The Limits of Pan Privacy and Shuffle Privacy for Learning and Estimation*

Albert Cheu
cheu.a@northeastern.edu
Khoury College, Northeastern University
Boston, Massachusetts, USA

Jonathan Ullman
jullman@ccs.neu.edu
Khoury College, Northeastern University
Boston, Massachusetts, USA

ABSTRACT

There has been a recent wave of interest in intermediate trust models for differential privacy that eliminate the need for a fully trusted central data collector, but overcome the limitations of local differential privacy. This interest has led to the introduction of the shuffle model (Cheu et al., EUROCRYPT 2019; Erlingsson et al., SODA 2019) and revisiting the pan-private model (Dwork et al., ITCS 2010). The message of this line of work is that, for a variety of low-dimensional problems—such as counts, means, and histograms—these intermediate models offer nearly as much power as central differential privacy. However, there has been considerably less success using these models for high-dimensional learning and estimation problems.

In this work we prove the first non-trivial lower bounds for high-dimensional learning and estimation in both the pan-private model and the general multi-message shuffle model. Our lower bounds apply to a variety of problems—for example, we show that, private agnostic learning of parity functions over d bits requires $\Omega(2^{d/2})$ samples in these models, and privately selecting the most common attribute from a set of d choices requires $\Omega(d^{1/2})$ samples, both of which are exponential separations from the central model.

CCS CONCEPTS

• **Theory of computation → Distributed algorithms**; • **Mathematics of computing → Information theory**; • **Security and privacy → Privacy-preserving protocols**.

KEYWORDS

Differential Privacy, Pan-Privacy, Shuffle Model, Lower Bounds

ACM Reference Format:
Albert Cheu and Jonathan Ullman. 2021. The Limits of Pan Privacy and Shuffle Privacy for Learning and Estimation. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing (STOC '21), June 21–25, 2021, Virtual, Italy*. ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3406325.3450995

*The authors were supported by NSF grants CCF-1750640, CNS-1816028, CNS-1916020. A full version of this work can be found at https://arxiv.org/abs/2009.08000.

STOC '21, June 21–25, 2021, Virtual, Italy

ACM ISBN 978-1-4503-8053-9/21/06…$15.00
https://doi.org/10.1145/3406325.3450995

1 INTRODUCTION

The most widely accepted way to ensure individual privacy in the context of statistics and machine learning is *differential privacy* [23], which provides a strong guarantee that no individual user's data has a strong influence on the *output* of the computation that are visible to the attacker. Differentially private algorithms, however, are designed for a variety of different *trust models* that determine what output is visible. The strongest, and most commonly studied trust model is the *central model*, in which a single party is entrusted to collect raw data from the users, runs a differentially private computation, and only the final output of this computation is visible. On the other extreme, the weakest trust model is the *local model* [40], where we don't trust anyone to safeguard raw data, so each user applies differential privacy locally to their own data to compute a response, and each user's response is visible. While the central model allows for many powerful algorithms, the local model is much less powerful ([8, 14, 19, 40] et seq.) and significantly limits the accuracy of computations.

In principle there is no tradeoff between trust and power, as the user's can use cryptographic secure multiparty computation to implement any algorithm designed for the central model without any trusted party. However, general-purpose secure multiparty computation has several drawbacks, such as large computation and communication costs, multiple rounds of interaction, and requiring all users to remain live throughout the computation. Although there are more practical protocols implementing certain differentially private algorithms ([22] et seq.) so far these are restricted to relatively simple computations and are not practical for large-scale applications.

Thus, a recent focus has been on *intermediate trust models* that offer some of the best features of both the central model and the local model. Two models that have received significant attention are:

- The *shuffle model* [16, 29].[1] In this model, users introduce randomness into their own data, as in the local model. However the user's responses are then passed through a *secure shuffler* so the responses are visible but not identified with individual users. We consider the most general *multi-message* shuffle model where each user can send multiple responses that are shuffled independently. An equivalent model would use secure aggregation to ensure that only a histogram of

[1] More precisely, we consider a version of the shuffle model with an additional *robustness* property [3]. Although the property is not without loss of generality, and has not always been formalized in the literature, it is satisfied by all known natural shuffle protocols, and was one of the explicit motivations of studying the shuffle model [16]. For brevity we use only the term "shuffle model" in the introduction, and defer more discussion of this issue to Section 2.

the responses is visible. Secure shuffling and secure aggregation are significantly easier to achieve than general secure computation, and Google's PROCHLO system [9] is a scalable realization of this model.

- The *pan-private model* [24]. In this model, the users' data is processed in an online fashion by a central party. We trust this central party to process the data but not to store it in perpetuity, so we assume that at any one point in the stream, the party's internal state may become visible. This model captures, for example, a data collector who is well intentioned, and can be trusted to see raw data during process, but whose storage may be subject to breaches [1].

We visualize the models in Figure 1. At first glance, these two models seem unrelated, however a recent result of Balcer, Cheu, Joseph, and Mao [3] shows that, for a large class of problems that includes all the problems we study, any protocol in the shuffle model can be simulated in the pan-private model with only a small reduction in accuracy. So for purposes of this work, we can think of these models as being ordered from least powerful to most powerful as *local* $\preceq$ *shuffle* $\preceq$ *pan-private* $\preceq$ *central*.

Both the shuffle model ([16, 29] et seq.) and the pan-private model [1, 24, 44] provably allow much greater accuracy than the local model, while also requiring weaker trust than the central model. See Section 1.3 for a more specific overview of recent progress. However, these positive results are mostly limited to relatively simple functionalities, such as computing means and histograms over the user's data. We note that these are all problems that can be solved efficiently in the local model with reasonable, although larger, sample complexity. However, for problems such as learning parities and selecting the most common attribute, where the local model is most severely limited [19, 28, 40, 47], there is no evidence that either the pan-private or shuffle model can overcome these limitations. Our main contribution is to show that these limitations are inherent:

> *For many high-dimensional learning and estimation problems, the shuffle and pan-private models incur an exponential cost in sample complexity relative to the central model.*

For those familiar with differential privacy, our results can be interpreted as the statement *there is no analogue of the exponential mechanism in the pan-private or shuffle models*, as we prove lower bounds for problems that can be solved in the central model by applying the exponential mechanism.

Our specific lower bounds follow from a new general lower bound argument. We note that the two most common lower bounds techniques for the local model cannot prove lower bounds for the pan-private and shuffle models, so our lower bounds cannot be proven by any straightforward extension of existing lower bound techniques. Specifically, there is no non-trivial upper bound on the mutual information between the algorithm's inputs and outputs [2], so information-theoretic arguments [19, 42] do not apply. Moreover, these models can solve problems that would requite infinitely many statistical queries to solve, so the simulation of the local model in the statistical query model [40] cannot be extended to these more general models.

1.1 Results

Our main results are lower bounds for many closely related learning and estimation problems in both the pan-privacy and shuffle models of differential privacy. We note that throughout this work we adopt the standard model for studying privacy for distributional problems where we define the *accuracy* goal with respect to input satisfying certain distributional assumptions, but define *privacy* for a worst-case dataset. We begin by highlighting two important cases of our results.

Learning Parities. In this canonical learning problem, we are given a dataset consisting of n labeled examples $\{(x_i, y_i)\}$ sampled from some distribution $\mathbf{P}$ over the domain $\{\pm 1\}^d \times \{\pm 1\}$. The goal is to output a parity function $h_S(x) = \prod_{j \in S} x_j$ that predicts the labels nearly as well as any other parity function. Namely,

$$\mathop{\mathbb{P}}_{(x,y)\sim\mathbf{P}}(h_S(x) = y) \geq \max_T \mathop{\mathbb{P}}_{(x,y)\sim\mathbf{P}}(h_T(x) = y) - \alpha.$$

In the central model this problem can be solved privately to any constant level of accuracy with just $O(d)$ samples [40], whereas in the local model any algorithm solving this problem requires $\Omega(2^d)$ samples [28, 40].[2] We prove an exponential separation between the central model and the pan-privacy and shuffle models, showing that, for learning parities, these models are much more similar to the local model.

THEOREM 1.1. *(Informal) Any differentially private algorithm that learns parity functions in the pan-privacy model or the shuffle privacy model requires $\Omega(2^{d/2})$ samples in the worst-case.*

We also consider learning *sparse* parities, where our goal is to output some k-sparse parity function h_S, $|S| \leq k$ that competes with the best parity function on k variables. That is,

$$\mathop{\mathbb{P}}_{(x,y)\sim\mathbf{P}}(h_S(x) = y) \geq \max_{T:|T|\leq k} \mathop{\mathbb{P}}_{(x,y)\sim\mathbf{P}}(h_T(x) = y) - \alpha.$$

We show that learning k-sparse parities requires $\Omega(\sqrt{\binom{d}{\leq k}})$ samples where $\binom{d}{\leq k}$ denotes the number of k-sparse parities on d bits.

Selection. One of the most celebrated tools in central-model differential privacy is the *exponential mechanism* of McSherry and Talwar [43], which is a very general and very accurate method for optimizing a Lipschitz loss function over a discrete set of choices. The canonical problem solved by the exponential mechanism is the following *selection* problem: given a dataset consisting of n samples $\{x_i\}$ from some distribution $\mathbf{P}$ over the domain $\{0,1\}^d$, select a coordinate j such that the expected value of the j-th coordinate is as large as possible. Namely,

$$\mathop{\mathbb{E}}_{x\sim\mathbf{P}}(x_j) \geq \max_k \mathop{\mathbb{E}}_{x\sim\mathbf{P}}(x_k) - \alpha.$$

In the central model, the exponential mechanism solves this problem to any constant level of accuracy with just $O(\log d)$ samples, whereas in the local model any algorithm solving this problem requires $\Omega(d \log d)$ samples [19, 47]. Again, we show an exponential

[2] For specificity, we state lower bounds for the *non-interactive* local model of differential privacy, although, for every problem we consider, slightly weaker bounds are known to hold for interactive variants of the local model as well.

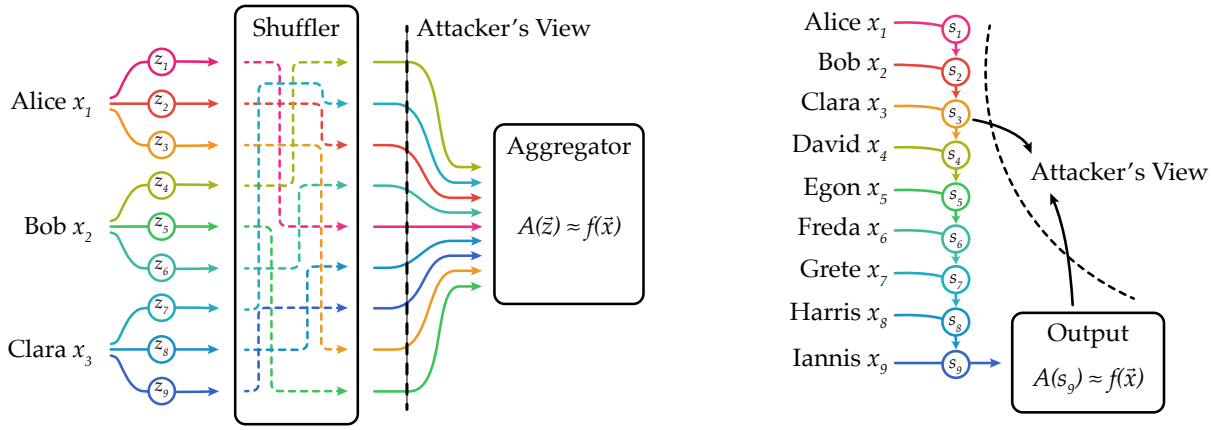

Figure 1: (Left) The multi-message shuffle model. The attacker's view consists of the entire set of messages, randomly shuffled. (Right) The pan-privacy model. The attacker's view consists of the output and the internal state at any single step, which is s_3 in this example.

separation between the central model and the pan-privacy and shuffle models, demonstrating that there is no general-purpose analogue of the exponential mechanism in these intermediate models.

Theorem 1.2. *(Informal) Any differentially private algorithm that solves selection to constant accuracy in the pan-privacy model or the shuffle privacy model requires $\Omega(\sqrt{d})$ samples in the worst-case.*

Variants of Differential Privacy. We emphasize that all of our lower bounds hold for the most general variant of differential privacy, (ε, δ)-differential privacy for $\delta \leq 1/n^{1.1}$, and obtain lower bounds for this variant is one of the main technical challenges addressed by our work. Thus, our results imply essentially the same lower bounds for pure differential privacy, concentrated differential privacy [12, 25], truncated concentrated differential privacy [10], Rényi differential privacy [45], and Gaussian differential privacy [18], none of which were known prior to our work.

