



Uniformity Testing Under User-Level Local Privacy

Clément L. Canonne   

The University of Sydney, Australia

Abigail Gentle   

The University of Sydney, Australia

Vikrant Singhal   

Harvard University, Cambridge, MA, USA

Abstract

We initiate the study of distribution testing under *user-level* local differential privacy, where each of n users contributes m samples from the unknown underlying distribution. This setting, albeit very natural, is significantly more challenging than the usual locally private setting, as for the same parameter ϵ the privacy guarantee must now apply to a full batch of m data points. While some recent work considers distribution *learning* in this user-level setting, nothing was known for even the most fundamental testing task, uniformity testing (and its generalization, identity testing).

We address this gap, by providing (nearly) sample-optimal user-level LDP algorithms for uniformity and identity testing. Motivated by practical considerations, our main focus is on the private-coin, symmetric setting, which does not require users to share a common random seed nor to have been assigned a globally unique identifier.

2012 ACM Subject Classification Security and privacy; Security and privacy \rightarrow Usability in security and privacy; Security and privacy \rightarrow Privacy protections; Theory of computation \rightarrow Theory of database privacy and security; Mathematics of computing \rightarrow Hypothesis testing and confidence interval computation

Keywords and phrases Differential Privacy, Local Differential Privacy, Uniformity Testing, Identity Testing, Hypothesis Testing, User-Level Differential Privacy, Person-Level Differential Privacy

Digital Object Identifier 10.4230/LIPIcs.ITCS.2026.33

Related Version *Full Version:* <https://arxiv.org/abs/2510.18379>

Funding *Clément L. Canonne:* Supported by an ARC DECRA (DE230101329).

Abigail Gentle: Supported by an Australian Government Research Training Program (RTP) Scholarship.

Vikrant Singhal: Supported in part by NSF grant BCS-2218803, the D3 Trustworthy AI Lab, and a grant from the Sloan foundation.

1 Introduction

We consider the problem of uniformity testing (equivalently, identity testing [29, 24]) of distributions in the setting where each of n distributed users hold m independent and identically distributed observations from some unknown common distribution. This naturally captures many real-world statistical scenarios, such as when data is distributed among many users’ personal devices.

Consider the testing equivalent of the problem specified in [16], where the goal is to learn users’ most-used emoji. In this practical deployment, users are queried once per day with a prespecified privacy budget. Of course, people typically use a lot more than a single emoji during that time period, and so one would hope to obtain information about many emojis at once, from each user. Yet, despite users “sampling” from the “distribution” of emojis multiple times per day, the $m = 1$ setting explored in earlier literature can only make use of *one* sample.



© Clément L. Canonne, Abigail Gentle, and Vikrant Singhal;
licensed under Creative Commons License CC-BY 4.0

17th Innovations in Theoretical Computer Science Conference (ITCS 2026).

Editor: Shubhangi Saraf; Article No. 33; pp. 33:1–33:24

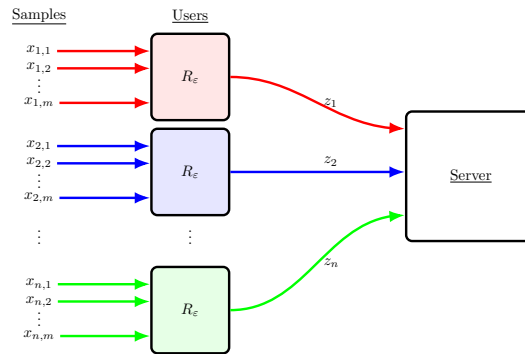
Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Can we leverage the fact that each user holds many samples to test the underlying distribution more efficiently, while still preserving each individual’s privacy as a whole?

To formalize this question, we work in the framework of Differential Privacy (DP) [26], specifically Local Differential Privacy (LDP) [33, 38], where data is made private before it leaves the device. This setting is of practical interest as data collection has grown massively, and many users are hesitant to trust a central curator with collecting and storing their data non-privately. Furthermore it has received theoretical attention as a well-parameterized model of learning under constrained information per-sample [2]. However, the usual setting of “item-level” (*i.e.*, single-sample) LDP is ill-suited to our goal, which is to capture the fact that each user can contribute many samples: naïvely, this would correspond to viewing the data of each user as an m -tuple, blowing up the domain size from k to k^m and leading to severely suboptimal algorithms. Instead, we will work in the more stringent setting of *user-level* LDP, whereby the privacy guarantee applies to the whole data held by any single user (see Figure 1 and Section 2 for an illustration and definition).



■ **Figure 1** Graphical representation of *user-level* local differential privacy. Each user holds m samples of some unknown distribution \mathbf{p} , and must be guaranteed privacy of all m samples at once. As pictured, each user sends a single private message to the server, which must somehow carry information about *all* of their samples.

As we elaborate in Section 1.1, user-level LDP is a much more stringent and challenging setting, as (roughly speaking) the algorithm’s output must remain similar even when the data of an entire user – a full batch of m samples! – is changed arbitrarily.

As our focus is on capturing practically motivated settings, we also pay special attention to the *type* of algorithm we design, and how realistic their deployment would be. For instance, *adaptive* protocols, in which users interact sequentially and adapt their output to the previous message observed, are typically undesirable, as they introduce latency and a host of technical implementation challenges.

For this reason, we consider *non-adaptive* protocols for this problem; which themselves come in several varieties. In particular a protocol can be *public-coin* or *private-coin*, and *symmetric* or *asymmetric*. A public-coin protocol assumes the existence of a shared random seed between all participants, while a private-coin protocol does not. Symmetric protocols are such that each user runs the same algorithm locally with the same parameters, while an asymmetric protocol allows some variation between users, as decided by the central data collector or curator. These models capture the level of coordination between the curator and the distributed users such that a *public-coin asymmetric* protocol is the most coordinated,

and a *private-coin symmetric* protocol is the least. For this reason we will progress through the models from most-coordination to least, as the latter model is more practical and thus preferable. Of course, if the $m = 1$ case is any indication, this practicality may come at a cost, in terms of per-user communication [9] required or overall sample complexity [2].

Can we obtain symmetric, private-coin testing algorithms for user-level LDP uniformity testing? What is the cost of achieving these two desirable features, in terms of sample and communication complexity?

To ground these distinctions and motivate the question further, one can consider the following tasks that are easy in one model, but hard in its complement. A public-coin protocol may assume that all users can apply the same randomly sampled hash function to their data, while a private-coin protocol requires that the hash is either determined in advance, or sampled independently and sent by each user to the server. An asymmetric protocol may easily partition the users into groups of equal size, while to achieve the same effect a symmetric protocol must have users randomly partition themselves and send which group they landed in. This runs into the canonical coupon collector problem, where some groups will be under-sampled, requiring a more thorough analysis, or more samples.

1.1 Overview of our Results and Techniques

In what follows, we focus on *uniformity* testing, that is, the task of testing whether the unknown distribution \mathbf{p} is uniform on the domain, or at distance at least α (in total variation distance) from uniform. As discussed in Section 1.2, by a now-standard reduction this directly implies the analogous statements for *identity* testing, where the reference distribution is a fixed, known reference \mathbf{q} instead of the uniform distribution \mathbf{u}_k .

Our setting (formally described in Section 2) involves n users, each holding $m \geq 1$ i.i.d. samples from an unknown distribution \mathbf{p} over a known domain of size $k \geq 2$. They engage in a distributed protocol, either public- or private-coin, to enable a central server to perform uniformity testing on this underlying distribution in a locally private manner, with privacy parameter $\varepsilon > 0$. Our specific focus will be on the regime $1 \leq m \leq \sqrt{k}/\alpha^2$, arguably the most natural and relevant, where each user has a possibly large number of observations, but not enough that they could run a uniformity testing algorithm by themselves:¹ we implicitly place ourselves in this regime throughout.

Our first “warm-up” result is a public-coin algorithm² which shows that, for uniformity testing, user-level LDP with m samples for each of n user behaves similarly to LDP with *one* sample for each of mn users:

¹ Indeed, if $m \gg \sqrt{k}/\alpha^2$, then each user has enough samples to perform uniformity testing by themselves, and so the question boils down to communicating the answer to the server in a locally private manner, for which $n = O(1/\varepsilon^2)$ users is sufficient. When m is even larger (specifically, at least the sample complexity of uniformity testing in the central DP model), then $n = 1$ is enough: a single user can run the DP algorithm and send the outcome.

² Throughout, and in line with the local privacy literature, we will use “algorithm” and “protocol” interchangeably.

► **Theorem 1** (See Theorem 8). *There exists an asymmetric, public-coin, user-level locally differentially private algorithm for uniformity testing (over domain of size k) which on privacy and distance parameters $\varepsilon > 0$ and $\alpha \in (0, 1]$ takes*

$$n = O\left(\frac{k}{m\alpha^2\varepsilon^2}\right)$$

users, each holding m i.i.d. samples from the unknown distribution and sending one bit of communication.

Whether this result seems unexpected or not to the reader, one interesting aspect is that this result essentially follows from combining, in a very simple way, two known algorithmic building blocks: the first is *domain compression* [6, 7], a type of hashing which, using shared randomness, allows us to reduce the domain size without increasing the total variation distance “too much”. The second is the user-level LDP coin *learning* (asymmetric) protocol of [8], which lets us learn the bias of a single Bernoulli (and so, *a fortiori*, test it). Combining the two is then rather straightforward, modulo some bookkeeping details: (1) use domain compression to reduce the uniformity testing instance from (k, α) to $(2, \alpha/\sqrt{k})$; then (2) (privately, with m samples) learn the bias of the resulting “coin” to accuracy $\alpha/(2\sqrt{k})$.

In view of the simplicity of the first algorithm, it is natural to expect the private-coin algorithm to be similarly straightforward. As we discuss shortly, this expectation turns out to be significantly optimistic: nonetheless, we are able to establish the following, providing an analogous result in the private-coin setting. (To interpret it, it is useful to remember that, for $m = 1$, the sample complexity of private-coin locally private uniformity testing is known to be $\Theta(k^{3/2}/(\alpha^2\varepsilon^2))$.)

