On Distributed Differential Privacy and Counting Distinct Elements

Lijie Chen[1]
Massachusetts Institute of Technology, Cambridge, MA, USA
lijieche@mit.edu

Badih Ghazi
Google Research, Mountain View, CA, USA
badihghazi@gmail.com

Ravi Kumar
Google Research, Mountain View, CA, USA
ravi.k53@gmail.com

Pasin Manurangsi
Google Research, Mountain View, CA, USA
pasin@google.com

Abstract

We study the setup where each of n users holds an element from a discrete set, and the goal is to count the number of distinct elements across all users, under the constraint of (ε, δ)-differentially privacy:

- In the non-interactive *local* setting, we prove that the additive error of any protocol is $\Omega(n)$ for any constant ε and for any δ inverse polynomial in n.
- In the *single-message shuffle* setting, we prove a lower bound of $\tilde{\Omega}(n)$ on the error for any constant ε and for some δ inverse quasi-polynomial in n. We do so by building on the moment-matching method from the literature on distribution estimation.
- In the *multi-message shuffle* setting, we give a protocol with at most one message per user in expectation and with an error of $\tilde{O}(\sqrt{n})$ for any constant ε and for any δ inverse polynomial in n. Our protocol is also robustly shuffle private, and our error of $\sqrt{n}$ matches a known lower bound for such protocols.

Our proof technique relies on a new notion, that we call *dominated protocols*, and which can also be used to obtain the first non-trivial lower bounds against multi-message shuffle protocols for the well-studied problems of selection and learning parity.

Our first lower bound for estimating the number of distinct elements provides the first $\omega(\sqrt{n})$ separation between global sensitivity and error in local differential privacy, thus answering an open question of Vadhan (2017). We also provide a simple construction that gives $\tilde{\Omega}(n)$ separation between global sensitivity and error in *two-party* differential privacy, thereby answering an open question of McGregor et al. (2011).

2012 ACM Subject Classification Security and privacy → Privacy-preserving protocols

Keywords and phrases Differential Privacy, Shuffle Model

Digital Object Identifier 10.4230/LIPIcs.ITCS.2021.56

Related Version A full version of the paper is available at `https://arxiv.org/abs/2009.09604`.

Acknowledgements We would like to thank Noah Golowich for numerous enlightening discussions about lower bounds in the multi-message $\mathrm{DP}_{\mathrm{shuffle}}$ model, and for helpful feedback. We also want to thank the anynomous ITCS reviewers for their helpful comments.

[1] Most of this work was done at Google Research, Mountain View, CA.

1 Introduction

Differential privacy (DP) [20, 19] has become a leading framework for private-data analysis, with several recent practical deployments [25, 39, 28, 3, 16, 1]. The most commonly studied DP setting is the so-called central (aka curator) model whereby a single authority (sometimes referred to as the analyst) is trusted with running an algorithm on the raw data of the users and the privacy guarantee applies to the algorithm's output.

The absence, in many scenarios, of a clear trusted authority has motivated the study of *distributed* DP models. The most well-studied such setting is the *local* model [31] (also [44]), denoted henceforth by $\mathrm{DP}_{\mathrm{local}}$, where the privacy guarantee is enforced at each user's output (i.e., the protocol transcript). While an advantage of the local model is its very strong privacy guarantees and minimal trust assumptions, the noise that has to be added can sometimes be quite large. This has stimulated the study of "intermediate" models that seek to achieve accuracy close to the central model while relying on more distributed trust assumptions. One such middle-ground is the so-called *shuffle* (aka *anonymous*) model [29, 8, 12, 24], where the users send messages to a *shuffler* who randomly shuffles these messages before sending them to the analyzer; the privacy guarantee is enforced on the shuffled messages (i.e., the input to the analyzer). We study both the local and the shuffle models in this work.

1.1 Counting Distinct Elements

A basic function in data analytics is estimating the number of distinct elements in a domain of size D held by a collection of n users, which we denote by $\mathsf{CountDistinct}_{n,D}$ (and simply by $\mathsf{CountDistinct}_n$ if there is no restriction on the universe size). Beside its use in database management systems, it is a well-studied problem in sketching, streaming, and communication complexity (e.g., [30, 9] and the references therein). In central DP, it can be easily solved with constant error using the Laplace mechanism [20]; see also [36, 15, 38, 14].

