Equivalence Testing: The Power of Bounded Adaptivity

Diptarka Chakraborty
National University of Singapore

Sourav Chakraborty
Indian Statistical Institute, Kolkata

Gunjan Kumar
National University of Singapore

Kuldeep S. Meel
University of Toronto

Abstract

Equivalence testing, a fundamental problem in the field of distribution testing, seeks to infer if two unknown distributions on $[n]$ are the same or far apart in the total variation distance. Conditional sampling has emerged as a powerful query model and has been investigated by theoreticians and practitioners alike, leading to the design of optimal algorithms albeit in a sequential setting (also referred to as adaptive tester). Given the profound impact of parallel computing over the past decades, there has been a strong desire to design algorithms that enable high parallelization. Despite significant algorithmic advancements over the last decade, parallelizable techniques (also termed non-adaptive testers) have $\tilde{O}(\log^{12} n)$ query complexity, a prohibitively large complexity to be of practical usage. Therefore, the primary challenge is whether it is possible to design algorithms that enable high parallelization while achieving efficient query complexity.

Our work provides an affirmative answer to the aforementioned challenge: we present a highly parallelizable tester with a query complexity of $\tilde{O}(\log n)$, achieved through a single round of adaptivity, marking a significant stride towards harmonizing parallelizability and efficiency in equivalence testing.

1 Introduction

Evaluating different properties of an unknown object is a fundamental challenge in statistics. When dealing with large objects, it becomes essential to determine these properties by making only a limited number of queries to the object. In the case of unknown objects being probability distributions, the goal is to assess whether the input distribution(s) possess specific properties or deviate significantly (i.e., "ε-far" for some $\varepsilon > 0$) from meeting them. All of this needs to be accomplished while minimizing the number of queries made (also known as *query complexity*) to the distribution(s). Probability distributions are crucial subjects of study, and distribution testing has remained central to sublinear algorithms and modern data analysis since its introduction [Goldreich et al., 1998, Goldreich and Ron, 2011, Batu et al., 2013].

Early investigations into distribution testing primarily utilized the SAMP query model, which only allows drawing samples from the given distribution(s). However, for testing many interesting properties, the SAMP model proves to be restrictive, as evidenced by strong polynomial (in domain size) lower bounds on the sample complexity. To overcome this limitation, several alternative query models have been proposed over the past decade. Among these models, the conditional sampling model (COND) [Chakraborty et al., 2013, Canonne et al., 2014] has been extensively studied. This model permits drawing samples from the input distribution(s) conditioned on any arbitrary subset of the domain. Various distribution testing problems have been explored under the COND model [Falahatgar et al., 2015, Kamath and Tzamos, 2019, Narayanan, 2021, Chakraborty et al., 2023] and certain variants of it like *subcube conditioning model* [Bhattacharyya and Chakraborty, 2018, Canonne et al., 2021, Chen et al., 2021]. Moreover, the COND model and its variants have recently found applications in the areas like formal methods and machine learning (e.g., [Chakraborty and Meel, 2019, Meel et al., 2020, Golia et al., 2022]).

In this work, we study the *equivalence testing* problem, one of the most fundamental problems in distribution testing. In this problem, given query access

Proceedings of the 27th International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238.

to two (unknown) distributions $\mathcal{P}$ and $\mathcal{Q}$, the objective is to decide whether they are equal or ε-far from each other in the *total variation distance*. For this problem, a query-optimal algorithm is already known in the COND model. However, the primary challenge with the query-efficient algorithm/tester in the COND model is its inherent sequential or adaptive nature. A tester is considered non-adaptive if it can generate all its queries based solely on the input parameter (in this case, the domain size) and its internal randomness, without relying on previous query responses. Non-adaptive testers are generally favored in practical situations because they can make multiple queries simultaneously.

Exploring the balance between the degree of adaptivity and query complexity is a captivating area of research. This curiosity prompted [Canonne and Gur, 2018] to delve deeper into adaptive testing, introducing a nuanced approach by permitting a limited number of adaptive stages or rounds. This multi-stage or bounded-round adaptivity concept finds resonance in various other problems, including group testing [Du et al., 2000, Damaschke et al., 2013, Eberhardt et al., 2020], submodular function maximization [Balkanski and Singer, 2018, Chekuri and Quanrud, 2019], compressed sensing and sparse recovery [Nakos et al., 2018, Kamath and Price, 2019], multi-armed bandits problem [Agarwal et al., 2017]. In this work, we initiate the study of bounded-round adaptivity in the context of the equivalence testing problem and provide a query-efficient one-round adaptive tester.

2 Notations and Preliminaries

Throughout this paper, we consider distributions over the domain $[n] := \{1, 2, \cdots, n\}$. For any $i, j \in [n]$ and $S \subseteq [n]$, for brevity we use $i \cup j$ and $i \cup S$ to denote the sets $\{i\} \cup \{j\}$ and $\{i\} \cup S$ respectively.

Given a distribution $\mathcal{D}$ over $[n]$ and $i \in [n]$, we use $\mathcal{D}(i)$ to represent the probability mass function of i. Similarly, for $S \subseteq [n]$, we use $\mathcal{D}(S)$ to denote $\sum_{i \in S} \mathcal{D}(i)$. For any $\gamma \in (0, 1)$, a $\pm\gamma$ estimate of a quantity (say d) means a number $\tilde{d} \in [d - \gamma, d + \gamma]$.

The total variation distance between two distributions $\mathcal{P}$ and $\mathcal{Q}$, denoted by $d_{TV}(\mathcal{P}, \mathcal{Q})$ is defined as

$$d_{TV}(\mathcal{P}, \mathcal{Q}) := \frac{1}{2} \sum_{i \in [n]} |\mathcal{P}(i) - \mathcal{Q}(i)|.$$

If the variation distance between two distributions is more than ε, then we say the distributions are ε-far (or just far, when it is clear from the context).

The Binomial distribution, with parameters $n \in \mathbb{Z}^+$ and $p \in [0, 1]$ denoted by $\mathsf{Bin}(n, p)$ is the distribution of the number of successes in n independent experiments, where each experiment yields a Boolean outcome, with success occurring with probability p and failure with probability $1 - p$.

Definition 1 (COND Query Model)**.** *A conditional sampling oracle for a distribution $\mathcal{D}$ is defined as follows: the oracle takes as input a subset $S \subseteq [n]$ and returns an element $j \in S$, such that the probability that $j \in S$ is returned is equal to $\mathcal{D}(j)/\mathcal{D}(S)$ if $\mathcal{D}(S) > 0$ and $1/|S|$ if $\mathcal{D}(S) = 0$.*

We denote such a conditional query by $\mathsf{COND}_{\mathcal{D}}(S)$.

The formal definition of a k-round adaptive tester is given in [Canonne and Gur, 2018]. For completeness, we present the formal definition of a one-round adaptive tester for equivalence in the COND Query Model.

Definition 2. *Given conditional query access to distributions $\mathcal{P}$ and $\mathcal{Q}$ (over domain $[n]$), and given tolerance parameter ε as inputs, a one-round adaptive tester $\mathcal{A}$ makes conditional queries to the distributions in two rounds:*

1. *In the first round, the algorithm $\mathcal{A}$ (without making any queries to the distributions) selects a set of subsets (say $S_1^0, \ldots S_{q_1}^0$ of $[n]$ and then makes the conditional queries $\mathsf{COND}_{\mathcal{P}}(S_1^0), \ldots, \mathsf{COND}_{\mathcal{P}}(S_{q_1}^0)$ and $\mathsf{COND}_{\mathcal{Q}}(S_1^0), \ldots, \mathsf{COND}_{\mathcal{Q}}(S_{q_1}^0)$.*
2. *In the second round, based on the answers to the queries it has received in the first round, it selects another set of subsets (say $S_1^1, \ldots S_{q_2}^1$ of $[n]$ and then makes the conditional queries $\mathsf{COND}_{\mathcal{P}}(S_1^1), \ldots, \mathsf{COND}_{\mathcal{P}}(S_{q_2}^1)$ and $\mathsf{COND}_{\mathcal{Q}}(S_1^1), \ldots, \mathsf{COND}_{\mathcal{Q}}(S_{q_2}^1)$.*

Finally, based on the answers to all the $2(q_1 + q_2)$ queries, $\mathcal{A}$ outputs with the following guarantee:

- *if $\mathcal{P}$ and $\mathcal{Q}$ are identical, then with probability at least $2/3$, $\mathcal{A}$ outputs* Accept*, and*
- *if $d_{TV}(\mathcal{P}, \mathcal{Q}) \geq \varepsilon$, then with probability at least $2/3$, $\mathcal{A}$ outputs* Reject.

The query/sample complexity of the algorithm is $2(q_1 + q_2)$.

In our proof, we will extensively use concentration lemmas. In particular, we will use the following version of Chernoff bound.

Lemma 3 (Additive Chernoff bound)**.** *Let $X_1, \ldots, X_m$ be m iid random variables, each X_i*

takes value in $\{0,1\}$ *and* $\mathbb{E}[X_i] = p$. *Then for any* $\gamma \in (0,1)$,

$$\Pr\left[\left|\sum_{i\in[m]} X_i/m - p\right| \geq \gamma\right] < e^{-\gamma^2 m}.$$

3 Related Work

In the standard SAMP model, the query complexity of the equivalence testing problem is $\Theta(\max(n^{2/3}/\varepsilon^{4/3}, \sqrt{n}/\varepsilon^2))$ [Chan et al., 2014, Batu et al., 2013, Valiant, 2011], which is prohibited in most practical applications. The COND model turns out to be beneficial in this context, enabling to require only $\tilde{O}(\log\log n)$ samples [Falahatgar et al., 2015], which has recently been shown to be optimal [Chakraborty et al., 2024].

Unfortunately, the above-mentioned optimal tester in the COND model is sequential. In simpler terms, the tester is adaptive, meaning that each query (indexed as t for any $t \geq 1$) in the COND model depends on the answers to the preceding $t-1$ queries. Designing a parallel, ideally entirely non-adaptive, tester remains an enormous challenge. [Kamath and Tzamos, 2019] introduced a non-adaptive tester for the equivalence testing problem, which required $\tilde{O}(\log^{12} n/\varepsilon^2)$ queries[1]. However, the substantial poly-logarithmic dependency on the domain size is impractical in many real-world applications. Moreover, the best-known lower bound for the query complexity of non-adaptive testers is $\Omega(\log n)$ [Acharya et al., 2018], indicating considerable room for improvement in the upper bound. One exciting question is to make the tester as less adaptive as possible while attaining the optimal non-adaptive query complexity. Such a question motivates the researchers to study the trade-off between the number of adaptive rounds and the query complexity (for testing various properties). The work [Canonne and Gur, 2018] led to the establishment of a "hierarchy theorem" examining the impact of the number of adaptive rounds.