► **Theorem 2** (1-bit user-level LDP uniformity testing). *There exists a private-coin, user-level locally differentially private algorithm for uniformity testing (over domain size k) which for small privacy parameter $\varepsilon > 0$, and distance parameter $\alpha \in (0, 1]$ takes*

$$n = O\left(\frac{k^{3/2} \log(k/r)}{m\alpha^2\varepsilon^2}\right)$$

users, each holding m i.i.d. samples from the unknown distribution. If the protocol is asymmetric, $r = \varepsilon$ and each user sends 1 bit of communication. If the protocol is symmetric, then $r = \varepsilon\alpha m$ and each user sends $O(\log k)$ bits of communication.

One may wonder whether this trade-off between symmetry and communication complexity is inherent: by a simple modification of an argument of Acharya and Sun [9] (originally in the context of (item-level) locally private distribution *learning*), we provide strong evidence that this trade-off is necessary for at least some parameter regime,³ showing that this is the case for $m = 1$:

► **Proposition 3.** *Any symmetric, private-coin algorithm for uniformity testing (over domain of size k) with distance parameter $\alpha \leq 1/k$ and $m = 1$ sample per user requires at least $\log_2 k$ bits of communication per user. (This holds regardless of whether the algorithm is locally private or not.)*

³ We cannot hope to show this trade-off for *all* values of m , since for $m \gg \sqrt{k}/\alpha^2$, as discussed before, there is a trivial, private-coin, symmetric protocol with a single bit per user.

For completeness, we give a proof of this result in the full version. While only stated for very small α , this result shows that one cannot achieve constant communication complexity per user with symmetric private-coin protocols.

Underlying our results is a technical lemma, likely of independent interest, which gives a way to compress many samples into one bit. This compression, which is likely to also find applications in the (non-private) bandwidth-constrained model, enables us to reduce the task to the much simpler task of privatizing a single bit, rather than a higher-dimensional message. Phrased in terms of differential privacy, this considerably reduces the sensitivity of our randomizers.

► **Lemma 4** (Informal statement of Lemma 12). *Let $X \sim \text{Bin}(m, 1/2 + \alpha)$ with $\alpha \in [-1/2, 1/2]$. Then, the indicator variable*

$$Y := \mathbb{1}_{\{X \geq m/2\}}$$

is distributed as $Y \sim \text{Bern}(1/2 + \beta)$ where

$$|\beta| = \Omega(\min(\sqrt{m}\alpha, 1)).$$

We briefly describe how the above lemma may lead to Theorem 2. The first (by now somewhat standard) idea is to use a good error-correcting code such as the Hadamard code to define a family of k sets⁴ $\chi_1, \dots, \chi_k \subseteq [k]$, each of size $k/2$, such that

$$\sum_{j=1}^k \left(\mathbf{p}(\chi_j) - \frac{1}{2} \right)^2 \geq d_{\text{TV}}(\mathbf{p}, \mathbf{u}_k)^2 \quad (1)$$

(note that since each set has size $k/2$, if $\mathbf{p} = \mathbf{u}_k$ then the sum is 0). Then we can partition the users into k groups, where the users of group j “monitor” χ_j : by computing how many of their m samples fall into their assigned set χ_j . That is, each user of a given group j now observes a random variable distributed as

$$\text{Bin}(m, \mathbf{p}(\chi_j)),$$

where we can rewrite $\mathbf{p}(\chi_j) = 1/2 + \alpha_j$. By (1), we then have that $\sum_{j=1}^k \alpha_j^2$ is either 0 (if \mathbf{p} is uniform) or at least α^2 (\mathbf{p} is far from it). Now, our user has a $\text{Bin}(m, \mathbf{p}(\chi_j))$ in hand, and we would like them to send a single bit: this is where Theorem 2 comes in, letting the user send the bit obtained by thresholding their observation at $m/2$. This yields a bit which has either bias 0 (if \mathbf{p} is uniform, since m is odd) or *some* bias β_j such that $\beta_j = \Omega(\min(\sqrt{m}\alpha_j, 1))$.

By piecing together (and centreing) the bits from one user of each of the k groups, we can view it as one k -dimensional random bit vector in $\{-1, 1\}^k$, with mean $(\beta_1, \dots, \beta_k)$.

This enables us to reduce our multi-sample-per-user uniformity testing problem to (privately) testing whether a *product distribution* over $\{-1, +1\}^k$ is uniform or has *mean vector* with norm at least

$$\sum_{j=1}^k \beta_j^2 \gtrsim \sum_{j=1}^k \min(m\alpha_j^2, 1). \quad (2)$$

The good news is that we now longer have to worry about user-level privacy: each user only needs to make their *one* output bit ε -LDP, which is easy to achieve via Randomized Response: by standard arguments, this only changes the mean testing problem by replacing the parameter $\beta^2 := \sum_{j=1}^k \beta_j^2$ by $O(\varepsilon^2 \beta^2)$. All that remains after that is to use an out-of-the-box (non-private) algorithm for mean testing of Rademacher-product distributions such as the ones in [20, 21], (not quite) establishing Theorem 2.

⁴ For simplicity of presentation, we implicitly assume in this overview that k is a power of 2, that m is odd, and that everything is a multiple of what it needs to be.

Are we there yet? There are still, unfortunately, a few annoying issues with the above outline. The first is the min in (2): we would *like* to lower bound $\sum_{j=1}^k \beta_j^2$ by $m \sum_{j=1}^k \alpha_j^2 \geq m\alpha^2$, but this only allows us to get $\min(m\alpha^2, 1)$, which is (much) weaker for large m . Moreover, this is actually unavoidable in our reduction to mean testing of Rademacher-products!

The key insight is that this is only an issue when some of the sets χ_j have very large bias (high or low probability) under \mathbf{p} , in which case $\sqrt{m}\alpha_j \gg 1$. We do not have control over this – it can happen! – but this should be an easy case to detect. And indeed, we can run, on a small number of users, an alternative protocol to detect whether there exists a $1 \leq j^* \leq k$ with $\alpha_{j^*} = |\mathbf{p}(\chi_{j^*}) - 1/2| \gg 1/\sqrt{m}$. By standard concentration arguments for maximum of Binomials, losing only a $\log k$ factor, we can argue that, with high probability, this will not lead to erroneously rejecting the uniform distribution ($\mathbf{p}(\chi_j) = 1/2$ for all j), but *will* detect if any of these $\mathbf{p}(\chi_j)$ is overly biased. If this test passes, then whether \mathbf{p} is uniform or not, we know that all α_j are small enough, and so $\sum_{j=1}^k \beta_j^2 \gtrsim \frac{m\alpha^2}{\log k}$.

This leads us to the second annoying issue: namely, that the above outline leads to a good sample complexity, but runs two distinct sub-protocols, one of them partitioning our n users in k distinct groups: this is very much an *asymmetric* protocol, with users running $k + 1$ distinct randomizers depending on their identity. To handle this, there is an “obvious”, general-purpose solution: let each user pick uniformly at random which of these $k + 1$ randomizers to run, and send both the (private) output and the (non-private) index of the randomizer they picked. By a coupon collector argument, this works with high probability, at the cost of a logarithmic factor in the number of users (and an $O(\log k)$ additional bits of communication per user).

As Proposition 3 asserts, the latter is necessary; the former, however, is very much wasteful. To avoid paying this logarithmic factor, we provide a generalization of the result for mean testing for Rademacher-product distributions, which allows for a different (and random) number of observations per coordinate and which we believe is of independent interest (Theorem 16). Its analysis, while intuitively simple, is rather technical and relies on a symmetrization argument by negative association: we provide it in the full version.

Finally, we observe that while our focus is privacy, we obtain novel results for the *bandwidth-constrained* setting [4] as well, where each user has limited communication budget with which to share their data.

Naïve approaches: “why did you not simply do this?” When considering this model, the first and most natural approach is to consider the composition theorems of differential privacy. Simple composition of differential states that sending T messages, each with ε -differential privacy, results in a final privacy loss of $T \cdot \varepsilon$. We use these theorems by pretending that there are nm total users, each with $m = 1$, and each guaranteed privacy $\varepsilon' = \varepsilon'(\varepsilon, m)$. Even in the most generous case when our sample complexity is some $n \propto 1/\varepsilon$, this results in no apparent gains from the additional samples over simply picking one of the m uniformly at random. In reality the situation is worse, as we frequently have $n \propto 1/\varepsilon^2$, meaning that sending m messages, each with privacy parameter ε/m will result in an *even worse* final sample complexity.

There also exists *advanced composition* [28, 27] that say that the same composition satisfies *approximate*⁵ (ε, δ) -differential privacy with final $\varepsilon' = \varepsilon\sqrt{2T \log(1/\delta)} + T\varepsilon(e^\varepsilon - 1)$. Applying this once again to some arbitrary problem with $n \propto 1/\varepsilon^2$, we see that our final sample complexity is again $nm \propto m/\varepsilon^2$, yielding no gains.

⁵ We do not further consider approximate-DP in this work, however one can informally think of δ as the “probability of not being ε -DP”.

As mentioned earlier, another simple approach exists when m is large enough for each user to (non-privately) test on their own. In this regime we have each user test locally, reporting a bit that indicates `accept` or `reject`. We then simply learn the average bit using ε -LDP, for which $n = O(1/\varepsilon^2)$ is sufficient. Therefore, for the very specific regime when $m > \sqrt{k}/\alpha^2$ and $n > 1/\varepsilon^2$, we have a final sample complexity $nm = O(\sqrt{k}/(\alpha^2\varepsilon^2))$. The existence of this approach allows us to assume $m \leq \sqrt{k}/\alpha^2$ for the remainder of this work.