We obtain new results on (ε, δ)-DP protocols for $\mathsf{CountDistinct}$ in the local and shuffle settings[2].

1.1.1 Lower Bounds for Local DP Protocols

Our first result is a lower bound on the additive error of $\mathrm{DP}_{\mathrm{local}}$ protocols[3] for counting distinct elements.

▶ **Theorem 1.** *For any $\varepsilon = O(1)$, no public-coin $(\varepsilon, o(1/n))$-$\mathrm{DP}_{\mathrm{local}}$ protocol can solve*[4] *$\mathsf{CountDistinct}_{n,n}$ with error $o(n)$.*

The lower bound in Theorem 1 is asymptotically tight[5]. Furthermore, it answers a question of Vadhan [42, Open Problem 9.6], who asked if there is a function with a gap of $\omega(\sqrt{n})$ between its (global) sensitivity and the smallest achievable error by any $\mathrm{DP}_{\mathrm{local}}$

---

[2] For formal definitions, please refer to Section 2. We remark that, throughout this work, we consider the *non-interactive* local model where all users apply the *same* randomizer (see Definition 15). We briefly discuss in Section 1.4 possible extensions to interactive local models. See the full version for how to generalize our results to the relaxed setting where each user can apply different randomizers to their inputs.

[3] See Section 2 for the the formal (standard) definition of public-coin DP protocols. Note that private-coin protocols are a sub-class of public-coin protocols, so all of our lower bounds apply to private-coin protocols as well.

[4] Throughout this work, we say that a randomized algorithm solves a problem with error e if with probability 0.99 it incurs error at most e.

[5] The trivial algorithm that always outputs 0 incurs an error n.

protocol.[6] As the global sensitivity of the number of distinct elements is 1, Theorem 1 exhibits a (natural) function for which this gap is as large as $\Omega(n)$. While Theorem 1 applies to the constant ε regime, it turns out we can prove a lower bound for much less private protocols (i.e., having a much larger ε value) at the cost of polylogarithmic factors in the error:

▶ **Theorem 2.** *For some $\varepsilon = \ln(n) - O(\ln\ln n)$ and $D = \Theta(n/\operatorname{polylog}(n))$, no public-coin $(\varepsilon, n^{-\omega(1)})$-$\mathrm{DP}_{\mathrm{local}}$ protocol can solve $\mathsf{CountDistinct}_{n,D}$ with error $o(D)$.*

To prove Theorem 2, we build on the *moment matching* method from the literature on (non-private) distribution estimation, namely [43, 45], and tailor it to CountDistinct in the $\mathrm{DP}_{\mathrm{local}}$ setting (see Section 3.1 for more details on this connection). The bound on the privacy parameter ε in Theorem 2 turns out to be very close to tight: the error drops quadratically when ε exceeds $\ln n$. This is shown in the next theorem:

▶ **Theorem 3.** *There is a $(\ln(n) + O(1))$-$\mathrm{DP}_{\mathrm{local}}$ protocol solving $\mathsf{CountDistinct}_{n,n}$ with error $O(\sqrt{n})$.*

1.1.2 Lower Bounds for Single-Message Shuffle DP Protocols

In light of the negative result in Theorem 2, a natural question is whether CountDistinct can be solved in a weaker distributed DP setting such as the shuffle model. It turns out that this is not possible using any shuffle protocol where each user sends no more than 1 message (for brevity, we will henceforth denote this class by $\mathrm{DP}^1_{\mathrm{shuffle}}$, and more generally denote by $\mathrm{DP}^k_{\mathrm{shuffle}}$ the variant where each user can send up to k messages). Note that the class $\mathrm{DP}^1_{\mathrm{shuffle}}$ includes any method obtained by taking a $\mathrm{DP}_{\mathrm{local}}$ protocol and applying the so-called *amplification by shuffling* results of [24, 6].

