Near-optimal algorithms for private estimation and sequential testing of collision probability

Róbert Busa-Fekete **Umar Syed**
Google Research

Abstract

We present new algorithms for estimating and testing *collision probability*, a fundamental measure of the spread of a discrete distribution that is widely used in many scientific fields. We describe an algorithm that satisfies (α, β)-local differential privacy and estimates collision probability with error at most ε using $\tilde{O}\left(\frac{\log(1/\beta)}{\alpha^2\varepsilon^2}\right)$ samples for $\alpha \leq 1$, which improves over previous work by a factor of $\frac{1}{\alpha^2}$. We also present a sequential testing algorithm for collision probability, which can distinguish between collision probability values that are separated by ε using $\tilde{O}(\frac{1}{\varepsilon^2})$ samples, even when ε is unknown. Our algorithms have nearly the optimal sample complexity, and in experiments we show that they require significantly fewer samples than previous methods.

1 INTRODUCTION

A key property of a discrete distribution is how widely its probability mass is dispersed over its support. One of the most common measures of this dispersal is *collision probability*. Let $\mathbf{p} = (p_1, \ldots, p_k)$ be a discrete distribution. The collision probability of $\mathbf{p}$ is defined $C(\mathbf{p}) = \sum_{i=1}^{k} p_i^2$.

Collision probability takes its name from the following observation. If X and X' are independent random variables with distribution $\mathbf{p}$ then $C(\mathbf{p}) = \Pr[X = X']$, the probability that the values of X and X' coincide. If a distribution is highly concentrated then its collision probability will be close to 1, while the collision probability of the uniform distribution is $1/k$.

Proceedings of the 28[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

Collision probability has played an important role in many scientific fields, although each time it is rediscovered it is typically given a different name. In ecology, collision probability is called the *Simpson index* and serves as a metric for species diversity (Simpson, 1949; Leinster, 2021). In economics, collision probability is known as the *Herfindahl–Hirschman index*, which quantifies market competition among firms (Herfindahl, 1997), and also the *Gini diversity index*, a measure of income and wealth inequality (Gini, 1912). Collision probability is also known as the *second frequency moment*, and is used in database optimization engines to estimate self join size (Cormode and Garofalakis, 2016). In statistical mechanics, collision probability is equivalent to *Tsallis entropy of second order*, which is closely related to Boltzmann–Gibbs entropy (Tsallis, 1988). The negative logarithm of collision probability is *Rényi entropy of second order*, which has many applications, including assessing the quality of random number generators (Skorski, 2017) and determining the number of reads needed to reconstruct a DNA sequence (Motahari et al., 2013). Collision probability has also been used by political scientists to determine the effective size of political parties (Laakso and Taagepera, 1979).

Collision probability is *not* equivalent to Shannon entropy, the central concept in information theory and another common measure of the spread of a distribution. However, collision probability has a much more intuitive interpretation, and is also easier to estimate. Specifically, estimating the Shannon entropy of a distribution with support size k requires $\Omega\left(\frac{k}{\log k}\right)$ samples (Valiant and Valiant, 2011), while the sample complexity of estimating collision probability is independent of k. Additionally, the negative logarithm of the collision probability of a distribution is a lower bound on its Shannon entropy, and this lower bound becomes an equality for the uniform distribution.

1.1 Our contributions

We present novel algorithms for estimating and testing the collision probability of a distribution.

Private estimation: We give an algorithm for estimating collision probability that satisfies (α, β)-*local differential privacy*.[1] As in previous work, our algorithm is *non-interactive*, which means that there is only a single round of communication between users and a central server, and *communication-efficient*, in the sense that each user sends $O(1)$ bits to the server (in fact, just 1 bit). If $\alpha \leq 1$ then our algorithm needs $\tilde{O}\left(\frac{\log(1/\beta)}{\alpha^2\varepsilon^2}\right)$ samples to output an estimate that has ε additive error, which nearly matches the optimal sample complexity and improves on previous work by an $O\left(\frac{1}{\alpha^2}\right)$ factor (Bravo-Hermsdorff et al., 2022). We also present experiments showing that our mechanism has significantly lower sample complexity in practice.

Sequential testing: We give an algorithm for determining whether collision probability is equal to a given value c_0 or differs from c_0 by at least $\varepsilon > 0$, assuming that one of those conditions holds. Our algorithm needs $\tilde{O}(\frac{1}{\varepsilon^2})$ samples to make a correct determination, which nearly matches the optimal sample complexity. Importantly, ε is *not* known to the algorithm. In other words, the algorithm automatically adapts to easy cases by drawing fewer samples. In experiments, we show that our algorithm uses significantly fewer samples than a worst-case optimal sequential testing algorithm due to (Daskalakis and Kawase, 2017), and also outperforms algorithms designed for the batch setting, in which the number of samples is specified in advance (Canonne, 2022a).

There is a common thread connecting our algorithms that also represents the source of their advantage over previous methods. Any set of n samples from a distribution contains $\binom{n}{2} = \Theta(n^2)$ pairs of samples whose values could potentially collide. Our algorithms examine $\Theta(n^2)$ of these pairs when estimating the collision probability of the distribution, while previous methods used only $O(n)$ pairs. Examining more pairs allows our algorithms to extract more information from a given set of samples, but also significantly complicates the algorithms' analysis, since the pairs are not all disjoint and therefore are not statistically independent.

Private sequential testing: The hashing technique we use in our private estimation algorithm can be plugged into our sequential testing algorithm, which results in a private sequential tester. This algorithm has a worse dependency on the privacy parameters α and β than our private estimator, but depends on the error parameter ϵ in the same way as our non-private sequential testing algorithm. To our best knowledge, this is the first private sequential testing algorithm for testing collision probability.

For simplicity, in the main body of this paper we state all theorems using big-O notation, reserving more detailed theorem statements and proofs for the Appendix.

2 RELATED WORK

The collision probability of a distribution is equal to its second frequency moment, and frequency moment estimation has been widely studied in the literature on data streams, beginning with the seminal work of Alon et al. (1999). The optimal algorithm for estimating the collision probability of a distribution (rather than a non-random data stream) was given by Crouch et al. (2016), but their algorithm is not private. Locally differentially private estimation of the frequency moments of a distribution was first studied by Butucea and Issartel (2021), who gave a non-interactive mechanism for estimating any positive frequency moment. The sample complexity of their mechanism depends on the support size of the distribution, and they asked whether this dependence could be removed. Their conjecture was affirmatively resolved for collision probability by Bravo-Hermsdorff et al. (2022), but removing the dependence on support size led to a much worse dependence on the privacy parameter. It has remained an open question until now whether this trade-off is necessary.

Property and closeness testing has a rich literature (Acharya et al., 2019a, 2013; Diakonikolas et al., 2015; Goldreich and Ron, 2000; Canonne, 2022b), but the sequential setting is studied much less intensively. Existing algorithms for sequential testing almost always define closeness in terms of total variation distance, which leads to sample complexities on the order $O(\sqrt{k}/\epsilon^2)$, where k is the support size of the distribution and the distribution is separated from the null hypothesis by ϵ in terms of total variation distance (Daskalakis and Kawase, 2017; Oufkir et al., 2021). By contrast, all of our results are entirely independent of k, making our approach more suitable when the support size is very large.

There are several batch testing approaches which are based on collision statistics. Most notably, the optimal uniform testing algorithm of Paninski (2003) distinguishes the uniform distribution from a distribution that is ϵ far from uniform in terms of total variation distance with a sample complexity $\Theta(\sqrt{k}/\epsilon^2)$. How-

[1] Instead of denoting the privacy parameters by ε and δ, as is common in the privacy literature, we will use them to denote error and probability, as is common in the statistics literature.

ever, in the batch setting, the parameter ϵ is given to the testing algorithm as input.

3 PRELIMINARIES

We study two problems related to learning the collision probability $C(\mathbf{p}) = \sum_i p_i^2$ of an unknown distribution $\mathbf{p} = (p_1, \ldots, p_k)$.

In the **private estimation problem**, a set of N users each possess a single sample drawn independently from distribution $\mathbf{p}$. We are given an error bound $\varepsilon > 0$ and confidence level $\delta \in [0, 1]$. A central server must compute an estimate $\hat{C}$ that satisfies $|\hat{C} - C(\mathbf{p})| \leq \varepsilon$ with probability at least $1-\delta$ while preserving the privacy of the users' samples. A *mechanism* is a distributed protocol between the server and the users that privately computes this estimate. The execution of a mechanism can depend on the samples, and the output of a mechanism is the entire communication transcript between the server and the users. Mechanism M satisfies (α, β)-*local differential privacy* if for each user i and all possible samples $x_1, \ldots, x_N, x_i'$ we have

$$\Pr[M(x_1, \ldots, x_N) \in \mathcal{O}] \leq e^{\alpha} \Pr[M(x_1, \ldots, x_{i-1}, x_i', x_{i+1}, \ldots, x_N) \in \mathcal{O}] + \beta,$$

where $\mathcal{O}$ is any set of possible transcripts between the server and the users. In other words, if the privacy parameters α and β are small then changing the sample of a single user does not significantly alter the distribution of the mechanism's output. Local differential privacy is the strongest version of differential privacy, and is suitable for a setting where the server is untrusted (Dwork et al., 2014). The *sample complexity* of the mechanism is the expected number of users $n = \mathrm{E}[N]$ whose samples are observed by the server.

In the **sequential testing problem**, we are given a confidence level $\delta \in [0, 1]$ and the promise that exactly one of the following two hypotheses hold: The *null hypothesis* is that $C(\mathbf{p}) = c_0$, while the *alternative hypothesis* is that $|C(\mathbf{p}) - c_0| \geq \varepsilon > 0$. An algorithm must decide which hypothesis is correct based on samples from $\mathbf{p}$. Instead of fixing the number of samples in advance, the algorithm draws independent samples from $\mathbf{p}$ one at a time, and after observing each sample, decides to either reject the null hypothesis or to continue sampling. If the null hypothesis is false then the algorithm must reject it, and if the null hypothesis is true then the algorithm must not stop sampling, and each of these events must occur with probability at least $1-\delta$. Importantly, while c_0 is known to the algorithm, ε is not known, and thus the algorithm must adapt to the difficulty of the problem. The *sample complexity* of the algorithm is the number of observed samples N if the null hypothesis is false, a random variable.

4 PRIVATE ESTIMATION

In this section we describe a distributed protocol for privately estimating the collision probability of a distribution. In our protocol, a set of users each draw a sample from the distribution, and then share limited information about their samples with a central server, who computes an estimate of the collision probability while preserving the privacy of each user's sample.

As discussed in Section 1, the collision probability of a distribution is the probability that two independent samples from the distribution will coincide. Therefore the most straightforward strategy the server could employ would be to collect all the users' samples and count the number of pairs of samples containing a collision. However, this approach would not be privacy-preserving.

Instead, in Mechanism 1 below, each user applies a one-bit hash function to their private sample and shares only their hash value with the server. The server computes a statistic that essentially counts the number of collisions among all pairs of hash values, and then applies a bias correction to form an estimate of the collision probability. To increase the robustness of this estimate, the server first partitions the hash values into groups and uses the median estimate from among the groups.

The hashing procedure in Mechanism 1 is carefully designed to both preserve user privacy and also yield an accurate estimate. On the one hand, if each user privately chose an independent hash function, then their hash values would be entirely uncorrelated and contain no useful information about the underlying distribution. On the other hand, if every user applied the same hash function to their sample, then the server could invert this function and potentially learn some user's sample. Instead, in Mechanism 1, the server sends the same hash function to all users, but each user prepends their sample with a independently chosen *salt*, or random integer, before applying the hash function. Salts are commonly used in cryptographic protocols to enhance security, and they play a similar role in our mechanism. The number of possible salts serves as a trade-off parameter between the privacy and accuracy of our mechanism, with more salts implying a stronger privacy guarantee.

The theorems in this section provide guarantees about the privacy and accuracy of Mechanism 1.

Theorem 1. *Mechanism 1 satisfies* (α, β)*-local differential privacy.*

Mechanism 1 Private estimation for collision probability

Parameters: Expected number of users n, desired relative error $\varepsilon_{\text{rel}} \in (0,1]$, failure probability $\delta \in (0,1]$, privacy parameters $\alpha \geq 0, \beta \in (0,1]$.

Server draws N_j from Poisson(m) for each $j \in \{1,\dots,g\}$, where $g = \frac{160 \log \frac{1}{\delta}}{\varepsilon_{\text{rel}}^2}$ and $m = \frac{n}{g}$.
Server partitions users into g groups, where each group j has size N_j.
Server transmits to each user i their assigned group j_i.
Server transmits random hash function $h : \{0,1\}^* \mapsto \{-1,+1\}$ to each user.
Each user i chooses salt s_i uniformly at random from $\{1,\dots,r\}$, where $r = 6\left(\frac{e^\alpha+1}{e^\alpha-1}\right)^2 \log \frac{4}{\beta}$.
Each user i draws sample x_i from distribution $\mathbf{p}$.
Each user i sends hash value $v_i = h(\langle j_i, s_i, x_i\rangle)$ to the server.
Server lets $V_j = \sum_{i \in I_j} v_i$ for each group j, where I_j is the set of users in group j.
Server lets $C_j = \frac{r(V_j^2 - m)}{m^2}$ for each group j.
Server partitions the g groups into $a = 8 \log \frac{1}{\delta}$ supergroups, each containing $b = \frac{g}{a}$ groups.
Server lets $\bar{C}_\ell = \frac{1}{b} \sum_{j \in J_\ell} C_j$ for each supergroup ℓ, where J_ℓ is the set of groups in supergroup ℓ.
Server outputs $\hat{C}$, the median of $\bar{C}_1, \dots, \bar{C}_a$.