A slightly less naïve approach (and why it is not enough). Another approach one could consider is one based on testing via repetition, used in [23, Section 2] for uniformity testing in the streaming setting. Assuming that $m \leq k$, we can run the following “privatized” version of this protocol. For each user, i , and each domain element $j \in [k]$, let $N_{i,j}$ the number of samples of user i that are j . Then each user computes the statistic,

$$Z_i = \frac{1}{k} \sum_{j=1}^k \mathbb{1}_{\{N_{i,j}=0\}}$$

and applies an additive ε -LDP noise mechanism. Then, the final statistic computed by the server would be the average of the private versions of all the Z_i 's, *i.e.*, the private version of $Z = \frac{1}{n} \sum_{i=1}^n Z_i$. To make each Z_i private, each user would need to add noise to Z_i that is calibrated to the sensitivity, Δ , of Z_i (*i.e.*, the effect of changing all the m samples of that user on Z_i in the worst case – if Z'_i is the new statistic after changing all of user i 's samples, then the sensitivity is the maximum possible value of $|Z_i - Z'_i|$). In this case, the sensitivity is $(m-1)/k$. The private version of Z_i is then simply $\tilde{Z}_i = Z_i + \tau_i$, where $\tau_i \sim \text{Lap}(\frac{m}{\varepsilon})$, and correspondingly the private version of Z is $\tilde{Z} = \frac{1}{n} \sum_{i=1}^n \tilde{Z}_i = Z + \frac{\tau}{n}$, where $\tau = \tau_1 + \dots + \tau_n$.

Now, by linearity of expectations, $\mathbb{E}[\tilde{Z}] = \mathbb{E}[Z] + \mathbb{E}[\tau/n] = \mathbb{E}[Z]$. Then for \mathbf{p} α -far from \mathbf{u} , following the proof of [23, Section 2] we get

$$\mathbb{E}_{\mathbf{p}}[\tilde{Z}] - \mathbb{E}_{\mathbf{u}}[\tilde{Z}] \geq \frac{m^2\alpha^2}{4ek^2} =: T$$

and, since τ and Z are independent, $\text{Var}_{\mathbf{p}}(\tilde{Z}), \text{Var}_{\mathbf{u}}(\tilde{Z}) \leq \frac{2m^2}{nk^3} + \frac{2\Delta^2}{n\varepsilon^2}$. By Chebyshev's inequality, we then get that a protocol which computes this \tilde{Z} at the server and thresholds its value at $\mathbb{E}_{\mathbf{u}}[\tilde{Z}] + T/2$ gives a valid (private, private-coin) uniformity testing algorithm as long as

$$nm^2 \gtrsim \frac{k}{\alpha^4} + \frac{k^2}{\varepsilon^2\alpha^2}.$$

Even though this is an improvement over the previously outlined methods, the sample complexity still falls short of what we are hoping for. One may hope to improve it by using techniques like the sensitivity-reducing mapping from [13, 12] that could potentially reduce the sensitivity Δ from m/k to $1/k$. However, even in the best possible case, that would only improve the sample complexity to

$$nm^2 \gtrsim \frac{k}{\alpha^4} \quad \text{and} \quad nm^4 \gtrsim \frac{k^2}{\varepsilon^2\alpha^2}.$$

As we see, even this best-case scenario would not improve the dependence on k , and be very costly for small m .

1.2 Prior work

Uniformity testing was first considered in the context of theoretical computer science by [30], and the optimal sample complexity $\Theta(\sqrt{k}/\alpha^2)$ was obtained in [35]. These results have since been generalized to the *identity* testing problem, where the reference distribution is not uniform, a problem later proven to be formally equivalent to uniformity testing [25, 29, 24]. The task has since also been considered under (differential) privacy, first in the *central* model of differential privacy [18], where the tight sample complexity was later shown to be $\Theta(\sqrt{k}/\alpha^2 + \sqrt{k}/(\alpha\sqrt{\varepsilon}) + k^{1/3}/(\alpha^{4/3}\varepsilon^{2/3}) + 1/\alpha\varepsilon)$ [10, 14].

Under the more restrictive model of *local* differential privacy, the question of testing was raised in [36] and has since been wholly resolved. It is known now that the tight sample complexity for non-interactive private-coin protocols is $\Theta(k^{3/2}/(\alpha^2\varepsilon^2))$, while non-interactive public-coin protocols achieve $\Theta(k/(\alpha^2\varepsilon^2))$ [2, 5, 1]. Furthermore, allowing interactivity cannot improve the sample complexity beyond the bound given in the public-coin setting [3, 15, 37].

In the *shuffle* model of differential privacy, where users' responses are permuted before being received by the central curator, an upper bound of $O(k^{3/4}\sqrt{\log(1/\delta)}/(\alpha\varepsilon) + \sqrt{k}/\alpha^2)$ was shown in [22]. The best known lower bounds were derived from a connection to *pan-privacy* [17].

The user-level setting is a natural generalization of any distributed statistical problem. In practice, it is restrictive to consider users holding only one sample from the distribution of interest whether we are testing, learning, or performing any other distributed task. For this reason, distribution *learning* with multiple samples was studied under *bandwidth constraints*, where an intricate interplay between the number of samples per-user and the number of bits available to communicate the samples was shown [4]. Under user-level differential privacy, these results were extended to show that (among other things) for small ε , one achieves a sample complexity for learning of $n = O(k^2/(m\alpha^2\varepsilon^2))$. Comparing this to the case when $m = 1$ (For example [11]) we can see that the risk decreases linearly with m . We could say that for the task of learning, having m samples per-user is *as beneficial as having m times more users to query*.

Even though there is no direct reduction between the two settings, and (as inferred from the previous discussion) the situation, as m grows, becomes quite subtle, this gives strong evidence that our results which show a sample complexity improvement by a factor m are tight, at least in the most relevant parameter regime for m .

Organization

The remainder of this paper is organized as follows. Section 2 establishes necessary facts and definitions we use throughout the paper, including the necessary background on differential privacy, distribution testing, and some concentration inequalities important to this work. In Section 3, we present a result for the public-coin setting, combining known theorems from distribution testing and user-level private learning. Section 4 introduces our main technical results in the form of non-private protocols that our private protocols are built upon. In particular, we show new algorithms for various testing problems, as well as several technical lemmata we believe are of independent interest. Due to the length and technical depth of this section, we defer the question of establishing privacy to Section 5, where we show how to adapt the results of the previous section to the user-level local differential privacy setting. Finally, we conclude with some open questions surrounding user-level local differential privacy. Wherever a proof, lemma, or algorithm is deferred from the main body, we include an appropriate reference to the full version.

2 Preliminaries

We use n to refer to the number of users participating in the protocol (or in the dataset) and m to refer to the number of samples (or data points) each user has. Additionally, we use k to denote the domain size, ε to denote the privacy parameter, α to denote the accuracy parameter, and β to denote the probability of failure (in terms of accuracy) of our protocol. We also write $[k] := \{1, 2, \dots, k\}$.

Next, for a distribution \mathbf{p} , $\mathbf{p}^{\otimes m}$ refers to the m -dimensional product distribution with each marginal \mathbf{p} . For two distributions \mathbf{p} and \mathbf{q} over the same (countably infinite) domain Ω , we denote by $d_{\text{TV}}(\mathbf{p}, \mathbf{q})$ the total variation distance between them, defined as

$$d_{\text{TV}}(\mathbf{p}, \mathbf{q}) = \sup_{S \subseteq \Omega} (\mathbf{p}(S) - \mathbf{q}(S)) = \frac{1}{2} \sum_{x \in \Omega} |\mathbf{p}(x) - \mathbf{q}(x)|.$$

In the context of Bernoulli and Binomial distributions with parameter p , we call their deviation from $1/2$ (i.e., $p - 1/2$) their *bias*.

Finally, we use the notation \gtrsim , \lesssim , and \asymp to denote the (sometimes slightly more convenient) analogues of the $\Omega(\cdot)$, $O(\cdot)$, and $\Theta(\cdot)$ notation: specifically, for two sequences $(a_n)_n$, $(b_n)_n$ indexed by some parameter n , we write $a_n \lesssim b_n$ if there exists $C > 0$ such that $a_n \leq C \cdot b_n$ for every $n \geq 0$, with the inequality reversed for \gtrsim . $a_n \asymp b_n$ then denotes that both $a_n \lesssim b_n$ and $a_n \gtrsim b_n$ hold. Throughout, \wedge and \vee denote minimum and maximum: $a \wedge b = \min(a, b)$, and $a \vee b = \max(a, b)$.

2.1 Differential Privacy

We first provide the necessary notions and results from the differential privacy (DP), starting with the definition of *local* differential privacy (LDP).

► **Definition 5** (Local Differential Privacy [32, 38]). *For $\varepsilon > 0$, a randomized algorithm $Q: \mathcal{X} \rightarrow \mathcal{Y}$ provides ε -Local Differential Privacy if for all $i, j \in \mathcal{X}$,*

$$\max_{y \in \mathcal{Y}} \frac{\Pr[Q(i) = y]}{\Pr[Q(j) = y]} \leq e^\varepsilon.$$

While Local Differential Privacy is somewhat involved to define for interactive protocols, where each user can send (in an adaptive manner) several messages, it is simpler in our setting. We consider non-interactive protocols, where each user only sends one message to the server. When each user holds several datapoints (that is, $\mathcal{X} = [k]^m$), the above definition then directly corresponds to the *user-level* LDP guarantee considered in this paper.

We will rely on the (binary) *Randomized Response* mechanism, which is an optimal ε -local differential privacy protocol for 1-bit inputs [33]. Formally, let Q be the local randomizer for this protocol where $Q(x)$ is a random variable, and $Q(y | x)$ is the probability of seeing output y on input x .

$$Q(y | x) = \begin{cases} \frac{e^\varepsilon}{e^\varepsilon + 1} & y = x \\ \frac{1}{e^\varepsilon + 1} & y \neq x. \end{cases}$$

2.2 Distribution and Uniformity Testing

Here, we formally state the problem and setting. We are interested in the problem of *uniformity testing* with multiple samples per user. Specifically, given a discrete distribution \mathbf{p} over k elements, which we assume without loss of generality is over the domain $[k]$, each

33:10 Uniformity Testing Under User-Level Local Privacy

user $i = 1, \dots, n$ holds a multi-sample $X_i \sim \mathbf{p}^{\otimes m}$ which is an m -dimensional vector of i.i.d. samples from \mathbf{p} . We are interested in algorithms that can distinguish between the cases where $d_{\text{TV}}(\mathbf{p}, \mathbf{u}) = 0$ and $d_{\text{TV}}(\mathbf{p}, \mathbf{u}) > \alpha$ via the fewest possible number of users n . As mentioned in the introduction, barring some kind of *information constraint* per-user, this problem reduces to the well-studied case where $m = 1$ by having each user send all m of their samples and treating each as its own message. This problem becomes non-trivial when we are either *bandwidth-constrained* or *privacy-constrained*. In the former, users have up to ℓ bits with which to communicate their samples. In the latter, the user is constrained by *differential privacy*.