More Applications. In our work we also prove tight lower bounds for several closely related, natural problems that have been studied in the literature on differential privacy:

- *Estimating k-Sparse Parities* for $1 \leq k \leq d$. Here we are given samples $\{x_i\}$ from a distribution $\mathbf{P} \in \{\pm 1\}^d$, and the goal is to output a set of estimates $\{a_T\}_{\substack{T \subseteq [d] \\ |T| \leq k}}$ such that
$$\left| a_T - \mathop{\mathbb{E}}_{x \sim \mathbf{P}} \left(\prod_{j \in T} x_j \right) \right| \leq \alpha$$
for every T.
- *d-wise Simple Hypothesis Testing.* Here we are given samples $\{x_i\}$ from a distribution $\mathbf{P} \in \mathcal{Q}$ where $\mathcal{Q} = \{\mathbf{Q}_1, \ldots, \mathbf{Q}_d\}$ is a known set of d hypotheses satisfying $\mathrm{d_{TV}}(\mathbf{Q}_i, \mathbf{Q}_j) \geq \alpha$, and the goal is to determine which of these distributions is $\mathbf{P}$.
- *1-Sparse Mean Estimation.* Here we are given samples $\{x_i\}$ from a distribution $\mathbf{P} \in \{\pm 1\}^d$ with mean μ, with the promise that $\|\mu\|_0 = 1$, and the goal is to output $\hat{\mu}$ such that $\|\mu - \hat{\mu}\|_\infty \leq \alpha$.

We summarize our lower bounds and compare to the local and central models in Table 1; due to page limits, we will only present the analysis for simple hypothesis testing and parity learning. Our analysis of the other problems (found in the full version of this work) are similar to the analysis found here. We also stress that, while the focus of this work is on lower bounds and not algorithms, all of our lower bounds are easily seen to be tight up to logarithmic factors with respect to trivial statistical query algorithms [41] that can be implemented in both the pan-private and shuffle models of privacy.

1.2 Techniques

Our results are all a consequence of a very general lower bound for algorithms in these models. For simplicity, we will restrict this discussion to pan-private algorithms, as lower bounds for shuffle privacy will then follow from a general transformation from the shuffle model to the pan-privacy model due to Balcer, Cheu, Joseph, and Mao [3]. Also, in this discussion we will ignore the parameter δ for brevity, but, crucially, our results apply for moderately small $\delta > 0$.

Let $\{\mathbf{P}_v\}_{v \in \mathcal{V}}$ be some family of distributions over the domain $\mathcal{X}$, let V be uniform over $\mathcal{V}$, and let
$$\mathbf{U} = \mathop{\mathbb{E}}_{v \sim V} (\mathbf{P}_v)$$
be the uniform mixture of these distributions. We will give lower bounds that show no (ε, δ)-differentially private algorithm in the pan-private or shuffle models can distinguish n i.i.d. samples from $\mathbf{U}^n$ from data drawn from the mixture $\mathbf{P}_V^n$, where we chose $v \sim V$ uniformly and then sample from $\mathbf{P}_v^n$. We will, of course, choose the family $\{\mathbf{P}_v\}$ so that any algorithm solving one of the problems above, must distinguish $\mathbf{U}^n$ from $\mathbf{P}_V^n$, which is how we will obtain sample-complexity lower bounds.

Table 1: Summary of our sample-complexity lower bounds for the pan-privacy and shuffle privacy models, in comparison to the local and central models. For brevity, all lower bounds are stated for accuracy $\alpha = 1/100$ and $(1, n^{-2})$-differential privacy. See the formal theorems for more general statements. All our lower bounds are tight up to polylogarithmic factors. We use the notation $\binom{d}{\le k} = \sum_{i=1}^{k} \binom{d}{i}$.

Problem	Parameters	Local Privacy	Pan/Shuffle Privacy (This Work)	Central Privacy
Learning Parities	Dimension d Sparsity k	$\Omega(\binom{d}{\le k} \log \binom{d}{\le k})$ [28]	$\Omega\left(\sqrt{\binom{d}{\le k}}\right)$ Thms 5.6/5.7	$O(\log \binom{d}{\le k})$ [40]
Selection	Dimension d	$\Omega(d \log d)$ [19]	$\Omega(\sqrt{d})$ See full ver.	$O(\log d)$ [43]
Estimating Parities	Dimension d Sparsity k	$\Omega(\binom{d}{\le k} \log \binom{d}{\le k})$ [28]	$\Omega\left(\sqrt{\binom{d}{\le k}}\right)$ See full ver.	$\tilde{O}(\sqrt{d} \log \binom{d}{\le k})$ [36]
d-Wise Simple Hypothesis Testing	d Hypotheses	$\Omega(d \log d)$ [35]	$\Omega(\sqrt{d})$ Thms 4.2/4.4	$O(\log d)$ [11]
1-Sparse Mean Estimation	Dimension d	$\Omega(d \log d)$ [19]	$\Omega(\sqrt{d})$ See full ver.	$O(\log d)$ [Folklore]

For background, we recap the way to use this setup to prove lower bounds in the (non-interactive) local model of differential privacy. Here, one chooses the data from the mixture $\mathbf{P}_V^n$, and a lemma of Duchi, Jordan, and Wainwright [19] gives a bound on the mutual information between the output of the protocol Π and the identity of the random mixture component V:

$$I(\Pi(\mathbf{P}_V^n); V) = O(n \cdot \varepsilon^2 \cdot \|\{\mathbf{P}_v\}\|_{\infty\to 2}^2) \tag{1}$$

where

$$\|\{\mathbf{P}_v\}\|_{\infty\to 2}^2 = \sup_{f:\mathcal{X}\to[\pm 1]} \mathop{\mathbb{E}}_{v\sim V}\left(\left(\mathop{\mathbb{E}}_{x\sim \mathbf{P}_v}(f(x)) - \mathop{\mathbb{E}}_{x\sim \mathbf{U}}(f(x))\right)^2\right)$$

is the crucial quantity determining how hard these distributions are to distinguish subject to local differential privacy. For intuition, note that this quantity satisfies the relationship

$$\begin{aligned}\|\{\mathbf{P}_v\}\|_{\infty\to 2}^2 &\le \mathop{\mathbb{E}}_{v\sim V}\left(\sup_{f:\mathcal{X}\to[\pm 1]}\left(\mathop{\mathbb{E}}_{x\sim \mathbf{P}_v}(f(x)) - \mathop{\mathbb{E}}_{x\sim \mathbf{U}}(f(x))\right)^2\right)\\ &= 4 \cdot \mathop{\mathbb{E}}_{v\sim V}\left(\mathrm{d}_{\mathrm{TV}}(\mathbf{P}_v, \mathbf{U})^2\right),\end{aligned}$$

but it can be much smaller than $4 \cdot \mathbb{E}_{v\sim V}(\mathrm{d}_{\mathrm{TV}}(\mathbf{P}_v, \mathbf{U})^2)$, which is crucial for proving tight lower bounds.

Given this lemma, and a construction of a hard distribution family such that $\|\{\mathbf{P}_v\}\|_{\infty\to 2}^2$ is small, it is not hard to deduce a lower bound on the number of samples n required to identify the specific mixture component V. It is also not too difficult to construct a family of hard distributions for all of our problems of interest (see Lemma 3.10). We note that all of the lower bounds in the "local model" column of Table 1 are proven via this approach.

With this state-of-affairs, it is tempting to try to argue that a mutual-information bound analogous to (1) holds for pan-private or shuffle model algorithms. However, Balcer and Cheu [2] constructed a family of distributions and a pan-private algorithm such that the mutual information $I(\Pi; V)$ can be unbounded, showing that the purely information-theoretic approach used to prove lower bounds for the local model cannot work for pan-privacy.[3]

Nonetheless, we prove the following indistinguishability lemma for pan-private algorithms:

$$\mathrm{d}_{\mathrm{TV}}(\Pi(\mathbf{U}^n), \Pi(\mathbf{P}_V^n)) \le O(n \cdot \varepsilon \cdot \|\{\mathbf{P}_v\}\|_{\infty\to 2}) \tag{2}$$

Although this bound is quantitatively somewhat weaker than (1)—in ways that are actually crucial to avoid proving false statements—it is nonetheless sufficient to give tight lower bounds for all of the problems we consider. The value of this lemma is that, even though the information-theoretic bounds that are used in the local model are false for the pan-private model, the exact same constructions of hard distributions can be used to obtain lower bounds for pan-privacy!

The proof of this lemma uses a hybrid argument, where we transition between data sampled from $\mathbf{U}^n$ and data sampled from $\mathbf{P}_V^n$. Namely, we fix a value of i between 0 and n and consider the case where the first i inputs are sampled from $\mathbf{U}^i$ and the remaining $n-i$ inputs are sampled from $\mathbf{P}^{n-i}$. We then bound the total variation distance between the i-th case and the $(i+1)$-st case and apply the triangle inequality. In each step, we carefully argue that the total variation distance between the two cases follows from a careful application of (1) to the algorithm that computes the internal state after viewing the first i inputs, which is why we ultimately get a bound of a similar form.

[3] The algorithm showing pan-private algorithms can have unbounded mutual information crucially uses the full generality of (ε, δ)-differential privacy for $\delta > 0$, however, even for stricter variants of differential privacy where the mutual information is bounded, we don't know how to obtain a mutual-information bound as strong as (1) for any of these variants.

1.3 Related Work

Comparison to the Concurrent Works of [15] and [7]. A concurrent and independent work of Chen, Ghazi, Kumar, and Manurangsi [15] proves lower bounds for selection and learning parity in the multi-message shuffle model. Their lower bounds depend on the number of messages, and are only non-trivial when the number of messages is relatively small, whereas our lower bounds do not require any bound on the number of messages. For example, their lower bound for selection is $\Omega(d/m)$, where m is the number of messages, while our lower bound for selection is $\Omega(\sqrt{d})$ for any number of messages, and our lower bound is matched by a trivial algorithm that sends d messages. Compared to ours, their lower bounds do not require the shuffle protocol to be robust, although robustness was a motivating feature of the shuffle model that is discussed in the early work on the subject [16, 29]. Their work also does not consider the pan-privacy model, and their arguments do not seem to apply to that model.

Another concurrent and independent work of Beimel, Haitner, Nissim, and Stemmer [7] proves lower bounds for multi-message shuffle protocols that use a small number of messages. They show that if an m-message shuffle protocol is private when run with for n users, then each user's messages reveals at most $\approx n^m$ bits of information about their input, which allows them to prove non-trivial lower bounds when m is quite small. Beimel et al. also describe shuffle protocols that use multiple rounds of communication; we do not consider interactivity in this work.

The Shuffle Model. The shuffle model was introduced concurrently in works by Cheu et al. [16] and Erlingsson et al. [29]. These works were both inspired by Google's PROCHLO system [9], which implements a more general algorithmic paradigm called *encode, shuffle, and analyze*. Much of the work in this model has focused on constructing optimal algorithms for problems like binary sums [16, 30], real-valued sums [4, 5, 32–34], histograms and heavy-hitters [2, 16, 31], and uniformity testing [3]. Another complementary set of works have given general *amplification theorems* showing that if each user applies a differentially private randomizer to their data, then the shuffle protocol using the randomizer satisfies differential privacy with stronger parameters [4, 29].

Almost all prior lower bounds for the shuffle model apply only to a special case of the model where each user sends only a single response, the so-called *single-message shuffle model*. Cheu et al. [16] showed that if a protocol is private in this restricted model, then each user's response satisfies local differential privacy, for which we already have strong lower bounds. Their approach was refined by Ghazi et al. [31], who obtained stronger bounds for single-message protocols. Balle et al. [4] proved a lower bound for computing real-valued sums in the single-message model. In contrast, our lower bounds hold for the general *multi-message shuffle model*, where each user may send an arbitrary number of messages that are shuffled independently. Note that in this model, the user's individual responses need not satisfy any local differential privacy [2]. An early lower bound for the multi-message shuffle model is due to Ghazi et al. [30], and applies to computing binary sums subject to pure differential privacy and a strong communication constraint. We emphasize that our lower bounds do not impose any restriction on the number of messages or the amount of communication.

The Pan-Private Model. The pan-privacy model was introduced by Dwork et al. [24] as a model of differential privacy for streaming algorithms, and they constructed pan-private algorithms for classic streaming problems like distinct elements. Their algorithm was subsequently improved by Mir et al. [44], who also gave the first lower bounds for this model. We note that their technique gives lower bounds for worst-case inputs, whereas our technique gives lower bounds for distributional problems.