For the classical equivalence testing problem, in this paper, we make significant strides toward achieving optimal (non-adaptive) query complexity of $O(\log n)$ by allowing only one round of adaptivity.

Comarison of our algorithm to $\tilde{O}(\log^{12} n)$-query algorithm by [Kamath and Tzamos, 2019]: [Kamath and Tzamos, 2019] provided a non-adaptive tester for the equivalence testing problem, which requires $\tilde{O}(\log^{12} n)$ queries. Let us briefly describe their algorithm and compare it with ours. They construct $\tilde{O}(\log^6 n)$-many sets of varying sizes. Subsequently, they compare the empirical conditional distribution over these subsets for both distributions, by performing $\tilde{O}(\log^6 n)$ conditional sampling queries on each subsets. If these empirical distributions exhibit significant differences, their algorithm returns reject. On the other hand, if the empirical distributions are close for all such subsets, the algorithm returns accept.

Despite the similarity in the construction of sets of varying sizes, our algorithm differs in both description as well as analysis. In our approach, a key component is the role of tuple (i, S) as a certificate that two distributions are far, whereas their algorithm solely considers subsets S. Rather than arguing that the empirical distribution over these subsets will differ significantly (if the distributions are far apart), we argue that certain detectable properties will differ with respect to the tuple (i, S). This additional dimension allows us to reduce the number of queries significantly. However, it is important to note that our algorithm involves a single round of adaptivity. As a result, our algorithm is incomparable to theirs in this regard.

4 An Efficient One-Round Adaptive Algorithm

Our main contribution is a one-round adaptive tester for the equivalence testing problem in the COND model that makes at most $\tilde{O}(\log n)$ queries.

Theorem 4. *There exists an algorithm which, given* COND *access to two distributions* $\mathcal{P}$ *and* $\mathcal{Q}$ *on* $[n]$ *and a parameter* $\varepsilon > 0$, *makes at most* $O\left(\frac{\log n(\log\log n)\log 1/\varepsilon}{\varepsilon^9}\right)$ *queries to the oracle and is one-round adaptive, and decides whether* $\mathcal{P} = \mathcal{Q}$ *or* $d_{TV}(\mathcal{P},\mathcal{Q}) \geq \varepsilon$ *with probability at least* $2/3$.

In the remaining part of this paper, we will prove the above theorem. We start with a high-level idea behind our algorithm, and then we provide a formal description of the algorithm with a detailed analysis.

4.1 High-Level Overview

The first attempt to design an equivalence tester is to pick a sample, say i, from $\mathcal{P}$ and compare the probability mass $\mathcal{P}(i)$ and $\mathcal{Q}(i)$ in $\mathcal{P}$ and $\mathcal{Q}$ respectively. If $\mathcal{P}$ and $\mathcal{Q}$ are far (in total variation distance), then with enough probability, the probability mass of i in both distributions is significantly different. However, the issue is that estimating the probability mass of the element i can be very expensive. To bypass this issue, the idea is to sample a subset S of the domain, hoping the following:

[1] Note that $\tilde{O}(f(n))$ notation hides $poly(\log f(n))$ terms.

- the probability mass of S in $\mathcal{P}$ is comparable to that of the mass of S in $\mathcal{Q}$, and
- the probability mass of i (in $\mathcal{P}$) is similar to the probability mass of S.

Assuming that all the above two statements hold, one can use conditional sampling (conditioned on $S \cup i$) to compare $\mathcal{P}(i)/\mathcal{P}(S \cup i)$ and $\mathcal{Q}(i)/\mathcal{Q}(S \cup i)$. Since, we assumed $\mathcal{P}(S)$ is similar to $\mathcal{Q}(S)$ so with high enough probability $\mathcal{P}(i)/\mathcal{P}(S \cup i)$ and $\mathcal{Q}(i)/\mathcal{Q}(S \cup i)$ will be different enough. And since we assumed that $\mathcal{P}(i)$ is comparable to $\mathcal{P}(S)$, one can estimate $\mathcal{P}(i)/\mathcal{P}(S \cup i)$ and can upper bound $\mathcal{Q}(i)/\mathcal{Q}(S \cup i)$ using only a few samples, which should be sufficient to distinguish $\mathcal{P}$ from $\mathcal{Q}$ if the the two distributions are far. But the issue is how to take care of the two assumptions.

Firstly, for the second assumption, since we don't know the quantity $\mathcal{P}(i)$ beforehand, trying to pick a S with similar probability seems unrealistic. For this, we pick a collection of sets, $S_1, \ldots, S_{\log n}$, where the set S_k is obtained by picking each element of the domain with probability $1/2^k$. This ensures that the expected value of $\mathcal{P}(S_k)$ is $1/2^k$. So, irrespective of what the value of $\mathcal{P}(i)$ is, there exists (with high probability) a S^* such that $\mathcal{P}(S^*)$ is comparable to $\mathcal{P}(i)$. Thus, the hope is to go over all the sets and estimate the $\mathcal{P}(i)/\mathcal{P}(S_k \cup i)$ and $\mathcal{Q}(i)/\mathcal{Q}(S_k \cup i)$ for all the $\log n$ sets, as long as the ratios are within a particular range. Assuming that $\mathcal{P}(S)$ and $\mathcal{Q}(S)$ is comparable and $\mathcal{P}$ and $\mathcal{Q}$ are far, the value of $\mathcal{P}(i)/\mathcal{P}(S^* \cup i)$ and $\mathcal{Q}(i)/\mathcal{Q}(S^* \cup i)$ will be different enough.

For the above argument to go through, we need the other assumption that the probability mass of S in $\mathcal{P}$ is comparable to that of the mass of S in $\mathcal{Q}$. Since the sets S_k are obtained by independently drawing elements from the domain, one expects the assumption to hold. While the expected weight of S according to $\mathcal{P}$ and $\mathcal{Q}$ will be the same, we need to prove concentration. The concentration is hard to achieve in this case. In other words, when the random set S_k is drawn the expected value of $\mathcal{P}(S_k) = \mathcal{Q}(S_k) = 1/2^k$, and either of the two cases can happen:

- (Case 1) The value of the random variable $\mathcal{P}(S_k) - \mathcal{Q}(S_k)$ is concentrated around 0, or
- (Case 2) There is large "tail probability," because of which concentration is not possible. (The notion of tail probability is formalized in Section 4.3).

If Case 1 holds, that is, the value of the random variables $\mathcal{P}(S_k)$ and $\mathcal{Q}(S_k)$ are concentrated around the expectation then the argument of the previous para goes through and we will be able to distinguish $\mathcal{P}$ from $\mathcal{Q}$ by estimating $\mathcal{P}(i)/\mathcal{P}(S^* \cup i)$ and $\mathcal{Q}(i)/\mathcal{Q}(S^* \cup i)$.

On the other hand, if Case 2 holds, then it means that the tail probability is high, and if this happens, it means $\mathcal{P}$ and $\mathcal{Q}$ are far. This case can be caught by estimating the tail probability. This is what is done in our algorithm EstTail.

So our main tester EquivTester first picks a number of samples according to $\mathcal{P}$ and then constructs $\tilde{O}(\log n)$ sets S_k. For each set S_k and for each sample i it estimates the difference between $\mathcal{P}(i)/\mathcal{P}(S_k \cup i)$ and $\mathcal{Q}(i)/\mathcal{Q}(S_k \cup i)$ and also the tail probability (using EstTail). If either of the two estimates is large, the algorithm rejects $\mathcal{P}$ and $\mathcal{Q}$, and if the algorithm does not reject in all the iterations, then the algorithm accepts.

Note that the power of conditional samples is used in estimating the values of $\mathcal{P}(i)/\mathcal{P}(S_k \cup i)$ and $\mathcal{Q}(i)/\mathcal{Q}(S_k \cup i)$ and also for estimating the tail probabilities. Regarding the amount of adaptiveness used in the algorithm EquivTester, we observe that once the sets S_k's are fixed, the rest of the samples (conditional samples) can be drawn in parallel.

4.2 Algorithm Description

Our algorithm, EquivTester, takes as input, two distributions $\mathcal{P}$ and $\mathcal{Q}$, and a parameter $\varepsilon > 0$. It returns Accept if $\mathcal{P} = \mathcal{Q}$ and Reject if their total variation distance $d_{TV}(\mathcal{P}, \mathcal{Q})$ is greater than ε, both with at least $2/3$ probability.

EquivTester samples $O(1/\varepsilon)$ points from $\mathcal{P}$, with the set of all such points denoted by E (line 4). It then constructs subset S_t for each t in $\{1, 1/2, 1/4, \ldots, 1/n\}$, such that each element from $[n]$ is included in S_t with probability t (lines 5–6). The algorithm then employs two subroutines, EstProb and EstTail, for each tuple (i, S) (where $i \in E$ and $S = S_t$ for some t) (lines 8–16). We invoke EstProb to estimate corresponding conditional probabilities $\mathcal{P}(i)/\mathcal{P}(i \cup S)$ and $\mathcal{Q}(i)/\mathcal{Q}(i \cup S)$ and return Reject if the difference between conditional probabilities is far (lines 9–12). If the difference is not far, we invoke EstTail (to estimate the tail probability TP, formally defined in Section 4.3) and again, we reject if the difference between the tail probabilities of (i, S) for $\mathcal{P}$ and $\mathcal{Q}$ is far (lines 13–16). Finally, if for all tuples (i, S), all these estimates are close, EquivTester returns Accept(line 17).

Query complexity: We now establish an upper bound on the number of calls to the COND oracle by the EquivTester algorithm. Note that each invocation of EstProb results in $m = O\left(\frac{(\log \log n) \log 1/\varepsilon}{\varepsilon^8}\right)$

calls to the COND oracle, and each invocation of EstTail leads to $mb = 40000m$ calls to the COND oracle. Given that EquivTester invokes both EstProb and EstTail at most $|E \times \mathcal{S}| \leq 20 \log n/\varepsilon$ times, the total number of calls made to the COND oracle is at most $O\left(\frac{\log n(\log\log n)\log 1/\varepsilon}{\varepsilon^9}\right)$.