Theorem 2. *The sample complexity of Mechanism 1 is n. Moreover, if $n \geq \Omega\left(\left(\frac{e^\alpha+1}{e^\alpha-1}\right)^2 \frac{\log \frac{1}{\beta} \log \frac{1}{\delta}}{C(\mathbf{p})\varepsilon_{rel}^2}\right)$ then the estimate $\hat{C}$ output by Mechanism 1 satisfies $|\hat{C} - C(\mathbf{p})| \leq \varepsilon_{rel} C(\mathbf{p})$ with probability $1-\delta$.*

To simplify comparison to previous work, we state a straightforward corollary of Theorem 2 that converts its relative error guarantee to an absolute error guarantee.

Corollary 1. *If $\alpha \leq 1$, $\varepsilon_{rel} = \frac{\varepsilon}{C(\mathbf{p})} \in (0,1]$ and $n \geq \Omega\left(\frac{C(\mathbf{p}) \log \frac{1}{\beta} \log \frac{1}{\delta}}{\alpha^2 \varepsilon^2}\right)$ then the estimate $\hat{C}$ output by Mechanism 1 satisfies $|\hat{C} - C(\mathbf{p})| \leq \varepsilon$ with probability $1-\delta$.*

4.1 Lower bound

The next theorem proves that the sample complexity bound in Corollary 1 is tight for small α up to logarithmic factors.

Theorem 3. *Let $\hat{C}_{\alpha,n}(\mathbf{p})$ be a collision probability estimate returned by an $(\alpha,0)$-locally differentially private mechanism that draws n samples from distribution $\mathbf{p}$. If $\varepsilon \leq 1, \alpha \leq \frac{23}{35}$ and $n \leq o\left(\frac{1}{\alpha^2\varepsilon^2}\right)$ then there exists a distribution $\mathbf{p}$ such that $\mathrm{E}\left[|\hat{C}_{\alpha,n}(\mathbf{p}) - C(\mathbf{p})|\right] \geq \varepsilon$.*

4.2 Reduction to private distribution estimation

A natural alternative to Mechanism 1 would be an indirect approach that privately estimates the distribution itself, and then computes the collision probability of the estimated distribution. We prove a theoretical separation between these two approaches, by showing that this reduction does not preserve optimality. Specifically, we show that even if the mechanism for private distribution estimation has the optimal sample complexity, using it as a subroutine for collision probability estimation may require a number of samples that depends on the support size of the distribution. By contrast, the sample complexity of our method is independent of support size (see Corollary 1).

Let $[k] = \{1,\dots,k\}$ be the sample space. Let Δ_k be the set of all distributions on $[k]$. Let $A : [k]^n \to \Delta_k$ denote an algorithm that inputs n samples, one per user, and outputs an estimated distribution. An (α,β)-local differentially private algorithm A^* is (α,β)-*minimax optimal* if

$$A^* \in \arg\min_{A \in \mathcal{A}_{\alpha,\beta}} \max_{\mathbf{p} \in \Delta^k} \mathrm{E}_{x_1,\dots,x_n \sim \mathbf{p}^n}[\|A(x_1,\dots,x_n) - \mathbf{p}\|_1]$$

where $\mathcal{A}_{\alpha,\beta}$ is the set of all (α,β)-local differentially private algorithms for estimating distributions.

Theorem 4. *For all $\alpha \in (0,1)$ there exists an $(\alpha,0)$-minimax optimal algorithm $A^* : [k]^n \to \Delta_k$ and a distribution $\mathbf{p} \in \Delta^k$ such that if each $x_i \in [k]$ is drawn independently from $\mathbf{p}$ and $\mathrm{E}[|C(A^*(x_1,\dots,x_n)) - C(\mathbf{p})|] \leq \varepsilon$ then $n \geq \Omega\left(\min\left\{\frac{k^2}{\alpha^2}, \frac{k}{\alpha^2\varepsilon}\right\}\right)$.*

4.3 Comparison to previous work

Butucea and Issartel (2021) gave a non-interactive $(\alpha,0)$-locally differentially private mechanism for estimating collision probability with sample complexity $\tilde{O}\left(\frac{(\log k)^2}{\alpha^2\varepsilon^2}\right)$ and communication complexity $O(k)$. Bravo-Hermsdorff et al. (2022) gave a non-interactive mechanism with the same privacy guarantee, sample complexity $\tilde{O}\left(\frac{1}{\alpha^4\varepsilon^2}\right)$, and communication complexity

$O(1)$.[2] Thus the latter mechanism is better suited to distributions with very large support sizes, but is a worse choice when the privacy parameter α is very small. Our mechanism combines the advantages of these mechanisms, at the expense of a slightly weaker privacy guarantee and an additional $O(C(\mathbf{p}) \log \frac{1}{\beta})$ samples.

Notably, the earlier mechanism due to Bravo-Hermsdorff et al. (2022) is also based on counting collisions among salted hash values. But there are key differences between the mechanisms which lead to our improved sample complexity. In their mechanism, the server assigns salts to the users, each user adds noise to their hash value, and the server counts hash collisions among $\frac{n}{2}$ disjoint user pairs. In our mechanism, the salts are chosen privately, no additional noise is added to the hash values, and the server counts hash collisions among $\Theta(n^2)$ user pairs. Using more available pairs to count collisions is a more efficient use of data (although it significantly complicates the analysis, as the pairs are not all independent), and choosing the salts privately eliminates the need for additional randomness, which improves the accuracy of the estimate.

5 SEQUENTIAL TESTING

In this section we describe an algorithm for sequentially testing whether $C(\mathbf{p}) = c_0$ (the null hypothesis) or $|C(\mathbf{p}) - c_0| \geq \varepsilon > 0$ (the alternative hypothesis), where c_0 is given but ε is unknown. Algorithm 2 below draws samples from the distribution $\mathbf{p}$ one at a time. Whenever the algorithm observes a sample x_i it updates a running estimate of $|C(\mathbf{p}) - c_0|$ based on the all-pairs collision frequency observed so far. The algorithm compares this estimate to a threshold that shrinks like $\Theta\big(\sqrt{i^{-1} \log\log i}\big)$ and rejects the null hypothesis as soon as the threshold is exceeded. Although our algorithm is simple to describe, its proof of correctness is non-trivial, as it involves showing that a sequence of dependent random variables (the running estimates) become concentrated. Our proof uses a novel decoupling technique to construct martingales based on the running estimates.

The next theorem provides a guarantee about the accuracy of Algorithm 2.

Theorem 5. *If $C(\mathbf{p}) = c_0$ then Algorithm 2 does not reject the null hypothesis with probability $1 - \delta$. If $|C(\mathbf{p}) - c_0| \geq \varepsilon$ then Algorithm 2 rejects the null hypothesis after observing N samples, where $N \in O\left(\frac{1}{\varepsilon^2} \log\log \frac{1}{\varepsilon} \log \frac{1}{\delta}\right)$ with probability $1 - \delta$.*

[2]Note that Bravo-Hermsdorff et al.'s original NeurIPS paper claimed $\tilde{O}\left(\frac{1}{\alpha^2 \varepsilon^2}\right)$ sample complexity, but a more recent version on Arxiv claims $\tilde{O}\left(\frac{1}{\alpha^4 \varepsilon^2}\right)$ sample complexity and explains that the original version contained mistakes. See References for a link to the Arxiv version.

Algorithm 2 Sequential testing of collision probability

Given: Null hypothesis value c_0, confidence level $\delta \in [0, 1]$.
for $i = 1, 2, 3, \ldots$ **do**
  Draw sample x_i from distribution $\mathbf{p}$.
  Let $T_i = \sum_{j=1}^{i-1} \mathbf{1}\{x_i = x_j\} - 2(i-1)c_0$.
  if $\left|\frac{2}{i(i-1)} \sum_{j=1}^{i} T_j\right| > 3.2\sqrt{\frac{\log\log i + 0.72 \log(20.8/\delta)}{i}}$ **then**
    Reject the null hypothesis.
  end if
end for

The $\log\log \frac{1}{\varepsilon}$ factor in Theorem 5 results from our application of a confidence interval due to Howard et al. (2021) that shrinks like $\Theta\big(\sqrt{i^{-1} \log\log i}\big)$. Note that $\log\log \frac{1}{\varepsilon} < 4$ if $\varepsilon \geq 10^{-10}$, so this factor is negligible for nearly all problem instances of practical interest.

We remark that our proof technique bears some superficial resemblance to the approach used in recent work by Oufkir et al. (2021). They make use of the fact that for any random variable T taking values from $\mathbb{N}$ and for all $T \in \mathbb{N}_+$, it holds that $\mathbb{E}[T] \leq N + \sum_{t \geq N} \mathbb{P}(T \geq t)$. Then with a carefully selected N and Chernoff bounds with infinite many applications of union bound implies upper bound on the expected sample complexity. By contrast, we construct a test martingale that is specific to collision probability and apply an anytime or time-uniform concentration bound to the martingale introduced by Waudby-Smith and Ramdas (2020).

5.1 Lower bound

The next theorem proves that sample complexity bound in Theorem 5 is tight up to log-log factors.

Theorem 6. *Let N be the number of samples observed by a sequential testing algorithm for collision probability. For all $\varepsilon, \delta \in [0, 1]$ there exists a distribution $\mathbf{p}$ and $c_0 \in [0, 1]$ such that $|C(\mathbf{p}) - c_0| \geq \varepsilon$ and if the algorithm rejects the null hypothesis with probability $1 - \delta$ then $\mathrm{E}[N] \geq \Omega\left(\frac{\log(1/\delta)}{\varepsilon^2}\right)$.*

5.2 Comparison to previous work

Daskalakis and Kawase (2017) described a general sequential testing algorithm that can be adapted to estimate collision probability and has a worst-case sample complexity comparable to Algorithm 2. However, their approach has two major disadvantages relative to ours which lead to much higher sample complexities in

practice, and which we empirically confirm in Section 6. First, their approach is based on a simple "doubling trick": They repeatedly invoke a non-sequential testing algorithm on subsequences of samples with successively smaller values of the error tolerance ϵ, and stop when the testing algorithm rejects. This is a wasteful use of samples compared to our approach, as stopping cannot occur within a subsequence, and everything learned from previous subsequences is discarded. Second, applying their approach to collision probability testing requires partitioning the n samples into $\frac{n}{2}$ disjoint pairs, so that the observed collisions are independent of each other. By contrast, our approach uses observed collisions among all $\Theta(n^2)$ pairs of samples to estimate collision probability, which significantly complicates the theoretical analysis, but leads to better empirical performance.

5.3 Private sequential testing

In this section, we apply the private estimation technique to sequential testing. The most obvious approach to private sequential testing is to apply Mechanism 1 repeatedly with an exponentially increasing sample size, and then apply the union bound over the repetitions. We call this algorithm the Doubling Tester (as it is based on the doubling trick (Auer et al., 1995; Cesa-Bianchi and Lugosi, 2006)), and it has the following sample complexity.

Theorem 7. *If $C(\mathbf{p}) = c_0$ then the Doubling Tester is (α, β)-local differential privacy and does not reject the null hypothesis with prob. $1-\delta$. If $|C(\mathbf{p}) - c_0| \geq \varepsilon$ then the Doubling Tester rejects the null hypothesis after observing N_1 samples if $\alpha > 1$ and N_2 if $\alpha \leq 1$, where $N_1 \in O\left(\left(\frac{e^\alpha+1}{e^\alpha-1}\right)^2 \frac{\log^2 \frac{1}{\varepsilon} \log \frac{1}{\beta} \log \frac{1}{\delta}}{\varepsilon^2}\right)$ and $N_2 \in O\left(\frac{\log^2 1/\varepsilon}{\varepsilon^2} \cdot \frac{\log \frac{1}{\beta} \log \frac{1}{\delta}}{\alpha^2}\right)$ respectively, with prob. $1-\delta$.*

Another approach is based on the observation that the hashing of Mechanism 1 only depends on the privacy parameters. So we can apply the same hashing to the input of Algorithm 2. The only difference is that the hashing introduces a bias to the test statistic, which has to be corrected, and the scale of the statistic becomes of order $\log \frac{1}{\beta}$. We call this the Private Sequential Tester (PSQ), which is described in Appendix 17, and it has the following sample complexity.

Theorem 8. *If $C(\mathbf{p}) = c_0$ then the PSQ algorithm is (α, β)-local differential privacy and does not reject the null hypothesis with probability $1-\delta$. If $|C(\mathbf{p}) - c_0| \geq \varepsilon$ then the PSQ algorithm rejects the null hypothesis after observing N_1 samples if $\alpha > 1$ and N_2 if $\alpha \leq 1$, where $N_1 \in O\left(\left(\frac{e^\alpha+1}{e^\alpha-1}\right)^4 \frac{\log\log \frac{1}{\varepsilon} \log^2 \frac{1}{\beta} \log \frac{1}{\delta}}{\varepsilon^2}\right)$ and $N_2 \in O\left(\frac{\log\log 1/\varepsilon}{\varepsilon^2} \cdot \frac{\log^2 \frac{1}{\beta} \log \frac{1}{\delta}}{\alpha^2}\right)$ respectively, with probability $1-\delta$.*

Based on Theorem 8, the Private Sequential Tester has lower sample complexity than the Doubling Tester, if $\log \frac{1}{\varepsilon}$ is much larger than $\log \frac{1}{\beta}$. This is often the case since, for example, for a close-to-uniform distribution with large domain size d, in which case the interesting parameter budget is $\epsilon < 1/d$.