In the case where ε is any constant and δ is inverse quasi-polynomial in n, the improvement in the error for $\mathrm{DP}^1_{\mathrm{shuffle}}$ protocols compared to $\mathrm{DP}_{\mathrm{local}}$ is at most polylogarithmic factors:

▶ **Theorem 4.** *For all $\varepsilon = O(1)$, there are $\delta = 2^{-\operatorname{polylog}(n)}$ and $D = n/\operatorname{polylog}(n)$ such that no public-coin (ε, δ)-$\mathrm{DP}^1_{\mathrm{shuffle}}$ protocol can solve $\mathsf{CountDistinct}_{n,D}$ with error $o(D)$.*

We note that Theorem 4 essentially answers a more general variant of Vadhan's question: it shows that even for $\mathrm{DP}^1_{\mathrm{shuffle}}$ protocols (which include $\mathrm{DP}_{\mathrm{local}}$ protocols as a sub-class) the gap between sensitivity and the error can be as large as $\tilde{\Omega}(n)$.

The proof of Theorem 4 follows by combining Theorem 2 with the following connection between $\mathrm{DP}_{\mathrm{local}}$ and $\mathrm{DP}^1_{\mathrm{shuffle}}$:

▶ **Lemma 5.** *For any $\varepsilon = O(1)$ and $\delta \le \delta_0 \le 1/n$, if the randomizer R is (ε, δ)-$\mathrm{DP}^1_{\mathrm{shuffle}}$ on n users, then R is $\left(\ln n - \ln(\Theta_\varepsilon(\log \delta_0^{-1} / \log \delta^{-1})), \delta_0\right)$-$\mathrm{DP}_{\mathrm{local}}$.*

We remark that Lemma 5 provides a stronger quantitative bound than the qualitatively similar connections in [12, 27]; specifically, we obtain the term $\ln(\Theta_\varepsilon(\log \delta_0^{-1} / \log \delta^{-1}))$, which was not present in the aforementioned works. This turns out to be crucial for our purposes, as this term gives the $O(\ln\ln n)$ term necessary to apply Theorem 2.

[6] To the best of our knowledge, the largest previously known gap between global sensitivity and error was $O(\sqrt{n})$, which is achieved, e.g., by binary summation [11]. For CountDistinct, the lower bound of [21] on pan-private algorithms against two intrusions along with the equivalence shown by [2] between this model and sequential local DP, imply a lower bound of $\Omega(n)$ against *pure* DP protocols. A lower bound against approximate DP protocols can then be obtained via the transformation of [10]; however, this lower bound would only hold for an ε bounded strictly below one (e.g., 1/4), whereas our lower bound in Theorem 1 holds for ε an arbitrarily large constant.

1.1.3 A Communication-Efficient Shuffle DP Protocol

In contrast with Theorem 4, Balcer et al. [5] recently gave a $\mathrm{DP}_{\mathrm{shuffle}}$ protocol for $\mathsf{CountDistinct}_{n,D}$ with error $O(\sqrt{D})$. Their protocol sends $\Omega(D)$ messages per user. We instead show that an error of $\tilde{O}(\sqrt{D})$ can still be guaranteed with each user sending *in expectation* at most one message each of length $O(\log D)$ bits.

▶ **Theorem 6.** *For all $\varepsilon \leq O(1)$ and $\delta \leq 1/n$, there is a public-coin (ε, δ)-$\mathrm{DP}_{\mathrm{shuffle}}$ protocol that solves $\mathsf{CountDistinct}_n$ with error $\sqrt{\min(n, D)} \cdot \mathrm{poly}(\log(1/\delta)/\varepsilon)$ where the expected number of messages sent by each user is at most one.*

In the special case where $D = o(n/\operatorname{poly}(\varepsilon^{-1}\log(\delta^{-1})))$, we moreover obtain a *private-coin* $\mathrm{DP}_{\mathrm{shuffle}}$ protocol achieving the same guarantees as in Theorem 6 (see the full version for a formal statement). Note that Theorem 6 is in sharp contrast with the lower bound shown in Theorem 4 for $\mathrm{DP}^1_{\mathrm{shuffle}}$ protocols. Indeed, for δ inverse quasi-polynomial in n, the former implies a public-coin protocol with less than one message per-user *in expectation* having error $\tilde{O}(\sqrt{n})$ whereas the latter proves that no such protocol exists, even with error as large as $\tilde{\Omega}(n)$, if we restrict each user to send one message *in the worst case*.