Making EquivTester one-round adaptive: For the sake of improved presentation of EquivTester, we have opted not to group together the conditional queries that can be made simultaneously. We now modify EquivTester by re-arranging the order of the conditional queries and making it a one-round adaptive algorithm. First, we note that the construction of $\mathcal{S} = \{S_t : t \in \{1, 1/2, \ldots, 1/n\}\}$ does not require any call to the COND oracle.

To convert EquivTester into a one-round algorithm, all conditional queries executed in line 4 of EquivTester and line 1 of EstTail can be made simultaneously. This is possible since these queries are either of the form $\mathsf{COND}_{\mathcal{P}}([n])$ or $\mathsf{COND}_{\mathcal{P}}(S)$ and $\mathsf{COND}_{\mathcal{Q}}(S)$ for some $S \in \mathcal{S}$ and as noted before, the set $\mathcal{S}$ can be constructed beforehand.

Then, we can make all the remaining conditional queries simultaneously. These queries are of the form: (i) $\mathsf{COND}_{\mathcal{P}}(i \cup S)$ and $\mathsf{COND}_{\mathcal{Q}}(i \cup S)$ (line 1 of EstProb) where $i \sim \mathsf{COND}_{\mathcal{P}}([n])$ is the outcome of query made in the previous round and $S \in \mathcal{S}$ is available beforehand, (ii) $\mathsf{COND}_{\mathcal{P}}(i \cup j)$ and $\mathsf{COND}_{\mathcal{Q}}(i \cup j)$ (line 3 of EstTail) where again $i \sim \mathsf{COND}_{\mathcal{P}}([n])$ and $j \sim \mathsf{COND}_{\mathcal{D}}(S)$ (for some $S \in \mathcal{S}$ and $\mathcal{D} \in \{\mathcal{P}, \mathcal{Q}\}$) are the outcomes from the queries in the first round.

4.3 Technical Analysis

Before we formally present the analysis of the correctness and complexity of EquivTester, we define the tail probability, TP. Given a distribution $\mathcal{D}$, a tuple (i, S) where $i \in [n]$ and $S \subseteq [n]$, and parameters $\beta \in (0,1)$ and $b \in \mathbb{Z}^+$, the tail probability $\mathsf{TP}(\mathcal{D}, i, S, \beta, b)$ is defined as follows:

It is the probability that a random sample $j \sim \mathsf{COND}_{\mathcal{D}}(S)$ will occur no more than $\frac{1}{2} + \beta$ times in b independent queries of $\mathsf{COND}_{\mathcal{D}}(i, j)$. Formally, it can be expressed as:

$$\begin{aligned}
&\mathsf{TP}(\mathcal{D}, i, S, \beta, b) := \\
&\quad \Pr_{j \sim \mathsf{COND}_{\mathcal{D}}(S)}\left[\mathsf{Bin}\left(b, \frac{\mathcal{D}(j)}{\mathcal{D}(j) + \mathcal{D}(i)}\right) \leq (\frac{1}{2} + \beta)b\right] \\
&\quad = \sum_{j \in S} \frac{\mathcal{D}(j)}{\mathcal{D}(S)} \Pr\left[\mathsf{Bin}\left(b, \frac{\mathcal{D}(j)}{\mathcal{D}(j) + \mathcal{D}(i)}\right) \leq (\frac{1}{2} + \beta)b\right].
\end{aligned}$$

We now analyze our algorithms. The subroutine $\mathsf{EstProb}(\mathcal{D}, i, S, m)$ takes as input a distribution $\mathcal{D}$, $i \in [n]$, $S \subseteq [n]$ and a parameter $m \in \mathbb{Z}^+$. It uses a straightforward estimator to return $\pm\gamma$ estimate of $\frac{\mathcal{D}(i)}{\mathcal{D}(i \cup S)}$ with probability at least $1 - e^{-\gamma^2 m}$.

Lemma 5. *For an arbitrary $i \in [n]$, $S \subseteq [n]$, $\gamma \in (0,1)$ and $m \geq 1$ and distribution $\mathcal{D}$, the $\mathsf{EstProb}(\mathcal{D}, i, S, m)$ returns $\pm\gamma$ estimate of $\frac{\mathcal{D}(i)}{\mathcal{D}(i \cup S)}$ with probability at least $1 - e^{-\gamma^2 m}$.*

Algorithm 1 $\mathsf{EquivTester}(\mathcal{P}, \mathcal{Q}, \varepsilon)$

Input: A pair of distribution $\mathcal{P}, \mathcal{Q}$ on $[n]$, $\varepsilon > 0$
Output: Accept with prob. $2/3$ if $\mathcal{P} = \mathcal{Q}$, Reject with prob. $2/3$ if $d_{TV}(\mathcal{P}, \mathcal{Q}) \geq \varepsilon$.

1: $\gamma \leftarrow \frac{\varepsilon^4}{L}$ where $L \leftarrow 10^{15}$ is a large constant.
2: $m \leftarrow 100(\log\log n)\log\frac{1}{\varepsilon}/\gamma^2$.
3: $\beta = 0.05, b = 100/\beta^2$.
4: Sample $20/\varepsilon$ points from $\mathcal{P}$. Let E be the set of such points.
5: **for** $t \in \{1, \frac{1}{2}, \frac{1}{4}, \ldots, \frac{1}{n}\}$ **do**
6: Construct set S_t by picking each element of $[n]$ independently with probability t.
7: Let $\mathcal{S} = \{S_t | t \in \{1, \frac{1}{2}, \frac{1}{4}, \ldots, \frac{1}{n}\}\}$.
8: **for** all tuple $(i, S) \in E \times \mathcal{S}$ **do**
9: $\mathsf{ep}_1 = \mathsf{EstProb}(\mathcal{P}, i, S, m)$.
10: $\mathsf{ep}_2 = \mathsf{EstProb}(\mathcal{Q}, i, S, m)$.
11: **if** $|\mathsf{ep}_1 - \mathsf{ep}_2| > 2\gamma$ **then**
12: **return** Reject
13: $\mathsf{et}_1 = \mathsf{EstTail}(\mathcal{P}, i, S, \beta, b, m)$.
14: $\mathsf{et}_2 = \mathsf{EstTail}(\mathcal{Q}, i, S, \beta, b, m)$.
15: **if** $|\mathsf{et}_1 - \mathsf{et}_2| > 2\gamma$ **then**
16: **return** Reject
17: **return** Accept.

Algorithm 2 $\mathsf{EstProb}(\mathcal{D}, i, S, m)$

Input: distribution $\mathcal{D}$ on $[n]$, $i \in [n]$, $S \subseteq [n]$, parameter $m \geq 1$
Output: estimate of $\frac{\mathcal{D}(i)}{\mathcal{D}(i \cup S)}$

1: Sample $j_1, \ldots, j_m \sim \mathsf{COND}_{\mathcal{D}}(i \cup S)$.
2: return $\frac{\sum_{k \in [m]} \mathbb{1}_{j_k = i}}{m}$

The subroutine $\mathsf{EstTail}(\mathcal{D}, i, S, \beta, b, m)$ takes as input a distribution $\mathcal{D}$, $i \in [n]$, $S \subseteq [n]$ and parameters $\beta \in (0,1), b, m \in \mathbb{Z}^+$. It returns, in a straightforward way, $\pm\gamma$ estimate of $\mathsf{TP}(\mathcal{D}, i, S, \beta, b, m)$ with high probability.

Lemma 6. *For any distribution $\mathcal{D}$ on $[n]$, $i \in [n]$, $S \subseteq [n]$, parameters $\beta \in (0,1)$, $b, m \in \mathbb{Z}^+$, the $\mathsf{EstTail}(\mathcal{D}, i, S, \beta, b, m)$ returns $\pm\gamma$ estimate of $\mathsf{TP}(\mathcal{D}, i, S, \beta, b)$ with probability at least $1 - e^{-\gamma^2 m}$.*

Algorithm 3 $\mathsf{EstTail}(\mathcal{D}, i, S, \beta, b, m)$

Input: distribution $\mathcal{D}$ on $[n]$, $i \in [n]$, $S \subseteq [n]$, parameters $\beta > 0$, $b, m \geq 1$

Output: estimate of $\mathsf{TP}(\mathcal{D}, i, S, \beta, b)$

1: Sample $j_1, \ldots, j_m \sim \mathsf{COND}_{\mathcal{D}}(S)$.
2: **for** $k \in [m]$ **do**
3: Sample $y_1, \ldots, y_b \sim \mathsf{COND}_{\mathcal{D}}(\{j_k, i\})$.
4: **if** $\frac{\sum_{\ell \in [b]} \mathbb{1}_{y_\ell = j_k}}{b} \leq \frac{1}{2} + \beta$ **then**
5: $Z_k = 1$.
6: **else**
7: $Z_k = 0$
8: $Z = \frac{\sum_{k \in [m]} Z_k}{m}$.
9: return Z.

The proofs of both Lemma 5 and Lemma 6 are by a standard application of additive Chernoff bound (Lemma 3), and deferred to the supplementary materials.

For brevity, from now on, we use $\mathsf{EstProb}(\mathcal{D}, i, S)$ for $\mathsf{EstProb}(\mathcal{D}, i, S, m)$, $\mathsf{EstTail}(\mathcal{D}, i, S)$ for $\mathsf{EstTail}(\mathcal{D}, i, S, \beta, b, m)$ and $\mathsf{TP}(\mathcal{D}, i, S)$ for $\mathsf{TP}(\mathcal{D}, i, S, \beta, b, m)$.

We now prove the first part of our main theorem, i.e., if $\mathcal{P} = \mathcal{Q}$, then $\mathsf{EquivTester}$ returns Accept with high probability.