6 EXPERIMENTS

We compared our mechanism for private collision probability estimation (Mechanism 1) to the recently proposed mechanism from Bravo-Hermsdorff et al. (2022). As discussed in Section 4.3, we expect Mechanism 1 to outperform their mechanism when the support size of the distribution is large and the privacy requirement is strict. We also compared to the indirect method described in Section 4.2: Privately estimate the distribution itself, and then compute the collision probability of the estimated distribution. In our experiments we use an open-source implementation of a private heavy hitters algorithm due to Cormode et al. (2021).[3]

In Figure 1 we use each mechanism to privately estimate the collision probability of two distributions supported on 1000 elements: the uniform distribution ($p_i = 1/k$) and the power law distribution ($p_i \propto 1/i$). Our simulations show that Mechanism 1 has significantly lower error for small values of the privacy parameters α and β.

We next compared our sequential testing algorithm (Algorithm 2) to the algorithm from Daskalakis and Kawase (2017). We ran experiments comparing the two algorithms on the power law distribution ($p_i \propto 1/i$) and exponential distribution ($p_i \propto \exp(-i)$), with the results depicted in Figure 2. Consistent with our discussion in Section 5.2, we found that as each tester's null hypothesis approaches the true collision probability, the empirical sample complexity of Daskalakis and Kawase (2017)'s algorithm becomes much larger than the empirical sample complexity of Algorithm 2.

We also compared Algorithm 2 to two batch testing algorithms, both of which are described in a survey by Canonne (2022a):

- **Plug-in:** Form empirical distribution $\hat{\mathbf{p}}$ from samples $x_1, \ldots, x_n$ and let $\hat{C} = C(\hat{\mathbf{p}})$.
- **U-statistics:** Let $\hat{C} = \frac{2}{n(n-1)} \sum_{i<j} \mathbf{1}\{x_i = x_j\}$ be the all-pairs collision frequency.

[3] https://github.com/Samuel-Maddock/pure-LDP

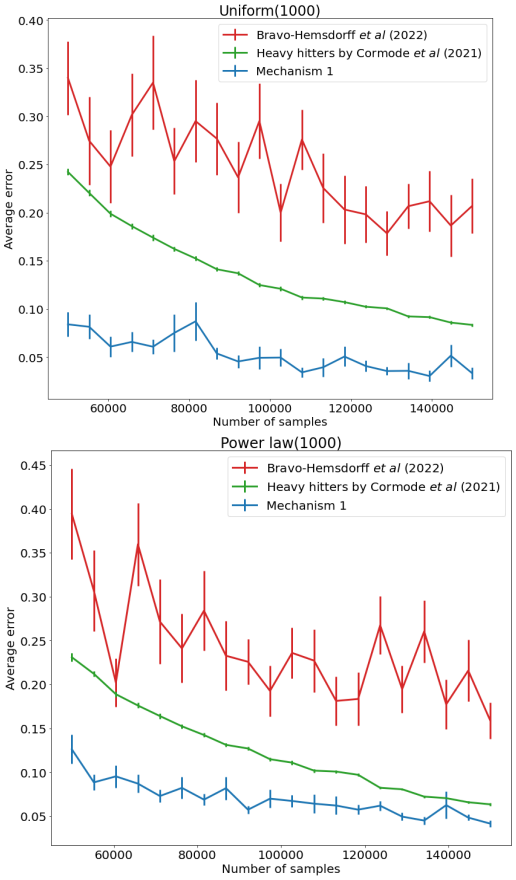

Figure 1: Sample complexity of private collision probability estimation mechanisms for $\alpha = 0.25$. Both mechanisms use the MD5 hash function and confidence level $\delta = 0.1$. For Mechanism 1 we let $\beta = 10^{-5}$. Error bars are one standard error.

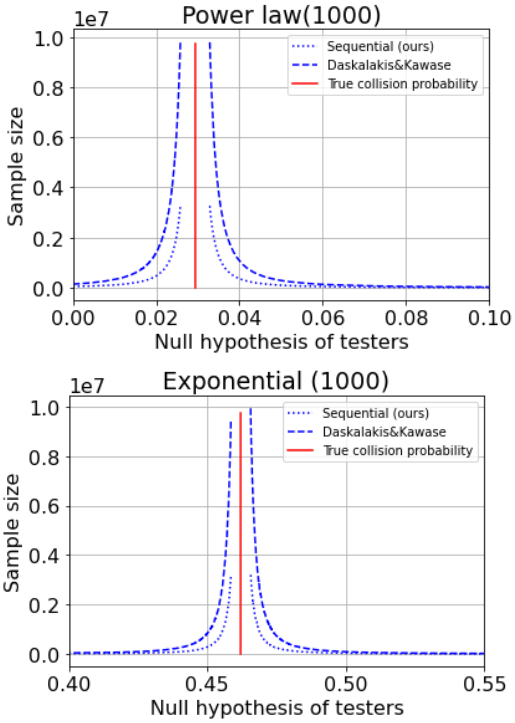

Figure 2: Sample complexity of our sequential tester (Algorithm 2) compared to the sample complexity of Daskalakis and Kawase (2017)'s sequential tester adapted for collision probability testing.

Each batch testing algorithm takes as input both the null hypothesis value c_0 and a tolerance parameter ε, and compares $|\hat{C} - c_0|$ to ε to decide whether to reject the null hypothesis $C(\mathbf{p}) = c_0$. The sample complexity of a batch testing algorithm is determined via worst-case theoretical analysis in terms of ε. On the other hand, sequential testing algorithms automatically adapt their sample complexity to the difference $|C(\mathbf{p}) - c_0|$.

In Figure 3 we evaluate batch and our sequential testing algorithms on both on the uniform distribution and power law distributions. We use 20 different support sizes for each distribution, evenly spaced on a log scale between 10 and 10^6 inclusively. Varying the support size also varies $|C(\mathbf{p}) - c_0|$.

As expected, when $|C(\mathbf{p}) - c_0|$ is large, our sequential testing algorithm requires many fewer samples than the batch algorithm to reject the null hypothesis, and as $|C(\mathbf{p}) - c_0|$ shrinks the number of samples required sharply increases (see grey areas in Figure 3). In all cases our sequential testing algorithm is never outperformed by the batch testing algorithms.

Note that in Figure 3 the plug-in tester has a worse sample complexity than the U-statistics tester. Since these sample complexities are determined by theoretical analysis, we experimentally confirmed that this discrepancy is not simply an artifact of the analysis. In Figure 4 we run simulations comparing the algorithms in terms of their error $|\hat{C} - C(\mathbf{p})|$, and find that the plug-in tester is also empirically worse than the U-statistics tester.

6.1 Private sequential tester

In the last set of experiments, we evaluated the private sequential testing algorithm that is introduced in Subsection 5.3 and defined in Algorithm 3. We refer to this algorithm as PSQ. As a baseline, we ran Mechanism 1 with the so-called "doubling trick", with the initial sample complexity set to 2^{13}, and if the confidence interval of the estimator was not tight enough to make decision, *i.e.* reject or accept, then we doubled the sample complexity and re-ran Mechanism 1. We refer to this approach as Doubling Sequential Tester. The

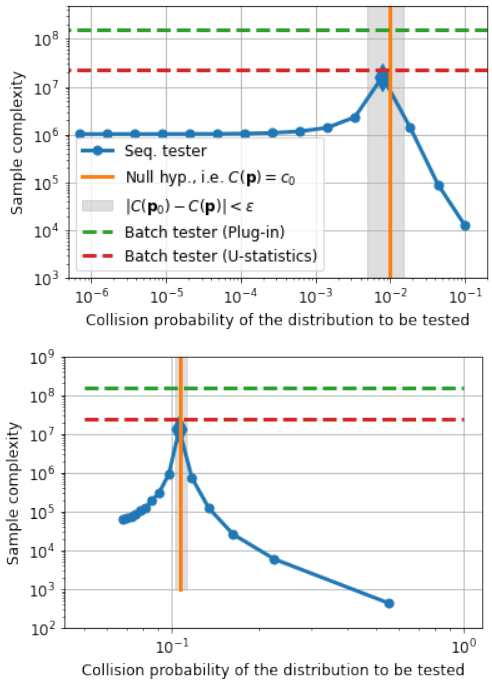

Figure 3: Sample complexity of the sequential tester compared to the sample complexity of the batch testers. For the batch testers, the tolerance parameter ϵ is set to 0.01.

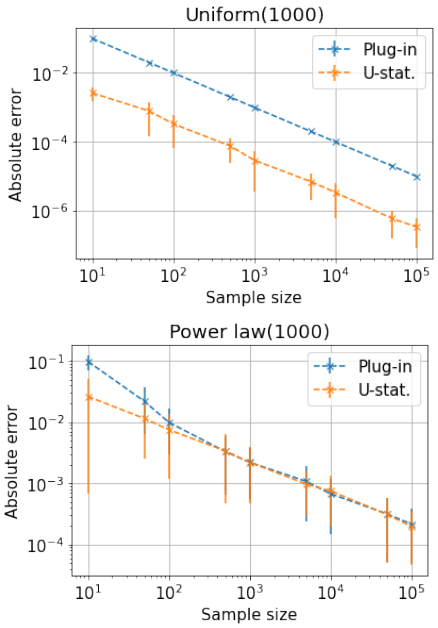

Figure 4: Empirical absolute error of plug-in and U-statistic estimators when the data is generated from uniform distribution and power law with domain size 1000.

sample complexity of private sequential algorithms are show in Figure 5 on the same problem instance that

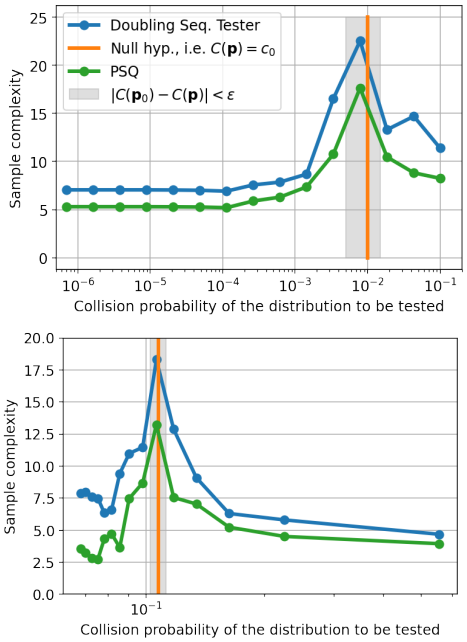

Figure 5: Sample complexity of private sequential testing algorithms with respect to the non-private estimator. The sample complexity of private sequential testers is divided by the sample complexity of the non-private sequential tester from Figure 3 and the multiplicative factors are shown.

were used for evaluating the non-private testers in Figure 4. Unsurprisingly, private testers require more samples than the non-private ones, and so we plotted the multiplicative factor between the sample complexity of private and non-private testers. The Doubling Sequential Tester requires 10x more samples if the distribution to be tested close to the null hypothesis. The PSQ algorithm typically requires fewer samples, however it still needs 3-17x times more samples than the non-private tester.

7 CONCLUSIONS AND FUTURE WORK

We introduced a locally differentially private estimator for collision probability that is near-optimal in a minimax sense and empirically superior to the state-of-the-art method introduced by Bravo-Hermsdorff et al. (2022). Our method is based on directly estimating the collision probability using all pairs of observed samples, unlike in previous work. We also introduced a near-optimal sequential testing algorithm that is likewise based on directly estimating the collision probability, and requires far fewer samples than minimax optimal testing algorithms for many problem instances and can be combined with our private estimator.

References

Jayadev Acharya, Ashkan Jafarpour, Alon Orlitsky, and Ananda Suresh. A competitive test for uniformity of monotone distributions. In Carlos M. Carvalho and Pradeep Ravikumar, editors, *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, volume 31 of *Proceedings of Machine Learning Research*, pages 57–65, Scottsdale, Arizona, USA, 29 Apr–01 May 2013. PMLR.

Jayadev Acharya, Alon Orlitsky, Ananda Theertha Suresh, and Himanshu Tyagi. The complexity of estimating rényi entropy. In *Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms*, pages 1855–1869. SIAM, 2014.

Jayadev Acharya, Clement Canonne, Cody Freitag, and Himanshu Tyagi. Test without trust: Optimal locally private distribution testing. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pages 2067–2076. PMLR, 16–18 Apr 2019a.

Jayadev Acharya, Ziteng Sun, and Huanyu Zhang. Hadamard response: Estimating distributions privately, efficiently, and with little communication. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1120–1129. PMLR, 2019b.

Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. *Journal of Computer and System Sciences*, 58(1):137–147, 1999.

P. Auer, N. Cesa-Bianchi, Y. Freund, and R.E. Schapire. Gambling in a rigged casino: The adversarial multi-armed bandit problem. In *Proceedings of IEEE 36th Annual Foundations of Computer Science*, pages 322–331, 1995.

Gecia Bravo-Hermsdorff, Róbert Busa-Fekete, Mohammad Ghavamzadeh, Andres Munoz Medina, and Umar Syed. Private and communication-efficient algorithms for entropy estimation. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 15382–15393. Curran Associates, Inc., 2022. URL `https://arxiv.org/pdf/2305.07751.pdf`.

Róbert Busa-Fekete, Dimitris Fotakis, Balázs Szörényi, and Emmanouil Zampetakis. Identity testing for mallows model. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 23179–23190, 2021. URL `https://proceedings.neurips.cc/paper/2021/hash/c315f0320b7cd4ec85756fac52d78076-Abstract.html`.

Cristina Butucea and Yann Issartel. Locally differentially private estimation of functionals of discrete distributions. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 24753–24764. Curran Associates, Inc., 2021. URL `https://proceedings.neurips.cc/paper/2021/file/cf8c9be2a4508a24ae92c9d3d379131d-Paper.pdf`.

Clément L. Canonne. Topics and techniques in distribution testing: A biased but representative sample. *Found. Trends Commun. Inf. Theory*, 19(6): 1032–1198, nov 2022a. ISSN 1567-2190. doi: 10.1561/0100000114. URL `https://doi.org/10.1561/0100000114`.

Clément L Canonne. *Topics and techniques in distribution testing*. Now Publishers, 2022b.