A strengthening of $\mathrm{DP}_{\mathrm{shuffle}}$ protocols called *robust* $\mathrm{DP}_{\mathrm{shuffle}}$ *protocols*[7] was studied by [5], who proved an $\Omega\left(\sqrt{\min(D, n)}\right)$ lower bound on the error of any protocol solving $\mathsf{CountDistinct}_{n,D}$. Our protocols are robust $\mathrm{DP}_{\mathrm{shuffle}}$ and, therefore, achieve the optimal error (up to polylogarithmic factors) among all robust $\mathrm{DP}_{\mathrm{shuffle}}$ protocols, while only sending at most one message per user in expectation.

1.2 Dominated Protocols and Multi-Message Shuffle DP Protocols

The technique underlying the proof of Theorem 1 can be extended beyond $\mathrm{DP}_{\mathrm{local}}$ protocols for $\mathsf{CountDistinct}$. It applies to a broader category of protocols that we call *dominated*, defined as follows:

▶ **Definition 7.** *We say that a randomizer $R \colon \mathcal{X} \to \mathcal{M}$ is* (ε, δ)-dominated*, if there exists a distribution $\mathcal{D}$ on $\mathcal{M}$ such that for all $x \in \mathcal{X}$ and all $E \subseteq \mathcal{M}$,*

$$\Pr[R(x) \in E] \leq e^{\varepsilon} \cdot \Pr_{\mathcal{D}}[E] + \delta.$$

In this case, we also say R is (ε, δ)-dominated by $\mathcal{D}$. We define (ε, δ)-dominated protocols in the same way as (ε, δ)-$\mathrm{DP}_{\mathrm{local}}$, except that we require the randomizer to be (ε, δ)-dominated instead of being (ε, δ)-DP.

Note that an (ε, δ)-$\mathrm{DP}_{\mathrm{local}}$ randomizer R is (ε, δ)-dominated: we can fix a $y^* \in \mathcal{X}$ and take $\mathcal{D} = R(y^*)$. Therefore, our new definition is a relaxation of $\mathrm{DP}_{\mathrm{local}}$.

We show that *multi-message* $\mathrm{DP}_{\mathrm{shuffle}}$ protocols are dominated, which allows us to prove the first non-trivial lower bounds against $\mathrm{DP}^{O(1)}_{\mathrm{shuffle}}$ protocols.

Before formally stating this connection, we recall why known lower bounds against $\mathrm{DP}^1_{\mathrm{shuffle}}$ protocols [12, 27, 4] do not extend to $\mathrm{DP}^{O(1)}_{\mathrm{shuffle}}$ protocols.[8] These prior works use the connection stating that any (ε, δ)-$\mathrm{DP}^1_{\mathrm{shuffle}}$ protocol is also $(\varepsilon + \ln n, \delta)$-$\mathrm{DP}_{\mathrm{local}}$ [12,

[7] Roughly speaking, they are $\mathrm{DP}_{\mathrm{shuffle}}$ protocols whose transcript remains private even if a constant fraction of users drop out from the protocol.

[8] We remark that [26] developed a technique for proving lower bounds on the *communication complexity* (i.e., the number of bits sent per user) for multi-message protocols. Their techniques do not apply to our setting as our lower bounds are in terms of the number of messages, and do not put any restriction on the message length. Furthermore, their technique only applies to *pure*-DP where $\delta = 0$, whereas ours applies also to *approximate*-DP where $\delta > 0$.

Theorem 6.2]. It thus suffices for them to prove lower bounds for $\mathrm{DP}_{\mathrm{local}}$ protocols with low privacy requirement (i.e., $(\varepsilon + \ln n, \delta)$-$\mathrm{DP}_{\mathrm{local}}$), for which lower bound techniques are known or developed. For ε-$\mathrm{DP}^1_{\mathrm{shuffle}}$ protocols, [4] showed that they are also ε-$\mathrm{DP}_{\mathrm{local}}$; therefore, lower bounds on $\mathrm{DP}_{\mathrm{local}}$ protocols automatically translate to lower bounds on pure-$\mathrm{DP}^1_{\mathrm{shuffle}}$ protocols. To apply this proof framework to $\mathrm{DP}^{O(1)}_{\mathrm{shuffle}}$ protocols, a natural first step would be to connect $\mathrm{DP}^{O(1)}_{\mathrm{shuffle}}$ protocols to $\mathrm{DP}_{\mathrm{local}}$ protocols. However, as observed by [4, Section 4.1], there exists an ε-$\mathrm{DP}^{O(1)}_{\mathrm{shuffle}}$ protocol that is not $\mathrm{DP}_{\mathrm{local}}$ for any privacy parameter. That is, there is no analogous connection between $\mathrm{DP}_{\mathrm{local}}$ protocols and multi-message $\mathrm{DP}_{\mathrm{shuffle}}$ protocols, even if the latter can only send $O(1)$ messages per user.