More recently, Amin, Joseph, and Mao [1] revisited the model from the perspective of finding an intermediate trust model between local and central privacy, which is the perspective we adopt in this work. They also gave an algorithm for *uniformity testing* and a matching lower bound for algorithms satisfying pure differential privacy, which is (ε, δ)-privacy with $\delta = 0$. Theirs is the first lower bound in this model for any distributional problem. As we discussed above, their information-theoretic arguments are inherently limited to pure differential privacy, whereas ours apply to differential privacy in general.

The initial work on pan-privacy considered a more general model where the attacker can view the internal state at two or more arbitrary steps, however [1] showed that this model is equivalent to the local model with sequential interaction. Our lower bounds apply to the weakest model, where the attacker can view the state at just a single time step.

Lower Bounds Techniques in the Local and Central Model. We briefly summarize the techniques for proving lower bounds in the more well studied models of differential privacy. The first lower bounds for local differential privacy were proven by Kasiviswanathan et al. [40], who proved that the local model is equivalent, up to polynomial factors, to the statistical queries model [41]. Balcer and Cheu [2] showed that the shuffle and pan-private model do not admit such a characterization. Recently Edmonds, Nikolov, and Ullman [28] gave a nearly tight characterization of the sample complexity of query release and agnostic learning in the non-interactive local model. Subsequent work led to stronger lower bounds for specific problems in the local model [6, 8, 14, 19–21, 37, 38], including interactive variants of the local model. This line of work primarily uses information-theoretic arguments that were first introduced by McGregor et al. [42] in the context of two-party differential privacy. However, as we discussed earlier, these approaches cannot give strong lower bounds for the pan-private and shuffle model, and the main novelty in our work is finding alternative lower-bound arguments for these intermediate models that do not require strong information bounds.

There are two main approaches to proving lower bounds for high-dimensional problems in the central model of differential privacy. The first are reconstruction attacks, introduced by Dinur and Nissim ([17] et seq.). These attacks only apply when computing some statistics to very high accuracy, and thus cannot give non-trivial lower bounds for distributional problems where the accuracy can never be smaller than the sampling error. The other main approach is based on tracing attacks ([13, 27, 46] et seq.). Although tracing attacks give tight lower bounds for the central model, but the lower bounds we prove for more restricted models are exponentially larger, and do not seem to be provable using tracing attacks. We refer the reader to [26] for a survey of these lower bounds.

2 PRELIMINARIES

2.1 Notational Conventions

We use boldface letters to denote probability distributions, capital letters in plain math text denote random variables, and calligraphic letters denote sets. We reserve M for randomized algorithms and Π for distributed protocols. Throughout this work, we use the notation $[k] := \{1, 2, \ldots, k\}$.

2.2 Differential Privacy

We define a *dataset* $\vec{x} \in \mathcal{X}^n$ to be an ordered tuple of n rows where each row is drawn from a data universe $\mathcal{X}$ and corresponds to the data of one user. Two datasets $\vec{x}, \vec{x}\,' \in \mathcal{X}^n$ are *neighbors*, denoted as $\vec{x} \sim \vec{x}\,'$, if they differ in at most one row.

Definition 2.1 (Differential Privacy [23])**.** An algorithm $M : \mathcal{X}^n \to \mathcal{R}$ satisfies (ε, δ)*-differential privacy* if, for every pair of neighboring datasets $\vec{x}$ and $\vec{x}\,'$ and every event $C \subseteq \mathcal{R}$,

$$\mathbb{P}\big(M(\vec{x}) \in C\big) \leq e^{\varepsilon} \cdot \mathbb{P}\big(M(\vec{x}\,') \in C\big) + \delta.$$

The *central model* of differential privacy refers to the case where the algorithm M is allowed to depend arbitrarily on $\vec{x}$ with no further restrictions.

2.3 The Pan-Private Model

A *pan-private* algorithm observes the data as a *stream*. At each step, the algorithm receives a datapoint that it uses to update its internal state, and this process repeats until the stream is exhausted and a final output is computed. We say that two streams $\vec{x}$ and $\vec{x}\,'$ are *neighbors* if they differ in at most one element. Pan-privacy models an attacker who observes the final output of the algorithm, as well as the internal state at any one step in the stream, and requires that the joint distribution of these two pieces of information is differentially private.

Definition 2.2 (Online Algorithm)**.** An *online algorithm* M is defined by a sequence of internal algorithms $M_1, M_2, \ldots$ and an output algorithm M_O. On input $\vec{x}$, the first function $M_1 : \mathcal{X} \to \mathcal{I}$ maps x_1 to a state s_1 and the remaining functions M_i map x_i and the previous state s_{i-1} to a new state s_i. At the end of the stream, M publishes a final output by executing $M_O : \mathcal{I} \to \mathcal{O}$ on its final internal state.

Definition 2.3 (Pan-privacy [1, 24])**.** Given an online algorithm M, let $M_{\mathcal{I}}(\vec{x})$ denote its internal state after processing stream $\vec{x}$, and let $\vec{x}_{\leq t}$ be the first t elements of $\vec{x}$. We say M is (ε, δ)*-pan-private* if, for every pair of neighboring streams $\vec{x}$ and $\vec{x}\,'$, every time t and every set of internal state, output state pairs $T \subseteq \mathcal{I} \times \mathcal{O}$,

$$\begin{aligned} &\mathbb{P}_{M}\big(\big(M_{\mathcal{I}}(\vec{x}_{\leq t}), M_O(M_{\mathcal{I}}(\vec{x}))\big) \in T\big) \\ &\leq e^{\varepsilon} \cdot \mathbb{P}_{M}\big(\big(M_{\mathcal{I}}(\vec{x}\,'_{\leq t}), M_O(M_{\mathcal{I}}(\vec{x}\,'))\big) \in T\big) + \delta. \end{aligned} \tag{3}$$

See Figure 1 for a diagram.

Note that any pan-private algorithm can trivially be implemented in the central model. Our definition of pan-privacy is the specific variant given by Amin et al. [1]. This version guarantees record-level privacy (uncertainty about the presence of any single stream element) rather than user-level privacy (uncertainty about the presence of any one data universe element). We use this variant because for the problems we consider it is natural to model each user as contributing a single element of the stream.

2.4 The Shuffle Model

In the *shuffle model*, each user individually randomizes their own data to produce a series of messages. Unlike the local model, where these messages would be identified with the user who produced them, we allow the users to send their messages to a *secure shuffler* that collects all the messages of all the users and randomly permutes them.[4] The shuffle model captures an attacker who observes the messages after they are shuffled, and we require this shuffled set of messages to satisfy differential privacy. An equivalent model would allow the attacker observes only a histogram of the messages.

Definition 2.4 (Shuffle Model [16])**.** A protocol Π in the *shuffle model* consists of three randomized algorithms:

- A *randomizer* $\Pi_R : \mathcal{X} \to \mathcal{Y}^*$ mapping data to (possibly variable-length) vectors. The length of the vector is the number of messages sent. If, on all inputs, the probability of sending a single message is 1, then the protocol is said to be *single-message*. Otherwise, the protocol is *multi-message*.
- A *shuffler* $\Pi_S : \mathcal{Y}^* \to \mathcal{Y}^*$ that applies a uniformly random permutation to all messages.
- An *analyzer* $\Pi_A : \mathcal{Y}^* \to \mathcal{O}$ that computes on a permutation of messages.

As the shuffler is the same in every protocol, we identify each shuffle protocol by $\Pi = (\Pi_R, \Pi_A)$. We define the honest execution on input $\vec{x} \in \mathcal{X}^n$ as

$$\Pi(\vec{x}) := \Pi_A(\Pi_S(\Pi_R(x_1), \ldots, \Pi_R(x_n))).$$

We denote the output of the shuffler as

$$(\Pi_S \circ \Pi_R^n)(\vec{x}) := \Pi_S(\Pi_R(x_1), \ldots, \Pi_R(x_n)).$$

We assume that users and the analyzer have access to n, as well as an arbitrary amount of public randomness.

We remark that we focus on *one-round* or *non-interactive* shuffle protocols: each user generates their messages in one time step, the shuffler permutes the messages for the analyzer in the next time step, and no other communication occurs between parties. Concurrent work by Beimel et al. [7] define a form of multi-round shuffle protocol but we do not consider it here.

It remains to define differential privacy in this model. We note that the output of the shuffler only follows the distribution $\Pi(\vec{x})$ if all users are following the protocol as specified. This assumption is undesirable because it means each user is reliant on other users to behave correctly. Thus we consider a *robust* variant of the shuffle model, where we require that the protocol remains private when only a constant fraction of users behave correctly, while the other users may behave arbitrarily. We emphasize all known natural protocols in this model satisfy the additional robustness condition, and the need for robustness was explicitly discussed in [16] as a feature of the model, so we consider the robust variant to be the most appropriate version of the model.

Definition 2.5 (Robust Shuffle Differential Privacy [3])**.** Fix $\gamma \in (0, 1]$. A protocol $\Pi = (R, A)$ is $(\varepsilon, \delta, \gamma)$*-robustly shuffle differentially*

[4] See [9] for a discussion of various choices of how to implement such a secure shuffler.

private if, for all $n \in \mathbb{N}$ and $\gamma' \geq \gamma$, the algorithm $\Pi_S \circ \Pi_R^{\gamma' n}$ is (ε, δ)-differentially private. In other words, Π guarantees (ε, δ)-shuffle privacy whenever at least a γ fraction of the intended number of users follow the protocol.

We remark that the above definition only explicitly handles drop-out attacks, where malicious users send no messages. However, dropping out is the worst malicious users can do. Combining arbitrary messages from malicious users with the messages of honest users can be viewed as a post-processing of $\Pi_S \circ \Pi_R^{\gamma n}$. If $\Pi_S \circ \Pi_R^{\gamma n}$ is already differentially private for the outputs of the γn users alone, then differential privacy's resilience to post-processing ensures that adding other messages does not affect this guarantee. Hence, it is without loss of generality to focus on drop-out attacks.

2.5 From Robust Shuffle Privacy to Pan-Privacy

[3] prove a reduction from robust shuffle privacy to pan-privacy in the context of uniformity testing and counting distinct elements. Here, we note that the technique can be applied to essentially any distributional problem, so we state it as a standalone theorem. Using this theorem we will be able to obtain lower bounds for the shuffle model from those we prove for the pan-private model.

We begin by establishing some notation. For any universe $\mathcal{X}$, let $\mathbf{U}$ denote any fixed distribution over $\mathcal{X}$. For any distribution $\mathbf{P}$ over $\mathcal{X}$ and any $b \in [0, 1]$, let $\mathbf{P}_{(b)}$ denote the mixture $b \cdot \mathbf{P} + (1 - b) \cdot \mathbf{U}$.

Theorem 2.6 (Generalization of [3]). *For any number of users n and any $(\varepsilon, \delta, 1/3)$-robustly shuffle private protocol Π, there exists an (ε, δ)-pan-private algorithm M^Π such that*

$$\mathrm{d_{TV}}(M^\Pi(\mathbf{U}^{n/3}), \Pi(\mathbf{U}^n)) = 0 \tag{4}$$

and, for any $\mathbf{P}$ over $\mathcal{X}$,

$$\mathrm{d_{TV}}(M^\Pi(\mathbf{P}^{n/3}), \Pi(\mathbf{P}^n_{(2/9)})) < \exp(-\Omega(n)). \tag{5}$$

In particular, if n is larger than some absolute constant,

$$\mathrm{d_{TV}}(M^\Pi(\mathbf{P}^{n/3}), \Pi(\mathbf{P}^n_{(2/9)})) < 1/6.$$

Algorithm 1: M^Π, an online algorithm built from a shuffle protocol

Input: Data stream $\vec{x} \in \mathcal{X}^{n/3}$; a shuffle protocol $\Pi = (\Pi_R, \Pi_A)$ that expects n inputs

Create initial state $S_0 \leftarrow (\Pi_S \circ \Pi_R^{n/3})(\mathbf{U}^{n/3})$
Sample $N' \sim \mathbf{Bin}(n, 2/9)$
Set $N' \leftarrow \min(N', n/3)$
For $i \in [n/3]$
  If $i \leq N'$**:** $W_i \leftarrow x_i$;
  Else $W_i \sim \mathbf{U}$;
  Create the state S_i by shuffling the messages from S_{i-1} with those from $\Pi_R(W_i)$
Create $\vec{Y}$ by shuffling the messages from $S_{n/3}$ with those from $\Pi_R^{n/3}(\mathbf{U}^{n/3})$
Return $\Pi_A(\vec{Y})$

Proof. We present a concise version of M^Π in Algorithm 1. Although it does not explicitly take the form specified by Definition 2.2, it is straightforward to decompose it into a sequence

$$(M_1, \ldots, M_{n/3}, M_O).$$

We give a high-level justification before detailing the proof. Prior to reading user data, it simulates the execution of the shuffle protocol on $n/3$ data samples drawn i.i.d. from $\mathbf{U}$. Upon reading each user's data, it generates messages using the local randomizer and adds to the shuffled set of messages. Each state of the shuffled set is private due to the robust privacy of the shuffle protocol: it contains a uniformly random permutation of messages from at least $n/3$ executions of the local randomizer. After reading user data, the algorithm injects $n/3$ more rounds of noisy messages. This serves to ensure that the output of the protocol is private. And observe that the shuffle protocol is run on a mixture between the true data distribution and the public distribution $\mathbf{U}$, where the mixture parameter is known to the algorithm. This means that the distribution of the output is shifted by a known and bounded factor.

