Electronic Colloquium on Computational Complexity, Report No. 131 (2018)

Anaconda: A Non-Adaptive Conditional Sampling Algorithm for Distribution Testing

Gautam Kamath*
EECS & CSAIL, MIT
g@csail.mit.edu

Christos Tzamos†
University of Wisconsin-Madison
tzamos@wisc.edu

July 16, 2018

Abstract

We investigate distribution testing with access to non-adaptive conditional samples. In the conditional sampling model, the algorithm is given the following access to a distribution: it submits a query set S to an oracle, which returns a sample from the distribution conditioned on being from S. In the non-adaptive setting, all query sets must be specified in advance of viewing the outcomes.

Our main result is the first polylogarithmic-query algorithm for equivalence testing, deciding whether two unknown distributions are equal to or far from each other. This is an exponential improvement over the previous best upper bound, and demonstrates that the complexity of the problem in this model is intermediate to the the complexity of the problem in the standard sampling model and the adaptive conditional sampling model. We also significantly improve the sample complexity for the easier problems of uniformity and identity testing. For the former, our algorithm requires only $\tilde{O}(\log n)$ queries, matching the information-theoretic lower bound up to a $O(\log \log n)$-factor.

Our algorithm works by reducing the problem from ℓ_1-testing to ℓ_∞-testing, which enjoys a much cheaper sample complexity. Necessitated by the limited power of the non-adaptive model, our algorithm is very simple to state. However, there are significant challenges in the analysis, due to the complex structure of how two arbitrary distributions may differ.

*Supported by NSF Award CCF-1551875, CCF-1617730, CCF-1650733, CCF-1741137, ONR N00014-12-1-0999, a Google Faculty Research Award, and a Simons Investigator award. Part of this work was done while the author was an intern at Microsoft Research, New England.

†This work was done while the author was a postdoc at Microsoft Research, New England.

ISSN 1433-8092

1 Introduction

Statistical hypothesis testing is one of the most classical problems in statistics, with a rich history over the past century. Over the last two decades, the problem has recently attracted the focus of theoretical computer scientists, primarily with a focus on rigorous, finite sample guarantees for distributions with large domain sizes. The seminal works of Goldreich and Ron [GR00] and Batu, Fortnow, Rubinfeld, Smith, and White [BFR+00] initiated this study of distribution testing, viewing distributions as a natural domain for property testing (see [Gol17] for coverage of this much broader field).

Since these works, distribution testing has enjoyed a wealth of study, resulting in a thorough understanding of the complexity of testing many distributional properties (see e.g. [BFF+01, BKR04, Pan08, ADJ+11, BFRV11, Val11, Rub12, ILR12, DDS+13, CDVV14, VV17, Wag15, AD15, ADK15, BV15, DKN15, DK16, Can16, BCG17, BC17a, DDK18, DKW18, DGPP18], and [Can15b] for a recent survey). For many problems, these works have culminated in sample-optimal algorithms.

In this paper, we will be concerned with the following three problems, defined on discrete distributions over $[n]$:

- **Uniformity Testing:** Given sample access to a distribution p, test whether $p = \mathcal{U}_n$ (the uniform distribution on $[n]$) or is far from it;
- **Identity Testing:** Given sample access to a distribution p and the description of a distribution q, test whether $p = q$ or is far from it;
- **Equivalence Testing:** Given sample access to distributions p and q, test whether they are equal to or far from each other.

Observe that each of these problems generalizes the previous, and thus are in increasing difficulty. All three of these problems have a sample complexity which is either $\Theta(n^{1/2})$ or $\Theta(n^{2/3})$. In other words, while these problems enjoy a sample complexity which is strongly sublinear in the domain size, in the absence of additional assumptions, information-theoretic lower bounds often necessitate a sample complexity which is polynomial in the size of the domain. When the domain is exceptionally large, this cost may be prohibitive for many of the inference tasks we wish to perform.

To circumvent these strong lower bounds, one may imagine oracle models where one has additional power when interacting with the distribution. Some examples include when the algorithm may query the PDF or CDF of the distribution [BDKR05, GMV06, RS09, CR14], or is given probability-revealing sample [OS18]. However, perhaps the most popular alternative model, and the one we consider in this paper, is the *conditional sampling* model. This model was recently introduced concurrently by Chakraborty, Fischer, Goldhirsh, and Matsliah [CFGM13, CFGM16] and Canonne, Ron, and Servedio [CRS14, CRS15]. The algorithm is able to *query* a distribution in the following way: it submits a query set S to an oracle, which returns a sample from the distribution conditioned on being from S. Additionally, we will distinguish between conditional sampling models where the algorithm's queries may be adaptive (COND) or non-adaptive (NACOND) [1]. In comparison, we will use SAMP to refer to the standard sampling model.

Conditional sampling often dramatically reduces the complexity of distribution testing problems. For example, given SAMP access to a distribution, the sample complexity of identity testing is $\Theta(\sqrt{n}/\varepsilon^2)$ [Pan08, VV17]. However, given COND access, the query complexity drops to $\tilde{\Theta}(1/\varepsilon^2)$ [FJO+15], completely removing the dependence on the support size. Motivated by the power of this model, there has been significant investigation into its implications on distribution testing [Can15a, FJO+15, ACK15b, FLV17, SSJ17, BC17b], as well as group testing [ACK15a], sublinear algorithms [GTZ17], and crowdsourcing [GTZ18].

At this point, we have a developed understanding of the power of the COND oracle with respect to the aforementioned distribution testing problems. Perhaps surprisingly, the relative complexities of certain problems have qualitatively different relationships between SAMP and COND. To be precise, the sample complexities of identity testing and equivalence testing in SAMP are $\Theta(n^{1/2})$[Pan08, VV17] and $\Theta(n^{2/3})$[CDVV14] respectively: there is a polynomial relationship between the two. However, their query complexities in COND are $\Theta(1)$[CRS15, FJO+15] and $\log^{\Theta(1)} \log n$[FJO+15, ACK15b] respectively: there is a "chasm" between the two complexities, as we go from no dependence on the domain size to a doubly logarithmic one.

[1] A formal definition of these concepts is given in Definition 1

However, the picture is much less clear when it comes to the non-adaptive NACOND model. We know that the complexity of identity testing is $\mathrm{poly}\log n$[CFGM13, ACK15b], though the upper and lower bounds are quite far from each other. On the other hand, the complexity of equivalence testing is far less clear: the best lower bound is $\Omega(\log n)$[ACK15b], and the best upper bound is $O(n^{2/3})$[CDVV14]. Given the interesting qualitative behavior observed for the COND model, this begs the following question:

Question 1. *What is the relationship of the query complexities of identity and equivalence testing in the* NACOND *model?*

In particular, are they polynomially related, as in the SAMP model? Or is there a larger gap between the two, as in the COND model? Stated another way, do we require both conditional samples and adaptivity *simultaneously* in order to reap the benefits for testing equivalence?

1.1 Results and Discussion

Our main result is a qualitative resolution to this problem: we give a $\mathrm{poly}\log n$-query algorithm for equivalence testing.

Theorem 1 (Non-Adaptive Equivalence Testing)**.** *There exists an algorithm which, given* NACOND *access to unknown distributions p, q on $[n]$, makes $\tilde{O}\left(\frac{\log^{12} n}{\varepsilon^2}\right)$ queries to the oracle on each distribution and distinguishes between the cases $p = q$ and $d_{\mathrm{TV}}(p, q) \geq \varepsilon$ with probability at least $2/3$.*

For the special case of uniformity testing, we have a sharper analysis, allowing us to obtain a $\tilde{O}(\log n)$ query algorithm, which nearly matches the $\Omega(\log n)$ lower bound of [ACK15b]:

Theorem 2 (Non-Adaptive Uniformity Testing)**.** *There exists an algorithm which, given* NACOND *access to an unknown distribution p on $[n]$, makes $\tilde{O}\left(\frac{\log n}{\varepsilon^2}\right)$ queries to the oracle on p and distinguishes between the cases $p = \mathcal{U}_n$ and $d_{\mathrm{TV}}(p, \mathcal{U}_n) \geq \varepsilon$ with probability at least $2/3$, where $\mathcal{U}_n$ is the uniform distribution on $[n]$.*

As a corollary of Theorem 2, we can obtain an improved upper bound for identity testing with an adaptation of the reduction from identity testing to uniformity testing of [CFGM16] (inspired by the bucketing techniques of [BFR$^+$00, BFF$^+$01]).

Theorem 3 (Non-Adaptive Identity Testing)**.** *There exists an algorithm which, given* NACOND *access to an unknown distribution p on $[n]$ and a description of a distribution q over $[n]$, makes $\tilde{O}\left(\frac{\log^2 n}{\varepsilon^2}\right)$ queries to the oracle on p and distinguishes between the cases $p = q$ and $d_{\mathrm{TV}}(p, q) \geq \varepsilon$ with probability at least $2/3$.*

Our results and a comparison with the complexity of testing in various oracle models are presented in Table 1.