In the user-level LDP setting, upon receiving n private responses Y_1, \dots, Y_n from the users, the server must distinguish between the two cases:

- $d_{\text{TV}}(\mathbf{p}, \mathbf{u}) = 0$, and
- $d_{\text{TV}}(\mathbf{p}, \mathbf{u}) > \alpha$,

with probability at least $2/3$, while satisfying ϵ -user-level LDP. (The threshold $2/3$ is somewhat arbitrary, and can be amplified to any $1 - \beta$ by standard arguments at a sample complexity cost of $O(\log(1/\beta))$.)

2.3 Useful Probability Tools

We first recall Cantelli's inequality, a one-sided version of Chebyshev's inequality:

► **Lemma 6** (Cantelli's inequality). *Let X be a real-valued random variable with finite variance. Then, for every $\lambda > 0$,*

$$\Pr[X \geq \mathbb{E}[X] + \lambda], \Pr[X \leq \mathbb{E}[X] - \lambda] \leq \frac{\text{Var}[X]}{\text{Var}[X] + \lambda^2}.$$

We will also require the following standard tail bound for subgaussian random variables (see *e.g.*, [31]):

► **Lemma 7.** *Let X_1, \dots, X_n be (not necessarily independent) σ^2 -subgaussian random variables with mean zero. Then*

$$\mathbb{E}\left[\max_{1 \leq i \leq n} X_i\right] \leq \sqrt{2\sigma^2 \log n}$$

and, for every $t > 0$,

$$\mathbb{P}\left\{\max_{1 \leq i \leq n} X_i \geq \sqrt{2\sigma^2(\log n + t)}\right\} \leq e^{-t}.$$

In particular, this implies that, for every $t > 0$,

$$\mathbb{P}\left\{\max_{1 \leq i \leq n} |X_i| \geq \sqrt{2\sigma^2(\log(2n) + t)}\right\} \leq e^{-t}.$$

3 Public-Coin via Domain Compression

In this section, as a warmup, we show how to establish our public-coin result, Theorem 1, leveraging existing algorithm building blocks, *domain compression* and *user-level coin estimation*. We restate it below:

► **Theorem 8** (LDP 1-bit asymmetric public coin uniformity testing). *There exists an asymmetric, public-coin, user-level locally differentially private algorithm for uniformity testing (over domain of size k) which on privacy and distance parameters $\varepsilon > 0$ and $\alpha \in (0, 1]$ takes*

$$n = O\left(\frac{k}{m\alpha^2\varepsilon^2}\right)$$

users, each holding m i.i.d. samples from the unknown distribution and sending one bit of communication.

Proof (Detailed sketch). The protocol works as follows: using public randomness, the n users jointly perform domain compression, *i.e.*, a type of hashing of the domain, reducing the domain size from k to 2. By the following lemma, with constant probability, this preserves the distance between distributions up to a \sqrt{k} factor:

► **Lemma 9** (Domain Compression [24]). *There exists absolute constants $c_1, c_2 > 0$ such that the following holds. For any $2 \leq \ell \leq k$ and any $\mathbf{p}, \mathbf{q} \in \Delta_k$,*

$$\Pr_{\Pi} \left[d_{\text{TV}}(\mathbf{p}_{\Pi}, \mathbf{q}_{\Pi}) \geq c_1 \sqrt{\frac{\ell}{k}} d_{\text{TV}}(\mathbf{p}, \mathbf{q}) \right] \geq c_2$$

where $\Pi = (\Pi_1, \dots, \Pi_{\ell})$ is a uniformly random partition of $[k]$ into ℓ subsets, and $\mathbf{p}_{\Pi} \in \Delta_{\ell}$ denotes the probability distribution on $[\ell]$ induced by \mathbf{p} and Π .

That is, if $\mathbf{p} = \mathbf{u}_k$, then this (always) results in a distribution uniform over a domain size 2 (*i.e.*, a fair coin), while if \mathbf{p} was α -far from uniform this results (with probability $\Omega(1)$) in a coin with bias $\alpha' = \Omega(\alpha/\sqrt{k})$.

All n users now having m samples from the same induced “coin”, all that remains is to learn the bias of this coin to accuracy $\alpha'/4$ in order to distinguish between the two cases. This can be done with the following existing protocol:

► **Theorem 10** ([4, Theorem 3.2]). *For $\varepsilon \in (0, 1]$, there exists a private-coin algorithm for person-level coin bias estimation with*

$$\mathbb{E}[(\hat{p} - p)^2] = O\left(\frac{1}{nm\varepsilon^2}\right)$$

assuming $n \geq C \cdot \log(m)/\varepsilon^2$, where $C > 0$ is an absolute constant.

Phrased differently, with the above one can privately learn the bias of a coin to an additive α' , with arbitrary (constant) probability, in the user-level LDP setting, using $n = O(1/(m\alpha^2\varepsilon^2))$ users.

Putting things together and recalling our setting of α' , the above protocol then succeeds with (small) constant probability, as long as

$$nm \gtrsim \frac{k}{\alpha^2\varepsilon^2}$$

“as claimed.” This is so far a symmetric protocol: but the probability of success, for distributions far from uniform, is quite small, as the domain compression only preserves the distances with small probability. To amplify the probability of success to $2/3$ by standard techniques, we repeat the protocol on disjoint batches of users and combine their answers via a majority vote, leading to an asymmetric protocol. Finally, note that the condition on n from Theorem 10 is indeed satisfied, as $\frac{k}{m\alpha^2\varepsilon^2} \gg \frac{\log(m)}{\varepsilon^2}$. ◀

4 Low-Bandwidth Private-Coin via Hadamard Matrices

In order to prove Theorem 2 we must first introduce our novel (non-private) algorithm for uniformity testing. Due to the technical depth of this section, and as the analysis follows straightforwardly in the private case, we defer the privacy analysis to Section 5 where we give a sketch of the proof. The complete proof of the private algorithm can be found in the full version, and largely follows the same path as the proof contained in this section, differing only in the application of binary randomized response at key points.

As discussed in Section 1, our high-level approach will proceed as follows: given n users, each holding m samples from some unknown discrete distribution \mathbf{p} over $k = 2^t$ elements, we first assign each user i a group G_j , for $2 \leq j \leq k$ (intentionally dropping a group so that we have $k - 1$ groups). As a first attempt, consider this assignment to be a deterministic function $j(i)$ that evenly partitions the n users among k groups. Later, to remove this coordination step, which would lead to an asymmetric protocol, we will instead have each users select the index j of their group uniformly at random. We describe our algorithm in stages: first, we give in Section 4.1 an asymmetric algorithm well-suited for “small” values of m , before providing a complementary approach to detect large variations from uniformity (Section 4.2), and explaining how to put them together in Section 4.3.

4.1 An algorithm for small m

Using the same indexing as for the groups, let χ_j be the set of indices where a 1 appears in the column j of the Hadamard matrix $H \in \{\pm 1\}^{k \times k}$, that is,

$$\chi_j := \{ r \in [k] : H_{rj} = +1 \}, \quad 2 \leq j \leq k. \quad (3)$$

The properties of Hadamard matrices ensure that each of its columns is a column vector of length 2^t , where half of the positions are $+1$, and the other half are -1 , and so $|\chi_j| = k/2$ for all j . Hereafter, we will write $\mathbf{p}(\chi_j)$ to denote the probability that a sample under \mathbf{p} falls in set χ_j , *i.e.*, $\mathbf{p}(\chi_j) = \sum_{r \in \chi_j} \mathbf{p}_r$. User i then computes

$$X_i = \sum_{\ell=1}^m \mathbb{1}_{\{x_\ell \in \chi_{j(i)}\}},$$

i.e., the number (out of their m samples) that lie within χ_j (the subset of the domain they are “monitoring”). One can observe the following distinction of cases:

- If $\mathbf{p} = \mathbf{u}_k$, then $X_i \sim \text{Bin}(m, 1/2)$.
- If $d_{\text{TV}}(\mathbf{p}, \mathbf{u}_k) > \alpha$, then $X_i \sim \text{Bin}(m, 1/2 + \alpha_j)$, for some α_j which will be related to α by Lemma 13.

We call α_j the *bias* observed by group G_j . User i , instead of directly sending X_i , will then compute the one-bit indicator

$$Y_i = \mathbb{1}_{\{X_i \geq \frac{m+1}{2}\}},$$

and send this to the server. We will show that, given $n = O\left(\frac{k^{3/2}}{m\alpha^2}\right)$ independently drawn samples Y_1, \dots, Y_n as above, there exists an algorithm that distinguishes between $\mathbf{p} = \mathbf{u}_k$ and \mathbf{p} α -far from \mathbf{u}_k (with probability at least $2/3$).

► **Theorem 11** (Asymmetric 1-bit multi-sample uniformity tester). *Given n users, each holding m samples of some unknown distribution \mathbf{p} on k elements. There exists an algorithm (Algorithm 1) that distinguishes between $\mathbf{p} = \mathbf{u}$ and $d_{\text{TV}}(\mathbf{p}, \mathbf{u}) > \alpha$ using*

$$n = O\left(\frac{k^{3/2}}{m\alpha^2 \wedge 1}\right),$$

samples. Moreover, each user only sends one bit.