Pan-privacy: For any user i and intrusion time t, we prove that $\left(M_I^\Pi(\vec{x}_{\leq t}), M_O^\Pi(M_I^\Pi(\vec{x}))\right)$—the adversary's view—is (ε, δ)-private *conditioned on arbitrary event $N' = n'$*. If $i > n'$, observe that the algorithm is completely independent of x_i. Otherwise, we shall leverage the robust privacy of Π.

We first consider the case where $t < i$. The state observed by the adversary, S_t, is independent of x_i so it will suffice to prove that $M_O^\Pi(M_I^\Pi(\vec{x}))$ is differentially private *conditioned on any event $S_t = s_t$*. Note that $M_O^\Pi(M_I^\Pi(\vec{x}))$ is obtained by running Π_A on the union of s_t and

$$(\Pi_S \circ \Pi_R^h)(x_{t+1}, \ldots, x_i, W_{i+1}, \ldots, W_{n/3}, \underbrace{\mathbf{U}, \ldots, \mathbf{U}}_{n/3 \text{ terms}}), \tag{6}$$

where $h = 2n/3 - t \geq n/3$. We can therefore invoke the robust shuffle privacy of Π.

Now we consider the case where $t \geq i$. Observe that $M_I^\Pi(\vec{x}_{\leq t})$ is equivalent to

$$(\Pi_S \circ \Pi_R^h)(\underbrace{\mathbf{U}_\mathcal{X}, \ldots, \mathbf{U}_\mathcal{X}}_{n/3 \text{ terms}}, x_1, \ldots, x_i, W_{i+1} \ldots, W_t),$$

where $h = n/3 + t > n/3$. We again invoke the robust shuffle privacy of Π. And, conditioned on any event $M_I^\Pi(\vec{x}_{\leq t}) = s_t$, we argue that $M_O^\Pi(M_I^\Pi(\vec{x}))$ is independent of x_i. This follows from our previous observation that $M_O^\Pi(M_I^\Pi(\vec{x}))$ is obtained by running Π_A on the union of s_t and (6); x_i is not an input to this function.

Bound on TV distance: In the case where the input $\vec{X}$ is drawn from $\mathbf{U}^{n/3}$, observe that every execution of Π_R made by M^Π is on an independent sample from $\mathbf{U}$. Because the output of the algorithm is obtained by running Π_A on n such executions, we immediately have $M^\Pi(\mathbf{U}^{n/3}) = \Pi(\mathbf{U}^n)$.

Otherwise, consider n samples from $\mathbf{P}_{(2/9)}$. The number of samples drawn from $\mathbf{P}$ is distributed as $\mathbf{Bin}(n, 2/9)$. By Hoeffding's bound, $\mathbb{P}(\mathbf{Bin}(n, 2/9) > n/3) < \exp(-\Omega(n))$. This means the TV distance between $\mathbf{Bin}(n, 2/9)$ and the distribution of N' is at most

$\exp(-\Omega(n))$. In turn, the TV distance between

$$\Pi(\mathbf{P}_{(2/9)}^n) = \Pi_A(\Pi_S(\overbrace{\underbrace{\Pi_R(\mathbf{P}),\ldots,\Pi_R(\mathbf{P})}_{\mathbf{Bin}(n,2/9)\text{ terms}},\Pi_R(\mathbf{U}),\ldots,\Pi_R(\mathbf{U})}^{n\text{ terms}}))$$

and

$$M^{\Pi}(\mathbf{P}^{n/3}) = \Pi_A(\Pi_S(\overbrace{\underbrace{\Pi_R(\mathbf{P}),\ldots,\Pi_R(\mathbf{P})}_{N'\text{ terms}},\Pi_R(\mathbf{U}),\ldots,\Pi_R(\mathbf{U})}^{n\text{ terms}}))$$

is at most $\exp(-\Omega(n))$ as well. This concludes the proof. □

3 MAIN LOWER BOUND

Let M be a pan-private algorithm. Let $\{\mathbf{P}_v\}_{v\in\mathcal{V}}$ be a family of distributions, V be uniform over $\mathcal{V}$, and $\mathbf{U} = \mathbb{E}_{v\sim V}(\mathbf{P}_v)$ be the uniform mixture over the distributions. Let $\mathbf{U}^n$ be the product distribution consisting of n copies of $\mathbf{U}$ and let $\mathbf{P}_V^n = \mathbb{E}_{v\sim V}(\mathbf{P}_v^n)$ be the mixture of product distributions. Note that $\mathbf{U} = \mathbf{P}_V^1$.

An important quantity that we will show measures how hard it is for pan-private algorithms to distinguish $\mathbf{U}^n$ from $\mathbf{P}_V^n$ is the $(\infty\to 2)$-*norm*[5] of $\{\mathbf{P}_v\}$, which defined as

$$\|\{\mathbf{P}_v\}\|_{\infty\to 2} = \sup_{f:\mathcal{X}\to[\pm 1]} \mathop{\mathbb{E}}_{v\sim V}\left(\left(\mathop{\mathbb{E}}_{x\sim\mathbf{P}_v}(f(x)) - \mathop{\mathbb{E}}_{x\sim U}(f(x))\right)^2\right)^{1/2}$$

The main goal of this section is to prove the following theorem.

Theorem 3.1. *If $\{\mathbf{P}_v\}_{v\in\mathcal{V}}$ is a family of distributions and M is an (ε,δ)-pan private algorithm such that*[6] *$\delta\log|\mathcal{V}|/\delta \ll \varepsilon^2\|\{\mathbf{P}_v\}\|_{\infty\to 2}^2$ and $\mathrm{d_{TV}}(M(\mathbf{P}_V^n), M(\mathbf{U}^n))$ is larger than a positive constant, then*

$$n \geq \Omega\left(\frac{1}{\varepsilon\|\{\mathbf{P}_v\}\|_{\infty\to 2}}\right)$$

More generally, $n \geq 1/O(\varepsilon\|\{\mathbf{P}_v\}\|_{\infty\to 2} + \sqrt{\delta\log|\mathcal{V}|/\delta})$

The main tool we use to prove Theorem 3.1 is the following information inequality.

Lemma 3.2. *For any (ε,δ)-pan private algorithm M,*

$$\mathrm{d_{TV}}(M(\mathbf{P}_V^n), M(\mathbf{U}^n)) \leq n\cdot\sqrt{\tfrac{1}{2}I_{\varepsilon,\delta}(\{\mathbf{P}_v\})}$$

where we define $I_{\varepsilon,\delta}(\{\mathbf{P}_v\}) = \sup_{\substack{M:\mathcal{X}\to\mathcal{R}\\(\varepsilon,\delta)\text{-}DP}} I(M(\mathbf{P}_V);V)$

Proof of Lemma 3.2. As a shorthand, let $\mathbf{Q}_i$ denote the distribution of $M(\mathbf{U}^i,\mathbf{P}_V^{n-i})$. This is the distribution of the algorithm's output on a data stream where the first i elements are sampled i.i.d. from $\mathbf{U}$ and the rest from $\mathbf{P}_V$. Note that $\mathbf{Q}_0 = M(\mathbf{P}_V^n)$ and $\mathbf{Q}_n = M(\mathbf{U}^n)$. By the triangle inequality we have

$$\mathrm{d_{TV}}(M(\mathbf{P}_V^n), M(\mathbf{U}^n)) = \mathrm{d_{TV}}(\mathbf{Q}_0,\mathbf{Q}_n) \leq \sum_{i=1}^n \mathrm{d_{TV}}(\mathbf{Q}_{i-1},\mathbf{Q}_i).$$

Thus, in order to prove the theorem it is enough to show that for every $i = 1,\ldots,n$,

$$\mathrm{d_{TV}}(\mathbf{Q}_{i-1},\mathbf{Q}_i) \leq \sqrt{\tfrac{1}{2}I_{\varepsilon,\delta}(\{\mathbf{P}_v\})} \tag{7}$$

Before proving (7), we give a simplified diagram of the relevant random variables in the two distributions $\mathbf{Q}_{i-1},\mathbf{Q}_i$ in Figure 2. For the purposes of comparing $\mathbf{Q}_{i-1}$ and $\mathbf{Q}_i$, we can group all of the inputs $X_1,\ldots,X_{i-1}\sim\mathbf{U}^{i-1}$ into one random variable and all of the inputs $X_{i+1\cdots n}\sim\mathbf{P}_V^{n-i}$ into another random variable. Moreover, in $\mathbf{Q}_{i-1}$, X_i is drawn from $\mathbf{P}_V$, for the same choice of V as $X_{i+1\cdots n}$, whereas in $\mathbf{Q}_i$, X_i is drawn from $\mathbf{U}$.

Now, observe that the random variable S_i has the same marginal distribution in both $\mathbf{Q}_{i-1},\mathbf{Q}_i$. This also holds for $X_{i+1\cdots n}$. But in $\mathbf{Q}_{i-1}$, S_i and $X_{i+1\cdots n}$ are correlated by the shared choice of V, while in $\mathbf{Q}_i$ they are independent. Moreover, S_n is a post-processing of the pair $(S_i, X_{i+1\cdots n})$. Thus, using $(S_i,X_{i+1\cdots n})$ to denote the joint distribution of $S_i(V)$ and $X_{i+1\cdots n}(V)$ in $\mathbf{Q}_{i-1}$, and applying the data-processing inequality, we have

$$\begin{aligned}\mathrm{d_{TV}}(\mathbf{Q}_{i-1},\mathbf{Q}_i) &\leq \mathrm{d_{TV}}((S_i,X_{i+1\cdots n}),(S_i\otimes X_{i+1\cdots n}))\\ &= \mathop{\mathbb{E}}_{s_i\sim S_i}\left(\mathrm{d_{TV}}(X_{i+1\cdots n}|_{S_i=s_i},X_{i+1\cdots n})\right)\end{aligned}$$

where the last step uses the following fact.

Fact 3.3. *For any random variables A,B,*

$$\mathrm{d_{TV}}((A,B),(A\otimes B)) = \mathop{\mathbb{E}}_{a\sim A}(\mathrm{d_{TV}}(B|_{A=a},B))$$

Next, since S_i and $X_{i+1\cdots n}$ are independent conditioned on V, we have

$$\mathop{\mathbb{E}}_{s_i\sim S_i}\left(\mathrm{d_{TV}}(X_{i+1\cdots n}|_{S_i=s_i},X_{i+1\cdots n})\right) \leq \mathop{\mathbb{E}}_{s_i\sim S_i}\left(\mathrm{d_{TV}}(V|_{S_i=s_i},V)\right)$$

where we use the following fact.