Nicoló Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge University Press, 2006.

Graham Cormode and Minos Garofalakis. Join sizes, frequency moments, and applications. In *Data Stream Management: Processing High-Speed Data Streams*, pages 87–102. Springer, 2016.

Graham Cormode, Samuel Maddock, and Carsten Maple. Frequency estimation under local differential privacy. *PVLDB Journal Proceedings*, 14(11): 2046–2058, 2021.

Michael Crouch, Andrew McGregor, Gregory Valiant, and David P Woodruff. Stochastic streams: Sample complexity vs. space complexity. In *24th Annual European Symposium on Algorithms (ESA 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

Constantinos Daskalakis and Yasushi Kawase. Optimal Stopping Rules for Sequential Hypothesis Testing. In *25th Annual European Symposium on Algorithms (ESA 2017)*, volume 87 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 32:1–32:14. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2017. URL `http://drops.dagstuhl.de/opus/volltexte/2017/7823`.

V. de la Peña and E. Giné. *Decoupling: From Dependence to Independence*. Probability and Its Applications. Springer New York, 1999.

Ilias Diakonikolas, Daniel M. Kane, and Vladimir Nikishkin. Testing identity of structured distributions. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*

2015, San Diego, CA, USA, January 4-6, 2015, pages 1841–1854, 2015.

Devdatt Dubhashi and Alessandro Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press, USA, 1st edition, 2009.

John C. Duchi, Martin J. Wainwright, and Michael I. Jordan. Minimax optimal procedures for locally private estimation. *CoRR*, abs/1604.02390, 2016.

Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4): 211–407, 2014.

Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, pages 1054–1067, 2014.

Corrado Gini. *Variabilità e mutabilità: contributo allo studio delle distribuzioni e delle relazioni statistiche.[Fasc. I.]*. Tipogr. di P. Cuppini, 1912.

Oded Goldreich and Dana Ron. On testing expansion in bounded-degree graphs. *Electron. Colloquium Comput. Complex.*, 7(20), 2000.

Orris C Herfindahl. *Concentration in the steel industry*. Columbia University, 1997.

Steven R. Howard, Aaditya Ramdas, Jon McAuliffe, and Jasjeet Sekhon. Time-uniform, nonparametric, nonasymptotic confidence sequences. *The Annals of Statistics*, 49(2):1055 – 1080, 2021.

Markku Laakso and Rein Taagepera. "Effective" number of parties: A measure with application to west europe. *Comparative Political Studies*, 12(1):3–27, 1979. doi: 10.1177/001041407901200101. URL `https://doi.org/10.1177/001041407901200101`.

Tom Leinster. *Entropy and Diversity: The Axiomatic Approach*. Cambridge University Press, 2021. doi: 10.1017/9781108963558.

Gábor Lugosi and Shahar Mendelson. Mean estimation and regression under heavy-tailed distributions: A survey. *Foundations of Computational Mathematics*, 19(5):1145–1190, 2019.

Abolfazl S Motahari, Guy Bresler, and NC David. Information theory of dna shotgun sequencing. *IEEE Transactions on Information Theory*, 59(10):6273–6289, 2013.

Aadil Oufkir, Omar Fawzi, Nicolas Flammarion, and Aurélien Garivier. Sequential algorithms for testing closeness of distributions. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 11655–11664. Curran Associates, Inc., 2021.

Liam Paninski. Estimation of entropy and mutual information. *Neural Comput.*, 15(6):1191–1253, 2003.

John Riordan. Moment Recurrence Relations for Binomial, Poisson and Hypergeometric Frequency Distributions. *The Annals of Mathematical Statistics*, 8(2):103 – 111, 1937. doi: 10.1214/aoms/1177732430. URL `https://doi.org/10.1214/aoms/1177732430`.

Edward H Simpson. Measurement of diversity. *nature*, 163(4148):688–688, 1949.

Maciej Skorski. Evaluating entropy for true random number generators: Efficient, robust and provably secure. In *International Conference on Information Security and Cryptology*, pages 526–541, 03 2017.

Constantino Tsallis. Possible generalization of boltzmann-gibbs statistics. *Journal of statistical physics*, 52(1):479–487, 1988.

Gregory Valiant and Paul Valiant. Estimating the unseen: An n/log(n)-sample estimator for entropy and support size, shown optimal via new clts. In *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*, STOC '11, page 685–694, New York, NY, USA, 2011. Association for Computing Machinery. doi: 10.1145/1993636.1993727. URL `https://doi.org/10.1145/1993636.1993727`.

Ian Waudby-Smith and Aaditya Ramdas. Estimating means of bounded random variables by betting, 2020. URL `https://arxiv.org/abs/2010.09686`.

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]
   The mathematical setting is described in Section 3. The private algorithm assume an access to a random bit generator which is referred to as salt. This is described in Section 4 and this assumption is commonly used in cryptography protocols. The definition of each algorithm can be found in the paper or in the supplementary material as pseudocode.
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   The focus of this paper is to understand the interplay of privacy parameter and sample

complexity. We addressed this question for the private algorithms and also analyzed the sample complexity for the non-private algorithm.

(c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [No]
We did not release the code associated with our experiments, as it is proprietary and relies on non-open-sourced or non-free libraries (which are not part of the main scientific contribution of the paper, but needed for the parallel computation).

2. For any theoretical claim, check if you include:

(a) Statements of the full set of assumptions of all theoretical results. [Yes]
All theorem are included in the main paper.

(b) Complete proofs of all theoretical results. [Yes]
Full proofs of all our theoretical results are given in the appendix.

(c) Clear explanations of any assumptions. [Yes]
The main contribution of the paper is clearly stated in the abstract and introduction, and reflected in the technical and experimental sections (Sections 3-5 and the associated Appendices.

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). [No]
We are not releasing the code, as it is proprietary.

(b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
We reported all hyperparameters of the algorithms as well as the data generation process in every case.

(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]
Sample complexity is reported in most of the case which is clearly described in Section 6. This is explained in the experimental setup in detail. We carried out each experiments over 100 repetitions to get confidence interval.

(d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [No] Each algorithm we introduced can be implemented on a CPU. Parallel computation and cluster setting is not revealed since we used proprietary library to run the same setting over repetitions to obtain confidence interval for the reported statistics.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include: We used synthetic data in our experiments.

(a) Citations of the creator If your work uses existing assets. [Not Applicable]

(b) The license information of the assets, if applicable. [Not Applicable]

(c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]

(d) Information about consent from data providers/curators. [Not Applicable]

(e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include: We did not use crowdsourcing or research with human subject.

(a) The full text of instructions given to participants and screenshots. [Not Applicable]

(b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]

(c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Supplementary material for "Near-optimal algorithms for private estimation and sequential testing of collision probability"

8 Proof of Theorem 1

Let X_i and V_i be the private value and hash value, respectively, for user i, and let H be the random hash function chosen by the server.[4] Let $\mathbf{X} = (X_1, \ldots, X_n)$ and $\mathbf{V} = (V_1, \ldots, V_n)$.

Recall that in the local model of differential privacy, the output of a mechanism is the entire communication transcript between the users and the server, which in the case of Mechanism 1 consists of the hash function and all the hash values.[5] Therefore our goal is to prove that for any subset $\mathcal{O}$ of possible values for $(\mathbf{V}, H)$ we have

$$\Pr[(\mathbf{V}, H) \in \mathcal{O} \mid \mathbf{X} = \mathbf{x}] \leq e^{\alpha} \Pr[(\mathbf{V}, H) \in \mathcal{O} \mid \mathbf{X} = \mathbf{x}'] + \beta. \tag{1}$$

where $\mathbf{x}$ and $\mathbf{x}'$ differ in one component.

The proof of Eq. (1) will need a couple of observations. First, given the hash function and private values, the hash values are mutually independent:

$$\Pr[\mathbf{V} = \mathbf{v} \mid H = h \wedge \mathbf{X} = \mathbf{x}] = \prod_i \Pr[V_i = v_i \mid H = h \wedge X_i = x_i]. \tag{2}$$

Second, the hash function is independent of the private values:

$$\Pr[H = h \mid \mathbf{X} = \mathbf{x}] = \Pr[H = h \mid \mathbf{X} = \mathbf{x}']. \tag{3}$$

We also need a definition. Fix user i^*. Suppose $\mathbf{x}$ and $\mathbf{x}'$ differ in component i^*. Define H_α to be the set of all hash functions such that if $h \in H_\alpha$ then

$$\Pr[V_{i^*} = v \mid H = h \wedge X_{i^*} = x_{i^*}] \leq e^{\alpha} \Pr[V_{i^*} = v \mid H = h \wedge X_{i^*} = x'_{i^*}] \tag{4}$$

for all v. Note that the definition of H_α depends implicitly on x_{i^*} and x'_{i^*}, but does not depend on X_{i^*}.

Combining the above we have

$$\begin{aligned}
&\Pr[\mathbf{V} = \mathbf{v} \wedge H = h \mid \mathbf{X} = \mathbf{x} \wedge H \in \mathcal{H}_\alpha] \\
&= \Pr[\mathbf{V} = \mathbf{v} \mid H = h \wedge \mathbf{X} = \mathbf{x} \wedge H \in \mathcal{H}_\alpha] \cdot \Pr[H = h \mid \mathbf{X} = \mathbf{x} \wedge H \in \mathcal{H}_\alpha] \\
&= \prod_i \Pr[V_i = v_i \mid H = h \wedge X_i = x_i \wedge H \in \mathcal{H}_\alpha] \cdot \Pr[H = h \mid \mathbf{X} = \mathbf{x} \wedge H \in \mathcal{H}_\alpha] && \because \text{Eq. (2)} \\
&= \prod_i \Pr[V_i = v_i \mid H = h \wedge X_i = x_i \wedge H \in \mathcal{H}_\alpha] \cdot \Pr[H = h \mid \mathbf{X} = \mathbf{x}' \wedge H \in \mathcal{H}_\alpha] && \because \text{Eq. (3)} \\
&\leq e^{\alpha} \prod_i \Pr[V_i = v_i \mid H = h \wedge X_i = x'_i \wedge H \in \mathcal{H}_\alpha] \cdot \Pr[H = h \mid \mathbf{X} = \mathbf{x}' \wedge H \in \mathcal{H}_\alpha] && \because \text{Eq. (4)} \\
&= e^{\alpha} \Pr[\mathbf{V} = \mathbf{v} \mid H = h \wedge \mathbf{X} = \mathbf{x}' \wedge H \in \mathcal{H}_\alpha] \cdot \Pr[H = h \mid \mathbf{X} = \mathbf{x}' \wedge H \in \mathcal{H}_\alpha] && \because \text{Eq. (2)} \\
&= e^{\alpha} \Pr[\mathbf{V} = \mathbf{v} \wedge H = h \mid \mathbf{X} = \mathbf{x}' \wedge H \in \mathcal{H}_\alpha].
\end{aligned}$$

[4] Note that this definition of V_i overrides the definition given in Mechanism 1. We will not use the latter definition in this proof.

[5] Strictly speaking, the transcript also includes the group assignments. However, these assignments can be chosen arbitrarily, as long as each group has the desired size, and do not impact the privacy of the mechanism in any way. Hence we omit them from the transcript for simplicity.

Summing both sides over $\mathcal{O}$ yields

$$\Pr[(\mathbf{V}, H) \in \mathcal{O} \mid \mathbf{X} = \mathbf{x} \wedge H \in \mathcal{H}_\alpha] \leq e^\alpha \Pr[(\mathbf{V}, H) \in \mathcal{O} \mid \mathbf{X} = \mathbf{x}' \wedge H \in \mathcal{H}_\alpha]. \tag{5}$$

Therefore we have

$$\begin{aligned}
\Pr[(\mathbf{V}, H) \in \mathcal{O} \mid \mathbf{X} = \mathbf{x}] = {} & \Pr[(\mathbf{V}, H) \in \mathcal{O} \mid \mathbf{X} = \mathbf{x} \wedge H \in \mathcal{H}_\alpha] \cdot \Pr[H \in \mathcal{H}_\alpha \mid \mathbf{X} = \mathbf{x}] \\
& + \Pr[(\mathbf{V}, H) \in \mathcal{O} \mid \mathbf{X} = \mathbf{x} \wedge H \notin \mathcal{H}_\alpha] \cdot \Pr[H \notin \mathcal{H}_\alpha \mid \mathbf{X} = \mathbf{x}] \\
\leq {} & \Pr[(\mathbf{V}, H) \in \mathcal{O} \mid \mathbf{X} = \mathbf{x} \wedge H \in \mathcal{H}_\alpha] \cdot \Pr[H \in \mathcal{H}_\alpha \mid \mathbf{X} = \mathbf{x}] \\
& + \Pr[H \notin \mathcal{H}_\alpha \mid \mathbf{X} = \mathbf{x}] \\
= {} & \Pr[(\mathbf{V}, H) \in \mathcal{O} \mid \mathbf{X} = \mathbf{x} \wedge H \in \mathcal{H}_\alpha] \cdot \Pr[H \in \mathcal{H}_\alpha \mid \mathbf{X} = \mathbf{x}'] && \because \text{Eq. (3)} \\
& + \Pr[H \notin \mathcal{H}_\alpha \mid \mathbf{X} = \mathbf{x}] \\
\leq {} & e^\alpha \Pr[(\mathbf{V}, H) \in \mathcal{O} \mid \mathbf{X} = \mathbf{x}' \wedge H \in \mathcal{H}_\alpha] \cdot \Pr[H \in \mathcal{H}_\alpha \mid \mathbf{X} = \mathbf{x}'] && \because \text{Eq. (5)} \\
& + \Pr[H \notin \mathcal{H}_\alpha \mid \mathbf{X} = \mathbf{x}] \\
= {} & e^\alpha \Pr[(\mathbf{V}, H) \in \mathcal{O} \wedge H \in \mathcal{H}_\alpha \mid \mathbf{X} = \mathbf{x}'] + \Pr[H \notin \mathcal{H}_\alpha \mid \mathbf{X} = \mathbf{x}] \\
\leq {} & e^\alpha \Pr[(\mathbf{V}, H) \in O \mid \mathbf{X} = \mathbf{x}'] + \Pr[H \notin \mathcal{H}_\alpha \mid \mathbf{X} = \mathbf{x}] \\
= {} & e^\alpha \Pr[(\mathbf{V}, H) \in O \mid \mathbf{X} = \mathbf{x}'] + \Pr[H \notin \mathcal{H}_\alpha] && \because \text{Eq. (3)}
\end{aligned}$$