(Note that this protocol, which follows the above outline, is asymmetric, as users are partitioned in k distinct groups of equal size, and users from different groups process their m samples differently.) This first algorithm is particularly well suited to the “small m regime” where $m \leq 1/\alpha^2$, where it achieves the desired number of users.

The rest of this subsection is dedicated to establishing Theorem 11. In view of this, we will need three ingredients: first, demonstrating that Y_i has bias $\sqrt{m}\alpha_j$ (this factor \sqrt{m} is where the dependence on m in the final bound will come from); second, showing that a distribution that is α -far from uniform induces bias in each group, such that $\sum_{j=1}^k \alpha_j^2 \approx \alpha^2$; and third, a known result on uniformity testing for Rademacher-product distributions. The proof proceeds immediately from these results: Each user sends their single bit Y_i with bias $\sqrt{m}\alpha_j$. We then take one bit from each group and arrange them into a vector. This vector simulates a sample from the product distribution with mean $\mu = (\mathbf{p}(\chi_1), \dots, \mathbf{p}(\chi_k))$. Applying Lemma 13 we get that the expected ℓ_2^2 norm of this vector is at least $m\alpha^2$. Plugging in an ℓ_2^2 -norm product tester as a black box, we conclude our proof.

As stated above, we begin by capturing the behaviour of Y_i as a function of each α_j . Recall that Y_i is an indicator variable that declares whether a binomial X_i exceeded the mean it “should” have in the uniform case: in this sense, going from X_i to Y_i converts a “many-bit” sample to a “single-bit” one. We show that for $X \sim \text{Bin}(m, 1/2 \pm \alpha)$ with small bias α , this indicator behaves as a Bernoulli with mean $\sqrt{m}\alpha$, and so this conversion from many bits to one still preserves both bias α and a dependence on m . For ease of exposition, we defer its proof to the end of the section.

► **Lemma 12** (From Many Bits, One). *Let $m = 2\ell - 1 \geq 1$ be an odd integer, and $\alpha \in [-1/2, 1/2]$. Define*

$$Y := \mathbb{1}_{\{X \geq \ell\}}$$

where $X \sim \text{Bin}(m, 1/2 + \alpha)$. Then $Y \sim \text{Bern}(1/2 + \beta)$, where

$$|\beta| = \Omega(\min(\sqrt{m}|\alpha|, 1)).$$

(Moreover, if $\alpha = 0$, then $\beta = 0$.)

While the bias α_j observed by each group is clearly distribution-dependent, we need to relate them explicitly to the distance parameter α . To do so, we use the known fact that multiplication by a Hadamard matrix is ℓ_2 -norm preserving.

► **Lemma 13** (Hadamard transform is norm-preserving). *As in the process described above, let \mathbf{p} be a distribution with $d_{\text{TV}}(\mathbf{p}, \mathbf{u}) \geq \alpha$. Let H be the $2^t \times 2^t$ Hadamard matrix with ± 1 entries. For each column $j \in \{2, 3, \dots, 2^t\}$ (excluding the all-ones column), define $\chi_j := \{i \in [k] : H_{ij} = +1\}$, and let $\alpha_j := \mathbf{p}(\chi_j) - \frac{1}{2}$ be the bias of column j under distribution \mathbf{p} . Then*

$$\sum_{j=1}^k \alpha_j^2 \geq \alpha^2.$$

33:14 Uniformity Testing Under User-Level Local Privacy

■ **Algorithm 1** Asymmetric Hadamard Protocol for Uniformity Testing.

Require: n users, each with vector $\vec{x}_i \in [k]^m$ holding m samples from distribution \mathbf{p} over $[k]$; Distance parameter α ; Hadamard matrix H of size $k \times k$

Ensure:

$\chi_j \leftarrow \{r \in [k] : H_{rj} = +1\}$ for each column $j \in [k]$

$N \leftarrow \frac{n}{k-1}$ ▷ Number of users in each group

Partition users into $k - 1$ groups G_2, \dots, G_k of size N each

for $j = 2$ to k **do**

for each user $i \in G_j$ **do**

$X_i \leftarrow \sum_{\ell=1}^m \mathbb{1}_{\{x_{i,\ell} \in \chi_j\}}$ ▷ Count samples in monitored subset

$Y_i \leftarrow \mathbb{1}_{\{X_i \geq \frac{m+1}{2}\}}$ ▷ Threshold to single bit

return Y_i

Server:

$Z_i \leftarrow 2(Y_i - 1/2)$ for all $i \in [n]$ ▷ Convert to Rademacher

for $\ell = 1$ to N **do**

Initialize $\vec{Z}^{(\ell)} \leftarrow (0, \dots, 0)$

for each group $j = 2$ to k **do**

$\vec{Z}_j^{(\ell)} \leftarrow$ next unused $Z_i \in G_j$

Run product distribution uniformity test on $\{\vec{Z}^{(1)}, \dots, \vec{Z}^{(N)}\}$ with parameter $\gamma \leftarrow \sqrt{m}\alpha \wedge 1$

return test result

(This follows, for instance, from [2, Lemma 3], combined with Cauchy–Schwarz to relate total variation and ℓ_2 distances.) For completeness, we provide a self-contained proof in the full version.

Finally, the third (and last) ingredient missing is an algorithm to test, given i.i.d. observations from a product distribution on k bits, whether the mean vector is zero (uniform distribution) or has large norm:

► **Lemma 14** (Uniformity testing on product distributions). *Fix any $d \geq 2$. There exists an algorithm that, given a parameter $\gamma \in (0, \sqrt{d}]$ and n i.i.d. samples from a product distribution \mathbf{p} on $\{-1, 1\}^d$ with $\mu := \mathbb{E}_{X \sim \mathbf{p}}[X] \in [-1, 1]^d$, has the following guarantees.*

- If $\|\mu\|_2 \leq \frac{\gamma}{2}$, the algorithm returns *accept* w.p. $\geq \frac{2}{3}$;
 - If $\|\mu\|_2 \geq \gamma$, the algorithm returns *reject* w.p. $\geq \frac{2}{3}$;
- as long as $n \geq C \frac{\sqrt{d}}{\gamma^2}$, for some absolute constant $C > 0$.

For a proof of this in the case $\gamma \in (0, 1]$, see, e.g., [20, Section 2.1] or [21, Lemma 4.2], which establish this along the way, while focusing on testing in total variation distance, or [19, Theorem 4.1], which provides slightly stronger guarantees. We note that while stated only for $\gamma \in (0, 1]$, the proofs above actually implicitly show the result for the whole range of γ . For completeness, we provide a self-contained proof (for the whole range of γ) in the full version.

With these three building blocks in hand, we are ready to analyze Algorithm 1:

Proof of Theorem 11. Recall that user i is deterministically assigned group index $j(i) \in \{2, \dots, k\}$ such that each group is of equal size.⁶ Where i is clear from context or not relevant we suppress this notation, letting $j := j(i)$. We have each user send their Y_i per Algorithm 1

⁶ The first column (and row) of the Hadamard matrix are all-ones, and can be ignored, or otherwise simulated if necessary, as any user’s behaviour in this group would be to deterministically send a 1.

and focus on the server's view. Clearly Y_i is distributed as some (yet unknown) Bernoulli $Y_i \sim \text{Bern}(\frac{1}{2} + \beta_j)$, where β_j depends on \mathbf{p} . Centering each of these, we get the Rademacher random variables

$$Z_i = 2(Y_i - 1/2) \in \{-1, 1\} \quad (1 \leq i \leq n)$$

each with mean $\mathbb{E}[Z_i] = \beta_{j(i)}$. Now, we wish to apply Lemma 14 which takes as input samples from some product distribution. To facilitate this we construct our own vector samples by concatenating one Z_i from each group. Each group G_j contains $N = n/(k-1)$ users. So, as described in Algorithm 1, we create the vectors $\{\vec{Z}^{(1)}, \dots, \vec{Z}^{(N)}\}$ so that $\vec{Z}_j^{(\ell)}$ holds the bit sent by the ℓ 'th user of group j .

Each $\vec{Z}^{(\ell)}$ therefore has mean vector $\mu = (\beta_1, \dots, \beta_k)$. Applying Lemma 12, we get that, for all $1 \leq j \leq k$, (1) if $\mathbf{p} = \mathbf{u}_k$, then $\beta_j = 0$; and (2) otherwise, we have $|\beta_j| = \Omega(\min(\sqrt{m}\alpha_j, 1))$. Combining this with Lemma 13, this vector μ satisfies:

- if $\mathbf{p} = \mathbf{u}$, then $\mu = \mathbf{0}^k$;
- if $d_{\text{TV}}(\mathbf{p}, \mathbf{u}) > \alpha$, then

$$\|\mu\|_2^2 = \sum_{j=1}^k \beta_j^2 \gtrsim \sum_{j=1}^k (m\alpha_j^2 \wedge 1) \quad (4)$$

and the RHS is at least $m\alpha^2 \wedge 1$. This is exactly the setting we need to invoke Lemma 14: setting $\gamma = \sqrt{m}\alpha \wedge 1$, and $N = n/(k-1)$, this yields a final sample complexity of $n = O\left(\frac{k^{3/2}}{m\alpha^2 \wedge 1}\right)$. ◀

Before proceeding to the next component of our algorithm, it remains to establish Lemma 12.

Proof of Lemma 12. We start with the ‘‘Moreover’’ statement: if $\alpha = 0$, then $X \sim \text{Bin}(m, 1/2)$. By symmetry, $m - X \sim \text{Bin}(m, 1/2)$, and since $m = 2\ell - 1$,

$$\Pr[X \geq \ell] = \Pr[m - X \geq \ell] = \Pr[X \leq m - \ell] = \Pr[X \leq \ell - 1]$$

from which $\Pr[X \geq \ell] = 1/2$.

Assume without loss of generality that $\alpha \geq 0$ (as otherwise we can consider $m - X$ instead). To establish the first part of the statement, we will distinguish between three cases, depending on how large α .

