set of size s can be simulated by s/k oracle access when conditioning on at most k-sized sets is allowed. This gives an upper bound of $O(\frac{n \log\log n}{k})$. Interestingly, the hard instance for the bounded-set conditioning to estimate the support size is when the support size is constant. We refer the readers to the full version for all the detailed proofs.

4 Conclusion

We investigate the power of conditioning for estimating the support size up to a multiplicative $(1 + \epsilon)$-factor. To date, there is a gap between the upper bound of $O(\log\log n + 1/\epsilon^2)$ and the lower bound of $\Omega(\sqrt{\log\log n})$ in the standard COND model. In this paper, we close this gap by providing a lower bound of $\Omega(\log\log n + \frac{1}{\epsilon^2 \log\frac{1}{\epsilon}})$. We actually show the lower bound in even a more powerful model, namely COND-PR, where in addition to the conditioning, one is also allowed to get the actual probability of the sampled elements (i.e., a combination of COND and PR-SAMP). In the dependency of ϵ, there is a small gap of $\log(1/\epsilon)$ factor, and we want to leave the problem of removing this factor from the lower bound term as an open problem.

It is quite surprising that the combination of COND and PR-SAMP does not yield more power compared to only the COND model in the context of the support size estimation. We thus continue our investigation by appending the algorithms with an even more powerful oracle that could also get the conditional probabilities of the sampled elements (not just the actual probabilities). We call this model COND-EVAL. This model turns out to be

more powerful in the context of several other important distribution testing problems. For the support size estimation, we show a lower bound of $\Omega(\log\log\log n + \frac{1}{\epsilon^2 \log \frac{1}{\epsilon}})$ in this COND-EVAL model. On the technical side, this paper introduces many new ideas, such as using continuous distribution (Dirichlet distribution) for constructing hard instances and applying information theory and communication complexity tools to conditional sampling models. We hope that such techniques could be useful for showing non-trivial lower bounds for other distribution testing problems as well.

For the support size estimation problem in the COND-EVAL model, currently, we only know of an $O(\log\log n)$ upper bound, whereas we could only show a lower bound of $\Omega(\log\log\log n)$. We would like to pose the problem of closing this gap as an interesting open problem. Another interesting open problem is to determine if an upper bound of $o(\log\log n)$ is possible in the COND-PR and COND-EVAL models, assuming a certain lower bound on the probability mass of each element in the support size (e.g., the probability of any element is either 0 or at least $1/n$).

References

1. Jayadev Acharya, Clément L Canonne, and Gautam Kamath. A chasm between identity and equivalence testing with conditional queries. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2015)*. Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2015.
2. Tugkan Batu, Sanjoy Dasgupta, Ravi Kumar, and Ronitt Rubinfeld. The complexity of approximating the entropy. *SIAM Journal on Computing*, 35(1):132–150, 2005.
3. Rishiraj Bhattacharyya and Sourav Chakraborty. Property testing of joint distributions using conditional samples. *ACM Transactions on Computation Theory (TOCT)*, 10(4):1–20, 2018.
4. David Blackwell and James B MacQueen. Ferguson distributions via pólya urn schemes. *The annals of statistics*, 1(2):353–355, 1973.
5. Eric Blais, Joshua Brody, and Kevin Matulef. Property testing lower bounds via communication complexity. *computational complexity*, 21(2):311–358, 2012.
6. Eric Blais, Clément L Canonne, and Tom Gur. Distribution testing lower bounds via reductions from communication complexity. *ACM Transactions on Computation Theory (TOCT)*, 11(2):1–37, 2019.
7. Cafer Caferov, Barış Kaya, Ryan O'Donnell, and AC Say. Optimal bounds for estimating entropy with pmf queries. In *International Symposium on Mathematical Foundations of Computer Science*, pages 187–198. Springer, 2015.
8. Clément Canonne and Ronitt Rubinfeld. Testing probability distributions underlying aggregated data. In *International Colloquium on Automata, Languages, and Programming*, pages 283–295. Springer, 2014.
9. Clément L Canonne. A survey on distribution testing: Your data is big. but is it blue? *Theory of Computing*, pages 1–100, 2020.
10. 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)*, pages 321–336. SIAM, 2021.
11. Clément L Canonne, Dana Ron, and Rocco A Servedio. Testing probability distributions using conditional samples. *SIAM Journal on Computing*, 44(3):540–616, 2015.
12. Amit Chakrabarti and Oded Regev. An optimal lower bound on the communication complexity of gap-hamming-distance. *SIAM Journal on Computing*, 41(5):1299–1317, 2012.
13. Sourav Chakraborty, Eldar Fischer, Yonatan Goldhirsh, and Arie Matsliah. On the power of conditional samples in distribution testing. *SIAM Journal on Computing*, 45(4):1261–1296, 2016.