It remains to show that $\Pr[H \notin \mathcal{H}_\alpha] \leq \beta$. Recall that $x_{i^*} \neq x'_{i^*}$ are the only different values in $\mathbf{x}$ and $\mathbf{x}'$. For any hash value v and salt s define the random variables

$$\begin{aligned}
p_{s,v}(H) &= \mathbf{1}\{H(\langle j_{i^*}, s, x_{i^*}\rangle) = v\} \\
p'_{s,v}(H) &= \mathbf{1}\{H(\langle j_{i^*}, s, x'_{i^*}\rangle) = v\}
\end{aligned}$$

Also define $\bar{p}_v(H) = \frac{1}{r}\sum_s p_{s,v}(H)$ and $\bar{p}'_v(H) = \frac{1}{r}\sum_s p'_{s,v}(H)$. Observe that

$$\begin{aligned}
\Pr[V_{i^*} = v \mid H = h \wedge X_{i^*} = x_{i^*}] &= \bar{p}_v(h) \\
\Pr[V_{i^*} = v \mid H = h \wedge X_{i^*} = x'_{i^*}] &= \bar{p}'_v(h)
\end{aligned}$$

since each user chooses their salt uniformly at random from $\{1, \ldots, r\}$. Therefore

$$\Pr[H \notin \mathcal{H}_\alpha] = \Pr\left[\exists v : \frac{\bar{p}_v(H)}{\bar{p}'_v(H)} > e^\alpha\right]. \tag{6}$$

We will analyze the right-hand side of Eq. (6). Clearly $\mathrm{E}[p_{s,v}(H)] = \mathrm{E}[p'_{s,v}(H)] = \frac{1}{2}$, since each $H(\langle j, s, x\rangle)$ is chosen uniformly at random from $\{-1, +1\}$. Also $\bar{p}_v(H)$ and $\bar{p}'_v(H)$ are each the average of r independent Boolean random variables, since each $H(\langle j, s, x\rangle)$ is chosen independently. Therefore, by the Chernoff bound, for all $\varepsilon \in [0, 1]$ and any hash value v

$$\begin{aligned}
\Pr\left[\bar{p}_v(H) \geq \frac{1}{2}(1+\varepsilon)\right] &\leq \exp\left(-\frac{\varepsilon^2 r}{6}\right) \\
\Pr\left[\bar{p}'_v(H) \leq \frac{1}{2}(1-\varepsilon)\right] &\leq \exp\left(-\frac{\varepsilon^2 r}{6}\right)
\end{aligned}$$

Fix $\varepsilon = \frac{e^\alpha - 1}{e^\alpha + 1}$. Observe that if $\bar{p}_v(H) \leq \frac{1}{2}(1+\varepsilon)$ and $\bar{p}'_v(H) \geq \frac{1}{2}(1-\varepsilon)$ then

$$\frac{\bar{p}_v(H)}{\bar{p}'_v(H)} \leq \frac{1+\varepsilon}{1-\varepsilon} = e^\alpha. \tag{7}$$

Continuing from Eq. (6)

$$\begin{aligned}
\Pr\left[\exists v : \frac{\bar{p}_v(H)}{\bar{p}'_v(H)} > e^\alpha\right] &\leq \sum_v \Pr\left[\frac{\bar{p}_v(H)}{\bar{p}'_v(H)} > e^\alpha\right] \\
&\leq \sum_v \Pr\left[\bar{p}_v(H) \geq \frac{1}{2}(1+\varepsilon) \vee \bar{p}'_v(H) \leq \frac{1}{2}(1-\varepsilon)\right] && \because \text{Eq. (7)}
\end{aligned}$$

$$\begin{aligned}
&\leq \sum_v 2\exp\left(-\frac{\varepsilon^2 r}{6}\right) && \because \text{Chernoff bound}\\
&= 4\exp\left(-\frac{\varepsilon^2 r}{6}\right)\\
&\leq \beta
\end{aligned}$$

where the last line follows from $r = 6\left(\frac{e^\alpha+1}{e^\alpha-1}\right)^2 \log\frac{4}{\beta} = \frac{6}{\varepsilon^2}\log\frac{4}{\beta}$.

9 Proof of Theorem 2

Proving the sample complexity guarantee is straightforward: Since $\mathrm{E}[N_j] = m$ for each group j, the expected number of users who participate in the mechanism is $\mathrm{E}\left[\sum_{j=1}^g N_j\right] = gm = n$.

To prove the error guarantee, we need the following proposition.

Proposition 1. *Recall the definitions of a, b, m, r in Mechanism 1. If*

$$\sigma^2 = \frac{3r^2}{bm^3} + \frac{20r^2}{bm^2} + \frac{16r}{bm}C(\mathbf{p}) + \frac{2}{b}C(\mathbf{p})^2$$

then the estimate $\hat{C}$ returned by Mechanism 1 satisfies

$$|\hat{C} - C(\mathbf{p})| \leq 2\sigma$$

with probability at least $1 - \exp(-\frac{a}{8})$.

Proof. Fix group j and let $M = N_j$. Let $M_{s,x}$ be the number of users in group j who select salt s and sample x. Therefore $M = \sum_{s,x} M_{s,x}$. Let $q_{s,x} = \frac{p_x}{r}$ be the probability that a user selects salt s and sample x. Let $\mathbf{M}$ and $\mathbf{q}$ be vectors whose components are the $M_{s,x}$'s and $q_{s,x}$'s, respectively. Observe that $\mathbf{M}$ is a sample from Multinomial$(M, \mathbf{q})$. Since M is Poisson(m) distributed, we have by the 'Poissonization trick' that each $M_{s,x}$ is independent and Poisson$(mq_{s,x})$ distributed. It is well-known (Riordan, 1937) that if Z is Poisson(λ) distributed then

$$\begin{aligned}
\mathrm{E}[Z^2] &= \lambda + \lambda^2\\
\mathrm{E}[Z^4] &= \lambda + 7\lambda^2 + 6\lambda^3 + \lambda^4
\end{aligned}$$

With a slight abuse of notation, let $C(\mathbf{z}) = \sum_i z_i^2$ for any non-negative vector $\mathbf{z}$. Let $V = V_j$. In their classic paper on sketches for data streams, Alon et al. (1999) analyzed V^2 as an estimator for $C(\mathbf{M})$. In particular they showed

$$\begin{aligned}
\mathrm{E}[V^2 \mid \mathbf{M}] &= C(\mathbf{M})\\
\mathrm{Var}[V^2 \mid \mathbf{M}] &\leq 2C(\mathbf{M})^2.
\end{aligned}$$

We will extend these results to express $\mathrm{E}[V^2]$ and $\mathrm{Var}[V^2]$ in terms of $C(\mathbf{p})$. We have

$$\mathrm{E}[V^2] = \mathrm{E}[\mathrm{E}[V^2 \mid \mathbf{M}]] = \mathrm{E}[C(\mathbf{M})].$$

We express $\mathrm{E}[C(\mathbf{M})]$ as

$$\begin{aligned}
\mathrm{E}[C(\mathbf{M})] &= \sum_{s,x} \mathrm{E}[M_{s,x}^2]\\
&= \sum_{s,x} mq_{s,x} + m^2 q_{s,x}^2\\
&= m + m^2 C(\mathbf{q})\\
&= m + \frac{m^2}{r}C(\mathbf{p})
\end{aligned}$$

where the last line used

$$C(\mathbf{q}) = \sum_{s,x} q_{s,x}^2 = \frac{1}{r^2} \sum_{s,x} p_x^2 = \frac{1}{r} C(\mathbf{p}).$$

Also, by the law of total variance

$$\begin{aligned}
\mathrm{Var}[V^2] &= \mathrm{E}[\mathrm{Var}[V^2 \mid \mathbf{M}]] + \mathrm{Var}[\mathrm{E}[V^2 \mid \mathbf{M}]] \\
&\leq 2\,\mathrm{E}[C(\mathbf{M})^2] + \mathrm{Var}[C(\mathbf{M})] \\
&= 2\,\mathrm{E}[C(\mathbf{M})^2] + \mathrm{E}[C(\mathbf{M})^2] - \mathrm{E}[C(\mathbf{M})]^2 \\
&= 3\,\mathrm{E}[C(\mathbf{M})^2] - \mathrm{E}[C(\mathbf{M})]^2.
\end{aligned}$$

Both $\mathrm{E}[C(\mathbf{M})]^2$ and $\mathrm{E}[C(\mathbf{M})^2]$ can be bounded in terms of $C(\mathbf{p})$. From above we conclude

$$\mathrm{E}[C(\mathbf{M})]^2 = m^2 + \frac{2m^3}{r} C(\mathbf{p}) + \frac{m^4}{r^2} C(\mathbf{p})^2.$$

We express $\mathrm{E}[C(\mathbf{M})^2]$ as

$$\begin{aligned}
\mathrm{E}[C(\mathbf{M})^2] &= \mathrm{E}\left[\left(\sum_{s,x} M_{s,x}^2\right)^2\right] \\
&= \mathrm{E}\left[\sum_{s,x,s',x'} M_{s,x}^2 M_{s',x'}^2\right] \\
&= \mathrm{E}\left[\sum_{s,x} M_{s,x}^4 + \sum_{s \neq s' \vee x \neq x'} M_{s,x}^2 M_{s',x'}^2\right] \\
&= \sum_{s,x} \mathrm{E}[M_{s,x}^4] + \sum_{s \neq s' \vee x \neq x'} \mathrm{E}[M_{s,x}^2]\,\mathrm{E}[M_{s',x'}^2] \\
&= \sum_{s,x} m q_{s,x} + 7m^2 q_{s,x}^2 + 6m^3 q_{s,x}^3 + m^4 q_{s,x}^4 \\
&\quad + \sum_{s \neq s' \vee x \neq x'} (m q_{s,x} + m^2 q_{s,x}^2)(m q_{s',x'} + m^2 q_{s',x'}^2) \qquad (\star)
\end{aligned}$$

where we used the independence of the $M_{s,x}$'s and the moments of the Poisson distribution. By expanding and rearranging terms we have

$$\begin{aligned}
(\star) &= \sum_{s,x} m q_{s,x} + 7m^2 q_{s,x}^2 + 6m^3 q_{s,x}^3 + m^4 q_{s,x}^4 \\
&\quad + \sum_{s \neq s' \vee x \neq x'} m^2 q_{s,x} q_{s',x'} + m^3 q_{s,x}^2 q_{s',x'} + m^3 q_{s,x} q_{s',x'}^2 + m^4 q_{s,x}^2 q_{s',x'}^2 \\
&\leq \sum_{s,x} m q_{s,x} + 7m^2 q_{s,x}^2 + 6m^3 q_{s,x}^3 + m^4 q_{s,x}^4 \\
&\quad + \sum_{s \neq s' \vee x \neq x'} 7m^2 q_{s,x} q_{s',x'} + 3m^3 q_{s,x}^2 q_{s',x'} + 3m^3 q_{s,x} q_{s',x'}^2 + m^4 q_{s,x}^2 q_{s',x'}^2 \\
&= m + \sum_{s,x,s',x'} 7m^2 q_{s,x} q_{s',x'} + 3m^3 q_{s,x}^2 q_{s',x'} + 3m^3 q_{s,x} q_{s',x'}^2 + m^4 q_{s,x}^2 q_{s',x'}^2 \\
&= m + 7m^2 + 6m^3 C(\mathbf{q}) + m^4 C(\mathbf{q})^2 \\
&= m + 7m^2 + \frac{6m^3}{r} C(\mathbf{p}) + \frac{m^4}{r^2} C(\mathbf{p})^2.
\end{aligned}$$

Therefore

$$\mathrm{Var}[V^2] \leq 3\,\mathrm{E}[C(\mathbf{M})^2] - \mathrm{E}[C(\mathbf{M})]^2 \leq 3m + 20m^2 + \frac{16m^3}{r} C(\mathbf{p}) + \frac{2m^4}{r^2} C(\mathbf{p})^2.$$

Putting everything together, we have for each group j

$$\mathrm{E}[C_j] = \mathrm{E}\left[\frac{r(V_j^2 - m)}{m^2}\right] = C(\mathbf{p})$$
$$\mathrm{Var}[C_j] = \mathrm{Var}\left[\frac{r(V_j^2 - m)}{m^2}\right] \le \frac{r^2}{m^4}\mathrm{Var}[V_j^2] = \frac{3r^2}{m^3} + \frac{20r^2}{m^2} + \frac{16r}{m}C(\mathbf{p}) + 2C(\mathbf{p})^2.$$