In contrast, the next lemma captures the connection between multi-message $\mathrm{DP}_{\mathrm{shuffle}}$ and dominated protocols.

▶ **Lemma 8.** *If R is (ε, δ)-$\mathrm{DP}^k_{\mathrm{shuffle}}$ on n users, then it is $(\varepsilon + k(1 + \ln n), \delta)$-dominated.*

By considering dominated protocols and using Lemma 8, we obtain the first lower bounds for *multi-message* $\mathrm{DP}_{\mathrm{shuffle}}$ protocols for two well-studied problems: Selection and ParityLearning.

1.2.1 Lower Bounds for Selection

The Selection problem on n users is defined as follows. The ith user has an input $x_i \in \{0,1\}^D$ and the goal is to output an index $j \in [D]$ such that $\sum_{i=1}^{n} x_{i,j} \geq \left(\max_{j^*} \sum_{i=1}^{n} x_{i,j^*}\right) - n/10$. Selection is well-studied in DP (e.g., [17, 40, 41]) and its variants are useful primitives for several statistical and algorithmic problems including feature selection, hypothesis testing and clustering. In central DP, the exponential mechanism of [35] yields an ε-DP algorithm for Selection when $n = O_\varepsilon(\log D)$. On the other hand, it is known that any (ε, δ)-$\mathrm{DP}_{\mathrm{local}}$ protocol for Selection with $\varepsilon = O(1)$ and $\delta = O(1/n^{1.01})$ requires $n = \Omega(D \log D)$ users [41]. Moreover, [12] obtained a $(\varepsilon, 1/n^{O(1)})$-$\mathrm{DP}^D_{\mathrm{shuffle}}$ protocol for $n = \tilde{O}_\varepsilon(\sqrt{D})$. By contrast, for $\mathrm{DP}^1_{\mathrm{shuffle}}$ protocols, a lower bound of $\Omega(D^{1/17})$ was obtained in [12] and improved to $\Omega(D)$ in [27].

The next theorem give a lower bounds for Selection that holds against approximate-$\mathrm{DP}^k_{\mathrm{shuffle}}$ protocols. To the best of our knowledge, this is the first lower bound even for $k = 2$ (and even for the special case of pure protocols, where $\delta = 0$).

▶ **Theorem 9.** *For any $\varepsilon = O(1)$, any public-coin $(\varepsilon, o(1/D))$-$\mathrm{DP}^k_{\mathrm{shuffle}}$ protocol that solves Selection requires $n \geq \Omega\left(\frac{D}{k}\right)$.*

We remark that combining the advanced composition theorem for DP and known $\mathrm{DP}_{\mathrm{shuffle}}$ aggregation algorithms, one can obtain a $(\varepsilon, 1/\operatorname{poly}(n))$-$\mathrm{DP}^k_{\mathrm{shuffle}}$ protocol for Selection with $\tilde{O}(D/\sqrt{k})$ samples for any $k \leq D$ (see the full version for details).

1.2.2 Lower Bounds for Parity Learning

In ParityLearning, there is a hidden random vector $s \in \{0,1\}^D$, each user gets a random vector $x \in \{0,1\}^D$ together with the inner product $\langle s, x\rangle$ over $\mathbb{F}_2$, and the goal is to recover s. This problem is well-known for separating PAC learning from the Statistical Query (SQ) learning model [32]. In DP, it was studied by [31] who gave a central DP protocol (also based on the exponential mechanism) computing it for $n = O(D)$, and moreover proved a lower bound of $n = 2^{\Omega(D)}$ for any $\mathrm{DP}_{\mathrm{local}}$ protocol, thus obtaining the first exponential separation between the central and local settings.