Model	Uniformity	Identity	Equivalence
SAMP	$\Theta\left(\frac{\sqrt{n}}{\varepsilon^2}\right)$ [Pan08, VV17]	$\Theta\left(\frac{\sqrt{n}}{\varepsilon^2}\right)$ [Pan08, VV17]	$\Theta\left(\max\left(\frac{n^{2/3}}{\varepsilon^{4/3}}, \frac{n^{1/2}}{\varepsilon^2}\right)\right)$ [CDVV14]
NACOND	$\tilde{O}\left(\frac{\log n}{\varepsilon^2}\right)$ [this work] $\Omega(\log n)$ [ACK15b]	$\tilde{O}\left(\frac{\log^2 n}{\varepsilon^2}\right)$ [this work] $\Omega(\log n)$ [ACK15b]	$\tilde{O}\left(\frac{\log^{12} n}{\varepsilon^2}\right)$ [this work] $\Omega(\log n)$ [ACK15b]
COND	$\tilde{O}\left(\frac{1}{\varepsilon^2}\right)$ [CRS15] $\Omega\left(\frac{1}{\varepsilon^2}\right)$[CRS15]	$\tilde{O}\left(\frac{1}{\varepsilon^2}\right)$ [FJO$^+$15] $\Omega\left(\frac{1}{\varepsilon^2}\right)$[CRS15]	$\tilde{O}\left(\frac{\log\log n}{\varepsilon^5}\right)$ [FJO$^+$15] $\Omega\left(\sqrt{\log\log n}\right)$ [ACK15b]

Table 1: Summary of results, and a comparison of uniformity, identity, and equivalence testing in different sampling oracle models. Problems get harder as one moves up and to the right in this table.

We present a unified algorithm, ANACONDA, for both equivalence and uniformity testing, the only difference is in the choice of parameters. ANACONDA is quite simple to describe, requiring only four sentences in Section 1.2[2] We consider this algorithmic simplicity to be an advantage of ANACONDA, though we regret that its analysis is less simple – we elaborate on the technical challenges in Section 1.2.

Our bound for equivalence testing in the NACOND model is the first tailored to this setting. Specifically, the best upper bound was $O(n^{2/3})$ (for the harder problem of equivalence testing in the SAMP model [CDVV14]), and the best lower bound was $\Omega(\log n)$ (for the easier problem of uniformity testing in the NACOND model [ACK15b]). These results left open the question of the true complexity of equivalence testing: is it polynomial in $\log n$, or polynomial in n? Our algorithm gives an exponential improvement in the query complexity by showing that the former is true: equivalence testing enjoys significant savings in the query complexity when we switch from the SAMP to the NACOND oracle model.

More generally, as mentioned before, our results expose a qualitatively interesting relationship between identity and equivalence testing in the NACOND model. In the standard sampling model (SAMP), the complexity of these problems is known to be polynomially related ($\Theta(n^{1/2})$ versus $\Theta(n^{2/3})$). However, in the conditional sampling model with adaptivity (COND), there is a "chasm" between these two complexities: one has a constant query complexity, while the other has a complexity which is doubly logarithmic in n ($\Theta(1)$ versus $\operatorname{poly}\log\log n$). Our results demonstrate that when we remove adaptivity from the conditional sampling model (NACOND), the relationship is qualitatively quite different. In this setting, the "chasm" closes, and the complexity of both problems is once again polynomially related: both are $\operatorname{poly}\log n$. Interestingly, this complexity is intermediate to the complexity of the same problems in the SAMP and COND models, by an exponential factor on either side. These relationships are all summarized in Table 1. We note that our results further address an open problem of Fischer [Fis14], which inquires about the complexity of equivalence testing with conditional samples.

In terms of specific sample complexities, we observe that our upper bound for uniformity testing is nearly tight: our $\tilde{O}\left(\frac{\log n}{\varepsilon^2}\right)$ upper bound is complemented by the $\Omega(\log n)$ lower bound of [ACK15b]. It improves upon the algorithm of [CFGM13], which has query complexity $O\left(\frac{\log^{12.5} n}{\varepsilon^{17}}\right)$. Our algorithm for identity testing, with complexity $\tilde{O}\left(\frac{\log^2 n}{\varepsilon^2}\right)$, also significantly improves over theirs, which has a similar complexity as their algorithm for uniformity testing. We again mention that our bound for equivalence testing is exponentially better than the previous best algorithm for this problem (which is the $O(n^{2/3})$-query algorithm in the SAMP model of [CDVV14]).

1.2 Techniques and Proof Ideas

At the core of our approach is reducing the problem from ℓ_1-testing to ℓ_∞-testing, the latter of which is much cheaper in terms of sample complexity. In particular, throughout this exposition, keep in mind that one can estimate a distribution up to ε in ℓ_∞-distance at a cost of $O(1/\varepsilon^2)$ samples (cf. Lemma 1). In order to give intuition on how such an approach could possibly work, we focus on two very simple instances of uniformity testing. In the first instance, p is a distribution with a single "spike": for some $i^* \in [n]$, $p(i^*) = \frac{1}{n} + \varepsilon$, and for $i \neq i^*$, $p(i) = \frac{1-\varepsilon}{n}$. This can be detected by simply choosing $S = [n]$ and querying it with NACOND $O(1/\varepsilon^2)$ times: the empirical distribution $\hat{p}(i^*)$ will have a similar spike, betraying that the distribution is non-uniform. In the second instance, p is the "Paninski construction" (used as the lower bound in [Pan08]): a random half of the domain elements have probability $\frac{1+\varepsilon}{n}$, while the other half have probability $\frac{1-\varepsilon}{n}$. This can be detected by choosing S to be two random symbols, and again querying this subset $O(1/\varepsilon^2)$ times. With constant probability, the two symbols will be from different sets. While the ℓ_∞ distance from uniformity on each symbol is only $\frac{\varepsilon}{n}$, in this *conditional* distribution, it is increased to ε, allowing easy detection.

These two examples illustrate the heart of our approach: our algorithm, ANACONDA, attempts to find a query set in which the discrepancy of a single item is large in comparison to the total probability mass of the set. One of our key lemmas (Lemma 3) shows that this is possible with probability $\geq \Omega\left(\frac{1}{\log n}\right)$. While the two instances above are straightforward, a more careful analysis is required to avoid paying excess factors of $\log n$, particularly for uniformity and identity testing. That said, all the complexity is pushed to the analysis, and the algorithm itself is very simple to describe:

[2]Perhaps if we tried harder, we could describe it in two sentences, plus the word "repeat."

First, the algorithm chooses a random power of two between 2 and n – roughly, this serves as a "guess" for (the inverse of) the size of the set which represents the discrepany between the distributions. Next, the algorithm chooses a random set $S \subseteq [n]$ of this size. Finally, it performs NACOND queries to S (on both distributions, for equivalence testing), in order to form an empirical distribution (which is accurate in ℓ_∞-distance) and check whether there is a discrepant symbol or not. This process is repeated several times, and if we fail to ever detect a discrepant symbol, we can conclude that the distributions are equal.

Since uniformity testing is relatively well-behaved, the key lemma mentioned above (Lemma 3) does most of the work. This is because in this setting, once we have a handle on the distribution of the discrepancy, it is easy to reason about how much of the mass from the uniform distribution is contained in a query set. We require a few additional concentration arguments on the total discrepancy and probability mass contained in the query set, as well as a separate analysis for the case where $|S|$ needs to be small and this concentration does not hold.

We then leverage our algorithm for uniformity testing to provide an algorithm for identity testing. This uses the reduction of [CFGM13][3], which partitions the domain so that the conditional distribution on each part is close to uniform, and tests for identity on each part. This requires a non-adaptive identity tester for distributions which are close to uniform (in ℓ_∞-distance) – we show our analysis for uniformity testing can be adapted to handle this case. Our application crucially modifies their reduction in order to minimize the sample complexity, as ANACONDA can test against distributions which are further from uniform than theirs ($O(1/n)$, rather than $O(\varepsilon/n)$).

Finally, we turn to the most technically difficult problem of equivalence testing. This case turns out to be more challenging, as we must simultaneously reason about $p(i), p(S \setminus i), q(i)$, and $q(S \setminus i)$ – as mentioned prior, it is much easier to control the latter two quantities for uniformity testing. To establish our result, we must argue that ANACONDA identifies a set S where both differences $p(i) - q(i)$ and $p(S \setminus i) - q(S \setminus i)$ have opposite signs and are simultaneously relatively large compared to the magnitudes of $p(i), p(S \setminus i), q(i)$, and $q(S \setminus i)$ (Proposition 2). We consider the distribution of the discrepancy $p - q$, with a case analysis depending on the relationship between the "typical" magnitudes of the positive and negative discrepancies. If these magnitudes are close, then we can select a "smaller" set S (where "smaller" is defined based on these magnitudes) which has a reasonable probability of including a positively and negatively discrepant element of these magnitudes (Lemma 6). On the other hand, if these magnitudes are far, then with an appropriate choice of the size of the set S, there is a significant chance that our set will contain an element i with significant positive discrepancy $p(i) - q(i)$, while the total discrepancy in the set $p(S \setminus i) - q(S \setminus i)$ is very negative (Case 2 in Lemma 7). Despite all these technicalities, we emphasize that the algorithm itself is still quite simple; in particular, it is identical to the algorithm for uniformity testing (modulo some parameter modifications).