- First case: $\alpha \geq 2/\sqrt{m}$. By Cantelli's inequality (Lemma 6), since $\mathbb{E}[X] = \ell - 1/2 + m\alpha$,

$$\Pr[X < \ell] = \Pr\left[X < \mathbb{E}[X] - \left(m\alpha - \frac{1}{2}\right)\right] \leq \frac{\text{Var}[X]}{\text{Var}[X] + \left(m\alpha - \frac{1}{2}\right)^2} \leq \frac{1}{1 + m\alpha^2} \leq \frac{1}{5}$$

using $m\alpha > 1$ and $\text{Var}[X] \leq m/4$. This shows that $\Pr[X \geq \ell] \geq 4/5$, *i.e.*, $\beta \geq 3/10$.

- Second case: $\alpha < 3/m$.⁷ In this case, the mean of X only differs by $\alpha m < 3 = O(1)$ from $m/2$, the mean of a standard Binomial $\tilde{X} \sim \text{Bin}(m, 1/2)$ (and the modes of the two distributions are either the same or very close integers), so the change in probability mass between the two is quite subtle. We can provide a coupling between X and \tilde{X} as follows:

$$X = \tilde{X} + \tilde{Z}$$

⁷ The choice of the constant 3 in $3/m$ (instead of the more natural $1/m$) may appear somewhat arbitrary: this specific value for the cut-off will be useful, for technical reasons, in the third and last case.

33:16 Uniformity Testing Under User-Level Local Privacy

where the distribution of \tilde{Z} , conditioned on \tilde{X} , is $\tilde{Z} \sim \text{Bin}(\tilde{X}, 2\alpha)$. One can check that this satisfies $X \sim \text{Bin}(m, 1/2 + \alpha)$ (i.e., this is a valid coupling) and directly implies that $X \geq \tilde{X}$ a.s. Now, we have

$$\begin{aligned} \Pr[X \geq \ell] &\geq \Pr[\tilde{X} \geq \ell] + \Pr[\tilde{X} = \ell - 1, \tilde{Z} \geq 1] \\ &= \frac{1}{2} + \Pr[\tilde{X} = \ell - 1] \Pr[\text{Bin}(\ell - 1, 2\alpha) \geq 1] \\ &= \frac{1}{2} + \frac{\binom{2\ell-1}{\ell-1}}{2^m} \cdot (1 - (1 - 2\alpha)^{\ell-1}) \\ &= \frac{1}{2} + \Theta\left(\frac{1}{\sqrt{\ell}} \cdot \ell\alpha\right), \end{aligned}$$

using in the last step that $\alpha = O(1/\ell)$. This shows that in this case $\beta = \Theta(\sqrt{\ell}\alpha) = \Theta(\sqrt{m}\alpha)$.

- Third case: $3/m \leq \alpha < 2/\sqrt{m}$. In this regime, we can rely on the Gaussian approximation, as quantified by the Berry–Esseen theorem (see, e.g., [34, Section 11.5], which guarantees that the CDF F of the normalized version of X ,

$$X' := \frac{X - \mathbb{E}[X]}{\sqrt{\text{Var}[X]}} = \frac{2X - m(1 + 2\alpha)}{\sqrt{m(1 - 4\alpha^2)}}$$

is pointwise close to the CDF Φ of a standard Gaussian $Z \sim \mathcal{N}(0, 1)$:

$$\sup_{x \in \mathbb{R}} |F(x) - \Phi(x)| \leq \frac{C}{\sqrt{m}},$$

for some absolute constant $C > 0$ (one can take $C = 0.56$). In particular, this implies, in our case, that

$$\begin{aligned} \Pr[X < \ell] &= \Pr\left[X' < -\frac{2m\alpha - 1}{\sqrt{m(1 - 4\alpha^2)}}\right] \\ &\leq \Pr\left[X' < -\frac{\sqrt{m}\alpha}{\sqrt{1 - 4\alpha^2}}\right] && \text{(as } 2m\alpha - 1 \geq m\alpha\text{)} \\ &\leq \Pr[X' < -\sqrt{m}\alpha] \\ &\leq \Pr[Z < -\sqrt{m}\alpha] + \frac{C}{\sqrt{m}} && \text{(Berry–Esseen)} \\ &\leq \frac{1}{2} - \frac{\sqrt{m}\alpha}{5} + \frac{C}{\sqrt{m}} && \text{(Studying } \Phi \text{ for } \sqrt{m}\alpha \in [0, 2]\text{)} \\ &\leq \frac{1}{2} - \frac{1}{100}\sqrt{m}\alpha \end{aligned}$$

the last step using that $m\alpha \geq 3$ and $C \leq 0.56$. This shows that, in this regime as well, $\beta = \Omega(\sqrt{m}\alpha)$.

This concludes the distinction of cases, and the proof. ◀

4.2 An algorithm for large m

The above approach gives the “right” sample complexity under the restriction that

$$m\alpha_j^2 \leq 1, \quad \forall j \in [k],$$

■ **Algorithm 2** Symmetric protocol for large m .

Require: n users, each with vector $\vec{x}_i \in [k]^m$ holding m samples from distribution \mathbf{p} over $[k]$; Hadamard matrix H of size $k \times k$.

Ensure: With high probability detect when $\exists j^* \alpha_{j^*} > \Omega(\sqrt{\log(nk)/m})$.

$\chi_j \leftarrow \{r \in [k] : H_{rj} = +1\}$ for each column $j \in [k]$

$T \leftarrow \frac{1}{2} \sqrt{m \ln(20nk)}$

for each user $i \in [n]$ **do**

for each $j = 2$ to k **do**

$V_j^{(i)} \leftarrow \sum_{\ell=1}^m \mathbb{1}_{\{x_{i,\ell} \in \chi_j\}}$

 ▷ Count samples in each subset

if $\max_{2 \leq j \leq k} |V_j^{(i)} - \frac{m}{2}| > T$ **then**

 Set $v_i \leftarrow 1$

else

 Set $v_i \leftarrow 0$

Server:

if $\sum_{i=1}^n v_i \geq \frac{n}{2}$ **then**

▷ Majority vote

return reject

else

return accept

where $\alpha_j = \mathbf{p}(\chi_j) - \frac{1}{2}$. We here provide a different protocol, which works well when at least one of the $|\alpha_j|$'s is large. The main idea is to have each user just check *any* of the k sets χ_j receives a lot more (or less) than half of their m samples, namely, $\frac{m}{2} \pm \Omega(T)$ for some suitable threshold $T = \Theta(\sqrt{m \log(nk)})$. If \mathbf{p} is uniform, then this is highly unlikely, as by Lemma 7 the maximum deviation of k subgaussian random variables (here, Binomials) from their mean is less than T with overwhelming probability. But if \mathbf{p} is not uniform *and* one of the $|\alpha_j|$'s is large, then the number of samples falling in the corresponding set χ_j is a very biased Binomial, and for every user this set will receive $\frac{m}{2} \pm \Omega(T)$ samples with high constant probability. We formalize this idea in Algorithm 2, starting with the non-private version; and prove the following theorem:

► **Theorem 15.** *There exists an algorithm (Algorithm 2) with the following guarantees:*

- If $\mathbf{p} = \mathbf{u}_k$, then the center outputs **accept** with probability at least $9/10$;
- If there exists $1 \leq j^* \leq k$ such that $\alpha_{j^*} = \Omega(\sqrt{\log(nk)/m})$, then the center outputs **reject** with probability at least $9/10$.

Moreover, each person sends only one bit.

Proof. In what follows, we set $R := 2 \frac{T}{\sqrt{m}} = \sqrt{\ln(20nk)}$.

- If \mathbf{p} is uniform, then $\mathbf{p}(\chi_j) = 1/2$ for all j . By standard tail bounds for subgaussian random variables, this means that, for every $1 \leq i \leq n$

$$\Pr \left[\max_{1 \leq j \leq k} |V_j^{(i)} - \frac{m}{2}| > T \right] \leq \frac{1}{10n}$$

given our setting of T (specifically, we apply Lemma 7 with $t = \ln(10n)$ to the k mean-zero random variables $(V_j^{(i)} - \frac{m}{2})_j$, which are centered Binomials, and thus $\frac{m}{4}$ -subgaussian). By a union bound, we get $\Pr[\exists i, v_i = 1] \leq 1/10$, and the server outputs **accept** with probability at least $9/10$.

33:18 Uniformity Testing Under User-Level Local Privacy

- Turning to the other case, assume there exists $j^* \in [k]$ such that $|\alpha_{j^*}| > \frac{R}{\sqrt{m}}$. Then, for every person i , $\left| \mathbb{E} \left[V_{j^*}^{(i)} \right] - \frac{m}{2} \right| > R\sqrt{m}$, and

$$\max_{1 \leq j \leq k} |V_j^{(i)} - \frac{m}{2}| \geq |V_{j^*}^{(i)} - \frac{m}{2}| > T,$$

where the last inequality holds with probability at least $2/3$ by Chebyshev (as $R \geq \sqrt{\ln(20k)}$ is large enough, for large enough k). This implies that an expected $\frac{9}{10}n$ of the v_i 's will be equal to 1: more precisely, $\sum_{i=1}^n v_i \sim \text{Bin}(n, \tau)$ with $\tau \geq 9/10$. The probability that such a Binomial is less than $n/2$ is at most $1/10$ (and actually $e^{-\Omega(n)}$), and so the center will reject with probability at least $9/10$ (and actually $1 - e^{-\Omega(n)}$). ◀

4.3 Combined Algorithm for all values of m

There exists a critical regime of parameters that Algorithm 1 struggles with, leading to suboptimal sample complexity; and these are handled exactly by Algorithm 2. We can handle both regimes as follows: have the server run both protocols, the former with $n_1 = n - 1$ and the latter with $n_2 = 1$ users, and return **accept** if, and only if, both of them return **accept**. Assume

$$n \gtrsim \frac{k^{3/2} \log k}{m\alpha^2} \vee k.$$

Then, we have the following distinction of cases:

- If $\mathbf{p} = \mathbf{u}_k$, then both protocols **accept** with probability at least $9/10$ each, so overall the center returns **accept** with probability at least $8/10$;
- If $d_{\text{TV}}(\mathbf{p}, \mathbf{u}_k) > \alpha$, then
 - If there exists $j^* \in [k]$ such that

$$\alpha_{j^*} = \Omega\left(\sqrt{\log(k)/m}\right)$$

then, by Theorem 15, the second protocol (with one user) outputs **reject** with probability at least $9/10$, in which case the center outputs **reject**.