Observe that the C_j's are all independent, because they are calculated from disjoint samples and salts, and also because a group index is prepended to each input to the hash function, causing the hash values to be independent across groups. Therefore for each supergroup ℓ

$$\mathrm{E}[\bar{C}_\ell] = \frac{1}{b}\sum_{j\in J_\ell}\mathrm{E}[C_j] = C(\mathbf{p})$$
$$\mathrm{Var}[\bar{C}_\ell] = \frac{1}{b^2}\sum_{j\in J_\ell}\mathrm{Var}[C_j] \le \frac{3r^2}{bm^3} + \frac{20r^2}{bm^2} + \frac{16r}{bm}C(\mathbf{p}) + \frac{2}{b}C(\mathbf{p})^2 = \sigma^2$$

where the last equality follows from the definition of σ^2. We know by the analysis of the median-of-means estimator (Lugosi and Mendelson, 2019) that

$$\Pr[|\hat{C} - C(\mathbf{p})| \ge 2\sigma] \le \exp\left(-\frac{a}{8}\right)$$

which proves the proposition. □

We are now ready to complete the proof of the error guarantee. From Proposition 1 we have

$$\sigma^2 = \frac{3r^2}{bm^3} + \frac{20r^2}{bm^2} + \frac{16r}{bm}C(\mathbf{p}) + \frac{2}{b}C(\mathbf{p})^2 \le \frac{23r^2}{bm^2} + \frac{16r}{bm}C(\mathbf{p}) + \frac{2}{b}C(\mathbf{p})^2.$$

Plugging $m = \frac{n}{ab}$ and $b = \frac{20}{\varepsilon_{\mathrm{rel}}^2}$ into the previous inequality yields

$$\sigma^2 \le \frac{460a^2r^2}{n^2\varepsilon_{\mathrm{rel}}^2} + \frac{16ar}{n}C(\mathbf{p}) + \frac{\varepsilon_{\mathrm{rel}}^2}{10}C(\mathbf{p})^2.$$

Plugging $n \ge \frac{1280r\log\frac{1}{\delta}}{\varepsilon_{\mathrm{rel}}^2 C(\mathbf{p})} = \frac{160ar}{\varepsilon_{\mathrm{rel}}^2 C(\mathbf{p})}$ into the previous inequality yields

$$\sigma^2 \le \frac{460\varepsilon_{\mathrm{rel}}^2}{160^2}C(\mathbf{p})^2 + \frac{\varepsilon_{\mathrm{rel}}^2}{10}C(\mathbf{p})^2 + \frac{\varepsilon_{\mathrm{rel}}^2}{10}C(\mathbf{p})^2 = \left(\frac{460}{160^2} + \frac{2}{10}\right)\varepsilon_{\mathrm{rel}}^2 C(\mathbf{p})^2.$$

Since $\exp(-\frac{a}{8}) = \delta$ we have by Proposition 1

$$|\hat{C} - C(\mathbf{p})| \le 2\sigma \le \left(2\sqrt{\frac{460}{160^2} + \frac{2}{10}}\right)\varepsilon_{\mathrm{rel}}C(\mathbf{p}) < \varepsilon_{\mathrm{rel}}C(\mathbf{p})$$

with probability at least $1 - \delta$.

10 Proof of Corollary 1

If $\alpha \le 1$ then $\frac{e^\alpha+1}{e^\alpha-1} \le O\left(\frac{1}{\alpha}\right)$ because $e^\alpha + 1 \le O(1)$ for all $\alpha \le 1$ and $1+\alpha \le e^\alpha$ for all $\alpha \in \mathbb{R}$. Also let $\varepsilon_{\mathrm{rel}} = \frac{\varepsilon}{C(\mathbf{p})}$ in the statement of Theorem 2.

11 Proof of Theorem 3

We make use of the lower bound for local differential privacy introduced by Duchi et al. (2016) which relies on a privatized version of Le Cam's two point method. Accordingly, we construct a pair of problem instances $\mathbf{p}_0$ and $\mathbf{p}_1$ for which $d_C(\mathbf{p}_0, \mathbf{p}_1) = |C(\mathbf{p}_0) - C(\mathbf{p}_1)| \geq \Omega(\tau)$ and at the same time $d_{\mathrm{KL}}(\mathbf{p}_0, \mathbf{p}_1) \in \Theta(\tau^2)$. Specifically, let

$$\mathbf{p}_0 = \left(\frac{1}{2(K-1)}, \ldots, \frac{1}{2(K-1)}, \frac{1}{2}\right) \quad \text{and} \quad \mathbf{p}_1 = \left(\frac{1-\tau}{2(K-1)}, \ldots, \frac{1-\tau}{2(K-1)}, \frac{1+\tau}{2}\right) \tag{8}$$

The KL divergence between $\mathbf{p}_0$ and $\mathbf{p}_1$ is

$$d_{\mathrm{KL}}\left(\mathbf{p}_0, \mathbf{p}_1\right) = \frac{1}{2} \log \frac{1}{1-\tau^2} = \Theta(\tau^2)$$

and the absolute difference between their collision probability is

$$d_C(\mathbf{p}_0, \mathbf{p}_1) = |C(\mathbf{p}_0) - C(\mathbf{p}_1)| = \frac{\tau}{2}\left(1 + \left(\frac{\tau}{2} - \frac{1}{2(K-1)}\right)\right) \geq \tau/2$$

Proposition 1 of Duchi et al. (2016) immediately implies the following Corollary.

Corollary 2. *Let θ be an estimator of $C(\mathbf{p})$ which receives n observations from an α-locally differential private channel Q with $\alpha \in [0, 23/35]$,* i.e.*, channel Q is a conditional probability distribution which maps each observation x_i to a probability distribution on some finite discrete domain $\mathcal{Z}$. We will denote the privatized data by $Z_i \sim Q(\cdot|x_i)$. Then for any pair of distributions $\mathbf{p}_0$ and $\mathbf{p}_1$ such that $d_C(\mathbf{p}_0, \mathbf{p}_1) \geq \tau/2$, it holds that*

$$\inf_Q \inf_\theta \sup_{\mathbf{p}} \mathbb{E}_{Q,\mathbf{p}}\left[d_C(\mathbf{p}, \theta(Z_1, \ldots, Z_n))\right] \geq \frac{\tau}{4}\left(1 - \sqrt{2\alpha^2 n d_{KL}\left(\mathbf{p}_0, \mathbf{p}_1\right)}\right)$$

Corollary 2 applied to the the pair of distribution defined in (8) with $\tau = 1/(\alpha\sqrt{n})$ implies that Mechanism 1 is minimax optimal in terms of ϵ and α by achieving a sample complexity that is $O(1/(\alpha\sqrt{n}))$.

12 Proof of Theorem 4

In the simplest version of the k-RAPPOR algorithm (Erlingsson et al., 2014), each user i with private value $x_i \in [k]$ first constructs a vector $\mathbf{v}_i \in \{0,1\}^k$ that has 1 in component x_i and 0 elsewhere, and then reports the noisy vector $\hat{\mathbf{v}}_i$ formed by independently flipping each bit in $\mathbf{v}_i$ with probability $\frac{1}{e^{\alpha/2}+1}$. Since x_i is drawn from $\mathbf{p}$, the probability that component x of $\hat{\mathbf{v}}_i$ is 1 is equal to q_x, where

$$q_x = p_x \cdot \frac{e^{\alpha/2}}{e^{\alpha/2}+1} + (1-p_x) \cdot \frac{1}{e^{\alpha/2}+1}.$$

The distribution $\tilde{\mathbf{p}} = A(x_1, \ldots, x_n)$ estimated by k-RAPPOR is defined by $\tilde{p}_x = a\hat{p}_x - b$ for all $x \in [k]$, where $\hat{\mathbf{p}}$ is the empirical average of the $\hat{\mathbf{v}}_i$'s, and $a = \frac{e^{\alpha/2}+1}{e^{\alpha/2}-1}$ and $b = \frac{1}{e^{\alpha/2}-1}$ serve to debias the noise. In other words, $\mathrm{E}[\tilde{p}_x] = p_x$ for all $x \in [k]$. For all $\alpha \in (0,1)$ this version of the k-RAPPOR algorithm is $(\alpha, 0)$-minimax optimal (Acharya et al., 2019b). Let $\mathbf{p}$ be the uniform distribution. Let $\varepsilon_x = p_x - \tilde{p}_x$. We have

$$\begin{aligned}
\mathrm{E}[|C(\tilde{\mathbf{p}}) - C(\mathbf{p})|] &= \mathrm{E}\left[\left|\sum_x \tilde{p}_x^2 - \sum_x p_x^2\right|\right] \\
&= \mathrm{E}\left[\left|\sum_x (p_x - \varepsilon_x)^2 - \sum_x p_x^2\right|\right] \\
&= \mathrm{E}\left[\left|\sum_x \varepsilon_x^2 - 2\sum_x \varepsilon_x p_x\right|\right] \\
&\geq \mathrm{E}\left[\left|\sum_x \varepsilon_x^2\right|\right] - \mathrm{E}\left[\left|2\sum_x p_x \varepsilon_x\right|\right] && \because \text{Triangle inequality}
\end{aligned}$$

$$
\begin{aligned}
&= \sum_x \mathrm{E}\left[\varepsilon_x^2\right] - 2\,\mathrm{E}\left[\left|\sum_x p_x \varepsilon_x\right|\right] \\
&\geq \sum_x \mathrm{E}\left[\varepsilon_x^2\right] - 2\,\mathrm{E}\left[\sqrt{\sum_x p_x^2}\sqrt{\sum_x \varepsilon_x^2}\right] && \because \text{Cauchy-Schwarz} \\
&= \sum_x \mathrm{E}\left[\varepsilon_x^2\right] - \frac{2}{\sqrt{k}}\,\mathrm{E}\left[\sqrt{\sum_x \varepsilon_x^2}\right] && \because \mathbf{p} \text{ is uniform} \\
&\geq \sum_x \mathrm{E}\left[\varepsilon_x^2\right] - \frac{2}{\sqrt{k}}\sqrt{\sum_x \mathrm{E}\left[\varepsilon_x^2\right]} && \because \text{Jensen's inequality} \\
&= \sum_x \mathrm{Var}\left[\tilde{p}_x\right] - \frac{2}{\sqrt{k}}\sqrt{\sum_x \mathrm{Var}\left[\tilde{p}_x\right]} && \because \text{Definition of } \varepsilon_x \text{ and } \mathrm{E}[\tilde{p}_x] = p_x \\
&= a^2 \sum_x \mathrm{Var}\left[\hat{p}_x\right] - \frac{2a}{\sqrt{k}}\sqrt{\sum_x \mathrm{Var}\left[\hat{p}_x\right]} && \because \text{Definition of } \tilde{p}_x \\
&= \frac{a^2}{n}\sum_x q_x(1-q_x) - \frac{2a}{\sqrt{kn}}\sqrt{\sum_x q_x(1-q_x)} && \because \hat{p}_x \text{ is average of } n \text{ independent samples}
\end{aligned}
$$

It is clear from the definition of q_x that

$$\frac{1}{e^{\alpha/2}+1} \leq q_x \leq \frac{e^{\alpha/2}}{e^{\alpha/2}+1}.$$

We also have $\frac{e^{\alpha/2}}{e^{\alpha/2}+1} \leq \frac{2}{3}$ and $\frac{1}{e^{\alpha/2}+1} \geq \frac{1}{3}$, since $\alpha \in (0,1)$. Therefore, continuing from above, we have

$$
\begin{aligned}
\mathrm{E}[|C(\tilde{\mathbf{p}}) - C(\mathbf{p})|] &\geq \frac{a^2}{n}\sum_x q_x(1-q_x) - \frac{2a}{\sqrt{kn}}\sqrt{\sum_x q_x(1-q_x)} && \text{From above} \\
&\geq \frac{a^2}{n}\cdot k \cdot \frac{1}{e^{\alpha/2}+1}\left(1 - \frac{e^{\alpha/2}}{e^{\alpha/2}+1}\right) - \frac{2a}{\sqrt{kn}}\sqrt{k \cdot \frac{e^{\alpha/2}}{e^{\alpha/2}+1}\left(1 - \frac{1}{e^{\alpha/2}+1}\right)} && \because \text{Bounds on } q_x \\
&\geq \frac{a^2 k}{9n} - \frac{4a}{3\sqrt{n}} && \because \alpha \in (0,1) \\
&\geq \frac{4k}{9\alpha^2 n} - \frac{8}{\alpha\sqrt{n}} && \because \text{Definition of } a
\end{aligned}
$$

where the last inequality follows because $1 + z \leq e^z \leq 1 + 2z$ and $2 \leq e^z + 1 \leq 3$ for all $z \in (0, \frac{1}{2})$. Therefore if $\mathrm{E}[|C(\tilde{\mathbf{p}}) - C(\mathbf{p})|] \leq \varepsilon$ we have

$$n \geq \frac{1}{\varepsilon}\left(\frac{4k}{9\alpha^2} - \frac{8\sqrt{n}}{\alpha}\right).$$

If $n > \frac{k^2}{1296\alpha^2}$ we are done. Otherwise $n \leq \frac{k^2}{1296\alpha^2}$, which implies $\sqrt{n} \leq \frac{k}{36\alpha}$. Replacing $\sqrt{n}$ in the above expression with this upper bound shows that $n \geq \frac{2k}{9\alpha^2\varepsilon}$, which completes the proof.

13 Proof of Theorem 5

First let us define

$$U_m = \frac{2}{m(m-1)}\sum_{i=1}^{m}\sum_{j=1}^{i-1}\mathbf{1}\left\{X_i = X_j\right\} \tag{9}$$

based on $\{X_1, \ldots, X_m\}$. The sequence $U_1, U_2, , \ldots$ are dependent sequences, since each of them depends on all previous observations, thus we shall apply a decoupling technique to obtain a martingale sequence which we can

use in a sequential test. Based on U_m, let us define

$$\bar{U}_m := \sum_{i=1}^{m} \sum_{j=1}^{i-1} g_{\mathbf{p}}(X_i, X_j)$$

with

$$g_{\mathbf{p}}(X_i, X_j) = \mathbf{1}\{X_i = X_j\} - \Pr(X_i = X_j | X_i) - \Pr(X_i = X_j | X_j) + C(\mathbf{p}) \ .$$

This decoupling technique is motivated by Theorem 8.1.1 of de la Peña and Giné (1999), since the kernel function g has became centered and degenerate, *i.e.*, $\mathrm{E}[g_{\mathbf{p}}(X_i, X_j)|X_j] = \mathrm{E}[g_{\mathbf{p}}(X_i, X_j)|X_i] = 0$, which implies that $\bar{U}_n$ is a zero-mean martingale with $n \geq 2$ as follows.