We give a lower bound for ParityLearning that hold against approximate-$\mathrm{DP}^{k}_{\mathrm{shuffle}}$ protocols:

▶ **Theorem 10.** *For any $\varepsilon = O(1)$, if P is a public-coin $(\varepsilon, o(1/n))$-$\mathrm{DP}^{k}_{\mathrm{shuffle}}$ protocol that solves ParityLearning with probability at least 0.99, then $n \geq \Omega(2^{D/(k+1)})$.*

Our lower bounds for ParityLearning can be generalized to the Statistical Query (SQ) learning framework of [32] (see the full version for more details).

Independent Work

In a recent concurrent work, Cheu and Ullman [13] proved that robust $\mathrm{DP}_{\mathrm{shuffle}}$ protocols solving Selection and ParityLearning require $\Omega(\sqrt{D})$ and $\Omega(2^{\sqrt{D}})$ samples, respectively. Their results have no restriction on the number of messages sent by each user, but they only hold against the special case of *robust* protocols. Our results provide stronger lower bounds when the number of messages per user is less than $\sqrt{D}$, and apply to the most general $\mathrm{DP}_{\mathrm{shuffle}}$ model without the robustness restriction.

1.3 Lower Bounds for Two-Party DP Protocols

Finally, we consider another model of distributed DP, called the *two-party* model [33], denoted $\mathrm{DP}_{\text{two-party}}$. In this model, there are two parties, each holding part of the dataset. The DP guarantee is enforced on the view of each party (i.e., the transcript, its private randomness, and its input). See the full version for a formal treatment.

McGregor et al. [33] studied the $\mathrm{DP}_{\text{two-party}}$ and proved an interesting separation of $\Omega_\varepsilon(n)$ between the global sensitivity and ε-DP protocol in this model. However, this lower bound does not extend to the approximate-DP case (where $\delta > 0$); in this case, the largest known gap (also proved in [33]) is only $\tilde{\Omega}_\varepsilon(\sqrt{n})$, and it was left as an open question if this can be improved[9]. We answer this question by showing that the gap of $\tilde{\Omega}_\varepsilon(n)$ holds even against approximate-DP protocols:

▶ **Theorem 11.** *For any $\varepsilon = O(1)$ and any sufficiently large $n \in \mathbb{N}$, there is a function $f\colon \{0,1\}^{2n} \to \mathbb{R}$ whose global sensitivity is one and such that no $(\varepsilon, o(1/n))$-$\mathrm{DP}_{\text{two-party}}$ protocol can compute f to within an error of $o(n/\log n)$.*

The above bound is tight up to a logarithmic factors in n, as it is trivial to achieve an error of n.

The proof of Theorem 11 is unlike others in the paper; in fact, we only employ simple reductions starting from the hardness of inner product function already shown in [33]. Specifically, our function is a sum of blocks of inner product modulo 2. While this function is not symmetric, we show that it can be easily symmetrized (see the full version for details).

1.4 Discussions and Open Questions

In this work, we study DP in distributed models, including the local and shuffle settings. By building on the moment matching method and using the newly defined notion of dominated protocols, we give novel lower bounds in both models for three fundamental problems: CountDistinct, Selection, and ParityLearning. While our lower bounds are (nearly) tight in a large setting of parameters, there are still many interesting open questions, three of which we highlight below:

[9] The conference version of the paper [33] actually claimed to also have a lower bound $\Omega_\varepsilon(n)$ for the approximate-DP case as well. However, it was later found to be incorrect; see [34] for more discussions.