14 Sourav Chakraborty and Kuldeep S Meel. On testing of uniform samplers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33(01), pages 7777–7784, 2019.

15 Xi Chen, Rajesh Jayaram, Amit Levi, and Erik Waingarten. Learning and testing junta distributions with sub cube conditioning. In *Conference on Learning Theory*, pages 1060–1113. PMLR, 2021.

16 Remi Delannoy and Kuldeep S Meel. On almost-uniform generation of sat solutions: The power of 3-wise independent hashing. In *Proceedings of the 37th Annual ACM/IEEE Symposium on Logic in Computer Science*, 2022.

17 Moein Falahatgar, Ashkan Jafarpour, Alon Orlitsky, Venkatadheeraj Pichapati, and Ananda Theertha Suresh. Faster algorithms for testing under conditional sampling. In *Conference on Learning Theory*, pages 607–636. PMLR, 2015.

18 Moein Falahatgar, Ashkan Jafarpour, Alon Orlitsky, Venkatadheeraj Pichapati, and Ananda Theertha Suresh. Estimating the number of defectives with group testing. In *2016 IEEE International Symposium on Information Theory (ISIT)*, pages 1376–1380. IEEE, 2016.

19 Oded Goldreich and Dana Ron. On testing expansion in bounded-degree graphs. In *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation*, pages 68–75. Springer, 2011.

20 Priyanka Golia, Brendan Juba, and Kuldeep S. Meel. Efficient entropy estimation with applications to quantitative information flow. In *International Conference on Computer-Aided Verification (CAV)*, 2022.

21 Andrew D. Gordon, Thomas A. Henzinger, Aditya V. Nori, and Sriram K. Rajamani. Probabilistic programming. In *Future of Software Engineering Proceedings*, FOSE 2014, pages 167–181, New York, NY, USA, 2014. Association for Computing Machinery. doi:10.1145/2593882.2593900.

22 Sudipto Guha, Andrew McGregor, and Suresh Venkatasubramanian. Sublinear estimation of entropy and information distances. *ACM Transactions on Algorithms (TALG)*, 5(4):1–16, 2009.

23 Norman L Johnson, Samuel Kotz, and Narayanaswamy Balakrishnan. *Continuous univariate distributions, volume 2*, volume 289. John wiley & sons, 1995.

24 Gautam Kamath and Christos Tzamos. Anaconda: A non-adaptive conditional sampling algorithm for distribution testing. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 679–693. SIAM, 2019.

25 Kuldeep S Meel, Yash Pralhad Pote, and Sourav Chakraborty. On testing of samplers. *Advances in Neural Information Processing Systems*, 33:5753–5763, 2020.

26 Shyam Narayanan. On tolerant distribution testing in the conditional sampling model. In Dániel Marx, editor, *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms, SODA 2021, Virtual Conference, January 10–13, 2021*, pages 357–373. SIAM, 2021.

27 Krzysztof Onak and Xiaorui Sun. Probability–revealing samples. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 2018.

28 Ronitt Rubinfeld and Rocco A Servedio. Testing monotone high-dimensional distributions. *Random Structures & Algorithms*, 34(1):24–44, 2009.

29 C. E. Shannon. A mathematical theory of communication. *Bell system technical journal*, 27, 1948.

30 Gregory Valiant and Paul Valiant. Estimating the unseen: an n/log (n)-sample estimator for entropy and support size, shown optimal via new clts. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 685–694, 2011.

31 Yihong Wu and Pengkun Yang. Chebyshev polynomials, moment matching, and optimal estimation of the unseen. *The Annals of Statistics*, 47(2):857–883, 2019.