Fact 3.4. *If (A,B,C) are jointly distributed random variables and A and B are independent conditioned on C, then for every $a\in\mathrm{supp}(A)$, $\mathrm{d_{TV}}(B|_{A=a},B) \leq \mathrm{d_{TV}}(C|_{A=a},C)$.*

We prove Facts 3.3 and 3.4 in the full version of this work. From this point we can calculate

$$\begin{aligned}&\mathop{\mathbb{E}}_{s_i\sim S_i}\left(\mathrm{d_{TV}}(V|_{S_i=s_i},V)\right) &(8)\\ &\leq \sqrt{\mathop{\mathbb{E}}_{s_i\sim S_i}\left(\mathrm{d_{TV}}(V|_{S_i=s_i},V)^2\right)} &\text{(Jensen's Inequality)}\\ &\leq \sqrt{\mathop{\mathbb{E}}_{s_i\sim S_i}\left(\tfrac{1}{2}\cdot\mathrm{d_{KL}}(V|_{S_i=s_i}\|V)\right)} &\text{(Pinsker's Inequality)}\\ &= \sqrt{\mathop{\mathbb{E}}_{s_i\sim S_i}\left(\tfrac{1}{2}\cdot\mathrm{d_{KL}}((S_i,V)\|(S_i\otimes V))\right)} &\text{(chain rule for KL-divergence)}\\ &\leq \sqrt{\tfrac{1}{2}\cdot I(S_i;V)} &\text{(definition of mutual information)}\end{aligned}$$

Lastly, we argue that $I(S_i;V) \leq I_{\varepsilon,\delta}(\{\mathbf{P}_v\})$ using pan-privacy. The intuition is that pan privacy requires S_i to be (ε,δ)-differentially private as a function of the prefix $X_1,\ldots,X_i$. Moreover, $X_1,\ldots,X_{i-1}$ are drawn from the fixed distribution $\mathbf{U}^{i-1}$ that is independent from V. Therefore, we can fix the distribution of $X_1,\ldots,X_{i-1}$ and view S_i as an (ε,δ)-differentially private function of just X_i. Specifically, given an (ε,δ)-pan private algorithm M, and i, define the function

[5] We call this quantity the $(\infty\to 2)$-norm because it is equal to the better known $(\infty\to 2)$-norm, $\sup_z \|Mz\|_2/\|z\|_\infty$, of the matrix M defined by $M_{v,x} = \mathbf{P}_v(x) - U(x)$.

[6] We use $x \ll y$ to indicate that $x \leq cy$ for a sufficiently small numerical constant $c > 0$.

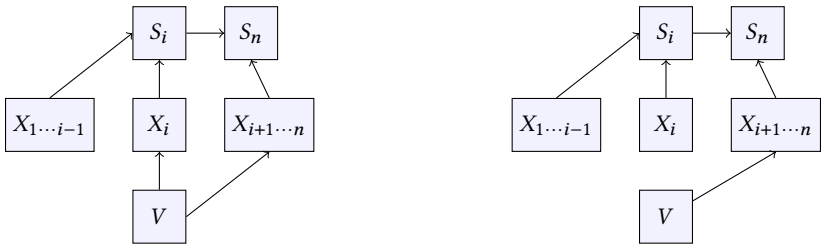

Figure 2: A simplified diagram of the relevant random variables in Q_{i-1} (left) and Q_i (right).

$f_i : \mathcal{X} \to \mathcal{R}$ as follows: $f_i(x)$ samples $X_1, \ldots, X_{i-1} \sim \mathbf{U}^{i-1}$, computes $s_1 = M_1(X_1)$, $s_2 = M_2(X_2, s_1)$, …, $s_{i-1} = M_{i-1}(X_{i-1}, s_{i-2})$, and outputs $r = M_i(x, s_{i-1})$. Pan-privacy guarantees that $f_i(x) = M_i(X_1, \ldots, X_{i-1}, x)$ is (ε, δ)-differentially private as a function of x. Note that $S_i|_{X_i=x}$ is distributed identically as $f_i(x)$. Therefore

$$\sqrt{\tfrac{1}{2}I(S_i; V)} = \sqrt{\tfrac{1}{2}I(M_i(\mathbf{P}_V); V)} \le \sqrt{\tfrac{1}{2}I_{\varepsilon,\delta}(\{\mathbf{P}_v\})}$$

Combining with the previous calculations gives

$$\mathrm{d_{TV}}(\mathbf{Q}_{i-1}, \mathbf{Q}_i) \le \sqrt{\tfrac{1}{2}I_{\varepsilon,\delta}(\{\mathbf{P}_v\})},$$

as desired. □

To use Lemma 3.2 we need a bound on the mutual information $I_{\varepsilon,\delta}(\{\mathbf{P}_v\})$. A result of Duchi, Jordan, and Wainwright [19], gives such a bound for the case of $\delta = 0$.

Lemma 3.5 ([19]). $I_{\varepsilon,0}(\{\mathbf{P}_v\}) \le O(\varepsilon^2 \|\{\mathbf{P}_v\}\|^2_{\infty\to 2})$.

We give a simple extension to the case of $\delta > 0$.

Lemma 3.6. $I_{\varepsilon,\delta}(\{\mathbf{P}_v\}) \le O(\varepsilon^2 \|\{\mathbf{P}_v\}\|^2_{\infty\to 2} + \delta \log |\mathcal{V}|/\delta)$.

Therefore, we will obtain Theorem 3.1 as an immediate corollary of Lemma 3.2 and Lemma 3.6. The proof of Lemma 3.6 from Lemma 3.5 relies on the following statement, which is an easy consequence of a structural result of Kairouz, Oh, and Viswanath [39].

Lemma 3.7. *If $M : \mathcal{X} \to \mathcal{R}$ is (ε, δ)-differentially private, then there is a $(2\varepsilon, 0)$-differentially private M' such that*

$$\forall x \in \mathcal{X}\ \mathrm{d_{TV}}(M(x), M'(x)) \le \delta$$

We prove this lemma in the full version of this work.

Proof of Lemma 3.6. Let M be any (ε, δ)-differentially private function with input $x \in \mathcal{X}$. Lemma 3.7 guarantees that there exists a mechanism M' that is $(2\varepsilon, 0)$-differentially private and satisfies

$$\forall x \in \mathcal{X}\ \mathrm{d_{TV}}(M(x), M'(x)) \le \delta$$

In particular, $\mathrm{d_{TV}}(M(\mathbf{P}_V), M'(\mathbf{P}_V)) \le \delta$. Therefore, there exists a joint distribution (M, M') such that $M = M(\mathbf{P}_V)$, $M' = M'(\mathbf{P}_V)$ and $\mathbb{P}(M \ne M') \le \delta$. Let B be the binary random variable $\mathbb{I}\{M \ne M'\}$. Thus, there is a joint distribution (M, M', B) such that $(B = 0 \implies R = R')$ and $\mathbb{P}(B \ne 0) \le \delta$. Therefore,

$$\begin{aligned}
&I(V; R)\\
&\le I(V; M, M', B)\\
&\le I(V; M, M' \mid B) + H(B)\\
&= I(V; M, M' \mid B = 0) \cdot \mathbb{P}(B = 0)\\
&\quad + I(V; M, M' \mid B = 1) \cdot \mathbb{P}(B = 1) + H(B)\\
&\le I(V; M') + H(V) \cdot \delta + H(B)\\
&= I(V; M') + O(\delta \log |\mathcal{V}| + \delta \log(1/\delta))\\
&\le I_{2\varepsilon,0}(\{\mathbf{P}_v\}) + O(\delta \log |\mathcal{V}| + \delta \log(1/\delta))\\
&= O(\varepsilon^2 \|\{\mathbf{P}_v\}\|^2_{\infty\to 2}) + O(\delta \log |\mathcal{V}| + \delta \log(1/\delta))
\end{aligned}$$

The lemma follows by rewriting the final line as $O(\delta \log |\mathcal{V}|/\delta)$. □

A Family of Hard Distributions. In order to apply Theorem 3.1 to a learning or optimization problem, we need a family of distributions $\{\mathbf{P}_v\}$ such that $\|\{\mathbf{P}_v\}\|_{\infty\to 2}$ is small and any accurate algorithm for the problem distinguishes $\mathbf{P}_V^n$ from $\mathbf{U}^n$. This subsection describes one such family we will use in most of our lower bound arguments.

Let $\mathcal{X} = \{\pm 1\}^d$ be the data domain. For a parameter $\alpha \in (0, 1/2)$, a non-empty set $\ell \subseteq [d]$, and a bit $b \in \{\pm 1\}^d$, we define the distribution $\mathbf{P}_{d,\ell,b,\alpha}$ to be uniform on $\{\pm 1\}^d$ except biased so that $\mathbb{E}_{x\sim\mathbf{P}_{d,\alpha,\ell,b,\alpha}}(\prod_{i\in t} x_i) = 2\alpha b$. Its probability mass function is

$$\mathbf{P}_{d,\ell,b,\alpha}(x) = \begin{cases} (1+2\alpha)2^{-d} & \text{if } \prod_{i\in t} x_i = b \\ (1-2\alpha)2^{-d} & \text{if } \prod_{i\in t} x_i = -b \end{cases} \tag{9}$$

Note that, for every non-empty $t' \ne t$, $\mathbb{E}_{x\sim\mathbf{P}_{d,\ell,b,\alpha}}(\prod_{i\in t'} x_i) = 0$.

For dimension d, a parameter $k \le d$, and $\alpha \in (0, 1/2)$, we define the family

$$\mathcal{P}_{d,k,\alpha} = \{\mathbf{P}_{d,\ell,b,\alpha} : t \subseteq [d], |t| \in [k], b \in \{\pm 1\}\} \tag{10}$$

Fact 3.8. *The size of $\mathcal{P}_{d,k,\alpha}$ is $2 \cdot \binom{d}{\le k}$ where $\binom{d}{\le k} = \sum_{j=1}^{k} \binom{d}{j}$.*

Fact 3.9. *The uniform mixture over $\mathcal{P}_{d,k,\alpha}$ is uniform over $\mathcal{X}$.*

The following lemma is implicit in many lower bounds for local differential privacy (e.g. [19, 28, 47]), although we reprove it here for completeness.

Lemma 3.10. *For every $d \in \mathbb{N}$, $k \le d$, and $\alpha \in (0, 1/2)$,*

$$\|\mathcal{P}_{d,k,\alpha}\|^2_{\infty\to 2} \le \frac{4\alpha^2}{\binom{d}{\le k}}$$

Proof. We first expand the definition of the $(\infty \to 2)$ norm. For brevity, we write $\sup_f$ in place of $\sup_{f:\mathcal{X}\to[\pm1]}$.

$$\begin{aligned}
&\|\mathcal{P}_{d,k,\alpha}\|_{\infty\to 2}^2 \\
&= \sup_f \sum_{\mathbf{P}\in\mathcal{P}_{d,k,\alpha}} \frac{1}{|\mathcal{P}_{d,k,\alpha}|}\cdot\left(\mathop{\mathbb{E}}_{x\sim\mathbf{P}}(f(x)) - \mathop{\mathbb{E}}_{x\sim\mathbf{U}}(f(x))\right)^2 \\
&= \sup_f \sum_{\substack{t\subseteq[d],|t|\in[k]\\ b\in\{\pm1\}}} \frac{1}{|\mathcal{P}_{d,k,\alpha}|}\cdot\left(\sum_{x\in\{\pm1\}^d} f(x)\cdot(\mathbf{P}_{d,\ell,b,\alpha}(x) - \mathbf{U}(x))\right)^2 \\
&= \sup_f \frac{1}{2\binom{d}{\le k}} \sum_{\substack{t\subseteq[d],|t|\in[k]\\ b\in\{\pm1\}}} \left(\sum_{x\in\{\pm1\}^d} f(x)\cdot(\mathbf{P}_{d,\ell,b,\alpha}(x) - \mathbf{U}(x))\right)^2 \qquad (11)
\end{aligned}$$

Note that (9) is equivalent to $\mathbf{P}_{d,\ell,b,\alpha}(x) = (1 + 2\alpha b\cdot\prod_{i\in t} x_i)2^{-d}$ and, via Fact 3.9, $\mathbf{U}(x) = 2^{-d}$. Thus,

$$\begin{aligned}
(11) &= \sup_f \frac{1}{2\binom{d}{\le k}}\cdot \sum_{\substack{t\subseteq[d],|t|\in[k]\\ b\in\{\pm1\}}}\left(\sum_{x\in\{\pm1\}^d} f(x)\cdot 2\alpha b\cdot\prod_{i\in t} x_i\cdot 2^{-d}\right)^2 \\
&= \sup_f \frac{2\alpha^2}{\binom{d}{\le k}}\cdot \sum_{\substack{t\subseteq[d],|t|\in[k]\\ b\in\{\pm1\}}}\left(\sum_{x\in\{\pm1\}^d} f(x)\cdot\prod_{i\in t} x_i\cdot 2^{-d}\right)^2 \\
&= \sup_f \frac{4\alpha^2}{\binom{d}{\le k}}\cdot \sum_{t\subseteq[d],|t|\in[k]}\left(\sum_{x\in\{\pm1\}^d} f(x)\cdot\prod_{i\in t} x_i\cdot 2^{-d}\right)^2 \\
&\le \sup_f \frac{4\alpha^2}{\binom{d}{\le k}}\cdot \sum_{t\subseteq[d]}\left(\sum_{x\in\{\pm1\}^d} f(x)\cdot\prod_{i\in t} x_i\cdot 2^{-d}\right)^2 \qquad (12)
\end{aligned}$$

Define $\hat{f}(t) := \mathop{\mathbb{E}}_{X\sim\mathbf{U}}(f(X)\cdot\prod_{i\in t} X_i)$, the Fourier transform over the Boolean hypercube. This is precisely the term being squared above. So we have

$$\begin{aligned}
(12) &= \frac{4\alpha^2}{\binom{d}{\le k}}\cdot \sup_{f:\mathcal{X}\to[\pm1]} \sum_{t\subseteq[d]} \hat{f}(t)^2 \\
&= \frac{4\alpha^2}{\binom{d}{\le k}}\cdot \sup_{f:\mathcal{X}\to[\pm1]} \mathop{\mathbb{E}}_{X\sim\mathbf{U}}\left(f(X)^2\right) \qquad \text{(Parseval's identity)} \\
&\le \frac{4\alpha^2}{\binom{d}{\le k}}
\end{aligned}$$

This concludes the proof. □

The following is an immediate corollary of Theorem 3.1, Lemma 3.10, and Fact 3.8.