Lemma 7. *If $\mathcal{P} = \mathcal{Q}$ then the algorithm returns* Accept *with probability at least $1 - o(1)$.*

Proof. For each $(i, S) \in E \times \mathcal{S}$, let $\mathsf{Bad}_1(\mathcal{P}, i, S)$ be the event that

$$\left| \mathsf{EstProb}(\mathcal{P}, i, S) - \frac{\mathcal{P}(i)}{\mathcal{P}(i \cup S)} \right| \geq \gamma$$

and $\mathsf{Bad}_2(\mathcal{P}, i, S)$ be the event that

$$|\mathsf{EstTail}(\mathcal{P}, i, S) - \mathsf{TP}(\mathcal{P}, i, S)| \geq \gamma.$$

Similarly, we define the events $\mathsf{Bad}_1(\mathcal{Q}, i, S)$ and $\mathsf{Bad}_2(\mathcal{Q}, i, S)$. Now, consider the event $\mathsf{Bad} := \bigcup_{(i,S) \in E \times \mathcal{S}} (\mathsf{Bad}_1(\mathcal{P}, i, S) \cup \mathsf{Bad}_2(\mathcal{P}, i, S) \cup \mathsf{Bad}_1(\mathcal{Q}, i, S) \cup \mathsf{Bad}_2(\mathcal{Q}, i, S))$.

From Lemma 5 (substituting $m = 100(\log\log n)\log\frac{1}{\varepsilon}/\gamma^2$), we have $\Pr[\mathsf{Bad}_1(\mathcal{P}, i, S)] \leq e^{-\gamma^2 m}$ and $\Pr[\mathsf{Bad}_1(\mathcal{Q}, i, S)] \leq e^{-\gamma^2 m}$. Similarly, from Lemma 6, we have $\Pr[\mathsf{Bad}_2(\mathcal{P}, i, S)] \leq e^{-\gamma^2 m}$ and $\Pr[\mathsf{Bad}_2(\mathcal{Q}, i, S)] \leq e^{-\gamma^2 m}$.

Since the total number of (i, S) tuples considered by the algorithm is at most $10 \log n \cdot \varepsilon^{-1}$, by a union bound, we have $\Pr[\mathsf{Bad}] \leq 10 \log n \cdot \varepsilon^{-1} \cdot 4 \cdot e^{-\gamma^2 m} \leq 1/(\log n)^{98}$ (since $e^{-\gamma^2 m} \leq \varepsilon/(\log n)^{100}$). Further, if the event Bad does not happen, then for all $(i, S) \in E \times \mathcal{S}$, we have $|\mathsf{EstTail}(\mathcal{P}, i, S) - \mathsf{EstTail}(\mathcal{Q}, i, S)| \leq 2\gamma$ and $|\mathsf{EstProb}(\mathcal{P}, i, S) - \mathsf{EstProb}(\mathcal{Q}, i, S)| \leq 2\gamma$. Hence, with probability at least $1 - o(1)$, $\mathsf{EquivTester}$ will return Accept. ☐

We now proceed towards showing the second part of our main theorem, if $d_{TV}(\mathcal{P}, \mathcal{Q}) \geq \varepsilon$, then $\mathsf{EquivTester}$ returns Reject with high probability.

Lemma 8. *If $d_{TV}(\mathcal{P}, \mathcal{Q}) \geq \varepsilon$, then* $\mathsf{EquivTester}$ *returns* Reject *with probability at least $2/3$.*

Proof. We start with the notion of a tuple $(i, S) \in E \times \mathcal{S}$ being a *distinguisher* for $\mathcal{P}$ and $\mathcal{Q}$, which will prove to be a sufficient condition for $\mathsf{EquivTester}$ to return Reject with high probability.

Definition 9. *A tuple $(i, S) \in E \times \mathcal{S}$ is called a* distinguisher *for $\mathcal{P}$ and $\mathcal{Q}$ if either of the following two conditions hold true:*

1. $\left| \frac{\mathcal{P}(i)}{\mathcal{P}(i \cup S)} - \frac{\mathcal{Q}(i)}{\mathcal{Q}(i \cup S)} \right| > 4\gamma$,
2. $|\mathsf{TP}(\mathcal{P}, i, S) - \mathsf{TP}(\mathcal{Q}, i, S)| > 4\gamma$.

To complete the proof, we rely on the following three lemmas, whose proofs we will provide later.

Lemma 10. *If $(i, S) \in E \times \mathcal{S}$ is a distinguisher for $\mathcal{P}$ and $\mathcal{Q}$, then* $\mathsf{EquivTester}$ *returns* Reject *with probability at least $1 - 4/(\log n)^{100}$.*

Lemma 11. *Let $c = 1000$. If $d_{TV}(\mathcal{P}, \mathcal{Q}) \geq \varepsilon$, then with probability at least $1 - e^{-6}$, there exists a non-empty set $\mathcal{T} \subseteq E$, such that for all $i^* \in \mathcal{T}$, we have*

1. $\sum_{j : \mathcal{P}(j) \leq \mathcal{P}(i^*)} \mathcal{P}(j) \geq 3\varepsilon/10$,
2. $\mathcal{P}(i^*) \geq (1 + \varepsilon/4)\mathcal{Q}(i^*)$,
3. $\mathcal{P}(i^*) \geq \frac{\varepsilon^3}{c^2 n}$.

Let us now consider the subset $\mathcal{T} \subseteq E$ from the above lemma.

Lemma 12. *For every $i^* \in \mathcal{T}$, with probability at least $4/5$, there exists a $S^* \in \mathcal{S}$, such that (i^*, S^*) is a distinguisher for $\mathcal{P}$ and $\mathcal{Q}$.*

We are now ready to finish the proof of Lemma 8. It now directly follows from Lemma 11, Lemma 12, and Lemma 10, that if $d_{TV}(\mathcal{P}, \mathcal{Q}) \geq \varepsilon$, then $\mathsf{EquivTester}$ does not return Reject with probability at most $e^{-6} + 1/5 + 4/(\log n)^{100} < 1/3$. ☐

Proof of Lemma 10

Proof. Analogous to the proof of Lemma 7, we now define the events $\mathsf{Bad}_1(\mathcal{P}), \mathsf{Bad}_2(\mathcal{P})$, and so on. For any distribution $\mathcal{D} \in \{\mathcal{P}, \mathcal{Q}\}$, let $\mathsf{Bad}_1(\mathcal{D})$ be the event that

$$\left|\mathsf{EstProb}(\mathcal{D}, i, S) - \frac{\mathcal{D}(i)}{\mathcal{D}(i \cup S)}\right| \geq \gamma$$

and $\mathsf{Bad}_2(\mathcal{D})$ be the event that

$$|\mathsf{EstTail}(\mathcal{D}, i, S) - \mathsf{TP}(\mathcal{D}, i, S)| \geq \gamma.$$

Let $\mathsf{Bad} := \cup_{\mathcal{D}\in\{\mathcal{P},\mathcal{Q}\}} \cup_{j\in\{1,2\}} \mathsf{Bad}_j(\mathcal{D})$.

Since $m = 100(\log\log n)\log\frac{1}{\varepsilon}/\gamma^2$, from Lemma 5 and Lemma 6, we have for $\mathcal{D} \in \{\mathcal{P}, \mathcal{Q}\}$, both $\Pr[\mathsf{Bad}_1(\mathcal{D})]$ and $\Pr[\mathsf{Bad}_2(\mathcal{D})]$ are at most $e^{-\gamma^2 m} \leq \varepsilon/(\log n)^{100}$. Thus by a union bound, the event Bad happens with probability at most $4\varepsilon/(\log n)^{100}$.

Since (i, S) is fixed in the context of this lemma, for brevity, we use ep_1 for $\mathsf{EstProb}(\mathcal{P}, i, S)$ and ep_2 for $\mathsf{EstProb}(\mathcal{Q}, i, S)$ for the remaining parts of the proof of this lemma. Similarly, we use et_1 for $\mathsf{EstTail}(\mathcal{P}, i, S)$ and et_2 for $\mathsf{EstTail}(\mathcal{Q}, i, S)$.

From now on, assume Bad does not happen. Since (i, S) is a distinguisher, either the item (1) or the item (2) in the Definition 9 holds. If the item (1) holds, then by the triangle inequality, we have

$$\begin{aligned}&|\mathsf{ep}_1 - \mathsf{ep}_2| \geq \\ &\left|\frac{\mathcal{P}(i)}{\mathcal{P}(i\cup S)} - \frac{\mathcal{Q}(i)}{\mathcal{Q}(i\cup S)}\right| - \left|\mathsf{ep}_1 - \frac{\mathcal{P}(i)}{\mathcal{P}(i\cup S)}\right| \\ &- \left|\mathsf{ep}_2 - \frac{\mathcal{Q}(i)}{\mathcal{Q}(i\cup S)}\right| > 2\gamma\end{aligned}$$

and thus $\mathsf{EquivTester}$ returns Reject.

Now, if the item (2) holds, then again, by the triangle inequality, we have

$$\begin{aligned}&|\mathsf{et}_1 - \mathsf{et}_2| \geq \\ &\quad |\mathsf{TP}(\mathcal{P}, i, S) - \mathsf{TP}(\mathcal{Q}, i, S)| \\ &\quad - |\mathsf{et}_1 - \mathsf{TP}(\mathcal{P}, i, S)| - |\mathsf{et}_2 - \mathsf{TP}(\mathcal{Q}, i, S)| > 2\gamma\end{aligned}$$

and thus $\mathsf{EquivTester}$ returns Reject. ☐

Proof of Lemma 11

Proof. Let $A = \{i \in [n] : \mathcal{P}(i) > \mathcal{Q}(i)\}$. We partition A into $A_1 = \{i \in A : \mathcal{P}(i) < \varepsilon^3/c^2 n\}$, $A_2 = \{i \in A \setminus A_1 : \mathcal{P}(i) < (1+\varepsilon/4)\mathcal{Q}(i)\}$ and $A_3 = A \setminus \{A_1 \cup A_2\}$. Note that any $i \in A_3$, by definition, will satisfy $\mathcal{P}(i) \geq (1+\varepsilon/4)\mathcal{Q}(i)$ and $\mathcal{P}(i) \geq \frac{\varepsilon^3}{c^2 n}$. We now lower bound $\mathcal{P}(A_3)$. Firstly,

$$\begin{aligned}d_{TV}(\mathcal{P}, \mathcal{Q}) &= \sum_{i\in A_1} |\mathcal{P}(i) - \mathcal{Q}(i)| + \sum_{i\in A_2} |\mathcal{P}(i) - \mathcal{Q}(i)| \\ &\quad + \sum_{i\in A_3} |\mathcal{P}(i) - \mathcal{Q}(i)| \\ &\leq n \cdot \frac{\varepsilon^3}{c^2 n} + \sum_{i\in A_2} \mathcal{P}(i)\cdot\frac{\varepsilon}{4} + \sum_{i\in A_3}\mathcal{P}(i) \\ &\leq \frac{\varepsilon^3}{c^2} + \frac{\varepsilon}{4} + \sum_{i\in A_3}\mathcal{P}(i).\end{aligned}$$

Since $d_{TV}(\mathcal{P}, \mathcal{Q}) \geq \varepsilon$, we have $\sum_{i\in A_3}\mathcal{P}(i) \geq 3\varepsilon/5$. Let $A_4 = \{i \in A_3 : \sum_{j\in A_3:\mathcal{P}(j)\leq\mathcal{P}(i)} \mathcal{P}(j) \geq 3\varepsilon/10\}$. Observe that every $i^* \in A_4$ satisfies all the items of this lemma. Now, we lower bound $\mathcal{P}(A_4)$.