1.3 Organization

The organization of the paper is as follows. In Section 2, we cover various preliminary definitions. In Section 3, we unveil ANACONDA. In Section 4, we analyze our algorithm for the special case of uniformity testing. This case is conceptually much simpler than equivalence testing, but exposes some of the key intuitions. In Section 5, we describe the full analysis for equivalence testing. In Section 6, we adapt the reduction of [CFGM16] to obtain a more efficient algorithm for identity testing. We conclude in Section 7 with some open problems for further investigation.

2 Preliminaries

In this paper, we will focus on discrete distributions over the support $[n]$. We denote the distributions of interest using p and q, where $p(i)$ is the probability placed by distribution p on symbol i. For a set S, let $p(S) = \sum_{i \in S} p(i)$. Furthermore, let p_S be the conditional distribution of p restricted to S, i.e., $p_S(i) = p(i)/p(S)$.

[3] We note that the reduction of [Gol16], from identity testing to uniformity testing, is not known to apply in either the NACOND or COND models.

We use the following definition of the conditional sampling model. Note that this uses the convention of [CFGM13] of sampling uniformly from query sets with 0 measure, rather than the convention of [CRS14] which immediately fails if given such a set, as the latter convention trivializes NACOND, reducing it to SAMP.

Definition 1. *A* conditional sampling *oracle for a distribution p is defined as follows: the oracle takes as input a* query set $S \subseteq [n]$*, and returns a symbol $i \in S$, where the probability that i is returned is equal to $p_S(i) = p(i)/p(S)$. If $p(S) = 0$, then a symbol $i \in S$ is returned uniformly at random.*

Given an adaptive conditional sampling *oracle (a* COND *oracle), the algorithm may query* adaptively*: before submitting each query set i, the algorithm is allowed to view the results of queries 1 through $i-1$. In contrast, given a* non-adaptive conditional sampling *oracle (a* NACOND *oracle), the algorithm must be non-adaptive: it must submit all query sets in advance of viewing any of their results.*

We will frequently use $z = (p - q)/\varepsilon$ to denote the "noise vector" between p and q, and $\bar{p} = (p+q)/2$. While the two cases in distribution testing that one considers are usually $p = q$ and $d_{\mathrm{TV}}(p, q) \geq \varepsilon$, for convenience of notation, we will generally assume the latter case to be $d_{\mathrm{TV}}(p, q) = \varepsilon$ – it is not hard to see that our analysis carries through whenever the algorithm is given a parameter ε which is less than the true total variation distance between p and q. With this in mind, when $p = q$, we have that $z = \vec{0}$, and when $d_{\mathrm{TV}}(p, q) = \varepsilon$, we have that $\|z\|_1 = 2$ and $\sum_{i \in [n]} z(i) = 0$. Let z^+ denote the "rectified" version of z, where $z^+(i) = \max(0, z(i))$ – here, in the latter case, $\|z^+\|_1 = \sum_{i \in [n]} z^+(i) = 1$. $z^-(i) = \max(0, -z(i))$ is defined similarly.

We will use log to refer to the logarithm with base 2 throughout this paper.

For our analysis, we will group indices into bins:

Definition 2. *The j-th bin for a vector x, denoted by $Bin_j(x)$, contains all indices whose values are in the range $[2^{-j}, 2^{-j+1})$, i.e. $Bin_j(x) \triangleq \{i \ : \ \frac{1}{2^j} \leq x(i) < \frac{1}{2^{j-1}}\}$.*

We will use the following distances on probability distributions:

Definition 3. *The* total variation distance *between distributions p and q is defined as*

$$d_{\mathrm{TV}}(p, q) = \frac{1}{2} \sum_{i \in [n]} |p(i) - q(i)| .$$

Definition 4. *The* Kolmogorov distance *between distributions p and q is defined as*

$$d_{\mathrm{K}}(p, q) = \max_{j \in [n]} \left| \sum_{i=1}^{j} p(i) - \sum_{i=1}^{j} q(i) \right| .$$

Note that, up to a factor of 2, Kolmogorov distance is equivalent to the ℓ_∞ distance between the vectors p and q. The Dvoretzky-Kiefer-Wolfowitz (DKW) inequality gives a generic algorithm for learning any distribution with respect to the Kolmogorov distance [DKW56].

Lemma 1 ([DKW56],[Mas90])**.** *Let $\hat{p}_m$ be the empirical distribution generated by m i.i.d. samples from a distribution p. We have that*

$$\Pr[d_{\mathrm{K}}(p, \hat{p}_m) \geq \varepsilon] \leq 2e^{-2m\varepsilon^2}.$$

In particular, if $m = \Omega(\log(1/\delta)/\varepsilon^2)$, then $\Pr[d_{\mathrm{K}}(p, \hat{p}_m) \geq \varepsilon] \leq \delta$*.*

3 Algorithm

Our algorithm, Anaconda, is presented in Algorithm 1. While it is phrased in terms of equivalence testing, it still works when a distribution q is explicitly given (i.e., identity testing), as one can simply simulate NACOND queries to q. It takes three parameters, T, m, and ε', which we will instantiate differently (as required by our analysis) for uniformity and equivalence testing.

The algorithm's behavior can roughly be summarized as follows. The algorithm first chooses a random size for a query set. It then chooses a random subset of the domain of this size. Next, it draws several conditional samples from this set, from both p and q. Finally, if it detects that a single element from the query set has a significantly discrepant probability mass under p and q, it outputs that the two distributions are far. It repeats this process several times, eventually outputting that the distributions are equal if it never discovers a discrepant element.

Algorithm 1 ANACONDA: An algorithm for testing equivalence given NACOND oracle access to p, q

1: **function** ANACONDA(ε, NACOND_p oracle, NACOND_q oracle, parameters T, m, ε')
2:     **for** $t = 1$ to T **do**
3:         Choose an integer $j \in \{1, \ldots, 2 \log n\}$ uniformly at random, and define $r \triangleq 2^j$.
4:         Choose a random set $S \subseteq [n]$, independently selecting each i to be in S with probability $1/r$.
5:         Perform m queries to NACOND_p and NACOND_q on the set S.
6:         Using these queries, form the empirical distribution $\hat{p}_S$ and $\hat{q}_S$.
7:         **if** $\exists i \in S$ such that $|\hat{p}_S(i) - \hat{q}_S(i)| \geq \varepsilon'$ **then**
8:             **return** $d_{\mathrm{TV}}(p, q) \geq \varepsilon$
9:         **end if**
10:     **end for**
11:     **return** $p = q$
12: **end function**

4 Analysis for Uniformity Testing

In this section, we will prove Theorem 2 by instantiating ANACONDA with parameters $T = \Theta(\log n)$, $m = \Theta(\log \log n/\varepsilon^2)$, and $\varepsilon' = \Theta(\varepsilon)$.

Our strategy will be as follows. We will argue that, with probability $\Omega(1/\log n)$, ANACONDA will select a set S with a single element that has significantly different mass under the uniform distribution and the distribution p_S. In this way, we will reduce the problem from ℓ_1-testing to ℓ_∞-testing, the latter of which is solvable with very few samples, by Lemma 1.

More precisely, we compare the probability assigned to a particular symbol i when performing a conditional sample on S, in the two cases where $p = \mathcal{U}_n$, and when $d_{\mathrm{TV}}(p, \mathcal{U}_n) = \varepsilon$. In the former case, the probability is $\frac{\mathcal{U}_n(i)}{\mathcal{U}_n(S)}$, while in the latter, it is $\frac{\mathcal{U}_n(i)+\varepsilon z(i)}{\mathcal{U}_n(S)+\varepsilon z(S)}$. Therefore, the difference in probability assigned is

$$\left| \frac{\mathcal{U}_n(i) + \varepsilon z(i)}{\mathcal{U}_n(S) + \varepsilon z(S)} - \frac{\mathcal{U}_n(i)}{\mathcal{U}_n(S)} \right|. \tag{1}$$

In the following two subsections, we will show that the following lemma:

Lemma 2. *If $d_{\mathrm{TV}}(p, \mathcal{U}_n) = \varepsilon$, then for each t,* ANACONDA *will select a set S which causes (1) to be $\geq \Omega(\varepsilon)$ with probability $\geq \Omega(1/\log n)$.*

Assuming this to be true for the moment, we will show how to complete the proof. Repeating this process $T = \Theta(\log n)$ times will guarantee that at least one iteration will choose an S containing a sufficiently discrepant element with probability $\geq 9/10$. We focus on the iteration where such an S is selected.