- **$\mathrm{DP}_{\mathrm{shuffle}}$ Lower Bounds for Protocols with Unbounded Number of Messages.** Our connection between $\mathrm{DP}_{\mathrm{shuffle}}$ and dominated protocols becomes weaker as $k \to \infty$ (Lemma 8). As a result, it cannot be used to establish lower bounds against $\mathrm{DP}_{\mathrm{shuffle}}$ protocols with a possibly unbounded number of messages. In fact, we are not aware of any separation between central DP and $\mathrm{DP}_{\mathrm{shuffle}}$ without a restriction on the number of messages and without the robustness restriction. This remains a fundamental open question. (In contrast, separations between central DP and $\mathrm{DP}_{\mathrm{local}}$ are well-known, even for basic functions such as binary summation [11].)
- **Lower Bounds against Interactive Local/Shuffle Model.** Our lower bounds hold in the *non-interactive* local and shuffle DP models, where all users send their messages simultaneously in a single round. While it seems plausible that our lower bounds can be extended to the *sequentially interactive* local DP model [17] (where each user speaks once but not simultaneously), it is unclear how to extend them to the fully interactive local DP model.
  The situation for $\mathrm{DP}_{\mathrm{shuffle}}$ however is more complicated. We remark that certain definitions could lead to the model being as powerful as the central model (in terms of achievable accuracy and putting aside communication constraints); see e.g., [29] on how to perform secure computations under a certain definition of the shuffle model. A very recent work provides a formal treatment of an interactive setting for the shuffle model [7].
- **$\mathrm{DP}^{1}_{\mathrm{shuffle}}$ Lower Bounds for CountDistinct with Larger δ.** All but one of our lower bounds hold as long as $\delta = n^{-\omega(1)}$, which is a standard assumption in the DP literature. The only exception is that of Theorem 4, which requires $\delta = 2^{-\Omega(\log^c n)}$ for some constant $c > 0$. It is interesting whether this can be relaxed to $\delta = n^{-\omega(1)}$.

2 Preliminaries

2.1 Notation

For a function $f \colon \mathcal{X} \to \mathbb{R}$, a distribution $\mathcal{D}$ on $\mathcal{X}$, and an element $z \in \mathcal{X}$, we use $f(\mathcal{D})$ to denote $\mathop{\mathbb{E}}_{x \leftarrow \mathcal{D}}[f(x)]$ and $\mathcal{D}_z$ to denote $\Pr_{x \leftarrow \mathcal{D}}[x = z]$. For a subset $E \subseteq \mathcal{X}$, we use $\mathcal{D}_E$ to denote $\sum_{z \in E} \mathcal{D}_z = \Pr_{x \leftarrow \mathcal{D}}[x \in E]$. We also use $\mathcal{U}_D$ to denote the uniform distribution over $\{0,1\}^D$.

For two distributions $\mathcal{D}_1$ and $\mathcal{D}_2$ on sets $\mathcal{X}$ and $\mathcal{Y}$ respectively, we use $\mathcal{D}_1 \otimes \mathcal{D}_2$ to denote their product distribution over $\mathcal{X} \times \mathcal{Y}$. For two random variables X and Y supported on $\mathbb{R}^D$ for $D \in \mathbb{N}$, we use $X + Y$ to denote the random variable distributed as a sum of two independent samples from X and Y. For any set $\mathcal{S}$, we denote by $\mathcal{S}^*$ the set consisting of sequences on $\mathcal{S}$, i.e., $\mathcal{S}^* = \cup_{n \ge 0} \mathcal{S}^n$. For $x \in \mathbb{R}$, let $[x]_+$ denote $\max(x, 0)$. For a predicate P, we use $\mathbb{1}[P]$ to denote the corresponding Boolean value of P, that is, $\mathbb{1}[P] = 1$ if P is true, and 0 otherwise.

For a distribution $\mathcal{D}$ on a finite set $\mathcal{X}$ and an event $\mathcal{E} \subseteq \mathcal{X}$ such that $\Pr_{z \leftarrow \mathcal{D}}[z \in \mathcal{E}] > 0$, we use $\mathcal{D}|\mathcal{E}$ to denote the conditional distribution such that

$$(\mathcal{D}|\mathcal{E})_z = \begin{cases} \frac{\mathcal{D}_z}{\Pr_{z \leftarrow \mathcal{D}}[z \in \mathcal{E}]} & \text{if } z \in \mathcal{E}, \\ 0 & \text{otherwise.} \end{cases}$$

Slightly overloading the notation, we also use $\alpha \cdot \mathcal{D}_1 + (1-\alpha) \cdot \mathcal{D}_2$ to denote the mixture of distributions $\mathcal{D}_1$ and $\mathcal{D}_2$ with mixing weights α and $(1-\alpha)$ respectively. Whether $+$ means mixture or convolution will be clear from the context unless explicitly stated.