Theorem 3.11. *Let $\mathbf{P}_{d,L,B,2\alpha}$ denote a distribution chosen uniformly at random from $\mathcal{P}_{d,k,\alpha}$ (where L is a uniformly random subset of $[d]$ with size $\le k$ and B is a uniformly random member of $\{\pm1\}$). If M is an (ε,δ)-pan private algorithm such that $\delta\log\binom{d}{\le k}/\delta \ll \alpha^2\varepsilon^2/\binom{d}{\le k}$ and $\mathrm{d_{TV}}(M(\mathbf{P}^n_{d,L,B,\alpha}), M(\mathbf{U}^n))$ is larger than a positive constant, then*

$$n \ge \Omega\left(\frac{\sqrt{\binom{d}{\le k}}}{\alpha\varepsilon}\right)$$

4 LOWER BOUNDS FOR SIMPLE HYPOTHESIS TESTING

In this section, we use Theorem 3.11 obtain lower bounds for the problem of simple hypothesis testing. We first prove a lower bound that holds under pan-privacy, then adapt it for robust shuffle privacy via Theorem 2.6.

Definition 4.1 (d-Wise Simple Hypothesis Testing)**.** Let d be any integer larger than 1 and let α be any real in the interval $(0, 1/2)$. An algorithm M solves *d-wise simple hypothesis testing with error α and sample complexity n* if, for any set of d distributions $\mathcal{P}$ satisfying $\mathrm{d_{TV}}(\mathbf{P},\mathbf{P}') \ge \alpha$ for every distinct pair $\mathbf{P},\mathbf{P}' \in \mathcal{P}$, when given n independent samples from an arbitrary $\mathbf{P}\in\mathcal{P}$ as input, the algorithm outputs $\mathbf{P}$ with probability $\ge 99/100$. This probability is over the randomness of the samples and of M.

Theorem 4.2. *If M is an (ε,δ)-pan-private algorithm that solves d-wise simple hypothesis testing with error α and $\delta\log d/\delta \ll \alpha^2\varepsilon^2/d$, then its sample complexity is $n = \Omega(\sqrt{d}/\alpha\varepsilon)$.*

Proof. Consider the set of distributions $\{\mathbf{U}\}\cup\mathcal{P}_{d,1,\alpha}$. Note that this is a family of $2d+1$ distributions. From Fact 3.8, its size is $2d+1$. In the full version of this work, we also prove the following lower bound on pairwise distances:

Claim 4.3. *For any $\mathbf{P}\neq\mathbf{P}'\in\{\mathbf{U}\}\cup\mathcal{P}_{d,1,\alpha}$, $\mathrm{d_{TV}}(\mathbf{P},\mathbf{P}')\ge\alpha$.*

The upshot is that $\{\mathbf{U}\}\cup\mathcal{P}_{d,1,\alpha}$ is a valid set of distributions for $(2d+1)$-wise hypothesis testing. We now argue that the accuracy of M for this problem instance implies that we can invoke Theorem 3.11.

To do so, let $\mathbf{P}_{d,L,B,\alpha}$ denote a distribution chosen uniformly at random from $\mathcal{P}_{d,1,\alpha}$. We show that the total variation distance between $M(\mathbf{U}^n)$ and $M(\mathbf{P}^n_{d,L,B,\alpha})$ is at least some positive constant.

$$\begin{aligned}
&\mathrm{d_{TV}}(M(\mathbf{U}^n), M(\mathbf{P}^n_{d,L,B,\alpha})) \\
&= \max_{\mathcal{P}\subseteq\{\mathbf{U}\}\cup\mathcal{P}_{d,k,\alpha}} \left|\mathbb{P}\left(M(\mathbf{U}^n)\in\mathcal{P}\right) - \mathbb{P}\left(M(\mathbf{P}^n_{d,L,B,\alpha})\in\mathcal{P}\right)\right| \\
&\ge \mathbb{P}\left(M(\mathbf{U}^n)\in\{\mathbf{U}\}\right) - \mathbb{P}\left(M(\mathbf{P}^n_{d,L,B,\alpha})\in\{\mathbf{U}\}\right) \\
&\ge \mathbb{P}\left(M(\mathbf{U}^n)\in\{\mathbf{U}\}\right) - \frac{1}{100} \\
&\ge \frac{99}{100} - \frac{1}{100} = \frac{49}{50}
\end{aligned}$$

To obtain the second inequality, we first observe that $\mathbf{P}_{d,t,b,\alpha}\neq\mathbf{U}$ for every t,b so $\mathbf{U}$ would be an incorrect output. Then we use the fact that M solves simple hypothesis testing: it is incorrect with probability at most $1/100$. The same reasoning yields the third inequality.

From Theorem 3.11, we conclude that $n = \Omega\left(\frac{1}{\varepsilon\|\mathcal{P}_{d,1,\alpha}\|_{\infty\to2}}\right) = \Omega\left(\sqrt{d}/\alpha\varepsilon\right)$.. This lower bound holds for a family of $2d+1$ distributions, so the claimed result follows by rescaling d. □

The next theorem adapts our proof to the robust shuffle privacy setting:

Theorem 4.4. *If Π is an $(\varepsilon, \delta, 1/3)$-robustly shuffle private protocol that solves d-wise simple hypothesis testing with error α and $\delta \log d/\delta \ll \alpha^2\varepsilon^2/d$, then its sample complexity is $n = \Omega(\sqrt{d}/\alpha\varepsilon)$.*

Proof. As before, let $\mathbf{P}_{d,L,B,\alpha}$ denote a distribution chosen uniformly at random from $\mathcal{P}_{d,1,\alpha}$. Let Π denote an algorithm in the shuffle model that solves $(2d+1)$-wise simple hypothesis testing with accuracy $2\alpha/9$.

Let M^{Π} denote the (ε, δ)-pan-private algorithm guaranteed by Theorem 2.6. We will lower bound the total variation distance between $M^{\Pi}(\mathbf{U}^{n/3})$ and $M^{\Pi}(\mathbf{P}^{n/3}_{d,L,B,\alpha})$.

$$\begin{aligned}
&\mathrm{d_{TV}}(M^{\Pi}(\mathbf{U}^{n/3}), M^{\Pi}(\mathbf{P}^{n/3}_{d,L,B,\alpha})) \\
&\geq \mathbb{P}\Big(M^{\Pi}(\mathbf{U}^{n/3}) \in \{\mathbf{U}\}\Big) - \mathbb{P}\Big(M^{\Pi}(\mathbf{P}^{n/3}_{d,L,B,\alpha}) \in \{\mathbf{U}\}\Big) \\
&\geq \mathbb{P}\big(\Pi(\mathbf{U}^{n}) \in \{\mathbf{U}\}\big) - \mathbb{P}\Big(\Pi(\mathbf{P}^{n}_{d,L,B,2\alpha/9}) \in \{\mathbf{U}\}\Big) - \frac{1}{6} \\
&\qquad\qquad \text{(Theorem 2.6)} \\
&\geq \frac{49}{50} - \frac{1}{6} = \frac{61}{75}
\end{aligned}$$

The third inequality comes from repeating the analysis in the proof of Theorem 4.2. Since M^{Π} is an (ε, δ)-pan-private algorithm such that

$$\mathrm{d_{TV}}(M^{\Pi}(\mathbf{U}^{n/3}), M^{\Pi}(\mathbf{P}^{n/3}_{d,L,B,\alpha}))$$

is at least a positive constant, we invoke Theorem 3.11 to conclude that $n = \Omega(\sqrt{d}/\alpha\varepsilon)$. The claimed theorem follows by rescaling α and d. □

5 LOWER BOUNDS FOR LEARNING SIGNED PARITY FUNCTIONS

In this section, we take $\mathcal{X} = \{\pm 1\}^{d+1}$ and interpret the bits at index $d+1$ to be labels of the strings. Our focus will be on signed parity functions: given a tuple $(\ell, b) \in 2^{[d]} \times \{\pm 1\}$ and a string $x \in \mathcal{X}$, we would like labels to predict the value $b \cdot \prod_{j\in\ell} x_j$. Specifically, for any distribution $\mathbf{P}$ over $\mathcal{X}$, we define error function

$$\mathrm{err}_{\mathbf{P}}(\ell, b) := \mathop{\mathbb{P}}_{X\sim\mathbf{P}}\left(b \cdot \prod_{j\in\ell} X_j \neq X_{d+1}\right),$$

to be the probability of misclassifying a random test example.

Definition 5.1. Let $\alpha \in (0, \frac{1}{2})$ be a parameter and let $1 \leq k \leq d$ be integers. An algorithm M *learns width-k signed parities with error α and sample complexity n* if it takes n independent samples from a distribution $\mathbf{P}$ over $\mathcal{X}$ and reports a tuple $(L, B) \in 2^{[d]} \times \{\pm 1\}$ such that, with probability at least $99/100$,

$$\mathrm{err}_{\mathbf{P}}(L, B) < \min_{\ell,b} \mathrm{err}_{\mathbf{P}}(\ell, b) + \alpha.$$

This probability is taken over the randomness of the samples and over M.

For this problem, we will use a variant of our family of distributions: for a parameter $\alpha \in [0, 1/2]$, a set $\ell \subseteq [d]$, and a bit $b \in \{\pm 1\}$,

Algorithm 2: M', an online algorithm

Input: Data stream $\vec{x} \in \mathcal{X}^m$; access to online algorithm $M : \mathcal{X}^n \to 2^{[d]} \times \{\pm 1\}$
Output: A random variable $Z \in \mathbb{R}$

$S_1 \leftarrow M_1(x_1)$
For $i \in [2, n]$
  $S_i \leftarrow M_i(x_i, S_{i-1})$
For $i \in [n+1, m]$
  If $i = n+1$ **:**
    $(\hat{L}, \hat{B}) \leftarrow M_Q(S_n)$
    $C \sim \mathbf{Lap}(1/\varepsilon)$
  Else
    $(\hat{L}, \hat{B}, C) \leftarrow S_{i-1}$
  If $\prod_{j\in\hat{L}} x_{i,j} = x_{i,d+1} \cdot \hat{B}$ **:**
    $C \leftarrow C + 1$
  $S_i \leftarrow (\hat{L}, \hat{B}, C)$
$L \sim \mathbf{Lap}(1/\varepsilon)$
Return $Z \leftarrow C + L$

we define the distribution $\mathbf{Q}_{d,\ell,b,\alpha}$ to have probability mass function

$$\mathbf{Q}_{d,\ell,b,\alpha}(x) = \begin{cases} (1+2\alpha)2^{-d-1} & \text{if } b \cdot \prod_{j\in\ell} x_j = x_{d+1} \\ (1-2\alpha)2^{-d-1} & \text{if } b \cdot \prod_{j\in\ell} x_j = -x_{d+1} \end{cases} \tag{13}$$

Fact 5.2. *For any $(\ell', b') \neq (\ell, b)$,*

$$\mathop{\mathbb{P}}_{X\sim\mathbf{Q}_{d,\ell,b,\alpha}}\left(b \cdot \prod_{j\in\ell} X_j = X_{d+1}\right) = \frac{1}{2} + \alpha$$

$$\mathop{\mathbb{P}}_{X\sim\mathbf{Q}_{d,\ell,b,\alpha}}\left(b' \cdot \prod_{j\in\ell'} X_j = X_{d+1}\right) \leq \frac{1}{2}$$

For dimension d, a parameter $k \leq d$, and $\alpha \in [0, 1/2]$, we define the family

$$\mathcal{Q}_{d,k,\alpha} = \{\mathbf{Q}_{d,\ell,b,\alpha} : \ell \subseteq [d], |\ell| \leq k, b \in \{\pm 1\}\} \tag{14}$$

Fact 5.3. *The size of $\mathcal{Q}_{d,k,\alpha}$ is $2\binom{d}{\leq k} + 2$.*

Fact 5.4. *The uniform mixture of $\mathcal{Q}_{d,k,\alpha}$ is uniform over $\mathcal{X}$.*

Lemma 5.5. *For every $d \in \mathbb{N}$, $k \leq d$, and $\alpha \in [0, 1/2]$,*

$$\|\mathcal{Q}_{d,k,\alpha}\|^2_{\infty\to 2} \leq \frac{4\alpha^2}{\binom{d}{\leq k}}$$

For brevity, we defer the proof of Lemma 5.5 to the the full version of this work.