Now if we draw $\Theta(\log \log n/\varepsilon^2)$ samples from p_S, Lemma 1 implies the empirical distribution $\hat{p}_S$ will approximate p_S in Kolmogorov distance up to an additive ε', with probability at least $1 - O\left(\frac{1}{\log n}\right)$, and thus Line 7 will correctly identify that $d_{\mathrm{TV}}(p, \mathcal{U}_n) = \varepsilon$. Therefore, with probability at least $4/5$, the algorithm will correctly detect in this case that $d_{\mathrm{TV}}(p, \mathcal{U}_n) = \varepsilon$.

We now examine what happens when $p = \mathcal{U}_n$. For each iteration t, the uniform distribution on S and p_S will be equal. We again invoke Lemma 1 with $\Theta(\log \log n/\varepsilon^2)$ samples, and use a union bound over all $T = \Theta(\log n)$ iterations. This implies that, with probability at least $9/10$, Line 7 will never identify an element which has $\geq \varepsilon'$ discrepancy, and thus the algorithm will output that $p = \mathcal{U}_n$ in Line 11.

It remains to prove Lemma 2. We break the analysis into two cases, which we address in the following two subsections. In Section 4.1, we handle the case where, for all $x \in \{z^-, z^+\}$, $\sum_{j=\log n/32+1}^{\log 5n} \sum_{i \in \mathrm{Bin}_j(x)} x(i) \geq 1/5$. This corresponds to the case where there are many symbols with small discrepancy from the uniform distribution, in both the positive and negative direction. In Section 4.2, we handle the complement of this case, where there exists an $x \in \{z^-, z^+\}$ for which $\sum_{j=1}^{\log n/32} \sum_{i \in \mathrm{Bin}_j(x)} x(i) \geq 3/5$. Roughly, this happens when there are not too many symbols which capture the discrepancy between the distributions.

4.1 Case I: Many Small Discrepancies

In this section, we prove Lemma 2 in the case where for all $x \in \{z^-, z^+\}$, $\sum_{j=\log n/32+1}^{\log 5n} \sum_{i \in \mathrm{Bin}_j(x)} x(i) \geq 1/5$. In short, the analysis can be summarized as follows: if the algorithm chooses a set S of size 2, it is likely to contain two elements with non-trivial discrepancy, and in both the positive and negative direction – this will suffice to make (1) be $\geq \Omega(\varepsilon)$.

We have the following proposition relating the size of a bin to the mass it contains, which is immediate from Definition 2.

Proposition 1. $2^{j-1} \sum_{i \in Bin_j} x(i) \leq |Bin_j(x)| \leq 2^j \sum_{i \in Bin_j} x(i)$.

This gives us the following lower bound on the number of symbols which are in bins $\log n/32 + 1$ through $\log 5n$:

$$\sum_{j=\log n/32+1}^{\log 5n} |\mathrm{Bin}_j(x)| \geq \sum_{j=\log n/32+1}^{\log 5n} 2^{j-1} \sum_{i \in \mathrm{Bin}_j(x)} x(i) \geq \frac{n}{32} \sum_{j=\log n/32+1}^{\log 5n} \sum_{i \in \mathrm{Bin}_j(x)} x(i) \geq \frac{n}{160} \tag{2}$$

In other words, for either $x \in \{z^-, z^+\}$, there are $\Omega(n)$ symbols with $x(i) \geq 1/5n$.

We complete the proof of Lemma 2 as follows. With probability $\frac{1}{2\log n}$, Anaconda will select $r = \log n$ in Line 3. Conditioning on this, with constant probability, the set S selected in Line 4 will be of size exactly 2. Further conditioning on this, due to (2), with constant probability S will consist of two symbols $i_1 \in \mathrm{Bin}_{j'}(z^+)$ and $i_2 \in \mathrm{Bin}_{j''}(z^-)$ for $\log n/32 + 1 \leq j', j'' \leq \log 5n$.

Without loss of generality, suppose that $z(i_1) \geq 0$ and $z(i_2) \leq 0$. Then (1) is the following:

$$\left|\frac{\mathcal{U}_n(i) + \varepsilon z(i)}{\mathcal{U}_n(S) + \varepsilon z(S)} - \frac{\mathcal{U}_n(i)}{\mathcal{U}_n(S)}\right| = \frac{\varepsilon n(z(i_i) - z(i_2))}{2(2 + \varepsilon n(z(i_1) + z(i_2)))} \geq \frac{\varepsilon n \cdot \frac{2}{5n}}{2(2 + \varepsilon n \cdot \frac{32}{n})} \geq \frac{\varepsilon}{68}. \tag{3}$$

This expression is $\geq \Omega(\varepsilon)$, and this event happens with probability $\geq \Omega(1/\log n)$, thus proving Lemma 2 in this case.

4.2 Case II: Not So Many Small Discrepancies

In this section, we prove Lemma 2 in the case where there exists an $x \in \{z^-, z^+\}$ for which $\sum_{j=1}^{\log n/32} \sum_{i \in \mathrm{Bin}_j(x)} x(i) \geq 3/5$. Without loss of generality, assume that this holds for z^+. Furthermore, we focus our analysis on the case where Anaconda picks an $r \leq \log n/32$. For the remainder of this proof, condition on this event, which happens with probability at least 1/4.

We will need the following key lemma:

Lemma 3. *Suppose $d_{\mathrm{TV}}(p, \mathcal{U}_n) = \varepsilon$. For each iteration t, with probability $\geq \frac{3}{20\log n/32}$, the algorithm will choose an r and a set S such that there exists $i \in S$ with $z^+(i) \geq 1/r$.*

Proof. For some fixed j, the probability of choosing j is $\frac{1}{\log n/32}$, and, conditioning on this j, the probability of picking any element from $\mathrm{Bin}_j(z^+)$ to be in S is $1 - \left(1 - \frac{1}{2^j}\right)^{|\mathrm{Bin}_j(z^+)|}$. By the law of total probability, we sum this over all bins to get the probability that the event of interest happens:

$$\frac{1}{\log n/32}\sum_{j\in[\log n/32]}1-\left(1-\frac{1}{2^j}\right)^{|\mathrm{Bin}_j(z^+)|} \geq \frac{1}{\log n/32}\sum_{j\in[\log n/32]}1-\exp\left(-\frac{|\mathrm{Bin}_j(z^+)|}{2^j}\right) \tag{4}$$

$$\geq \frac{1}{\log n/32}\sum_{j\in[\log n/32]}1-\exp\left(-\frac{1}{2}\sum_{i\in\mathrm{Bin}_j(z^+)}z^+(i)\right) \tag{5}$$

$$\geq \frac{1}{\log n/32}\sum_{j\in[\log n/32]}\frac{1}{4}\sum_{i\in\mathrm{Bin}_j(z^+)}z^+(i) \tag{6}$$

$$\geq \frac{3}{20\log n/32}. \tag{7}$$

(4) follows from the inequality $1-x \leq \exp(-x)$, (5) is due to Proposition 1, (6) is by the inequality $1-\exp(-x) \geq x/2$ (which holds for all $x \in [0,1]$), and (7) is by assumption ☐

We will require the following lemmata to complete the proof:

Lemma 4. *For any i and j,*

$$\Pr\left[\frac{1}{2\cdot 2^j} \leq \mathcal{U}_n\left(S\setminus i\right) \leq \frac{3}{2\cdot 2^j}\right] \geq 1-2/e^2.$$

Proof. Observe that the size of $S\setminus i$ is a sum of $n-1$ i.i.d. Bernoulli random variables with parameter $1/2^j$, and thus has expectation $\mu = \frac{n-1}{2^j}$. Then, by Chernoff bound, we have

$$\Pr\left[\frac{9}{16}\frac{(n-1)}{2^j} \leq |S\setminus i| \leq \frac{23}{16}\frac{(n-1)}{2^j}\right] \geq 1-2\exp\left(-\frac{49\mu}{768}\right) \geq 1-2/e^2.$$

The last inequality follows since $j \leq \log n/32$ for n larger than some absolute constant. Similarly, the lemma follows for n larger than some absolute constant by rescaling the size of the set by a factor of n. ☐

Lemma 5. *If $d_{\mathrm{TV}}(p,\mathcal{U}_n)=\varepsilon$, then for any i and j,*

$$\Pr\left[z\left(S\setminus i\right) \geq \frac{4}{2^j}\right] \leq 1/4.$$

Proof. Note that $z^+(S\setminus i)$ is a non-negative random variable. Its expectation $\mathbf{E}[z^+(S\setminus i)] \leq \mathbf{E}[z^+(S)] \leq 1/2^j$. The lemma follows by Markov's inequality, and by observing that the addition of any negative elements of z will only decrease $z(S\setminus i)$. ☐