Lemma 1. *$\bar{U}_2, \bar{U}_3, \ldots$ is a discrete-time martingale the filtration of which is defined $\mathcal{F}_t = \{X_1, \ldots, X_m\}$ and for all m,*

$$\mathrm{E}[Y_m(\mathbf{p})|\mathcal{F}_{m-1}] = 0$$

where $Y_m(\mathbf{p}) = \sum_{i=1}^{m-1} g_{\mathbf{p}}(X_m, X_i)$ if $m \geq 2$ and $Y_1 = 0$.

Proof. This decoupling is motivated by Theorem 8.1.1 of de la Peña and Giné (1999). First note that $\mathrm{E}[g_{\mathbf{p}}(X_i, X_j)|X_j] = \mathrm{E}[g_{\mathbf{p}}(X_i, X_j)|X_i] = 0$ by construction. This implies that

$$\mathrm{E}[Y_m(\mathbf{p})|\mathcal{F}_{m-1}] = \sum_{i=1}^{m-1} \mathrm{E}[g_{\mathbf{p}}(X_m, X_i)|\mathcal{F}_{m-1}] = \sum_{i=1}^{m-1} \mathrm{E}[g_{\mathbf{p}}(X_m, X_i)|X_i] = 0$$

for all $m \geq 2$. Since $\bar{U}_m = \sum_{i=1}^{m} Y_i(\mathbf{p})$, it holds that

$$\mathrm{E}\left[\bar{U}_m|\mathcal{F}_{m-1}\right] = \bar{U}_{m-1} + \mathrm{E}[Y_m(\mathbf{p})|\mathcal{F}_{m-1}] = \bar{U}_{m-1} \ .$$

Finally, it is straightforward that $\mathrm{E}[|Y_m(\mathbf{p})|] < \infty$ which implies that $\bar{U}_2, \bar{U}_3, \ldots$ is a discrete-time martingale by definition. □

The empirical sequence is $\bar{u}_m = \sum_{i=1}^{m} y_m(\mathbf{p})$ with

$$y_j(\mathbf{p}) = \sum_{i=1}^{m-1} \mathbf{1}\{x_i = x_j\} - \sum_{i=1}^{m-1} p_{x_i} - (m-1)p_{x_j} + (m-1)C(\mathbf{p})$$

which is a realization of a martingale with bounded difference such that $|\bar{U}_k - \bar{U}_{k-1}| = |Y_k| \leq 4m$ and $y_1(\mathbf{p}) = 0$. However we cannot compute the empirical sequence, since the parameters of distribution are not known. As a remedy, we further decompose $\bar{U}_n$ as the sum of two sequences based on the observation that

$$\mathbb{E}[p_{X_i}] = \sum_i p_{x_i}^2 = C(\mathbf{p})$$

which implies that $\sum_{i=1}^{m}(p_{X_i} - C(\mathbf{p}))$ which is again a zero-mean martingale sequence with the same filtration $\mathcal{F}_m$ such that the difference $|p_{X_i} - C(X)| < 1$ for all i. This motivates the following decomposition of $\bar{U}_n$ as

$$Y_j(\mathbf{p}) = \underbrace{\sum_{i=1}^{j-1} \mathbf{1}\{X_i = X_j\} - 2(j-1)C(\mathbf{p})}_{T_j(\mathbf{p})} + \underbrace{2(j-1)C(\mathbf{p}) - \sum_{i=1}^{j-1} p_{X_i} - (j-1)p_{X_j}}_{E_j(\mathbf{p})}$$

Note that $T_m(\mathbf{p})$ can be computed, and it is a zero-mean martingale sequence up to an error term $E_n(\mathbf{p})$ which we cannot be computed, since the parameters of the underlying distribution $\mathbf{p}$ is not available to the tester. Also note that $T_m(\mathbf{p})$ is a centralized version of U_m defined in (9). More detailed, we have that

$$\frac{2}{m(m-1)} \sum_{i=1}^{m} T_m(\mathbf{p}) = U_m - C(\mathbf{p})$$

which means that Algorithm 2 uses the sequence of $U_1, \ldots, U_m$ as test statistic which was our point of departure. Now we will focus on $E_m(\mathbf{p})$ and how it can be upper bounded.

Further note that $E_n(\mathbf{p})$ can be again decomposed into sequence of sums of zero mean-mean terms which we can upper bound with high probability. Due to the construction, it holds that

$$\sum_{i=1}^{m} Y_i(\mathbf{p}) = \sum_{i=1}^{m} T_i(\mathbf{p}) + \sum_{i=1}^{m} E_i(\mathbf{p})$$

We can apply the time uniform confidence interval of Howard et al. (2021) to the lhs which implies that it holds that

$$\Pr\left[\forall m \in \mathbb{N} : \left|\frac{2}{m(m-1)} \sum_{i-1}^{m} Y_i(\mathbf{p})\right| \geq \phi(i,\delta)\right] \leq \delta \ . \tag{10}$$

if the data is generated from $\mathbf{p}$ where

$$\phi(i,\delta) = 1.7\sqrt{\frac{\log\log i + 0.72\log(10.4/\delta)}{i}} \ .$$

Note that the confidence interval of Howard et al. (2021) applies to the sum of discrete time martingales where each term is sub-Gaussian. This also applies to $Y_i(\mathbf{p}$, since it is a bounded random variable.

Next we upper bound $\sum_i E_i(\mathbf{p})$. For doing so, we decompose each term as

$$E_i(\mathbf{p}) = \sum_{j=1}^{i-1} \big(F_2(\mathbf{p}) - p_{X_j}\big) + (i-1)(F_2(\mathbf{p}) - p_{X_i})$$

which implies

$$\begin{aligned}
\sum_{i-1}^{m} E_i(\mathbf{p}) &= \sum_{i=1}^{m}\sum_{j=1}^{i-1} \big(F_2(\mathbf{p}) - p_{X_j}\big) + \sum_{i=1}^{m}(i-1)(F_2(\mathbf{p}) - p_{X_i}) \\
&= \sum_{i=1}^{m-1}(m-i)\big(F_2(\mathbf{p}) - p_{X_i}\big) + \sum_{i=1}^{m}(i-1)(F_2(\mathbf{p}) - p_{X_i}) \\
&= m\sum_{i=1}^{m} \big(F_2(\mathbf{p}) - p_{X_i}\big)
\end{aligned}$$

Apply the time uniform confidence interval of Howard et al. (2021) to $E_i(\mathbf{p})$, we have that

$$\Pr\left[\forall m \in \mathbb{N} : \left|\frac{2}{m(m-1)} \sum_{i-1}^{m} E_i(\mathbf{p})\right| \geq \phi(i,\delta)\right] \leq \delta \ . \tag{11}$$

Due to union bound, we can upper bound the difference of $T_i(\mathbf{p})$ and $Y_i(\mathbf{p})$ using (11) and (10) as

$$\frac{2}{m(m-1)}\left|\sum_{i=1}^{m} Y_i(\mathbf{p}) - \sum_{i=1}^{m} T_i(\mathbf{p})\right| \leq 2\phi(i,\delta/2)$$

with probability at least $1-\delta$ for all m even if m is a random variable that depends on $X_1, \ldots, X_m$. This implies that if the observations is generated from a distribution with parameters $\mathbf{p}$, then $\frac{2}{m(m-1)} T_i(\mathbf{p})$ stays close to zero, including all distribution $\mathbf{p}_0$ such that $C(\mathbf{p}_0) = c_0$. This implies the correctness of Algorithm 2.

Finally note that

$$\left|\frac{2}{m(m-1)} \sum_{i=1}^{m} Y_m(\mathbf{p}) - \frac{2}{m(m-1)} \sum_{i=1}^{m} Y_i(\mathbf{p_0})\right| = |C(\mathbf{p}) - \underbrace{C(\mathbf{p}_0)}_{=c_0}|$$

for any $\mathbf{p}_0$ such that $C(\mathbf{p}_0) = c_0$ which implies the sample complexity bound. This concludes the proof.

14 Proof of Theorem 6

Before we proof the lower bound, we need to get a better understating of the relation of the total variation distance and $d_C(\mathbf{p}, \mathbf{p}') = |C(\mathbf{p}) - C(\mathbf{p}')|$

14.1 Total variation distance

The *total variation distance* between random variables X and Y is defined

$$|X - Y| = \frac{1}{2} \sum_z |\Pr[X = z] - \Pr[Y = z]|$$

where the sum is over the union of the supports of X and Y.

Theorem 9. *For any X and Y*

$$|C(X) - C(Y)| \leq 6\,|X - Y|\,.$$

Proof. Let $z_1, z_2, \ldots$ be an enumeration of the union of the supports of X and Y. Let $p_i = \Pr[X = z_i]$ and $q_i = \Pr[Y = z_i]$.

Assume without loss of generality $C(X) \leq C(Y)$. It suffices to prove $C(Y) \leq C(X) + 6|X - Y|$. Let $\delta_i = p_i - q_i$. We have

$$\begin{aligned}
C(X) &= \sum_i p_i^2 \\
&= \sum_i (q_i + \delta_i)^2 \\
&= \sum_i q_i^2 + 2\sum_i q_i \delta_i + \sum_i \delta_i^2 \\
&\leq \sum_i q_i^2 + 2\sum_i q_i |\delta_i| + \sum_i \delta_i^2 \\
&\leq \sum_i q_i^2 + 2\sum_i |\delta_i| + \sum_i \delta_i^2 \\
&\leq \sum_i q_i^2 + 2\sum_i |\delta_i| + \sum_i |\delta_i| \\
&= C(Y) + 6|X - Y|
\end{aligned}$$

and rearranging completes the proof. □

14.2 Lower bound

Based on of Lemma A.1 due to Oufkir et al. (2021), one can lower bound the stopping time of any sequential testing algorithm in expectation. Note that this lower bound readily applies to our setup and implies a lower bound for the expected sample complexity which is

$$\frac{\log 1/3\delta}{d_{\mathrm{KL}}(\mathbf{p}, \mathbf{p}')} \tag{12}$$

where

$$d_C(\mathbf{p}, \mathbf{p}') = |C(\mathbf{p}) - C(\mathbf{p}')| = \epsilon$$

In addition to this, the following Lemma lower bounds the sensitivity of KL divergence in terms of collision probability.

Lemma 2. *For any random variables X and X' with parameters $\mathbf{p}$ and $\mathbf{p}$, it holds*

$$d_C(\mathbf{p}, \mathbf{p}')^2 \leq 18 d_{KL}(\mathbf{p}, \mathbf{p}')$$

Proof. Pinsker's inequality and Theorem 9 implies this result. □

Lemma 2 applied to (12) implies that Theorem 5 achieves optimal sample complexity, since for any distribution for which

$$C(\mathbf{p}_0) = c_0$$

and

$$d_C(\mathbf{p}, \mathbf{p}') = \epsilon$$

the expected sample complexity of any tester is lower bounded by

$$\frac{\log 1/3\delta}{\epsilon^2}$$

This concludes the proof.

15 Batch testers used in the experiments

In the experimental study, we used two batch testers as baseline. Each of these testers are based on learning algorithm which means that using a learning algorithm, the collision probability is estimated with an additive error $\epsilon/2$ and then one can decide whether the true collision probability is close to c_0 or not. This approach is caller *testing-by-learning*. In this section, we present exact sample complexity bound for these batch testers and in addition to this, we show that these approaches are optimal in minimax sense for testing collision probability for discrete distributions. In this section we present the following results:

- We start by presenting a minmax lower bound for the batch testing problem which is $\Omega(\epsilon^{-2})$. In addition, we also show that the same lower bound applies to learning.
- In Subsection 15.2, we consider two estimators, i.e. plug-in and U-statistic, and we compute their sample complexity upper bound that are of order ϵ^{-2} and they differ only in constant. These are presented in Theorem 10 and 11, respectively.
- In Subsection 15.3, we present the testing-by-learning approach and discuss that the plug-in estimator is minmax optimal on a wide range of parameters.

15.1 Lower bound for estimation and testing

To construct lower bound for estimation and testing we consider the pair of distributions defined in (8) with $\tau = \epsilon$. In this case, we obtain two distributions such that $d_{\mathrm{KL}}(\mathbf{p}_0, \mathbf{p}_1) = \Theta(\epsilon^2)$ and $d_C(\mathbf{p}_0, \mathbf{p}_1) \geq \epsilon/2$. Then estimator lower bound can be obtained based on LeCam's theorem (See Appendix 16.1) which is $\Theta(1/\epsilon^2)$ as follows.

Corollary 3. *For any estimator $\hat{\theta}_n$ for Collision probability $F_2(\mathbf{p})$ based on $n \in o(1/\epsilon^2)$, there exist a discrete distribution $\mathbf{p}$ for which*

$$\mathbb{E}_P\left[\left|\hat{\theta}_n(\mathcal{D}_n) - F_2(\mathbf{p})\right|\right] \geq C \cdot \epsilon$$

where $C > 0$ does not depend on the distribution $\mathbf{p}$.

One can show a similar lower bound for testing using Neyman-Pearson lemma. We refer the reader to Section 3.1 of Canonne (2022b) for more detail. We recall this result here with d_C.