Theorem 5.6. *If $M = (M_1, \ldots, M_n, M_Q)$ is an (ε, δ)-pan-private algorithm that learns width-k signed parities with error α and*

$$\delta \log \tbinom{d}{\leq k}/\delta \ll \alpha^2\varepsilon^2 / \binom{d}{\leq k},$$

then its sample complexity is $n = \Omega(\sqrt{\binom{d}{\leq k}}/\alpha\varepsilon)$.

Proof. Analogous to previous proofs, let $\mathbf{Q}_{d,L,B,\alpha}$ denote a distribution chosen uniformly at random from $\mathcal{Q}_{d,k,\alpha}$. We argue that M implies an (ε,δ)-pan-private algorithm M' which takes $m = n+\Theta(1/\alpha\varepsilon)$ values from $\mathcal{X}$ as input and outputs a real number such that $\mathrm{d_{TV}}(M'(\mathbf{U}^m), M'(\mathbf{Q}^m_{d,L,B,\alpha}))$ is larger than a constant.

We specify M' in Algorithm 2. Although it does not explicitly have the structure in Definition 2.2, it is straightforward to decompose it into a sequence of algorithms. At a high level, M' has a training and a testing phase. In the training phase, it will execute M on the first n samples to obtain a signed parity function $(\hat{L},\hat{B})$. In the testing phase, M' will evaluate the function on the remaining samples and maintain a pan-private estimate of the number of correct predictions. If the samples are drawn from $\mathbf{U}$, then any choice of parity function makes a correct prediction with only $1/2$ probability. But if the samples are drawn from any distribution $\mathbf{Q}_{d,\ell,b,\alpha} \in \mathcal{Q}_{d,k,\alpha}$, we know that $(\hat{L},\hat{B}) = (\ell, b)$ with $\geq 99/100$ probability; conditioned on this event, our predictions will be correct with probability $1/2+\alpha$. Thus, the count of correct predictions will reliably differentiate between the two input cases.

Pan-privacy: We will first prove privacy for user i and intrusion time t. Recall that the adversary's view is $(M'_{\mathcal{I}}(\vec{x}_{\leq t}), M'_O(M'_{\mathcal{I}}(\vec{x})))$; for brevity, we shall use the notation (S_t, Z). If $i \leq n$ and $t \leq n$, the tuple is a post-processing of $(M_{\mathcal{I}}(\vec{x}_{\leq t}), M_O(M_{\mathcal{I}}(\vec{x})))$ which we know to be (ε,δ)-private. If $i \leq n$ but $t > n$, the adversary's view is a post-processing of $M_{\mathcal{I}}(\vec{x})$ which is again (ε,δ)-private.

If $i > n$ but $t \leq n$, the only influence S_t has on Z is the choice of $(\hat{L},\hat{B})$; it suffices to prove that Z is differentially private for any choice of $(\hat{L},\hat{B})$. Let $\mathbb{I}(\cdot)$ be the $\{0,1\}$ indicator function. Observe that $Z \sim \mathbf{Lap}(1/\varepsilon) + \sum_{u=n+1}^{m} \mathbb{I}(\prod_{j\in\hat{L}} x_{u,j} = x_{u,d+1}\cdot\hat{B}) + \mathbf{Lap}(1/\varepsilon)$. ε-differential privacy follows the observation that the summation is 1-sensitive and the privacy of the Laplace mechanism.

If $i > n$ and $t > n$, we consider two further cases. When $t \geq i$, observe that Z is a post-processing of S_t. Also observe that $S_t \sim \mathbf{Lap}(1/\varepsilon)+\sum_{u=n+1}^{t} \mathbb{I}(\prod_{j\in\hat{L}} x_{u,j} = x_{u,d+1}\cdot\hat{B})$. So we can again invoke the privacy of the Laplace mechanism. When $t < i$, we can show that Z is differentially private conditioned on any realization of $S_t = (\hat{L},\hat{B},C_t)$: because $Z \sim \mathbf{Lap}(1/\varepsilon) + C_t + \sum_{u=t+1}^{m} \mathbb{I}(\prod_{j\in\hat{L}} x_{u,j} = x_{u,d+1}\cdot\hat{B})$ and $i \in [t+1, m]$, we invoke the privacy of the Laplace mechanism one more.

Bound on TV distance: Now we show that the total variation distance between $M'(\mathbf{U}^m)$ and $M'(\mathbf{Q}^m_{d,L,B,\alpha})$ is larger than a constant. Notice that, for any $\tau \in \mathbb{R}$,

$$\begin{aligned}&\mathrm{d_{TV}}(M'(\mathbf{Q}^m_{d,L,B,\alpha}), M'(\mathbf{U}^m))\\ &\geq \mathbb{P}\Big(M'(\mathbf{Q}^m_{d,L,B,\alpha}) > \tau\Big) - \mathbb{P}\big(M'(\mathbf{U}^m) > \tau\big) \qquad (15)\end{aligned}$$

We will focus our attention on the first term:

$$\begin{aligned}&\mathbb{P}\Big(M'(\mathbf{Q}^m_{d,L,B,\alpha}) > \tau\Big)\\ &= \sum_{\substack{\ell\subseteq[d],|\ell|\leq k\\ b\in\{\pm1\}}} \mathbb{P}\Big(M'(\mathbf{Q}^m_{d,\ell,b,\alpha}) > \tau\Big)\cdot\mathbb{P}((L,B)=(\ell,b))\\ &= \sum_{\substack{\ell\subseteq[d],|\ell|\leq k\\ b\in\{\pm1\}}} \mathbb{P}\Big(M'(\mathbf{Q}^m_{d,\ell,b,\alpha}) > \tau \mid (\hat{L},\hat{B})=(\ell,b)\Big)\\ &\quad\cdot\mathbb{P}\Big((\hat{L},\hat{B})=(\ell,b)\Big)\cdot\mathbb{P}((L,B)=(\ell,b))\\ &\leq \sum_{\substack{\ell\subseteq[d],|\ell|\leq k\\ b\in\{\pm1\}}} \mathbb{P}\Big(M'(\mathbf{Q}^m_{d,\ell,b,\alpha}) > \tau \mid (\hat{L},\hat{B})=(\ell,b)\Big)\\ &\quad\cdot\frac{99}{100}\cdot\mathbb{P}((L,B)=(\ell,b))\end{aligned}$$

The final inequality comes from the fact that M learns parities. Notice that, conditioned on $(\hat{L},\hat{B}) = (\ell,b)$, Fact 5.2 implies $M'(\mathbf{Q}^m_{d,\ell,b,\alpha})$ is a sample from the convolution $\mathbf{Bin}(m-n, 1/2+\alpha) + \mathbf{Lap}(1/\varepsilon) + \mathbf{Lap}(1/\varepsilon)$ with probability $\geq 99/100$.

Meanwhile, note that the equality $\mathbb{P}_{X\sim\mathbf{U}}\Big(\prod_{j\in\ell} X_j = X_{d+1}\cdot b\Big) = 1/2$ holds for any parity function (ℓ,b). Consequently, the output of the algorithm $M'(\mathbf{U}^m)$ is a sample from the convolution $\mathbf{Bin}(m-n,1/2) + \mathbf{Lap}(1/\varepsilon) + \mathbf{Lap}(1/\varepsilon)$.

Because $m-n = \Theta(1/\alpha\varepsilon)$, we can use a Chernoff bound to argue that there is some τ where

$$\begin{aligned}\mathbb{P}\Big(M'(\mathbf{Q}^m_{d,\ell,b,\alpha})\Big) &\geq \frac{99}{100}\\ \mathbb{P}\big(M'(\mathbf{U}^m) > \tau\big) &\leq \frac{1}{100}\end{aligned}$$

By substitution,

$$\begin{aligned}(15) &\geq \left(\sum_{\substack{\ell\subseteq[d],|\ell|\leq k\\ b\in\{\pm1\}}} \frac{99}{100}\cdot\frac{99}{100}\cdot\mathbb{P}((L,B)=(\ell,b))\right) - \frac{1}{100}\\ &= \frac{99^2-100}{10000}\end{aligned}$$

Lemma 5.5 and Theorem 3.1 imply $m = \Omega\Big(\sqrt{\binom{d}{\leq k}}/\alpha\varepsilon\Big)$ and, in turn, $n = \Omega\Big(\sqrt{\binom{d}{\leq k}}/\alpha\varepsilon\Big)$. □

The next theorem adapts our proof to the robust shuffle privacy setting. For brevity, we defer the proof to the full version; the proof techniques are essentially the same as previously.

Theorem 5.7. *If Π is an $(\varepsilon,\delta,1/3)$-robustly shuffle private protocol that learns width-k signed parities with error α and $\delta\log\binom{d}{\leq k}/\delta \ll \alpha^2\varepsilon^2/\binom{d}{\leq k}$, then its sample complexity is $n = \Omega(\sqrt{\binom{d}{\leq k}}/\alpha\varepsilon)$.*

ACKNOWLEDGMENTS

We are grateful to Clément Canonne for many helpful discussions related to the proof of Lemma 3.2. We would also like to thank the anonymous reviewers for their constructive feedback.

REFERENCES

[1] Kareem Amin, Matthew Joseph, and Jieming Mao. 2020. Pan-Private Uniformity Testing. In *Conference on Learning Theory, COLT 2020, 9-12 July 2020, Virtual Event [Graz, Austria] (Proceedings of Machine Learning Research, Vol. 125)*, Jacob D. Abernethy and Shivani Agarwal (Eds.). PMLR, 183–218. https://arxiv.org/abs/1911.01452.

[2] Victor Balcer and Albert Cheu. 2020. Separating Local & Shuffled Differential Privacy via Histograms. In *1st Conference on Information-Theoretic Cryptography, ITC 2020, June 17-19, 2020, Boston, MA, USA (LIPIcs, Vol. 163)*, Yael Tauman Kalai, Adam D. Smith, and Daniel Wichs (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 1:1–1:14. https://doi.org/10.4230/LIPIcs.ITC.2020.1 https://arxiv.org/abs/1911.06879.

[3] Victor Balcer, Albert Cheu, Matthew Joseph, and Jieming Mao. 2021. Connecting Robust Shuffle Privacy and Pan-Privacy. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms, SODA 2021, Virtual Conference, January 10 - 13, 2021*, Dániel Marx (Ed.). SIAM, 2384–2403. https://doi.org/10.1137/1.9781611976465.142 https://arxiv.org/abs/2004.09481.

[4] Borja Balle, James Bell, Adrià Gascón, and Kobbi Nissim. 2019. The Privacy Blanket of the Shuffle Model. In *Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 11693)*, Alexandra Boldyreva and Daniele Micciancio (Eds.). Springer, 638–667. https://doi.org/10.1007/978-3-030-26951-7_22 https://arxiv.org/abs/1903.02837.

[5] Borja Balle, James Bell, Adrià Gascón, and Kobbi Nissim. 2020. Private Summation in the Multi-Message Shuffle Model. In *CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, USA, November 9-13, 2020*, Jay Ligatti, Xinming Ou, Jonathan Katz, and Giovanni Vigna (Eds.). ACM, 657–676. https://doi.org/10.1145/3372297.3417242 https://arxiv.org/abs/2002.00817.

[6] Raef Bassily and Adam Smith. 2015. Local, private, efficient protocols for succinct histograms. In *ACM Symposium on Theory of Computing (STOC '15)*. ACM, Portland, OR, USA, 127–135. https://doi.org/10.1145/2746539.2746632 https://arxiv.org/abs/1504.04686.

[7] Amos Beimel, Iftach Haitner, Kobbi Nissim, and Uri Stemmer. 2020. On the Round Complexity of the Shuffle Model. In *Theory of Cryptography - 18th International Conference, TCC 2020, Durham, NC, USA, November 16-19, 2020, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 12551)*, Rafael Pass and Krzysztof Pietrzak (Eds.). Springer, 683–712. https://doi.org/10.1007/978-3-030-64378-2_24 https://doi.org/10.1007/978-3-030-64378-2_24.