Note that, by Lemmas 3, 4, 5, if $d_{\mathrm{TV}}(p,\mathcal{U}_n)=\varepsilon$, with probability at least $\frac{1}{4}\cdot\frac{1}{\log n/32}\cdot\left(1-\frac{1}{4}-2/e^2\right) \geq \Omega(1/\log n)$, the following events happen simultaneously:

- $r \leq n/32$;
- $z(i) \geq 1/r$;
- $\mathcal{U}_n(i) = 1/n$;
- $z(S\setminus i) \leq 4/r$;
- $\frac{1}{2r} \leq \mathcal{U}_n(S\setminus i) \leq \frac{3}{2r}$;

We now show that a set S with all these properties will result in (1) being $\geq \Omega(\varepsilon)$:

$$\begin{aligned}\left|\frac{\mathcal{U}_n(i)+\varepsilon z(i)}{\mathcal{U}_n(S)+\varepsilon z(S)}-\frac{\mathcal{U}_n(i)}{\mathcal{U}_n(S)}\right| &= \varepsilon\left|\frac{z(i)\mathcal{U}_n(S\setminus i)-z(S\setminus i)\mathcal{U}_n(i)}{\mathcal{U}_n(S)(\mathcal{U}_n(S)+\varepsilon z(S))}\right| \\ &\geq \varepsilon\cdot\frac{1}{\mathcal{U}_n(S)(\mathcal{U}_n(S)+\varepsilon z(S))}\left(\frac{z(i)}{2r}-\frac{4}{rn}\right) \\ &\geq \varepsilon\cdot\frac{r}{2}\frac{1}{\frac{2}{r}+\varepsilon\left(\frac{4}{r}+z(i)\right)}\left(\frac{z(i)}{2r}-\frac{4}{rn}\right) \\ &\geq \varepsilon\cdot\frac{1}{\frac{2}{r}+\varepsilon\left(\frac{4}{r}+z(i)\right)}\left(\frac{z(i)}{4}-\frac{2}{n}\right)\end{aligned}$$

The analysis concludes by considering two cases. If $\varepsilon z(i) \geq \frac{2}{r}+\varepsilon\cdot\frac{4}{r}$, then we have the lower bound $\varepsilon\cdot\frac{1}{2\varepsilon z(i)}\left(\frac{z(i)}{4}-\frac{2}{n}\right) = \Omega(1) \geq \Omega(\varepsilon)$, as desired. Otherwise, we have the lower bound $\varepsilon\cdot\frac{r}{12}\left(\frac{z(i)}{4}-\frac{2}{n}\right) \geq \varepsilon\cdot\frac{r}{12}\left(\frac{1}{4r}-\frac{2}{n}\right) \geq \frac{\varepsilon}{96}$, which completes the proof.

5 Analysis for Equivalence Testing

In this section, we will prove Theorem 1 by instantiating Anaconda with parameters $T = \Theta(\log^6 n)$, $m = \tilde{\Theta}(\log^6 n/\varepsilon^2)$, and $\varepsilon' = \frac{\varepsilon}{\tilde{\Theta}(\log^3 n)}$.

We will require the following proposition, says if $d_{\mathrm{TV}}(p,q) = \varepsilon$ and Anaconda selects an appropriate set S, then it will detect the discrepancy.

Proposition 2. *Suppose that $d_{\mathrm{TV}}(p,q) = \varepsilon$ and that within the first T iterations a set S is identified such that for some $i \in S$ and some $c > 0$,*

$$\min\{z(i), z(i)-z(S)\} \geq \frac{p(S)+q(S)}{\tilde{O}(\log^c n)}.$$

Then, for $\varepsilon' = \frac{\varepsilon}{\tilde{O}(\log^c n)}$ and $m = \tilde{\Omega}\left(\frac{\log^{2c} n}{\varepsilon^2}\right)$, the algorithm outputs that $d_{\mathrm{TV}}(p,q) \geq \varepsilon$ with probability at least $1 - \frac{1}{\operatorname{poly}\log n}$.

Proof. We first argue that $|p_S(i) - q_S(i)| \geq \varepsilon\frac{\min\{z(i),z(i)-z(S)\}}{p(S)+q(S)}$.

We set $\bar{p} = \frac{p+q}{2}$. We have that $p = \bar{p} + z\frac{\varepsilon}{2}$, $q = \bar{p} - z\frac{\varepsilon}{2}$ and

$$\left|\frac{p(i)}{p(S)}-\frac{q(i)}{q(S)}\right| = \left|\frac{\bar{p}(i)+z(i)\frac{\varepsilon}{2}}{\bar{p}(S)+z(S)\frac{\varepsilon}{2}}-\frac{\bar{p}(i)-z(i)\frac{\varepsilon}{2}}{\bar{p}(S)-z(S)\frac{\varepsilon}{2}}\right| = \frac{\varepsilon}{2}\left|\frac{z(i)\bar{p}(S)-\bar{p}(i)z(S)}{\bar{p}^2(S)-(z(S)\frac{\varepsilon}{2})^2}\right| \geq \frac{\varepsilon}{2}\left|\frac{z(i)\bar{p}(S)-\bar{p}(i)z(S)}{\bar{p}^2(S)}\right|.$$

As $z(i)\bar{p}(S) - \bar{p}(i)z(S) \geq \bar{p}(S)\min\{z(i), z(i)-z(S)\}$, it follows that

$$|p_S(i) - q_S(i)| \geq \frac{\varepsilon}{2}\frac{\min\{z(i), z(i)-z(S)\}}{(p(S)+q(S))/2}$$

To complete the proof, we note that the condition implies that $|p_S(i) - q_S(i)| \geq \frac{\varepsilon}{\tilde{O}(\log^c n)}$ and thus by Lemma 1, $m = \tilde{\Omega}\left(\frac{\log^{2c} n}{\varepsilon^2}\right)$ suffices to detect (with failure probability $1/\operatorname{poly}\log n$) that $\|p_S - q_S\|_\infty > \varepsilon' = \frac{\varepsilon}{\tilde{O}(\log^c n)}$. ◻

To complete the proof, we will show that after $T = \operatorname{poly}\log n$ iterations, Algorithm 1 will choose a set S that satisfies the conditions of Proposition 2.

We define $\hat{z}$ to be the vector with $\hat{z}(i) = z(i)$ if $|z(i)| > \frac{p(i)+q(i)}{400\log n}$ and $\hat{z}(i) = 0$ otherwise. Roughly, this "zeroes out" the noise for any i where the noise vector z is too large in comparison to the signal vector $p+q$. Let b^+ be the measure on $\{1,\ldots,2\log n\}$ with mass $\hat{z}^+(\mathrm{Bin}_j(z^+))$ and equivalently define b^-. Notice that $|b^+|, |b^-| \in [1-\frac{1}{200\log n}, 1]$. This is because $\sum_{i:\hat{z}^+(i)=0} z^+(i) \leq \sum_i \frac{p(i)+q(i)}{400\log n} \leq \frac{1}{200\log n}$.

The next lemma shows that, if there are two bins (with respect to the positive and negative z vectors) which are both "heavy" and are close in index, then we will obtain an appropriate set S (for Proposition 2).

Lemma 6. *If $b^+(j) > \frac{1}{\tilde{O}(\log^\alpha n)}$ and $b^-(j') > \frac{1}{\tilde{O}(\log^\beta n)}$, for some j and j' with $2^{|j-j'|} = \tilde{O}(\log^\gamma n)$, then a single iteration of Algorithm 1 finds set S and $i \in S$ with $\min\{z(i), z(i) - z(S)\} \geq \frac{p(S)+q(S)}{\tilde{O}(\log^{\gamma+1} n)}$ with probability $\frac{1}{\tilde{O}(\log^{\alpha+\beta+\gamma+1} n)}$.*

Proof. With probability $\frac{1}{\tilde{O}(\log n)}$, an iteration of Algorithm 1 will choose $r = 2^{-\max\{j,j'\}-3}$. Given this value of r, a unique i with $\hat{z}^+(i) \in [2^{-j}, 2^{-j+1})$ and a unique i' with $\hat{z}^-(i') \in [2^{-j'}, 2^{-j'+1})$ are selected with probability $\frac{1}{\tilde{O}(\log^{\alpha+\beta+\gamma} n)}$. It holds that $z^-(i'), z^+(i) \in [8, \tilde{O}(\log^\gamma n)] \cdot r$ and their corresponding $p(i) + q(i) \leq O(\log n) \cdot z(i) \leq \tilde{O}(\log^{1+\gamma} n)r$ and $p(i') + q(i') \leq \tilde{O}(\log^{1+\gamma} n)r$.

By Markov's inequality, with probability at least 3/4, $z(S \setminus \{i, i'\}) \leq z^+(S \setminus \{i, i'\}) \leq 4r$. Similarly, with probability at least 3/4, $p(S \setminus \{i, i'\}) + q(S \setminus \{i, i'\}) \leq 8r$. By a union bound with probability 1/2 both hold simultaneously.