Corollary 4. *Let f an (ϵ, δ)-tester with sample complexity n. Then for any pair of distributions $\mathbf{p}_0$ and $\mathbf{p}_1$ such that $d_C(\mathbf{p}_0, \mathbf{p}_1) = \epsilon$, it holds that*

$$1 - 2\delta \leq d_{TV}(\mathbf{p}_0^{\otimes n}, \mathbf{p}_1^{\otimes n})$$

where $p^{\otimes n}$ is the n times product distribution from $\mathbf{p}$.

Using Pinsker's inequality it results in that

$$d_{\mathrm{TV}}(\mathbf{p}_0^{\otimes n}, \mathbf{p}_1^{\otimes n})^2 \leq \frac{1}{2} d_{\mathrm{KL}}(\mathbf{p}_0^{\otimes n}, \mathbf{p}_1^{\otimes n}) = \frac{n}{2} d_{\mathrm{KL}}(\mathbf{p}_0, \mathbf{p}_1) \ .$$

Accordingly, since we already constructed a pair of distributions for which $d_2(\mathbf{p}_0, \mathbf{p}_1) = \epsilon$ and $d_{\mathrm{KL}}(\mathbf{p}_0, \mathbf{p}_1) = \Omega(\epsilon^2)$, the sample compelxity lower bound for testing is also $\Omega(1/\epsilon^2)$.

15.2 Plug-in estimator versus U-statistic estimator

The first estimator is the plug-in estimator which estimates the distribution $\mathbf{p}$ by the normalized empirical frequencies $\widehat{\mathbf{p}} := \widehat{\mathbf{p}}(\mathcal{D}_m)$ and then the estimator is computed as

$$C(\widehat{\mathbf{p}}) = F_2(\widehat{\mathbf{p}}) = \sum_{i=1}^{K} \widehat{p}_i^2$$

In this section, we will other frequency moments of discrete distributions, therefore we will use $F_k(\mathbf{p})$ as the frequency moment of order k, which is the collision probability with $k = 2$.

The plug-in estimator is well-understood in the general case via lower and upper bound that are presented in Acharya et al. (2014). Here we recall an additive error bound under Poissonization which assumes that the sample size is chosen as $M \sim \mathrm{Poi}(m)$ and the data is then $\mathcal{D}_M$.

Theorem 10. *If*

$$m \geq \max\left\{\frac{1600 F_{3/2}(\mathbf{p})^2}{\epsilon^2}, \frac{8}{\epsilon^2}\log\frac{2}{\delta}\right\} = \frac{8}{\epsilon^2}\cdot\max\left\{200\cdot F_{3/2}(\mathbf{p})^2, \log\frac{2}{\delta}\right\} \ .$$

then

$$\mathbb{P}\left(|F_2(\widehat{\mathbf{p}}(\mathcal{D}_M)) - T_2(X)| \geq \epsilon\right) \leq \delta$$

where the dataset $\mathcal{D}_M$ is sampled with sample size $M \sim Poi(m)$.

Proof. Based on Theorem 9 of Acharya et al. (2014), the bias of the estimator with Poissonization is

$$|\mathbb{E}\left[F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))\right] - T_2(X)| \leq \frac{8}{m} + \frac{10}{\sqrt{m}} F_{3/2}(\mathbf{p})$$

and its variance is

$$\mathbb{V}\left[F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))\right] \leq \frac{64}{m^3} + \frac{4\cdot 17}{\sqrt{m}} F_{7/2}(\mathbf{p}) \ .$$

Thus

$$\begin{aligned}\mathbb{P}\left(|F_2(\widehat{\mathbf{p}}(\mathcal{D}_M)) - T_2(X)| \geq \epsilon\right) &= \mathbb{P}\left(|F_2(\widehat{\mathbf{p}}(\mathcal{D}_M)) - \mathbb{E}\left[F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))\right] + \mathbb{E}\left[F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))\right] - T_2(X)| \geq \epsilon\right) \\ &\leq \mathbb{P}\left(|F_2(\widehat{\mathbf{p}}(\mathcal{D}_M)) - \mathbb{E}\left[F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))\right]| \geq \epsilon - |\mathbb{E}\left[F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))\right] - T_2(X)|\right)\end{aligned}$$

where we applied the triangle inequality. Therefore if m is big enough, then it holds that

$$\frac{8}{m} + \frac{10}{\sqrt{m}} F_{3/2}(\mathbf{p}) \leq \frac{\epsilon}{2} \tag{13}$$

and also holds

$$\mathbb{P}\left(|F_2(\widehat{\mathbf{p}}(\mathcal{D}_M)) - \mathbb{E}\left[F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))\right]| \geq \epsilon/2\right) \leq \delta \tag{14}$$

thus the statement in the theorem holds. What remains is to compute a lower bound for m. (13) holds if

$$m \geq \max\left\{\frac{32}{\epsilon}, \frac{1600 F_{3/2}(\mathbf{p})^2}{\epsilon^2}\right\} \ .$$

Based on Bernstein's inequality, see Theorem 13 in Appendix 16, (14) hold if

$$m \geq \max\left\{\frac{8\log\frac{2}{\delta}}{\epsilon^2}, \frac{4736\cdot\sqrt[4]{\log\frac{2}{\delta}}}{\sqrt{\epsilon}}, \frac{6528\sqrt[3]{F_{7/2}(\mathbf{p})\log\frac{2}{\delta}}}{\epsilon^{4/3}}\right\}$$

which concludes the proof. To simplify the last terms, alternatively we can apply Hoeffding's inequality which yields that (14) holds whenever

$$m \geq \frac{8}{\epsilon^2}\log\frac{2}{\delta} \ .$$

Finally note that $32/\epsilon \leq 8/\epsilon^2 \log 2/\delta$ for any $\epsilon, \delta \in (0, 1]$ which concludes the proof. □

Theorem 11. *If*

$$m \geq \max\left\{\frac{32(F_3(X)-F_2(X)^2)}{\epsilon^2}\ln\frac{4}{\delta}, \frac{128+1/6}{\epsilon}\ln\frac{4}{\delta}\right\}$$

then

$$\mathbb{P}\left(|F_2(X)-U(\mathcal{D}_m)|\geq\epsilon\right)\leq\delta$$

Proof. Based on Theorem 9 of Acharya et al. (2014), the bias of the estimator with Poissonization is

$$|\mathbb{E}\left[F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))\right]-T_2(X)|\leq\frac{8}{m}+\frac{10}{\sqrt{m}}F_{3/2}(\mathbf{p})$$

and its variance is

$$\mathbb{V}\left[F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))\right]\leq\frac{64}{m^3}+\frac{4\cdot17}{\sqrt{m}}F_{7/2}(\mathbf{p})\ .$$

Thus

$$\begin{aligned}\mathbb{P}\left(|F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))-T_2(X)|\geq\epsilon\right)&=\mathbb{P}\left(|F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))-\mathbb{E}\left[F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))\right]+\mathbb{E}\left[F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))\right]-T_2(X)|\geq\epsilon\right)\\&\leq\mathbb{P}\left(|F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))-\mathbb{E}\left[F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))\right]|\geq\epsilon-|\mathbb{E}\left[F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))\right]-T_2(X)|\right)\end{aligned}$$

where we applied the triangle inequality. Therefore if m is big enough, then it holds that

$$\frac{8}{m}+\frac{10}{\sqrt{m}}F_{3/2}(\mathbf{p})\leq\frac{\epsilon}{2}\tag{15}$$

and also holds

$$\mathbb{P}\left(|F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))-\mathbb{E}\left[F_2(\widehat{\mathbf{p}}(\mathcal{D}_M))\right]|\geq\epsilon/2\right)\leq\delta\tag{16}$$

thus the statement in the theorem holds. What remains is to compute a lower bound for m. (15) holds if

$$m\geq\max\left\{\frac{32}{\epsilon},\frac{1600F_{3/2}(\mathbf{p})^2}{\epsilon^2}\right\}\ .$$

Based on Bernstein's inequality, see Theorem 13 in Appendix 16, (16) hold if

$$m\geq\max\left\{\frac{8\log\frac{2}{\delta}}{\epsilon^2},\frac{4736\cdot\sqrt[4]{\log\frac{2}{\delta}}}{\sqrt{\epsilon}},\frac{6528\sqrt[3]{F_{7/2}(\mathbf{p})\log\frac{2}{\delta}}}{\epsilon^{4/3}}\right\}$$

which concludes the proof. To simplify the last terms, alternatively we can apply Hoeffding's inequality which yields that (16) holds whenever

$$m\geq\frac{8}{\epsilon^2}\log\frac{2}{\delta}\ .$$

Finally note that $32/\epsilon\leq8/\epsilon^2\log2/\delta$ for any $\epsilon,\delta\in(0,1]$ which concludes the proof. □

Note that as soon as $(F_3(X)-F_2(X)^2)/5\leq\epsilon$, the second term of the sample complexity of Theorem 11 becomes dominant, and thus the sample complexity in these parameter regime is $O(\ln(1/\delta)/\epsilon)$. In addition to this, it is easy to see that the first tern of the sample complexity is zero when X is distributed uniformly.

15.3 Testing by learning

Testing by learning consists of estimating the parameter itself with a small additive error which allows us to distinguish between null H_0 and alternative hypothesis H_1. This approach had been found to be optimal in several testing problem Busa-Fekete et al. (2021), as it is also optimal in this case based on the lower bound presented in the previous section. We considered several estimators for Collision entropy which can be used in a batch testing setup by setting the sample size so as the additive error of the estimator is smaller than $\epsilon/2$. In this way, we can distinguish between H_0 and H_1 as expected. The confidence interval of each estimator does depend on some frequency moment of the underlying distribution which can be upper worst case upper bounded. For example, the plug-n estimator sample complexity m is $1600/\epsilon^2$ if $e^{-199}\leq\delta$.

16 Technical tools

16.1 LeCam's lower bound

Let $\hat{\theta}_n = \hat{\theta}(x_1, \ldots, x_n)$ such that $\hat{\theta}_n : (\Sigma^d)^n \mapsto \mathbb{R}$ be an estimator using n samples.

Theorem 12. *[Le Cam's theorem] Let $\mathcal{P}$ be a set of distributions. Then, for any pair of distributions $P_0, P_1 \in \mathcal{P}$, we have*

$$\inf_{\hat{\theta}} \max_{P \in \mathcal{P}} \mathbb{E}_P \left[d(\hat{\theta}_n(P), \theta(P)) \right] \geq \frac{d(\theta(P_0), \theta(P_1))}{8} e^{-n d_{KL}(P_0, P_1)},$$

where $\theta(P)$ is a parameter taking values in a metric space with metric d, and $\hat{\theta}_n$ is the estimator of θ based on n samples.

16.2 Bersntein's bound

The following form of Bernstein's bound can be derived from Theorem 1.4 of Dubhashi and Panconesi (2009).

Theorem 13. *(Bernstein's bound) Let $X_1, \ldots, X_n$ be i.i.d. random variables, and $\forall i \in [n], |X_i - \mathbb{E}[X_i]| \leq b$ and $\mathbb{E}[X_i] = \mu$. Let $\sigma^2 = \mathbb{V}[X_i]$. Then with probability at least $1 - \delta$ it holds that*

$$\left| \frac{1}{n} \sum_{i=1}^{n} X_i - \mu \right| \leq \sqrt{\frac{4\sigma^2 \ln \frac{2}{\delta}}{n}} + \frac{4b \ln \frac{2}{\delta}}{3n} \quad .$$

17 Private sequential tester

We present the private sequential tester algorithm. We use the same hashing which is used in Mechanism 1. The difference is that we do not create super group but the estimate for $C(\mathbf{p})$ is computed based on all users together. That is why the hashing in Line 6 of Algorithm 3 does depend only on the salt and observed sample. Since we use hashed data, the statistics is biased. Therefore we compute the biased null hypothesis c_0 in Line 2 and we also take into account that in the hash space the support of the test statistics scales with $2r$.

Algorithm 3 Private Sequential Tester (PSQ)

1: **Given:** Null hypothesis value c_0, confidence level $\delta \in [0, 1]$, privacy parameters $\alpha \geq 0, \beta \in [0, 1]$.
2: Set $c = \frac{c_0}{2r} + 1/2$ with $r = 6 \left(\frac{e^\alpha + 1}{e^\alpha - 1} \right)^2 \log \frac{4}{\beta}$
3: **for** $i = 1, 2, 3, \ldots$ **do**
4: User i chooses salt s_i uniformly at random from $\{1, \ldots, r\}$.
5: Draw sample x_i from distribution $\mathbf{p}$.
6: User i sends hash value $v_i = h(\langle s_i, x_i \rangle)$.
7: Let $T_i = \sum_{j=1}^{i-1} \mathbf{1}\{v_i = x_j\} - 2(i-1)c$.
8: **if** $\left| \frac{2}{i(i-1)} \sum_{j=1}^{i} T_j \right| > 3.2\sqrt{2r \cdot \frac{\log\log i + 0.72 \log(20.8/\delta)}{i}}$ **then**
9: Reject the null hypothesis.
10: **end if**
11: **end for**

Proof. We use the same privatization which is applied in Mechanism 1, but super groups are not used here because we do not use the technique of Median-of-Means in Algorithm 3.

First, it be shown that

$$F_2(X) = 2r \cdot \left(F_2(V) - \frac{1}{2} \right)$$

where X is the original random variable and V is the privatized by using hashing. This implies that if the expected value of $F(X_0) = c_0$, then the test statistic of Algorithm 3 is equal to $c_0/2r + 1/2$ which is computed

in Line 2. Accordingly, our algorithm is testing whether the U-statistics is close to this biased value $c_0/2r + 1/2$ or not.

Therefore as long as we construct a confidence time-uniform interval for $F_2(V)$, this readily implies a confidence interval for $F_2(X)$. However, we need to take into account that the support of $F_2(V)$ is $[0, 2r]$ which is $O(\log \frac{1}{\beta})$. This and Theorem 5 and Theorem 2 implies the sample complexity bound. The privacy guaranty implied by Theorem 1. □