Note that $\mathcal{P}(A_3 \setminus A_4) < 3\varepsilon/10$. Therefore, $\mathcal{P}(A_4) \geq 3\varepsilon/5 - 3\varepsilon/10 = 3\varepsilon/10$. Thus, the set $\mathcal{T} := E \cap A_4$ is empty with probability at most $(1 - 3\varepsilon/10)^{20/\varepsilon} < e^{-6}$ (since $(1-x)^r \leq e^{-xr}$). Therefore, with probability at least $1 - e^{-6}$, the set $\mathcal{T}$ is non-empty, and any $i^* \in \mathcal{T}$ satisfies all the items of this lemma. ☐

Proof of Lemma 12

Proof. Consider an arbitrary $i^* \in \mathcal{T}$. Our goal is to show that with high probability, there exists a $S^* \in \mathcal{S}$ such that (i^*, S^*) is a distinguisher. From the definition 9, it follows that if $|\mathcal{P}(i^*) - \mathcal{Q}(i^*)| > 4\gamma$ or $|\mathsf{TP}(\mathcal{P}, i^*, [n]) - \mathsf{TP}(\mathcal{Q}, i^*, [n])| > 4\gamma$, then $(i^*, [n])$ is a distinguisher (note that $[n] \in \mathcal{S}$). Therefore, we need to focus only on the following cases:

$$|\mathcal{P}(i^*) - \mathcal{Q}(i^*)| \leq 4\gamma \tag{1}$$

$$|\mathsf{TP}(\mathcal{P}, i^*, [n]) - \mathsf{TP}(\mathcal{Q}, i^*, [n])| \leq 4\gamma. \tag{2}$$

We now give the value of $t^* \in \{1, 1/2, \ldots, 1/n\}$, in terms of $\mathcal{P}(i^*)$ such that the tuple (i^*, S_{t^*}) will be a distinguisher with high probability, where recall S_{t^*} is a set constructed by picking each element in $[n]$ with probability t^*. For the same, first note that:

$$\mathcal{P}(i^*) - \frac{\mathcal{P}(i^*)}{1+\varepsilon/4} \leq \mathcal{P}(i^*) - \mathcal{Q}(i^*) \leq 4\gamma$$

where the first inequality is by Lemma 11 and the second by the Eq. 1. Immediately, we get:

$$\mathcal{P}(i^*) \leq \frac{32\gamma}{\varepsilon} \leq \frac{32\varepsilon^3}{L}. \tag{3}$$

Let $t' = \frac{c^2\mathcal{P}(i^*)}{\varepsilon^3}$. Since $\mathcal{P}(i^*) \leq \varepsilon^3/c^2 n$, we have $\frac{1}{n} \leq t' \leq 1$. Therefore, there exists $t^* \in \left\{\frac{1}{n}, \frac{2}{n}, \ldots, 1\right\}$ such that $t' \leq t^* < 2t'$, i.e.,

$$\frac{c^2\mathcal{P}(i^*)}{\varepsilon^3} \leq t^* < 2\frac{c^2\mathcal{P}(i^*)}{\varepsilon^3}. \tag{4}$$

Let $S^* = S_{t^*}$, i.e., S^* is the set constructed by picking each element in $[n]$ with probability t^*. Our goal now is to prove, with probability at least $4/5$, (i^*, S^*) is a distinguisher for $\mathcal{P}$ and $\mathcal{Q}$.

Let for any $j \in [n]$, we define $\mathsf{R}(\mathcal{D}, i^*, j)$ as

$$\mathcal{D}(j) \Pr\left[\mathsf{Bin}\left(b, \frac{\mathcal{D}(j)}{\mathcal{D}(j)+\mathcal{D}(i^*)}\right) \le \left(\frac{1}{2}+\beta\right) b\right]$$

and for any subset $S \subseteq [n]$

$$\mathsf{R}(\mathcal{D}, i^*, S) := \sum_{j \in S} \mathsf{R}(\mathcal{D}, i^*, j).$$

Therefore, by the definition of tail probability,

$$\mathsf{TP}(\mathcal{D}, i^*, S) = \frac{\mathsf{R}(\mathcal{D}, i^*, S)}{\mathcal{D}(S)}.$$

We want to show concentration on both $\mathsf{R}(\mathcal{D}, i^*, S^*)$ and $\mathcal{D}(S^*)$, for all $\mathcal{D} \in \{\mathcal{P}, \mathcal{Q}\}$ which then will give us the concentration inequalities for $\mathsf{TP}(\mathcal{D}, i^*, S^*)$.

Claim 13. *Let* Good_1 *be the event that for all* $\mathcal{D} \in \{\mathcal{P}, \mathcal{Q}\}$, *we have*

$$|\mathsf{R}(\mathcal{D}, i^*, S^*) - t^* \mathsf{R}(\mathcal{D}, i^*, [n])| \le \frac{8t^*\varepsilon\sqrt{\varepsilon \mathsf{R}(\mathcal{D}, i^*, [n])}}{c}.$$

Then $\Pr[\mathsf{Good}_1] \ge 9/10$.

Claim 14. *Let* Good_2 *be the event that all the following three conditions are satisfied*

1. $\mathcal{P}(S^*) \ge \frac{t^*\varepsilon}{9}$,
2. $\mathcal{P}(S^*) \le 200t^*$,
3. $\mathcal{Q}(S^*) \le 200t^*$.

We defer the proofs of the above two claims to the supplementary materials. Let $\mathsf{Good} = \mathsf{Good}_1 \wedge \mathsf{Good}_2$. Note that by union bound, $\Pr[\mathsf{Good}] \ge 4/5$.

We are now ready to complete the proof of Lemma 12. Assuming the event Good occurs, we now prove that (i^*, S^*) is a distinguisher. For the sake of contradiction, suppose (i^*, S^*) is not a distinguisher for $\mathcal{P}$ and $\mathcal{Q}$. Then the following claim holds, the proof of which is deferred to the supplementary materials.

Claim 15. *Assuming the event* Good, *and that* (i^*, S^*) *is not a distinguisher for* $\mathcal{P}$ *and* $\mathcal{Q}$, *we have* $\mathcal{Q}(S^*) > \mathcal{P}(S^*)\left(1+\frac{150\varepsilon}{c}\right)^{-1}$.

We now argue that $\left|\frac{\mathcal{P}(i)}{\mathcal{P}(i \cup S)} - \frac{\mathcal{Q}(i)}{\mathcal{Q}(i \cup S)}\right| > 4\gamma$, contradicting that (i^*, S^*) is not a distinguisher.

If $\frac{\mathcal{P}(S^*)}{\mathcal{P}(i^*)} > \frac{\mathcal{Q}(S^*)}{\mathcal{Q}(i^*)}$ then

$$\frac{\mathcal{P}(S^*)}{\mathcal{P}(i^*)} > \frac{\mathcal{Q}(S^*)}{\mathcal{Q}(i^*)} > \frac{\mathcal{P}(S^*)(1+\varepsilon/4)}{\mathcal{P}(i^*)(1+150\varepsilon/c)}, \tag{5}$$

(by Lemma 11 and Claim 15) which is a contradiction. Hence,

$$\frac{\mathcal{P}(S^*)}{\mathcal{P}(i^*)} \le \frac{\mathcal{Q}(S^*)}{\mathcal{Q}(i^*)}. \tag{6}$$

Recall that we would like to argue that $\left|\frac{\mathcal{P}(i^*)}{\mathcal{P}(i^*)+\mathcal{P}(S^*)} - \frac{\mathcal{Q}(i^*)}{\mathcal{Q}(i^*)+\mathcal{Q}(S^*)}\right| > 4\gamma$. Note that