When all of these events occur, which happens with probability at least $\frac{1}{\tilde{O}(\log^{\alpha+\beta+\gamma+1} n)}$ we get that:

$$\min\{z(i), z(i) - z(S)\} \geq 4r \quad \text{since} \quad z(i) - z(S) \geq z^-(i') - z(S \setminus \{i, i'\}) \geq 4r$$

The lemma follows by noting that $p(S) + q(S) \leq \tilde{O}(\log^{\gamma+1} n)r$. □

Finally, we have our main lemma required for the analysis. It leverages Lemma 6 to show that we can obtain an appropriate set S with reasonable probability.

Lemma 7. *If $d_{\mathrm{TV}}(p, q) = \varepsilon$, then a single iteration of Algorithm 1 finds set S and $i \in S$ with $\min\{z(i), z(i) - z(S)\} \geq \frac{p(S)+q(S)}{\tilde{O}(\log^3 n)}$ with probability $\frac{1}{\tilde{O}(\log^6 n)}$.*

Proof. Before we begin, we require the following two simple concentration lemmas:

Lemma 8. *Let $0 < a < b$, $X_i \sim Bernoulli(2^{-a})$ and let $1 > \sum_{i:x_i<2^{-b}} x_i \geq c$. Then, $\sum_{i:pxi<2^{-b}} X_i x_i > 2^{-a}(c - t2^{-(b-a)/2})$, with probability $1 - e^{-t}$.*

Proof. We apply the Chernoff bound on the variables $Z_i = X_i 2^b x_i$. We get that with probability $1 - e^{-t}$, $2^b \sum_{i:x_i<2^{-b}} X_i x_i > 2^{b-a} c - t 2^{(b-a)/2}$. Thus, $2^a \sum_{i:x_i<2^{-b}} X_i x_i > c - t 2^{-(b-a)/2}$ □

Lemma 9. *Let $a \geq 1$, $X_i \sim Bernoulli(2^{-a})$ and let $1 > \sum_{i:x_i>2^{-a}} x_i$. Then, $\sum_{i:x_i>2^{-a}} X_i x_i = 0$, with probability $\frac{1}{4}$.*

Proof. There are at most 2^a elements x_i and every element is selected independently with probability 2^{-a}. The probability that no element is chosen is $(1 - 2^{-a})^{2^a} \geq \frac{1}{4}$. □

We continue with the main proof. Consider two cases:

1. $d_{\mathrm{K}}(b^+, b^-) \leq \frac{1}{8 \log n}$.

   In this case, as $\sum b^+(j) > 2/3$, there will be a bin j with $b^+(j) \geq \frac{2/3}{2 \log n}$. As the $d_{\mathrm{K}}(b^+, b^-) \leq \frac{1}{8 \log n}$, the corresponding $b^-(j) \geq \frac{1}{3 \log n} - \frac{2}{8 \log n} \geq \frac{1}{12 \log n}$. Then, Lemma 6 implies that a good set will be identified with high probability.

2. $d_{\mathrm{K}}(b^+, b^-) > \frac{1}{8 \log n}$.

   In this case, there will be a bin j_r with $|\sum_{j \geq j_r} b^-(j) - \sum_{j \geq j_r} b^+(j)| \geq \frac{1}{8 \log n}$. Without loss of generality, $\sum_{j \geq j_r} b^+(j) < \sum_{j \geq j_r} b^-(j)$.

   Let j_l be the largest index such that $\frac{1}{8 \log n} < \sum_{j=j_l}^{j_r} b^+(j)$. Then there must exist a $j^* \in [j_l, j_r]$ such that $b^+(j^*) > \frac{1}{16 \log^2 n}$ as $|[j_l, j_r]| \leq 2 \log n$.

If there is a $j \in [j^*, j^*+2\log\log n]$, with $b^-(j) > \frac{1}{100\log n \log\log n}$, Lemma 6 implies that with probability $\frac{1}{O(\log^6 n)}$, $\min\{z(i), z(i)-z(S)\} \geq \frac{p(S)+q(S)}{\tilde{O}(\log^3 n)}$.

Otherwise, we have that $\sum_{j\geq j^*+2\log\log n} b^-(j) > \frac{1}{20\log n} + \sum_{j\geq j^*} b^+(j)$. We will show that in this case, when the algorithm selects $r = 2^{-j^*}$, a good set is identified with non-trivial probability.

With probability $\Omega(b^+(j^*)) = \frac{1}{O(\log^2 n)}$, a unique i with $\hat{z}^+(i) \in [2^{-j^*}, 2^{-j^*+1})$ is selected. It holds that $z^+(i) \in [1,2]\cdot r$ and the corresponding $p(i)+q(i) \leq O(\log n)\cdot z(i) \leq \tilde{O}(\log n) r$.

To complete the proof, we now provide bounds for $z(S\setminus\{i\})$ and $p(S\setminus\{i\})+q(S\setminus\{i\})$.

We decompose $z(S\setminus\{i\})$ into contributions from different sets of elements:

(a) $\hat{z}^+((S\setminus\{i\}) \cap (\bigcup_{j\geq j^*} \mathrm{Bin}_j(z^+))\}) \leq r\sum_{j\geq j^*} b^+(j) + \frac{r}{200\log n}$ with probability at least $\frac{1}{100\log n}$. This holds by Markov's inequality.

(b) $\hat{z}^+((S\setminus\{i\}) \cap (\bigcup_{j<j^*} \mathrm{Bin}_j(z^+))\}) = 0$ with probability 1/4. This holds by Lemma 9.

(c) $z^+(S\setminus\{i\}) - \hat{z}^+(S\setminus\{i\}) \leq 3\frac{r}{200\log n}$ with probability 2/3. This holds by Markov's inequality.

(d) $z^-(S\setminus\{i\}) \geq r\sum_{j\geq j^*+2\log\log n} b^-(j) - \frac{r}{200\log n}$ with probability 15/16. This holds by a concentration bound presented in Lemma 8.

Applying a union bound on cases (b)-(d), we get that they hold simultaneously with probability 1/8. Noting that

Thus, overall $-z(S\setminus\{i\}) \geq r\sum_{j\geq j^*+2\log\log n} b^-(j) - r\sum_{j\geq j^*} b^+(j) - \frac{5r}{200\log n} \geq \frac{r}{20\log n}$.

In addition, $z(i) \geq r$ and thus $\min\{z(i), z(i)-z(S)\} \geq \frac{r}{20\log n}$. With constant probability, we also have that $p(S)+q(S) \leq O(\log n)\cdot r$.

Thus, with probability $\frac{1}{O(\log^4 n)}$, $\min\{z(i), z(i)-z(S)\} \geq \frac{p(S)+q(S)}{\tilde{O}(\log^2 n)}$.

□

Finally, with Lemma 7 in hand, we combine it with Proposition 2 to complete the proof of Theorem 1.

Proof of Theorem 1: Set $T = \Theta(\log^6(n))$. Then Lemma 7 implies that, with constant probability, after T iterations, a set S will be identified such that for some $i \in S$,

$$\min\{z(i), z(i)-z(S)\} \geq \frac{p(S)+q(S)}{\tilde{O}(\log^3 n)}.$$

Proposition 2 then implies that for $\varepsilon' = \frac{\varepsilon}{\tilde{O}(\log^3 n)}$ and $m = \tilde{\Omega}\left(\frac{\log^6 n}{\varepsilon^2}\right)$, the algorithm correctly outputs that $d_{\mathrm{TV}}(p,q) \geq \varepsilon$ with probability at least $1 - \frac{1}{\mathrm{poly}\log n}$.

In contrast, when $d_{\mathrm{TV}}(p,q) = 0$, the algorithm incorrectly correctly outputs that $d_{\mathrm{TV}}(p,q) \geq \varepsilon$ with probability at most $\frac{1}{\mathrm{poly}\log n}$. □

6 Analysis for Identity Testing

In this section, we discuss how our results for uniformity testing imply Theorem 3 for identity testing. We adapt the reduction of [CFGM16], from non-adaptive identity testing to non-adaptive near-uniform identity testing. In particular, we use their Algorithm 4.2.2, with a few crucial differences – to describe these differences, we assume familiarity with the terminology of their paper.

In Line 1, they partition the domain using $Bucket(q, [n], \frac{\varepsilon}{30})$. We perform a less fine-grained partitioning, using $Bucket(q, [n], \frac{1}{100})$. Their bucketing defines M_0 as all i such that $q(i) < \frac{1}{n}$. We define it as all i such that $q(i) < \frac{\varepsilon}{100n}$.[4] The first modification will require a stronger near-uniform identity tester than the one in their paper, which can handle identity testing to any distribution q such that $\|q - \mathcal{U}_n\|_\infty \leq \frac{1}{100n}$. The second

[4] We note that the original definition of M_0 used in [CFGM13, CFGM16] appears to be an erratum, and a similar modification is required for the reduction to go through in their setting as well.

change implies that we do not have to do a near-uniform identity test on M_0 – either $\|z(M_0)\|_1 > \varepsilon/50$ and the discrepancy will be discovered in Line 3, or $\|z(M_0)\|_1 \le \varepsilon/50$, and this bucket can be ignored, as $\|z([n] \setminus M_0)\|_1 \ge 49\varepsilon/50$. As a result of these changes, there are only $\Theta(\log(n/\varepsilon))$ buckets in the partition, and we perform the tests in Line 2 with error bound $\frac{\delta \log(1+1/100)}{2\log(100n/\varepsilon)}$.