[8] Amos Beimel, Kobbi Nissim, and Eran Omri. 2008. Distributed private data analysis: Simultaneously solving how and what. In *International Cryptology Conference (CRYPTO '08)*. Springer, Santa Barbara, CA, USA, 451–468. https://doi.org/10.1007/978-3-540-85174-5_25 https://arxiv.org/abs/1103.2626.

[9] Andrea Bittau, Úlfar Erlingsson, Petros Maniatis, Ilya Mironov, Ananth Raghunathan, David Lie, Mitch Rudominer, Ushasree Kode, Julien Tinnes, and Bernhard Seefeld. 2017. PROCHLO: Strong Privacy for Analytics in the Crowd. In *ACM Symposium on Operating Systems Principles (SOSP '17)*. ACM, Shanghai, China, 441–459. https://doi.org/10.1145/3132747.3132769 https://arxiv.org/abs/1710.00901.

[10] Mark Bun, Cynthia Dwork, Guy N Rothblum, and Thomas Steinke. 2018. Composable and versatile privacy via truncated CDP. In *Annual ACM Symposium on Theory of Computing (STOC '18)*. ACM, Los Angeles, CA, USA, 74–86. https://doi.org/10.1145/3188745.3188946

[11] Mark Bun, Gautam Kamath, Thomas Steinke, and Zhiwei Steven Wu. 2019. Private Hypothesis Selection. In *Advances in Neural Information Processing Systems (NeurIPS '19)*. Vancouver, Canada, 156–167. https://arxiv.org/abs/1905.13229.

[12] Mark Bun and Thomas Steinke. 2016. Concentrated Differential Privacy: Simplifications, Extensions, and Lower Bounds. In *Theory of Cryptography - 14th International Conference, TCC 2016-B, Beijing, China, October 31 - November 3, 2016, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 9985)*, Martin Hirt and Adam D. Smith (Eds.). Beijing, China, 635–658. https://doi.org/10.1007/978-3-662-53641-4_24 https://arxiv.org/abs/1605.02065.

[13] Mark Bun, Jonathan Ullman, and Salil Vadhan. 2014. Fingerprinting Codes and the Price of Approximate Differential Privacy. In *ACM Symposium on the Theory of Computing (STOC '14)*. ACM, New York, NY, USA, 1–10. https://doi.org/10.1145/2591796.2591877 https://arxiv.org/abs/1311.3158.

[14] T.-H. Hubert Chan, Elaine Shi, and Dawn Song. 2012. Optimal Lower Bound for Differentially Private Multiparty Aggregation. In *European Symposium on Algorithms (ESA '12)*. Ljubljana, Slovenia, 277–288. https://eprint.iacr.org/2012/373.

[15] Lijie Chen, Badih Ghazi, Ravi Kumar, and Pasin Manurangsi. 2021. On Distributed Differential Privacy and Counting Distinct Elements. In *12th Innovations in Theoretical Computer Science Conference, ITCS 2021, January 6-8, 2021, Virtual Conference (LIPIcs, Vol. 185)*, James R. Lee (Ed.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 56:1–56:18. https://doi.org/10.4230/LIPIcs.ITCS.2021.56 https://arxiv.org/abs/2009.09604.

[16] Albert Cheu, Adam D. Smith, Jonathan R. Ullman, David Zeber, and Maxim Zhilyaev. 2019. Distributed Differential Privacy via Shuffling. In *Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 11476)*, Yuval Ishai and Vincent Rijmen (Eds.). Springer, Darmstadt, Germany, 375–403. https://doi.org/10.1007/978-3-030-17653-2_13 https://arxiv.org/abs/1808.01394.

[17] Irit Dinur and Kobbi Nissim. 2003. Revealing information while preserving privacy. In *Proceedings of the 22nd ACM Symposium on Principles of Database Systems (PODS '03)*. ACM, 202–210. https://doi.org/10.1145/773153.773173

[18] Jinshuo Dong, Aaron Roth, and Weijie J Su. 2019. Gaussian differential privacy. *arXiv preprint arXiv:1905.02383* (2019). https://arxiv.org/abs/1905.02383.

[19] John Duchi, Michael Jordan, and Martin Wainwright. 2013. Local Privacy and Statistical Minimax Rates. In *IEEE Symposium on Foundations of Computer Science (FOCS '13)*. IEEE Computer Society, Berkeley, CA, USA, 429–438. https://doi.org/10.1109/FOCS.2013.53 https://arxiv.org/abs/1302.3203.

[20] John Duchi and Ryan Rogers. 2019. Lower bounds for locally private estimation via communication complexity. In *Annual Conference on Learning Theory (COLT '19)*. JMLR.org, 1161–1191. https://arxiv.org/abs/1902.00582.

[21] John C. Duchi and Feng Ruan. 2018. The Right Complexity Measure in Locally Private Estimation: It is not the Fisher Information. *arXiv preprint arXiv:1806.05756* (2018).

[22] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. 2006. Our Data, Ourselves: Privacy Via Distributed Noise Generation. In *Advances in Cryptology - EUROCRYPT 2006, 25th Annual International Conference on the Theory and Applications of Cryptographic Techniques (Lecture Notes in Computer Science, Vol. 4004)*, Serge Vaudenay (Ed.). Springer, St. Petersburg, Russia, 486–503. https://doi.org/10.1007/11761679_29

[23] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating Noise to Sensitivity in Private Data Analysis. In *Conference on Theory of Cryptography (TCC '06)*. Springer, New York, NY, USA, 265–284. https://doi.org/10.1007/11681878_14

[24] Cynthia Dwork, Moni Naor, Toniann Pitassi, Guy N Rothblum, and Sergey Yekhanin. 2010. Pan-Private Streaming Algorithms. In *Innovations in Computer Science (ICS '10)*. Tsinghua University Press, Beijing, China, 66–80.

[25] Cynthia Dwork and Guy N Rothblum. 2016. Concentrated differential privacy. *arXiv preprint arXiv:1603.01887* (2016). https://arxiv.org/abs/1603.01887.

[26] Cynthia Dwork, Adam Smith, Thomas Steinke, and Jonathan Ullman. 2017. Exposed! A Survey of Attacks on Private Data. *Annual Review of Statistics and Its Application* 4 (2017), 61–84.

[27] Cynthia Dwork, Adam Smith, Thomas Steinke, Jonathan Ullman, and Salil Vadhan. 2015. Robust Traceability from Trace Amounts. In *IEEE Symposium on Foundations of Computer Science (FOCS '15)*. IEEE Computer Society, Berkeley, CA. https://doi.org/10.1109/FOCS.2015.46

[28] Alexander Edmonds, Aleksandar Nikolov, and Jonathan R. Ullman. 2020. The power of factorization mechanisms in local and central differential privacy. In *Proccedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020*, Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy (Eds.). ACM, 425–438. https://doi.org/10.1145/3357713.3384297 https://arxiv.org/abs/1911.08339.

[29] Úlfar Erlingsson, Vitaly Feldman, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Abhradeep Thakurta. 2019. Amplification by Shuffling: From Local to Central Differential Privacy via Anonymity. In *ACM-SIAM Symposium on Discrete Algorithms (SODA '19)*. ACM, San Diego, CA, USA, 2468–2479. https://doi.org/10.1137/1.9781611975482.151 https://arxiv.org/abs/1811.12469.

[30] Badih Ghazi, Noah Golowich, Ravi Kumar, Pasin Manurangsi, Rasmus Pagh, and Ameya Velingker. 2020. Pure Differentially Private Summation from Anonymous Messages. In *1st Conference on Information-Theoretic Cryptography (ITC '20)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik. https://doi.org/10.4230/LIPIcs.ITC.2020.15 https://arxiv.org/abs/2002.01919.

[31] Badih Ghazi, Noah Golowich, Ravi Kumar, Rasmus Pagh, and Ameya Velingker. 2020. On the Power of Multiple Anonymous Messages. In *Foundations of Responsible Computing (FORC '20)*. https://arxiv.org/abs/1908.11358.

[32] Badih Ghazi, Ravi Kumar, Pasin Manurangsi, and Rasmus Pagh. 2020. Private Counting from Anonymous Messages: Near-Optimal Accuracy with Vanishing Communication Overhead. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, 3505–3514. http://proceedings.mlr.press/v119/ghazi20a.html

[33] Badih Ghazi, Pasin Manurangsi, Rasmus Pagh, and Ameya Velingker. 2020. Private Aggregation from Fewer Anonymous Messages. In *Annual Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT '20)*. 798–827. https://doi.org/10.1007/978-3-030-45724-2_27 https://arxiv.org/abs/1909.11073.

[34] Badih Ghazi, Rasmus Pagh, and Ameya Velingker. 2019. Scalable and Differentially Private Distributed Aggregation in the Shuffled Model. *CoRR* abs/1906.08320 (2019). https://arxiv.org/abs/1906.08320.

[35] Sivakanth Gopi, Gautam Kamath, Janardhan Kulkarni, Aleksandar Nikolov, Zhiwei Steven Wu, and Huanyu Zhang. 2020. Locally Private Hypothesis Selection. In *Conference on Learning Theory, COLT 2020, 9-12 July 2020, Virtual Event [Graz, Austria] (Proceedings of Machine Learning Research, Vol. 125)*, Jacob D. Abernethy

and Shivani Agarwal (Eds.). PMLR, 1785–1816. https://arxiv.org/abs/2002.09465.

[36] Moritz Hardt and Guy Rothblum. 2014. A Multiplicative Weights Mechanism for Privacy-Preserving Data Analysis. In *IEEE Symposium on Foundations of Computer Science (FOCS '10)*. IEEE Computer Society, Las vegas, NV, USA, 61–70. https://doi.org/10.1109/FOCS.2010.85

[37] Matthew Joseph, Janardhan Kulkarni, Jieming Mao, and Zhiwei Steven Wu. 2018. Locally Private Gaussian Estimation. *arXiv preprint arXiv:1811.08382* (2018).

[38] Matthew Joseph, Jieming Mao, Seth Neel, and Aaron Roth. 2019. The role of interactivity in local differential privacy. In *IEEE Symposium on Foundations of Computer Science (FOCS '19)*. IEEE Computer Society, Baltimore, MD, USA, 94–105. https://doi.org/10.1109/FOCS.2019.00015 https://arxiv.org/abs/1904.03564.

[39] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. 2015. The composition theorem for differential privacy. In *International Conference on Machine Learning (ICML '15)*. PMLR, Lille, France, 1376–1385. https://arxiv.org/abs/1311.0776.

[40] Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. 2008. What Can We Learn Privately?. In *IEEE Symposium on Foundations of Computer Science (FOCS '08)*. IEEE Computer Society, Philadelphia, PA, USA, 531–540. https://doi.org/10.1109/FOCS.2008.27 https://arxiv.org/abs/0803.0924.

[41] Michael Kearns. 1998. Efficient noise-tolerant learning from statistical queries. *J. ACM* 45, 6 (1998), 983–1006. https://doi.org/10.1145/293347.293351

[42] Andrew McGregor, Ilya Mironov, Toniann Pitassi, Omer Reingold, Kunal Talwar, and Salil P. Vadhan. 2010. The Limits of Two-Party Differential Privacy. In *51th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2010, October 23-26, 2010, Las Vegas, Nevada, USA*. IEEE Computer Society, 81–90. https://doi.org/10.1109/FOCS.2010.14

[43] Frank McSherry and Kunal Talwar. 2007. Mechanism Design via Differential Privacy. In *IEEE Symposium on Foundations of Computer Science (FOCS '07)*. IEEE Computer Society, Las Vegas, NV, USA, 94–103. https://doi.org/10.1109/FOCS.2007.41

[44] Darakhshan Mir, Shan Muthukrishnan, Aleksandar Nikolov, and Rebecca N Wright. 2011. Pan-private algorithms via statistics on sketches. In *ACM Symposium on Principles of Database Systems (PODS '11)*. ACM, Athens, Greece, 37–48. https://arxiv.org/abs/1009.1544.

[45] Ilya Mironov. 2017. Rényi differential privacy. In *IEEE Computer Security Foundations Symposium (CSF '17)*. IEEE Computer Society, Santa Barbara, CA, USA, 263–275. https://arxiv.org/abs/1702.07476.

[46] Thomas Steinke and Jonathan R. Ullman. 2017. Tight Lower Bounds for Differentially Private Selection. In *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA, October 15-17, 2017 (FOCS '17)*, Chris Umans (Ed.). IEEE Computer Society, 552–563.

[47] Jonathan Ullman. 2018. Tight Bounds for Locally Differentially Private Selection. *arXiv preprint arXiv:1802.02638* (2018).