$$\begin{aligned}
&\left|\frac{\mathcal{P}(i^*)}{\mathcal{P}(i^*)+\mathcal{P}(S^*)} - \frac{\mathcal{Q}(i^*)}{\mathcal{Q}(i^*)+\mathcal{Q}(S^*)}\right| \\
&> \frac{\frac{\mathcal{P}(S^*)}{(1+\frac{150\varepsilon}{c})\mathcal{Q}(i^*)} - \frac{\mathcal{P}(S^*)}{\mathcal{P}(i^*)}}{\left(1+\frac{\mathcal{P}(S^*)}{\mathcal{P}(i^*)}\right)\left(1+\frac{\mathcal{Q}(S^*)}{\mathcal{Q}(i^*)}\right)} \quad \text{(Eq. 6, Claim 15)} \\
&= \frac{\frac{\mathcal{P}(S^*)}{(1+\frac{150\varepsilon}{c})} - \frac{\mathcal{Q}(i^*)\mathcal{P}(S^*)}{\mathcal{P}(i^*)}}{\left(1+\frac{\mathcal{P}(S^*)}{\mathcal{P}(i^*)}\right)(\mathcal{Q}(i^*)+\mathcal{Q}(S^*))} \\
&> \frac{\frac{\mathcal{P}(S^*)}{(1+\frac{150\varepsilon}{c})} - \frac{\mathcal{P}(S^*)}{1+\varepsilon/4}}{\left(1+\frac{\mathcal{P}(S^*)}{\mathcal{P}(i^*)}\right)\left(\frac{\mathcal{P}(i^*)}{1+\varepsilon/4}+\mathcal{Q}(S^*)\right)} \quad \text{(Lemma 11)} \\
&> \frac{\mathcal{P}(S^*)\varepsilon}{20}\left(1+\frac{\mathcal{P}(S^*)}{\mathcal{P}(i^*)}\right)^{-1}\left(\frac{\mathcal{P}(i^*)}{1+\varepsilon/4}+\mathcal{Q}(S^*)\right)^{-1} \\
&= \frac{\varepsilon}{20}\left(1+\frac{\mathcal{P}(i^*)}{\mathcal{P}(S^*)}\right)^{-1}\left(\frac{1}{1+\varepsilon/4}+\frac{\mathcal{Q}(S^*)}{\mathcal{P}(i^*)}\right)^{-1} \\
&> \frac{\varepsilon}{20}\left(1+\frac{9\mathcal{P}(i^*)}{t^*\varepsilon}\right)^{-1}\left(1+\frac{\mathcal{Q}(S^*)}{\mathcal{P}(i^*)}\right)^{-1} \quad \text{(Claim 14)} \\
&> \frac{\varepsilon}{20}\left(1+\frac{9t^*\varepsilon^3}{c^2t^*\varepsilon}\right)^{-1}\left(1+\frac{\mathcal{Q}(S^*)}{\mathcal{P}(i^*)}\right)^{-1} \quad \text{(Eq. 4)} \\
&> \frac{\varepsilon}{40}\left(1+\frac{\mathcal{Q}(S^*)}{\mathcal{P}(i^*)}\right)^{-1} \\
&> \frac{\varepsilon}{40}\left(1+\frac{200t^*}{\mathcal{P}(i^*)}\right)^{-1} \quad \text{(Claim 14)} \\
&> \frac{\varepsilon}{40}\left(1+\frac{400t^*c^2}{t^*\varepsilon^3}\right)^{-1} \quad \text{(Eq. 4)} \\
&> \frac{\varepsilon}{40\left(\frac{401c^2}{\varepsilon^3}\right)} > \frac{\varepsilon^4}{16040c^2} > 4\frac{\varepsilon^4}{L} = 4\gamma.
\end{aligned}$$

□

5 Conclusion

We considered the problem of equivalence testing of two distributions (over $[n]$) in the conditional sampling model. We presented a simple algorithm with sample complexity $\tilde{O}(\log n)$. While our algorithm is not fully non-adaptive, it is only one-round adaptive. This shows that even a limited amount of adaptiveness can help to significantly reduce the sample/query complexity. Our algorithm can also be modified slightly to obtain a fully adaptive algorithm with sample complexity $\tilde{O}(\log \log n)$, matching the best bound in this setting.

One limitation of our algorithm is the presence of large constants and a worsened dependency on the parameter ε compared to the previous algorithm by [Kamath and Tzamos, 2019]. Investigating methods to reduce this dependency on ε while maintaining the $\widetilde{O}(\log n)$ dependency with respect to the parameter n poses an intriguing open direction of research.

Acknowledgements

We thank the anonymous reviewers for their useful comments. D. Chakraborty is supported in part by an MoE AcRF Tier 2 grant (MOE-T2EP20221-0009), an MoE AcRF Tier 1 grant (T1 251RES2303), and a Google South & South-East Asia Research Award. K. S. Meel is 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].

References

[Acharya et al., 2018] Acharya, J., Canonne, C. L., and Kamath, G. (2018). A chasm between identity and equivalence testing with conditional queries. *Theory of Computing*, 14(19):1–46.

[Agarwal et al., 2017] Agarwal, A., Agarwal, S., Assadi, S., and Khanna, S. (2017). Learning with limited rounds of adaptivity: Coin tossing, multi-armed bandits, and ranking from pairwise comparisons. In *Conference on Learning Theory*, pages 39–75. PMLR.

[Balkanski and Singer, 2018] Balkanski, E. and Singer, Y. (2018). The adaptive complexity of maximizing a submodular function. In *Proceedings of the 50th annual ACM SIGACT symposium on theory of computing*, pages 1138–1151.

[Batu et al., 2013] Batu, T., Fortnow, L., Rubinfeld, R., Smith, W. D., and White, P. (2013). Testing closeness of discrete distributions. *Journal of the ACM (JACM)*, 60(1):1–25.

[Bhattacharyya and Chakraborty, 2018] Bhattacharyya, R. and Chakraborty, S. (2018). Property testing of joint distributions using conditional samples. *ACM Transactions on Computation Theory (TOCT)*, 10(4):1–20.

[Canonne et al., 2014] Canonne, C., Ron, D., and Servedio, R. A. (2014). Testing equivalence between distributions using conditional samples. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 1174–1192. SIAM.

[Canonne et al., 2021] Canonne, C. L., Chen, X., Kamath, G., Levi, A., and Waingarten, E. (2021). Random restrictions of high dimensional distributions and uniformity testing with subcube conditioning. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*. SIAM.

[Canonne and Gur, 2018] Canonne, C. L. and Gur, T. (2018). An adaptivity hierarchy theorem for property testing. *computational complexity*, 27:671–716.

[Chakraborty et al., 2024] Chakraborty, D., Chakraborty, S., and Kumar, G. (2024). Tight lower bound on equivalence testing in conditional sampling model. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 4371–4394. SIAM.

[Chakraborty et al., 2023] Chakraborty, D., Kumar, G., and Meel, K. S. (2023). Support size estimation: The power of conditioning. In *Proceedings of the 48th International Symposium on Mathematical Foundations of Computer Science, MFCS 2023, August 28 - September 01, 2023*.

[Chakraborty et al., 2013] Chakraborty, S., Fischer, E., Goldhirsh, Y., and Matsliah, A. (2013). On the power of conditional samples in distribution testing. In *Proceedings of the 4th conference on Innovations in Theoretical Computer Science*, pages 561–580.

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

[Chan et al., 2014] Chan, S.-O., Diakonikolas, I., Valiant, P., and Valiant, G. (2014). Optimal algorithms for testing closeness of discrete distributions. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 1193–1203. SIAM.

[Chekuri and Quanrud, 2019] Chekuri, C. and Quanrud, K. (2019). Parallelizing greedy for submodular set function maximization in matroids and beyond. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 78–89.

[Chen et al., 2021] Chen, X., Jayaram, R., Levi, A., and Waingarten, E. (2021). Learning and testing junta distributions with sub cube conditioning. In *Conference on Learning Theory*, pages 1060–1113. PMLR.

[Damaschke et al., 2013] Damaschke, P., Muhammad, A. S., and Triesch, E. (2013). Two new perspectives on multi-stage group testing. *Algorithmica*, 67:324–354.

[Du et al., 2000] Du, D., Hwang, F. K., and Hwang, F. (2000). *Combinatorial group testing and its applications*, volume 12. World Scientific.

[Eberhardt et al., 2020] Eberhardt, J. N., Breuckmann, N. P., and Eberhardt, C. S. (2020). Multi-stage group testing improves efficiency of large-scale covid-19 screening. *Journal of Clinical Virology*, 128:104382.

[Falahatgar et al., 2015] Falahatgar, M., Jafarpour, A., Orlitsky, A., Pichapati, V., and Suresh, A. T. (2015). Faster algorithms for testing under conditional sampling. In *Conference on Learning Theory*, pages 607–636. PMLR.

[Goldreich et al., 1998] Goldreich, O., Goldwasser, S., and Ron, D. (1998). Property testing and its connection to learning and approximation. *Journal of the ACM (JACM)*, 45(4):653–750.

[Goldreich and Ron, 2011] Goldreich, O. and Ron, D. (2011). On testing expansion in bounded-degree graphs. *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation: In Collaboration with Lidor Avigad, Mihir Bellare, Zvika Brakerski, Shafi Goldwasser, Shai Halevi, Tali Kaufman, Leonid Levin, Noam Nisan, Dana Ron, Madhu Sudan, Luca Trevisan, Salil Vadhan, Avi Wigderson, David Zuckerman*, pages 68–75.

[Golia et al., 2022] Golia, P., Juba, B., and Meel, K. S. (2022). A scalable Shannon entropy estimator. In *Computer Aided Verification: 34th International Conference, CAV 2022, Haifa, Israel, August 7–10, 2022, Proceedings, Part I*, pages 363–384. Springer.

[Kamath and Price, 2019] Kamath, A. and Price, E. (2019). Adaptive sparse recovery with limited adaptivity. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2729–2744. SIAM.

[Kamath and Tzamos, 2019] Kamath, G. and Tzamos, C. (2019). 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.

[Meel et al., 2020] Meel, K. S., Pote, Y. P., and Chakraborty, S. (2020). On testing of samplers. *Advances in Neural Information Processing Systems*, 33:5753–5763.

[Nakos et al., 2018] Nakos, V., Shi, X., Woodruff, D. P., and Zhang, H. (2018). Improved algorithms for adaptive compressed sensing. In *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

[Narayanan, 2021] Narayanan, S. (2021). On tolerant distribution testing in the conditional sampling model. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 357–373. SIAM.

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

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [**Yes**/No/Not Applicable]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [**Yes**/No/Not Applicable]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes/No/**Not Applicable**]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [**Yes**/No/Not Applicable]
   (b) Complete proofs of all theoretical results. [**Yes**/No/Not Applicable]
   (c) Clear explanations of any assumptions. [**Yes**/No/Not Applicable]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes/No/**Not Applicable**]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes/No/**Not Applicable**]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes/No/**Not Applicable**]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes/No/**Not Applicable**]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes/No/**Not Applicable**]
   (b) The license information of the assets, if applicable. [Yes/No/**Not Applicable**]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes/No/**Not Applicable**]
   (d) Information about consent from data providers/curators. [Yes/No/**Not Applicable**]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Yes/No/**Not Applicable**]
5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Yes/No/**Not Applicable**]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Yes/No/**Not Applicable**]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Yes/No/**Not Applicable**]

Supplementary Materials

6 MISSING PROOFS

Proof of Lemma 5. Note that for any $k \in [m]$, $\mathbb{1}_{j_k=i}$ is a Bernoulli random variable such that $\Pr[\mathbb{1}_{j_k=i} = 1] = \Pr[j_k = i] = \frac{\mathcal{D}(i)}{\mathcal{D}(i \cup S)}$. Also, note that $\mathbb{1}_{j_k=i}$ are iid for all $k \in [m]$. By additive Chernoff bound (Lemma 3), the value $\mathsf{ep} = \mathsf{EstProb}(\mathcal{D}, i, S, m)$ returned by the estimator satisfies:

$$\Pr\left[\left|\mathsf{ep} - \frac{\mathcal{D}(i)}{\mathcal{D}(i \cup S)}\right| > \gamma\right] \leq e^{-\gamma^2 m}.$$

□

Proof of Lemma 6. For any $k \in [m]$, we have

$$\begin{aligned}
&\Pr[Z_k = 1] \\
&= \sum_{j \in S} \Pr[j_k = j] \Pr\left[\sum_{\ell \in [b]} \mathbb{1}_{y_\ell = j} \leq (1/2 + \beta)\, b\right] \\
&= \sum_{j \in S} \frac{\mathcal{D}(j)}{\mathcal{D}(S)} \Pr\left[\mathsf{Bin}\left(b, \frac{\mathcal{D}(j)}{\mathcal{D}(i) + \mathcal{D}(j)}\right) \leq (1/2 + \beta) b\right] \\
&= \mathsf{TP}(\mathcal{D}, i, S, \beta, b).
\end{aligned}$$

Therefore, by additive Chernoff bound (Lemma 3), the value $\mathsf{et} = \mathsf{EstTail}\,(\mathcal{D}, i, S, \beta, b, m)$ returned by the algorithm satisfies

$$\Pr[|\mathsf{et} - \mathsf{TP}\,(\mathcal{D}, i, S, \beta, b)| > \gamma] \leq e^{-\gamma^2 m}.$$

□

To prove Claims 13, 14 and 15, we will use the following concentration inequality that directly follows from Bernstein's concentration inequality.