- Otherwise, then by (4) the mean of the product distribution in the first protocol is at least

$$\|\mu\|_2^2 \gtrsim \sum_{j=1}^k (m\alpha_j^2 \wedge 1) \geq \sum_{j=1}^k \left(\frac{m\alpha_j^2}{\log k} \wedge 1 \right) \gtrsim \frac{m\alpha^2}{\log k} \quad (5)$$

since $m\alpha_j^2 \lesssim \log k$ for all j , and $\sum_{j=1}^k \alpha_j^2 \geq \alpha^2$. Then, concluding as in Section 4 by invoking the uniformity testing algorithm for product distributions of Lemma 14 (which also handles the full range of distance parameter $\gamma^2 := \frac{m\alpha^2}{\log k} \in (0, k]$), the server rejects with probability at least $9/10$, as

$$\frac{n}{k} \gtrsim \frac{k^{1/2}}{\gamma^2} \vee 1.$$

Either way, at least one of the two tests outputs **reject** with probability $9/10$, and so does the center.

4.4 Symmetric Protocols via Generalized Product Testing

The above protocol is asymmetric in that, as stated, it needs users to be divided into k groups. Of these, $k - 1$ groups will run Algorithm 1, each with a different column. The last group (of only 1 user) runs Algorithm 2.

To resolve the asymmetry in Algorithm 1 we prove the following statement, which covers the setting we require to make our protocol symmetric: each of $n' := nd$ users independently selects a random coordinate and reports a sample from that coordinate. Formally, observations $(X_1, j_1) \dots, (X_{n'}, j_{n'})$ are obtained by choosing, independently for each $i \in [n']$, $j_i \in [d]$ uniformly at random, and $X_i \sim \mathbf{p}_{j_i}$. In this case, the numbers of times n_1, \dots, n_d each coordinate $j \in [d]$ is sampled are (correlated) $\text{Bin}(nd, 1/d)$ random variables.

► **Theorem 16.** *There exists an algorithm (available in the full version) which, given parameters $\gamma \in (0, \sqrt{d}]$, $n \geq 1$, and sample access to distributions $\mathbf{p}_1, \dots, \mathbf{p}_d$ on $\{-1, 1\}$, chooses n_1, \dots, n_d at random from a multinomial distribution with parameters $n' := nd$, d , and $(1/d, \dots, 1/d)$; and then is given n_j i.i.d. samples from each \mathbf{p}_j (where the samples are independent from the choice of n_i 's). Then, letting $\mu := \mathbb{E}_{X \sim \mathbf{p}_1 \otimes \dots \otimes \mathbf{p}_d}[X] \in [-1, 1]^d$, it has the following guarantees.*

- If $\|\mu\|_2 \leq \frac{\gamma}{2}$, the algorithm returns *accept* w.p. $\geq \frac{2}{3}$;
 - If $\|\mu\|_2 \geq \gamma$, the algorithm returns *reject* w.p. $\geq \frac{2}{3}$;
- as long as $n \geq C \frac{\sqrt{d}}{\gamma^2} \vee 1$ for some absolute constant $C > 0$.

We defer the proof to the full version.

As Theorem 16 takes exactly the same parameters, and has exactly the same guarantees as Lemma 14, we do not restate the proof of Theorem 11. We need only note that having each user randomly sample their own index j and send it to the center is indeed distributed as described above. This of course increases communication to $O(\log k)$ bits.

This alone does not make the entire protocol symmetric. As we said, we still have one user assigned to running Algorithm 2. However, this is easily addressed. At the cost of one extra bit of communication per user, we can simply have *all* users run this second test, and then let the center select arbitrarily one of the n outcomes to use. This finally gives a (non-private) algorithm, and can be summarized as follows:

► **Theorem 17** (Symmetric private-coin uniformity testing). *There exists a symmetric, private-coin (non-private) algorithm for uniformity testing (over domain size k) which on distance parameter $\alpha \in (0, 1]$ takes*

$$n = O\left(\frac{k^{3/2} \log k}{m\alpha^2} \vee k\right)$$

users, each holding m i.i.d. samples from the unknown distribution, and sending $O(\log k)$ bits of communication.

Note that the first term dominates when $m \leq O(\sqrt{k}/\alpha^2)$, which is the regime of interest (as otherwise a single user has enough samples to run a uniformity testing algorithm by themselves).

Of course, this result may still be somewhat underwhelming, as (albeit communication-efficient) the algorithm does not provide any privacy guarantee. In the next section, we will see how it can be easily adapted to yield our main result, Theorem 2.

5 Symmetric Locally Private Testing

We here sketch the analysis of the private analogues to the algorithms defined in Section 4, thus completing the proof of Theorem 2. We focus on making each of our two algorithms private before showing how they can be combined. In each case we will apply binary randomized response [33] to the bit returned by each user. As discussed before, the details of the analysis are provided in the full version of the paper.

Combining the two private algorithms gives us an asymmetric and symmetric protocol, each with sample complexity comparable to the $m = 1$ case with m times as many users, up to logarithmic factors.

Private Algorithm 1 for small m . We first introduce the private version of Algorithm 1, which handles the case when m is small. As described above, we make this algorithm private by an application of binary randomized response to the bit returned by each user. Consider the proof of Theorem 17, which itself follows the proof of Theorem 11. By applying randomized response to the bit sent by each user, one can quickly derive that each vector coordinate $\bar{Z}_j^{(\ell)}$ is now distributed as a Rademacher with mean $|\beta_j| = \Omega\left(2\frac{e^\varepsilon-1}{e^\varepsilon+1}(\sqrt{m}\alpha_j \wedge 1)\right)$. Thereafter, applying the same steps as in the proof of Theorem 11 yields an algorithm with sample complexity

$$n = O\left(\frac{k^{3/2}}{\varepsilon^2(m\alpha^2 \wedge 1)}\right).$$

Noting that as ε grows, this rapidly converges to the non-private result.

Private Algorithm 2 for large m . Recall that the second stage of our algorithm, defined in Section 4.2 handles the case when one of the subsets defined by the Hadamard matrix is overrepresented. Non-privately we only require that a single user perform this test for all of the subsets and report their response. Under local differential privacy this would not have any high-probability guarantee. Instead we use a known (and easily derived) bound for learning coins under binary randomized response. Specifically, that one can learn a Bernoulli through randomized response up to additive error α with success probability $2/3$ using $n = O(1/(\alpha^2\varepsilon^2))$ samples. This yields the following lemma:

► **Lemma 18** (Private version of Theorem 15). *Algorithm 2, when each user applies binary randomized response to their output, has the following guarantees:*

- If $\mathbf{p} = \mathbf{u}_k$, then the server outputs *accept* with probability at least $\frac{2e^\varepsilon+1}{3(e^\varepsilon+1)}$
- If there exists $1 \leq j^* \leq k$ such that $\alpha_{j^*} = \Omega\left(\sqrt{\log(nk)/m}\right)$, then the server outputs *reject* with probability at least $\frac{2e^\varepsilon+1}{3(e^\varepsilon+1)}$

Analysis of the Combined Algorithm. As in the non-private case, we have to consider how to combine these two algorithms. First, we consider the “easy” asymmetric case; allocating n_1 users to run the private analogue to Algorithm 1, we then only need $n_2 = O(1/\varepsilon^2)$ users to run the second protocol. As the number of users required by the second protocol is clearly dominated by the first, we retain much the same sample complexity up to a logarithmic factor. We defer the proof to the full version, however it suffices to say that we derive as a final sample complexity for the asymmetric algorithm

$$n \gtrsim \frac{k^{3/2}}{m\alpha^2(\varepsilon^2 \vee 1)} \log\left(\frac{k}{\varepsilon \vee 1}\right).$$

To make the protocol symmetric we can follow the same procedure described in Section 4.4 and have *all* users run the second protocol. This incurs two costs, (1) we must divide the privacy budget between these two protocols, and (2) we lose a logarithmic n factor in the final sample complexity.

Taking the path of least resistance, we assume that each user runs the private version of Algorithm 1 with privacy parameter $\varepsilon_1 = \varepsilon/2$, and likewise runs Algorithm 2 with $\varepsilon_2 = \varepsilon/2$. As such, we have $n_2 = n_1 = n$ and so gain an n_2 term in the log of Lemma 18. Setting n_2 to the sample complexity derived for the private analogue of Algorithm 2, one can see that we require

$$n \gtrsim \frac{k^{3/2}}{m\alpha^2(\varepsilon^2 \vee 1)} \log\left(\frac{k}{m\alpha(\varepsilon \vee 1)}\right).$$

users. Combining both bounds completes the proof of Theorem 2.

6 Conclusion

User-level locally private distribution testing is still far from being understood. We observe many phase transitions as ε and m vary. Consider the algorithm for testing in the central model of differential privacy discussed in Section 1.2, when m exceeds the stated sample complexity we see that the required number of “users” n goes to 1, and only 1 bit of communication is needed.

How exactly do these algorithms behave as a sliding scale between the local and central models of differential privacy? Characterizing the behaviour in each regime is an ongoing and important field of research.

Future work. Data generation is not always homogeneous: the distributions that users sample from are not truly identical; rather, it is more likely they are sampling from n distributions that could be similar or very far apart. User-level locally private distribution learning under limited heterogeneity is touched upon in [8], but we mirror their remark that this deserves further study.

A further heterogeneity that should be considered is the case when not all users hold the same number of samples m . In this case each may hold m_i for each $i \in [n]$. It is not at all obvious how this could be handled neatly, and general results for this model could greatly help practical implementations.