With these changes, mimicking the analysis of Theorem 4.2.1 of [CFGM16] gives the following theorem:

Theorem 4. *Suppose there exists a $k(n, \varepsilon, \delta)$-query algorithm, which, given* NACOND *access to an unknown distribution p over $[n]$ and a description of a distribution q over $[n]$ such that $\|q - \mathcal{U}_n\|_\infty \le \frac{1}{100n}$, distinguishes between the cases $p = q$ versus $d_{\mathrm{TV}}(p, q) \ge \varepsilon$ with probability $1 - \delta$.*

Then there exists an algorithm which, given NACOND *access to an unknown distribution p on $[n]$ and a description of a distribution q, makes $\tilde{O}\left(\log(n/\varepsilon) \cdot k\left(n, \varepsilon/2, \frac{\log(1+1/100)}{6\log(100n/\varepsilon)}\right) + \frac{\sqrt{\log(n/\varepsilon)}}{\varepsilon^2}\right)$ queries to the oracle on p and distinguishes between the cases $p = q$ and $d_{\mathrm{TV}}(p, q) \ge \varepsilon$ with probability at least $2/3$.*

In the rest of this section, we will sketch how the analysis of Theorem 2 can be extended to apply to any distribution q such that $\|q - \mathcal{U}_n\|_\infty \le \frac{1}{100n}$, while maintaining the same sample complexity:

Theorem 5 (Non-Adaptive Near-Uniform Identity Testing)**.** *There exists an algorithm which, given* NACOND *access to an unknown distribution p over $[n]$ and a description of a distribution q over $[n]$ such that $\|q - \mathcal{U}_n\|_\infty \le \frac{1}{100n}$, makes $\tilde{O}\left(\frac{\log n}{\varepsilon^2}\right)$ queries to the oracle on p and distinguishes between the cases $p = q$ versus $d_{\mathrm{TV}}(p, q) \ge \varepsilon$ with probability at least $2/3$.*

With this in hand, instantiating Theorem 4 with $k(n, \varepsilon, \delta) = \tilde{O}\left(\frac{\log n}{\varepsilon^2} \cdot \log(1/\delta)\right)$[5] gives Theorem 3.

Most of the analysis in Section 4 involves reasoning about the noise vector z, none of which changes for this setting. The exceptions are at the end of Sections 4.1 and 4.2, where we argue that (1) is large. We deal with the former case first – here, (1) can be written as

$$\varepsilon \cdot \left| \frac{z(i_1)q(i_2) - z(i_2)q(i_1)}{q(S)(q(S) + \varepsilon z(S))} \right| \ge \varepsilon \frac{2 \cdot \frac{1}{5n} \cdot \frac{99}{100n}}{\frac{202}{100n}\left(\frac{202}{100n} + \varepsilon \cdot \frac{32}{n}\right)} \ge \Omega(\varepsilon),$$

as desired. In the latter case, the proof follows with two minimal changes in the events that happen simultaneously (mentioned towards the end of the section). Instead of $\mathcal{U}_n(i) = 1/n$, we have that $q(i) \le 101/100n$. Also, instead of $\frac{1}{2r} \le \mathcal{U}_n(S \setminus i) \le \frac{3}{2r}$, we have that $\frac{1}{2r} \le q(S \setminus i) \le \frac{3}{2r}$. This can be proved by essentially the same argument as Lemma 4, but rescaling at the end by a factor of $100n/99$ or $100n/101$. With these changes, the argument is identical, and thus we have Theorem 5, implying Theorem 3.

7 Open Problems

In this paper, we managed to attain improved upper bounds for several testing problems in the NACOND model. However, there is still much room for improvement, since our upper bounds only match the lower bounds for the case of uniformity testing, where the complexity is known to be $\tilde{\Theta}(\log n)$.

A first question is to sharply characterize the complexity of general identity testing. While in the SAMP model, uniformity testing is known to be complete for identity testing [Gol16], a moment's thought indicates that the same reduction does not immediately hold for either the COND or NACOND model. This is (roughly) because Goldreich's reduction involves mapping the problem onto a larger domain, which would require more "granular" conditional samples than afforded by standard conditional sampling models in order for the reduction to go through. Therefore, it is plausible that testing identity to a general distribution q is *harder* than uniformity testing – this would be a qualitative difference in complexity which we are not aware of in any other sampling model.

Naturally, another question is to characterize the query complexity of equivalence testing. There are several possibilities here – it may be the same as that of uniformity or identity testing, or distinct from both. We would consider either of the former two to be surprising, as this would be qualitatively different behavior than either of the two neighboring oracle models (SAMP and COND).

[5]Note that a standard boosting applied to Theorem 5 gives a $1 - \delta$ probability of success at a multiplicative cost of $\log(1/\delta)$.

Acknowledgments

GK would like to thank Clément L. Canonne for many helpful discussions, Adam Bouland for discussing the connection between conditional sampling and quantum postselection, and Sam Elder for suggesting the name ANACONDA.

References

[ACK15a] Jayadev Acharya, Clément L. Canonne, and Gautam Kamath. Adaptive estimation in weighted group testing. In *Proceedings of the 2015 IEEE International Symposium on Information Theory*, ISIT '15, pages 2116–2120, Washington, DC, USA, 2015. IEEE Computer Society.

[ACK15b] 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.*, RANDOM '15, pages 449–466, Dagstuhl, Germany, 2015. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[AD15] Jayadev Acharya and Constantinos Daskalakis. Testing Poisson binomial distributions. In *Proceedings of the 26th Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '15, pages 1829–1840, Philadelphia, PA, USA, 2015. SIAM.

[ADJ+11] Jayadev Acharya, Hirakendu Das, Ashkan Jafarpour, Alon Orlitsky, and Shengjun Pan. Competitive closeness testing. In *Proceedings of the 24th Annual Conference on Learning Theory*, COLT '11, pages 47–68, 2011.

[ADK15] Jayadev Acharya, Constantinos Daskalakis, and Gautam Kamath. Optimal testing for properties of distributions. In *Advances in Neural Information Processing Systems 28*, NIPS '15, pages 3577–3598. Curran Associates, Inc., 2015.

[BC17a] Tuğkan Batu and Clément L. Canonne. Generalized uniformity testing. In *Proceedings of the 58th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '17, pages 880–889, Washington, DC, USA, 2017. IEEE Computer Society.

[BC17b] Rishiraj Bhattacharyya and Sourav Chakraborty. Property testing of joint distributions using conditional samples. *arXiv preprint arXiv:1702.01454*, 2017.

[BCG17] Eric Blais, Clément L. Canonne, and Tom Gur. Distribution testing lower bounds via reductions from communication complexity. In *Proceedings of the 32nd Computational Complexity Conference*, CCC '17, pages 28:1–28:40, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[BDKR05] Tuğkan Batu, Sanjoy Dasgupta, Ravi Kumar, and Ronitt Rubinfeld. The complexity of approximating the entropy. *SIAM Journal on Computing*, 35(1):132–150, 2005.

[BFF+01] Tuğkan Batu, Eldar Fischer, Lance Fortnow, Ravi Kumar, Ronitt Rubinfeld, and Patrick White. Testing random variables for independence and identity. In *Proceedings of the 42nd Annual IEEE Symposium on Foundations of Computer Science*, FOCS '01, pages 442–451, Washington, DC, USA, 2001. IEEE Computer Society.

[BFR+00] Tuğkan Batu, Lance Fortnow, Ronitt Rubinfeld, Warren D. Smith, and Patrick White. Testing that distributions are close. In *Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science*, FOCS '00, pages 259–269, Washington, DC, USA, 2000. IEEE Computer Society.

[BFRV11] Arnab Bhattacharyya, Eldar Fischer, Ronitt Rubinfeld, and Paul Valiant. Testing monotonicity of distributions over general partial orders. In *Proceedings of the 2nd Conference on Innovations in Computer Science*, ICS '11, pages 239–252, Beijing, China, 2011. Tsinghua University Press.

[BKR04] Tuğkan Batu, Ravi Kumar, and Ronitt Rubinfeld. Sublinear algorithms for testing monotone and unimodal distributions. In *Proceedings of the 36th Annual ACM Symposium on the Theory of Computing*, STOC '04, New York, NY, USA, 2004. ACM.