Lemma 16. *[Falahatgar et al., 2015] Consider a set G and a function $r : G \to \mathbb{R}_{\geq 0}$ such that $\max_{j \in G} r(j) \leq r_{max}$. Consider set S formed by selecting each element from G independently and uniformly randomly with probability r_0, then $\mathbb{E}[r(S)] = r_0 r(G)$ and with probability at least $1 - 2\lambda$,*

$$|r(S) - \mathbb{E}[r(S)]| \leq \sqrt{2 r_0 r_{max} r(G) \log \frac{1}{\lambda}} + r_{max} \log \frac{1}{\lambda}.$$

We now first state lower bounds for $\mathsf{R}(\mathcal{P}, i^*, [n])$ and $\mathsf{R}(\mathcal{Q}, i^*, [n])$ in the following claim.

Claim 17.

$$\mathsf{R}(\mathcal{P}, i^*, [n]) \geq \frac{\varepsilon}{9} \tag{7}$$

$$\mathsf{R}(\mathcal{Q}, i^*, [n]) \geq \frac{\varepsilon}{10} \tag{8}$$

Proof. Note that if $\mathcal{P}(j) \leq \mathcal{P}(i^*)$ then by additive Chernoff bound, $\Pr\left[\mathsf{Bin}\left(B, \frac{\mathcal{P}(j)}{\mathcal{P}(j)+\mathcal{P}(i^*)}\right) \leq (\frac{1}{2}+\beta)B\right] \geq 1 - \frac{1}{e^{\beta^2 B}} \geq 1 - \frac{1}{e^{100}}$. Hence, we have

$$\begin{aligned}\mathsf{R}(\mathcal{P}, i^*, [n]) &= \sum_{j\in[n]} \mathcal{P}(j) \Pr\left[\mathsf{Bin}\left(b, \frac{\mathcal{P}(j)}{\mathcal{P}(j)+\mathcal{P}(i^*)}\right) \leq (\frac{1}{2}+\beta)b\right] \\ &\geq (1 - \frac{1}{e^{100}}) \sum_{j:\mathcal{P}(j)\leq\mathcal{P}(i^*)} \mathcal{P}(j) \geq \frac{\varepsilon}{9}\end{aligned}$$

where the last inequality is because of item 1 of Lemma 11. Now we have,

$$\begin{aligned}\mathsf{R}(\mathcal{Q}, i^*, [n]) &= \mathsf{TP}(\mathcal{Q}, i^*, [n]) \\ &\geq (\mathsf{TP}(\mathcal{P}, i^*, [n]) - 4\gamma)(\text{ from inequality (2) }) \\ &= (\mathsf{R}(\mathcal{P}, i^*, [n]) - 4\gamma) \\ &\geq \varepsilon/10 (\text{as } \gamma = \varepsilon^4/L\ )\end{aligned}$$

□

Now we prove the Claims 13, 14 and 15.

Proof of Claim 13

Now we provide the proof of our Claim 13. Consider an indicator random variable $\mathbb{1}_{j\in S^*}$ for each $j \in [n]$ that takes the value 1 if j is included in S^* and 0 otherwise. We have

$$\mathbb{E}[\mathsf{R}(\mathcal{D}, i^*, S)] = \sum_{j\in[n]} \mathbb{E}[\mathbb{1}_{j\in S^*}]\mathsf{R}(\mathcal{D}, i^*, j) = t^*\mathsf{R}(\mathcal{D}, i^*, [n])$$

Note that if $\mathcal{D}(j) \geq \frac{3}{2}\mathcal{D}(i^*)$ then by additive Chernoff bound, we have $\mathsf{Bin}(b, \frac{\mathcal{D}(j)}{\mathcal{D}(j)+\mathcal{D}(i^*)}) \leq \frac{1}{2} + \beta$ with probability at most $\frac{1}{e^{(3/5-1/2-\beta)^2 b}} = 1/e^{100}$. Therefore, $\max_j \mathsf{R}(\mathcal{D}, i^*, j) \leq 2\mathcal{D}(i^*)$.

To apply Lemma 16, we have set $G = [n]$, $r_j = \mathsf{R}(\mathcal{D}, i^*, j)$ for all $j \in [n]$, $r_0 = t^*$ and $r_{max} = 2\mathcal{D}(i^*)$ and $\lambda = \frac{1}{c}$. Therefore, with probability at least $1 - 1/\lambda = 1 - 1/c$,

$$\begin{aligned}|\mathsf{R}(\mathcal{D}, i^*, S^*) - t^*\mathsf{R}(\mathcal{D}, i^*, [n])| &\leq \sqrt{2t^*(2\mathcal{D}(i^*))\mathsf{R}(\mathcal{D}, i^*, [n])(\log c)} + 2\mathcal{D}(i^*)(\log c) \\ &\leq \frac{8t^*\varepsilon\sqrt{\varepsilon \mathsf{R}(\mathcal{D}, i^*, [n])}}{c}\end{aligned}$$

where the last inequality is clear from the fact that $\mathcal{Q}(i^*) < \mathcal{P}(i^*)$ (Lemma 11) , $\mathcal{P}(i^*) \leq t^*\varepsilon^3/c^2$ (inequality (4)), $\mathcal{P}(i^*) \leq 32\varepsilon^3/L$ (inequality (3)) and $c = 1000, L = 10^{15}$.

Proof of Claim 14

Consider the set $G = \{j \in [n] : \mathcal{P}(j) \leq \mathcal{P}(i^*)\}$. Further, let $S^*_G = S^* \cap G$. Obviously, $\mathcal{P}(S^*) \geq \mathcal{P}(S^*_G)$. Further, $\mathbb{E}[\mathcal{P}(S^*_G)] = t^* \cdot (\sum_{j:\mathcal{P}(j)\leq\mathcal{P}(i^*)} \mathcal{P}(j)) \geq 3t^*\varepsilon/10$ (from Lemma 11).

Applying Lemma 16 to the set G with $r_0 = t^*, r_{max} = \mathcal{P}(i^*), r(G) = \mathcal{P}(G) \leq 1$ and $\lambda = \frac{1}{c}$, we have $\Pr[\mathcal{P}(S_G) < 3t^*\varepsilon/10 - 4t^*\varepsilon\sqrt{\varepsilon}/c] < \frac{1}{c}$. Note that $3t^*\varepsilon/10 - 4t^*\varepsilon\sqrt{\varepsilon}/c > \varepsilon/9$. Therefore,

$$\Pr[\mathcal{P}(S^*) < \frac{t\varepsilon}{9}] < \Pr[\mathcal{P}(S^*_G) < \frac{t\varepsilon}{9}] < \frac{1}{c}$$

Note that $\mathbb{E}[\mathcal{P}(S^*)] = \mathbb{E}[\mathcal{Q}(S^*)] = t^*$. Therefore, by Marhov's inequality, we have $\Pr[\mathcal{P}(S^*) > 200t^*] < 1/200$ and $\Pr[\mathcal{Q}(S^*) > 200t^*] < 1/200$. By union bound, our claim holds.

Proof of Claim 15

First, we show the following claim, which is a corollary of Claim 13.

Claim 18. *Assuming the event* Good, *we have*

$$|\mathsf{R}(\mathcal{P}, i^*, S^*) - \mathsf{R}(\mathcal{Q}, i^*, S^*)| \leq \frac{20t^*\varepsilon\sqrt{\varepsilon \min\{\mathsf{R}(\mathcal{P}, i^*, [n]), \mathsf{R}(\mathcal{Q}, i^*, [n])\}}}{c}. \tag{9}$$

Proof. By triangle inequality, we have

$$\begin{aligned}|\mathsf{R}(\mathcal{P}, i^*, S^*) - \mathsf{R}(\mathcal{Q}, i^*, S^*)| &\leq |\mathsf{R}(\mathcal{P}, i^*, S^*) - t^*\mathsf{R}(\mathcal{P}, i^*, [n])| + |\mathsf{R}(\mathcal{Q}, i^*, S^*) - t^*\mathsf{R}(\mathcal{Q}, i^*, [n])| + \\ &\quad t^*|\mathsf{R}(\mathcal{P}, i^*, [n]) - \mathsf{R}(\mathcal{Q}, i^*, [n])| \\ &\leq \frac{8t^*\varepsilon\sqrt{\varepsilon\mathsf{R}(\mathcal{P}, i^*, [n])}}{c} + \frac{8t^*\varepsilon\sqrt{\varepsilon\mathsf{R}(\mathcal{Q}, i^*, [n])}}{c} + 4t^*\gamma.\end{aligned}$$

The last inequality implies from Claim 13, the inequality (2) and that $\mathsf{TP}(\mathcal{P}, i^*, [n]) = \mathsf{R}(\mathcal{P}, i^*, [n])$. Further, we have

$$\begin{aligned}\left|\frac{8t^*\varepsilon\sqrt{\varepsilon\mathsf{R}(\mathcal{P}, i^*, [n])}}{c} - \frac{8t^*\varepsilon\sqrt{\varepsilon\mathsf{R}(\mathcal{Q}, i^*, [n])}}{c}\right| &= \frac{8t^*\varepsilon\sqrt{\varepsilon}}{c}\left|\sqrt{\mathsf{R}(\mathcal{P}, i^*, [n])} - \sqrt{\mathsf{R}(\mathcal{Q}, i^*, [n])}\right| \\ &\leq (8t^*\varepsilon^{3/2}/c)\frac{|\mathsf{R}(\mathcal{P}, i^*, [n]) - \mathsf{R}(\mathcal{Q}, i^*, [n])|}{\sqrt{\mathsf{R}(\mathcal{P}, i^*, [n])} + \sqrt{\mathsf{R}(\mathcal{Q}, i^*, [n])}} \\ &\leq (8t^*\varepsilon^{3/2}/c)\frac{12\gamma}{2\sqrt{\varepsilon}} \\ &\leq 50t^*\varepsilon\gamma/c.\end{aligned}$$

The second last inequality is because of inequalities (7) and (8).