References

- 1 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 Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, pages 2067–2076. PMLR, 16–18 April 2019. URL: <http://proceedings.mlr.press/v89/acharya19b.html>.
- 2 Jayadev Acharya, Clément L. Canonne, Cody Freitag, Ziteng Sun, and Himanshu Tyagi. Inference under information constraints III: local privacy constraints. *IEEE J. Sel. Areas Inf. Theory*, 2(1):253–267, 2021. doi:10.1109/JSAIT.2021.3053569.
- 3 Jayadev Acharya, Clément L. Canonne, Yuhan Liu, Ziteng Sun, and Himanshu Tyagi. Interactive inference under information constraints. *CoRR*, abs/2007.10976, 2020. arXiv:2007.10976.
- 4 Jayadev Acharya, Clément L. Canonne, Yuhan Liu, Ziteng Sun, and Himanshu Tyagi. Distributed estimation with multiple samples per user: Sharp rates and phase transition. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, 2021.

- 5 Jayadev Acharya, Clément L. Canonne, and Himanshu Tyagi. Inference under information constraints I: Lower bounds from chi-square contraction. *IEEE Trans. Inform. Theory*, 66(12):7835–7855, 2020. Preprint available at arXiv:abs/1812.11476. doi:10.1109/TIT.2020.3028440.
- 6 Jayadev Acharya, Clément L. Canonne, and Himanshu Tyagi. Inference under information constraints II: Communication constraints and shared randomness. *IEEE Transactions on Information Theory*, 2020. In press. Preprint available at arXiv:abs/1804.06952. doi:10.1109/TIT.2020.3028439.
- 7 Jayadev Acharya, Clément L. Canonne, Yanjun Han, Ziteng Sun, and Himanshu Tyagi. Domain compression and its application to randomness-optimal distributed goodness-of-fit. In Jacob Abernethy and Shivani Agarwal, editors, *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pages 3–40. PMLR, 09–12 July 2020. URL: <http://proceedings.mlr.press/v125/acharya20a.html>.
- 8 Jayadev Acharya, Yuhan Liu, and Ziteng Sun. Discrete distribution estimation under user-level local differential privacy. In *AISTATS*, volume 206 of *Proceedings of Machine Learning Research*, pages 8561–8585. PMLR, 2023. URL: <https://proceedings.mlr.press/v206/acharya23a.html>.
- 9 Jayadev Acharya and Ziteng Sun. Communication complexity in locally private distribution estimation and heavy hitters. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 51–60, Long Beach, California, USA, 09–15 June 2019. PMLR. URL: <http://proceedings.mlr.press/v97/acharya19c.html>.
- 10 Jayadev Acharya, Ziteng Sun, and Huanyu Zhang. Differentially private testing of identity and closeness of discrete distributions. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 6878–6891. Curran Associates, Inc., 2018. URL: <http://papers.nips.cc/paper/7920-differentially-private-testing-of-identity-and-closeness-of-discrete-distributions.pdf>.
- 11 Jayadev Acharya, Ziteng Sun, and Huanyu Zhang. Hadamard response: Estimating distributions privately, efficiently, and with little communication. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, pages 1120–1129. PMLR, 16–18 April 2019. URL: <http://proceedings.mlr.press/v89/acharya19a.html>.
- 12 Maryam Aliakbarpour, Arnav Burudgunte, Clément L. Canonne, and Ronitt Rubinfeld. Better private distribution testing by leveraging unverified auxiliary data. In *COLT*, volume 291 of *Proceedings of Machine Learning Research*, pages 22–63. PMLR, 2025. URL: <https://proceedings.mlr.press/v291/aliakbarpour25a.html>.
- 13 Maryam Aliakbarpour, Ilias Diakonikolas, Daniel Kane, and Ronitt Rubinfeld. Private testing of distributions via sample permutations. In *NeurIPS*, pages 10877–10888, 2019. URL: <https://proceedings.neurips.cc/paper/2019/hash/8e036cc193d0af59aa9b22821248292b-Abstract.html>.
- 14 Maryam Aliakbarpour, Ilias Diakonikolas, and Ronitt Rubinfeld. Differentially private identity and equivalence testing of discrete distributions. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 169–178, Stockholmsmässan, Stockholm Sweden, 10–15 July 2018. PMLR. URL: <http://proceedings.mlr.press/v80/aliakbarpour18a.html>.
- 15 Kareem Amin, Matthew Joseph, and Jieming Mao. Pan-private uniformity testing. In Jacob Abernethy and Shivani Agarwal, editors, *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pages 183–218. PMLR, 09–12 July 2020. URL: <http://proceedings.mlr.press/v125/amin20a.html>.
- 16 Apple Differential Privacy Team. Learning with Privacy at Scale, 2017. URL: <https://machinelearning.apple.com/research/learning-with-privacy-at-scale>.

- 17 Victor Balcer, Albert Cheu, Matthew Joseph, and Jieming Mao. Connecting robust shuffle privacy and pan-privacy. *CoRR*, abs/2004.09481, 2020. [arXiv:2004.09481](https://arxiv.org/abs/2004.09481).
- 18 Bryan Cai, Constantinos Daskalakis, and Gautam Kamath. Priv'it: Private and sample efficient identity testing. In *Proceedings of the 34th International Conference on Machine Learning*, ICML '17, pages 635–644. JMLR, Inc., 2017. URL: <http://proceedings.mlr.press/v70/cai17a.html>.
- 19 Clément L. Canonne, Xi Chen, Gautam Kamath, Amit Levi, and Erik Waingarten. Random restrictions of high dimensional distributions and uniformity testing with subcube conditioning. In *SODA*, pages 321–336. SIAM, 2021. doi:10.1137/1.9781611976465.21.
- 20 Clément L. Canonne, Ilias Diakonikolas, Daniel M. Kane, and Alistair Stewart. Testing Bayesian Networks. In Satyen Kale and Ohad Shamir, editors, *Proceedings of the 2017 Conference on Learning Theory*, volume 65 of *Proceedings of Machine Learning Research*, pages 370–448, Amsterdam, Netherlands, 07–10 July 2017. PMLR. URL: <http://proceedings.mlr.press/v65/canonne17a.html>.
- 21 Clément L. Canonne, Gautam Kamath, Audra McMillan, Jonathan Ullman, and Lydia Zakynthinou. Private identity testing for high-dimensional distributions. In *Advances in Neural Information Processing Systems 33*, 2020. To appear. Preprint available at [arXiv:abs/1905.11947](https://arxiv.org/abs/1905.11947).
- 22 Clément L. Canonne and Hongyi Lyu. Uniformity testing in the shuffle model: Simpler, better, faster. In Karl Bringmann and Timothy M. Chan, editors, *5th Symposium on Simplicity in Algorithms, SOSA@SODA 2022, Virtual Conference, January 10-11, 2022*, pages 182–202. SIAM, 2022. doi:10.1137/1.9781611977066.13.
- 23 Clément L. Canonne and Joy Qiping Yang. Simpler distribution testing with little memory. In Merav Parter and Seth Pettie, editors, *2024 Symposium on Simplicity in Algorithms, SOSA 2024, Alexandria, VA, USA, January 8-10, 2024*, pages 406–416. SIAM, 2024. doi:10.1137/1.9781611977936.37.
- 24 Clément L. Canonne. Topics and techniques in distribution testing: A biased but representative sample. *Foundations and Trends® in Communications and Information Theory*, 19(6):1032–1198, 2022. doi:10.1561/0100000114.
- 25 Ilias Diakonikolas and Daniel M. Kane. A new approach for testing properties of discrete distributions. In *57th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2016*. IEEE Computer Society, 2016. doi:10.1109/FOCS.2016.78.
- 26 Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography*, volume 3876 of *Lecture Notes in Comput. Sci.*, pages 265–284. Springer, Berlin, 2006. doi:10.1007/11681878_14.
- 27 Cynthia Dwork and Aaron Roth. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3-4):211–407, 2013. doi:10.1561/04000000042.
- 28 Cynthia Dwork, Guy N. Rothblum, and Salil Vadhan. Boosting and Differential Privacy. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 51–60, 2010. ISSN: 0272-5428. doi:10.1109/FOCS.2010.12.
- 29 Oded Goldreich. The uniform distribution is complete with respect to testing identity to a fixed distribution. *Electronic Colloquium on Computational Complexity (ECCC)*, 23:15, 2016. URL: <http://eccc.hpi-web.de/report/2016/015>.
- 30 Oded Goldreich and Dana Ron. On testing expansion in bounded-degree graphs. Technical Report TR00-020, Electronic Colloquium on Computational Complexity (ECCC), 2000.
- 31 Clement C. (<https://math.stackexchange.com/users/75808/clement> c). Tail bounds for maximum of sub-gaussian random variables. Mathematics Stack Exchange, 2023. URL:<https://math.stackexchange.com/q/4002713> (version: 2023-12-21). [arXiv:https://math.stackexchange.com/q/4002713](https://arxiv.org/abs/https://math.stackexchange.com/q/4002713).
- 32 Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What can we learn privately? *SIAM J. Comput.*, 40(3):793–826, 2011. doi:10.1137/090756090.

- 33 Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam D. Smith. What can we learn privately? *SIAM J. Comput.*, 40(3):793–826, 2011. doi:10.1137/090756090.
- 34 Ryan O’Donnell. *Analysis of Boolean functions*. Cambridge University Press, New York, 2014. doi:10.1017/CB09781139814782.
- 35 Liam Paninski. A coincidence-based test for uniformity given very sparsely sampled discrete data. *IEEE Transactions on Information Theory*, 54(10):4750–4755, 2008. doi:10.1109/TIT.2008.928987.
- 36 Or Sheffet. Locally private hypothesis testing. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4612–4621, Stockholmsmässan, Stockholm Sweden, 10–15 July 2018. PMLR. URL: <http://proceedings.mlr.press/v80/sheffet18a.html>.
- 37 Berrett Thomas and Butucea Cristina. Locally private non-asymptotic testing of discrete distributions is faster using interactive mechanisms. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS ’20, pages 3164–3173, Red Hook, NY, USA, 2020. Curran Associates Inc.
- 38 Stanley L. Warner. Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 60(309):63–69, 1965. doi:10.1080/01621459.1965.10480775.