[BV15] Bhaswar Bhattacharya and Gregory Valiant. Testing closeness with unequal sized samples. In *Advances in Neural Information Processing Systems 28*, NIPS '15, pages 2611–2619. Curran Associates, Inc., 2015.

[Can15a] Clément L. Canonne. Big data on the rise? - testing monotonicity of distributions. In *Proceedings of the 42nd International Colloquium on Automata, Languages, and Programming*, ICALP '15, pages 294–305, 2015.

[Can15b] Clément L. Canonne. A survey on distribution testing: Your data is big. but is it blue? *Electronic Colloquium on Computational Complexity (ECCC)*, 22(63), 2015.

[Can16] Clément L. Canonne. Are few bins enough: Testing histogram distributions. In *Proceedings of the 35th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, PODS '16, pages 455–463, New York, NY, USA, 2016. ACM.

[CDVV14] Siu-On Chan, Ilias Diakonikolas, Gregory Valiant, and Paul Valiant. Optimal algorithms for testing closeness of discrete distributions. In *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '14, pages 1193–1203, Philadelphia, PA, USA, 2014. SIAM.

[CF00] Kumar Chellapilla and David B. Fogel. Anaconda defeats Hoyle 6-0: a case study competing an evolved checkers program against commercially available software. In *Proceedings of the 2000 Congress on Evolutionary Computation*, CEC '00, pages 857–863, Washington, DC, USA, 2000. IEEE Computer Society.

[CFGM13] Sourav Chakraborty, Eldar Fischer, Yonatan Goldhirsh, and Arie Matsliah. On the power of conditional samples in distribution testing. In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science*, ITCS '13, pages 561–580, New York, NY, USA, 2013. ACM.

[CFGM16] 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.

[CR14] Clément L. Canonne and Ronitt Rubinfeld. Testing probability distributions underlying aggregated data. In *Proceedings of the 41st International Colloquium on Automata, Languages, and Programming*, ICALP '14, pages 283–295, 2014.

[CRS14] Clément L. Canonne, Dana Ron, and Rocco A. Servedio. Testing equivalence between distributions using conditional samples. In *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '14, pages 1174–1192, Philadelphia, PA, USA, 2014. SIAM.

[CRS15] Clément L. Canonne, Dana Ron, and Rocco A. Servedio. Testing probability distributions using conditional samples. *SIAM Journal on Computing*, 44(3):540–616, 2015.

[DDK18] Constantinos Daskalakis, Nishanth Dikkala, and Gautam Kamath. Testing Ising models. In *Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '18, Philadelphia, PA, USA, 2018. SIAM.

[DDS+13] Constantinos Daskalakis, Ilias Diakonikolas, Rocco A. Servedio, Gregory Valiant, and Paul Valiant. Testing k-modal distributions: Optimal algorithms via reductions. In *Proceedings of the 24th Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '13, pages 1833–1852, Philadelphia, PA, USA, 2013. SIAM.

[DGPP18] Ilias Diakonikolas, Themis Gouleakis, John Peebles, and Eric Price. Sample-optimal identity testing with high probability. In *Proceedings of the 45th International Colloquium on Automata, Languages, and Programming*, ICALP '18, pages 41:1–41:14, 2018.

[Dis16] Anaconda Software Distribution. Anaconda. Version 2-2.4.0. `https://anaconda.com`, November 2016.

[DK16] Ilias Diakonikolas and Daniel M. Kane. A new approach for testing properties of discrete distributions. In *Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '16, pages 685–694, Washington, DC, USA, 2016. IEEE Computer Society.

[DKN15] Ilias Diakonikolas, Daniel M. Kane, and Vladimir Nikishkin. Testing identity of structured distributions. In *Proceedings of the 26th Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '15, pages 1841–1854, Philadelphia, PA, USA, 2015. SIAM.

[DKW56] Aryeh Dvoretzky, Jack Kiefer, and Jacob Wolfowitz. Asymptotic minimax character of the sample distribution function and of the classical multinomial estimator. *The Annals of Mathematical Statistics*, 27(3):642–669, 09 1956.

[DKW18] Constantinos Daskalakis, Gautam Kamath, and John Wright. Which distribution distances are sublinearly testable? In *Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '18, pages 2747–2764, Philadelphia, PA, USA, 2018. SIAM.

[Fis14] Eldar Fischer. Problem 66: Distinguishing distributions with conditional samples. `https://sublinear.info/index.php?title=Open_Problems:66`, May 2014. Open Problem from Bertinoro Workshop on Sublinear Algorithms 2014.

[FJO$^+$15] Moein Falahatgar, Ashkan Jafarpour, Alon Orlitsky, Venkatadheeraj Pichapati, and Ananda Theertha Suresh. Faster algorithms for testing under conditional sampling. In *Proceedings of the 28th Annual Conference on Learning Theory*, COLT '15, pages 607–636, 2015.

[FLV17] Eldar Fischer, Oded Lachish, and Yadu Vasudev. Improving and extending the testing of distributions for shape-restricted properties. In *Proceedings of the 34th Symposium on Theoretical Aspects of Computer Science*, STACS '17, pages 31:1–31:14, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[FV12] Jan Fiedor and Tomáš Vojnar. Anaconda: a framework for analysing multi-threaded c/c++ programs on the binary level. In *International Conference on Runtime Verification*, RV '12, pages 35–41, Berlin, Heidelberg, 2012. Springer.

[GMV06] Sudipto Guha, Andrew McGregor, and Suresh Venkatasubramanian. Streaming and sublinear approximation of entropy and information distances. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '06, pages 733–742, Philadelphia, PA, USA, 2006. SIAM.

[Gol16] Oded Goldreich. The uniform distribution is complete with respect to testing identity to a fixed distribution. *Electronic Colloquium on Computational Complexity (ECCC)*, 23(15), 2016.

[Gol17] Oded Goldreich. *Introduction to Property Testing*. Cambridge University Press, 2017.

[GR00] Oded Goldreich and Dana Ron. On testing expansion in bounded-degree graphs. *Electronic Colloquium on Computational Complexity (ECCC)*, 7(20), 2000.

[GTZ17] Themistoklis Gouleakis, Christos Tzamos, and Manolis Zampetakis. Faster sublinear algorithms using conditional sampling. In *Proceedings of the 28th Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '17, pages 1743–1757, Philadelphia, PA, USA, 2017. SIAM.

[GTZ18] Themis Gouleakis, Christos Tzamos, and Manolis Zampetakis. Certified computation from unreliable datasets. In *Proceedings of the 31st Annual Conference on Learning Theory*, COLT '18, pages 3271–3294, 2018.

[Hey87] Eckhard W. Heymann. A field observation of predation on a moustached tamarin (saguinus mystax) by an anaconda. *International Journal of Primatology*, 8(2):193–195, 1987.

[ILR12] Piotr Indyk, Reut Levi, and Ronitt Rubinfeld. Approximating and testing k-histogram distributions in sub-linear time. In *Proceedings of the 31st ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, PODS '12, pages 15–22, New York, NY, USA, 2012. ACM.

[Mas90] P. Massart. The tight constant in the Dvoretzky-Kiefer-Wolfowitz inequality. *The Annals of Probability*, 18(3):1269–1283, 07 1990.

[Mer01] Laurie Mercier. *Anaconda: labor, community, and culture in Montana's smelter city*, volume 286. University of Illinois Press, 2001.

[MJSM+14] Onika Maraj, Jamal Jones, Jonathan Solone-Myvett, Ernest Clark, Marcos Palacios, and Anthony Ray. Anaconda. Music single, August 2014.

[OS18] Krzysztof Onak and Xiaorui Sun. Probability-revealing samples. In *Proceedings of the 21st International Conference on Artificial Intelligence and Statistics*, AISTATS '18, pages 84:2018–2026. JMLR, Inc., 2018.

[Pan08] Liam Paninski. A coincidence-based test for uniformity given very sparsely sampled discrete data. *IEEE Transactions on Information Theory*, 54(10):4750–4755, 2008.

[RS09] Ronitt Rubinfeld and Rocco A. Servedio. Testing monotone high-dimensional distributions. *Random Structures and Algorithms*, 34(1):24–44, 2009.

[Rub12] Ronitt Rubinfeld. Taming big probability distributions. *XRDS*, 19(1):24–28, 2012.

[SSJ17] Imdad S. B. Sardharwalla, Sergii Strelchuk, and Richard Jozsa. Quantum conditional query complexity. *Quantum Information & Computation*, 17(7& 8):541–566, 2017.

[Val11] Paul Valiant. Testing symmetric properties of distributions. *SIAM Journal on Computing*, 40(6):1927–1968, 2011.

[VV17] Gregory Valiant and Paul Valiant. An automatic inequality prover and instance optimal identity testing. *SIAM Journal on Computing*, 46(1):429–455, 2017.

[Wag15] Bo Waggoner. l_p testing and learning of discrete distributions. In *Proceedings of the 6th Conference on Innovations in Theoretical Computer Science*, ITCS '15, pages 347–356, New York, NY, USA, 2015. ACM.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il