Therefore, we have

$$\begin{aligned}&|\mathsf{R}(\mathcal{P}, i^*, S^*) - \mathsf{R}(\mathcal{Q}, i^*, S^*)| \\ &\leq \frac{8t^*\varepsilon\sqrt{\varepsilon\mathsf{R}(\mathcal{P}, i^*, [n])}}{c} + \frac{8t^*\varepsilon\sqrt{\varepsilon\mathsf{R}(\mathcal{Q}, i^*, [n])}}{c} + 4t^*\gamma \\ &\leq \frac{16t^*\varepsilon\sqrt{\varepsilon \min\{\mathsf{R}(\mathcal{P}, i^*, [n]), \mathsf{R}(\mathcal{Q}, i^*, [n])\}}}{c} + \left|\frac{8t^*\varepsilon\sqrt{\varepsilon\mathsf{R}(\mathcal{P}, i^*, [n])}}{c} - \frac{8t^*\varepsilon\sqrt{\varepsilon\mathsf{R}(\mathcal{Q}, i^*, [n])}}{c}\right| + 4t^*\gamma \\ &\leq \frac{16t^*\varepsilon\sqrt{\varepsilon \min\{\mathsf{R}(\mathcal{P}, i^*, [n]), \mathsf{R}(\mathcal{Q}, i^*, [n])\}}}{c} + (8t^*\varepsilon^{3/2}/c)\frac{12\gamma}{2\sqrt{\varepsilon}} + 4t^*\gamma \\ &\leq \frac{20t^*\varepsilon\sqrt{\varepsilon \min\{\mathsf{R}(\mathcal{P}, i^*, [n]), \mathsf{R}(\mathcal{Q}, i^*, [n])\}}}{c}.\end{aligned}$$

The last inequality holds because $\frac{16t^*\varepsilon\sqrt{\varepsilon\mathsf{R}(\mathcal{P}, i^*, [n])}}{c}$ is at least $16t^*\varepsilon\sqrt{\varepsilon\varepsilon/10}/c \geq t^*\varepsilon^2/c$ whereas $50t^*\varepsilon\gamma/c + 4t^*\gamma \leq 100t^*\gamma = 100t^*\varepsilon^4/L$.

□

Now we proceed with showing the proof of Claim 15. Recall that $\mathsf{TP}(\mathcal{D}, i^*, S^*) = \mathsf{R}(\mathcal{D}, i^*, S^*)/\mathcal{D}(S^*)$. We consider two cases.

1. $\frac{\mathsf{R}(\mathcal{P}, i^*, S^*)}{\mathcal{P}(S^*)} < \frac{\mathsf{R}(\mathcal{Q}, i^*, S^*)}{\mathcal{Q}(S^*)}$.

   By assumption that (i^*, S^*) is not a distinguisher, we have

$$4\gamma \geq \frac{\mathsf{R}(\mathcal{Q}, i^*, S^*)}{\mathcal{Q}(S^*)} - \frac{\mathsf{R}(\mathcal{P}, i^*, S^*)}{\mathcal{P}(S^*)}$$

$$\geq \mathsf{R}(\mathcal{P}, i^*, S^*)(\frac{1}{\mathcal{Q}(S^*)} - \frac{1}{\mathcal{P}(S^*)}) - \frac{20t^*\varepsilon\sqrt{\varepsilon \mathsf{R}(\mathcal{P}, i^*, [n])}}{c \cdot \mathcal{Q}(S^*)} (\text{from}(9))$$

This implies that

$$\mathsf{R}(\mathcal{P}, i^*, S^*)(\frac{1}{\mathcal{Q}(S^*)} - \frac{1}{\mathcal{P}(S^*)}) \leq 4\gamma + \frac{20t^*\varepsilon\sqrt{\varepsilon \mathsf{R}(\mathcal{P}, i^*, [n])}}{c \cdot \mathcal{Q}(S^*)}$$
$$\leq \frac{25t^*\varepsilon\sqrt{\varepsilon \mathsf{R}(\mathcal{P}, i^*, [n] \setminus \{i^*\})}}{c \cdot \mathcal{Q}(S^*)}$$

The last inequality follows because $4\gamma = 4\varepsilon^4/L$ is small compared to $\frac{20t^*\varepsilon\sqrt{\varepsilon \mathsf{R}(\mathcal{P}, i^*, [n])}}{c \cdot \mathcal{Q}(S^*)} \geq \frac{20t^*\varepsilon\sqrt{\varepsilon \cdot \varepsilon/9}}{c200t^*} \geq \varepsilon^2/200c$. Here, we used Claim 14 and inequality (7). Now,

$$\mathsf{R}(\mathcal{P}, i^*, S^*)(\frac{1}{\mathcal{Q}(S^*)} - \frac{1}{\mathcal{P}(S^*)}) \leq \frac{25t^*\varepsilon\sqrt{\varepsilon \mathsf{R}(\mathcal{P}, i^*, [n])}}{c \cdot \mathcal{Q}(S^*)}$$
$$\implies \mathcal{P}(S^*) - \mathcal{Q}(S^*) < \frac{25t\varepsilon\sqrt{\varepsilon \mathsf{R}(\mathcal{P}, i^*, [n])}\mathcal{P}(S^*)}{c \cdot \mathsf{R}(\mathcal{P}, i^*, S^*)}$$
$$\implies \mathcal{P}(S^*) - \mathcal{Q}(S^*) < \frac{30\varepsilon\sqrt{\frac{\varepsilon}{\mathsf{R}(\mathcal{P}, i^*, [n])}}\mathcal{P}(S^*)}{c}$$

The last inequality comes from fact that $\mathsf{R}(\mathcal{P}, i^*, S^*) \geq t^*\mathsf{R}(\mathcal{P}, i^*, [n])(1 - 8/c)$(from Claim 13). Finally, since $\mathsf{R}(\mathcal{P}, i^*, [n]) \geq \varepsilon/9$(from (7)), we have $\mathcal{P}(S^*) - \mathcal{Q}(S^*) < 90\varepsilon\mathcal{P}(S^*)/c$ and hence the claim follows.

2. $\frac{\mathsf{R}(\mathcal{P}, i^*, S^*)}{\mathcal{P}(S^*)} > \frac{\mathsf{R}(\mathcal{Q}, i^*, S^*)}{\mathcal{Q}(S^*)}$.

   From (9), we have

$$0 < \frac{\mathsf{R}(\mathcal{P}, i^*, S^*)}{\mathcal{P}(S^*)} - \frac{\mathsf{R}(\mathcal{Q}, i^*, S^*)}{\mathcal{Q}(S^*)} < \frac{\mathsf{R}(\mathcal{Q}, i^*, S^*)}{\mathcal{P}(S^*)} - \frac{\mathsf{R}(\mathcal{Q}, i^*, S^*)}{\mathcal{Q}(S^*)} + \frac{20t^*\varepsilon\sqrt{\varepsilon \mathsf{R}(\mathcal{Q}, i^*, [n])}}{c}$$

   This implies

$$\mathsf{R}(\mathcal{Q}, i^*, S^*)(\frac{\mathcal{P}(S^*) - \mathcal{Q}(S^*)}{\mathcal{P}(S^*)\mathcal{Q}(S^*)}) < \frac{20t^*\varepsilon\sqrt{\varepsilon \mathsf{R}(\mathcal{Q}, i^*, [n])}}{c \cdot \mathcal{P}(S^*)}.$$

   Hence, $(\frac{\mathcal{P}(S^*) - \mathcal{Q}(S^*)}{\mathcal{Q}(S^*)}) < \frac{20t^*\varepsilon\sqrt{\varepsilon \mathsf{R}(\mathcal{Q}, i^*, [n])}}{c \cdot \mathsf{R}(\mathcal{Q}, i^*, S^*)}$ which is at most $\frac{25t^*\varepsilon\sqrt{\varepsilon \mathsf{R}(\mathcal{Q}, i^*, [n])}}{ct^*\mathsf{R}(\mathcal{Q}, i^*, [n])}$ (from Claim 13)). Finally, $\frac{25t^*\varepsilon\sqrt{\varepsilon \mathsf{R}(\mathcal{Q}, i^*, [n])}}{ct^*\mathsf{R}(\mathcal{Q}, i^*, [n])} = \frac{25\varepsilon\sqrt{\frac{\varepsilon}{\mathsf{R}(\mathcal{Q}, i^*, [n])}}}{c} \leq \frac{90\varepsilon}{c}$ (from inequality (8)). Thus, we have $(\frac{\mathcal{P}(S^*) - \mathcal{Q}(S^*)}{\mathcal{Q}(S^*)}) \leq \frac{90\varepsilon}{c}$ and this directly implies the claim.

7 An $\tilde{O}(\log \log n)$-query fully adaptive algorithm

Our algorithm can also be modified slightly to obtain a fully adaptive algorithm with sample complexity $\tilde{O}(\log \log n)$. This matches the best-known bound in this setting by [Falahatgar et al., 2015]. In the original formulation, our algorithm sequentially examines all $\log n$ possible values of t to find a particular value, t^*. At t^*, one of our subroutines—either EstProb or EstTail—will Reject if the input distributions significantly differ. Employing a binary search for t^* reduces the number of queries to $\tilde{O}(\log \log n)$. However, this process requires adaptivity at each iteration.