2.2 Differential Privacy

We now recall the basics of differential privacy that we will need. Fix a finite set $\mathcal{X}$, the space of user reports. A *dataset* X is an element of $\mathcal{X}^*$, namely a tuple consisting of elements of $\mathcal{X}$. Let $\mathrm{hist}(X) \in \mathbb{N}^{|\mathcal{X}|}$ be the *histogram* of X: for any $x \in \mathcal{X}$, the xth component of $\mathrm{hist}(X)$ is the number of occurrences of x in the dataset X. We will consider datasets X, X' to be *equivalent* if they have the same histogram (i.e., the ordering of the elements $x_1, \ldots, x_n$ does not matter). For a multiset $\mathcal{S}$ whose elements are in $\mathcal{X}$, we will also write $\mathrm{hist}(\mathcal{S})$ to denote the histogram of $\mathcal{S}$ (so that the xth component is the number of copies of x in $\mathcal{S}$).

Let $n \in \mathbb{N}$, and consider a dataset $X = (x_1, \ldots, x_n) \in \mathcal{X}^n$. For an element $x \in \mathcal{X}$, let $f_X(x) = \frac{\mathrm{hist}(X)_x}{n}$ be the *frequency* of x in X, namely the fraction of elements of X that are equal to x. Two datasets X, X' are said to be *neighboring* if they differ in a single element, meaning that we can write (up to equivalence) $X = (x_1, x_2, \ldots, x_n)$ and $X' = (x_1', x_2, \ldots, x_n)$. In this case, we write $X \sim X'$. Let $\mathcal{Z}$ be a set; we now define the differential privacy of a randomized function $P \colon \mathcal{X}^n \to \mathcal{Z}$ as follows.

▶ **Definition 12** (Differential privacy (DP) [20, 19]). *A randomized algorithm $P \colon \mathcal{X}^n \to \mathcal{Z}$ is (ε, δ)-DP if for every pair of neighboring datasets $X \sim X'$ and for every set $\mathcal{S} \subseteq \mathcal{Z}$, we have*

$$\Pr[P(X) \in \mathcal{S}] \leq e^{\varepsilon} \cdot \Pr[P(X') \in \mathcal{S}] + \delta,$$

where the probabilities are taken over the randomness in P. Here, $\varepsilon \geq 0$ and $\delta \in [0, 1]$.

If $\delta = 0$, then we use ε-DP for brevity and informally refer to it as *pure-DP*; if $\delta > 0$, we refer to it as *approximate-DP*. We will use the following post-processing property of DP.

▶ **Lemma 13** (Post-processing, e.g., [22]). *If P is (ε, δ)-DP, then for every randomized function A, the composed function $A \circ P$ is (ε, δ)-DP.*

2.3 Shuffle Model

We briefly review the *shuffle model* of DP [8, 24, 12]. The input to the model is a dataset $(x_1, \ldots, x_n) \in \mathcal{X}^n$, where item $x_i \in \mathcal{X}$ is held by user i. A protocol $P \colon \mathcal{X} \to \mathcal{Z}$ in the shuffle model consists of three algorithms:

- The *local randomizer* $R \colon \mathcal{X} \to \mathcal{M}^*$ takes as input the data of one user, $x_i \in \mathcal{X}$, and outputs a sequence $(y_{i,1}, \ldots, y_{i,m_i})$ of *messages*; here m_i is a positive integer.
  To ease discussions in the paper, we will further assume that the randomizer R pre-shuffles its messages. That is, it applies a random permutation $\pi \colon [m_i] \to [m_i]$ to the sequence $(y_{i,1}, \ldots, y_{i,m_i})$ before outputting it.[10]
- The *shuffler* $S \colon \mathcal{M}^* \to \mathcal{M}^*$ takes as input a sequence of elements of $\mathcal{M}$, say $(y_1, \ldots, y_m)$, and outputs a random permutation, i.e., the sequence $(y_{\pi(1)}, \ldots, y_{\pi(m)})$, where $\pi \in S_m$ is a uniformly random permutation on $[m]$. The input to the shuffler will be the concatenation of the outputs of the local randomizers.
- The *analyzer* $A \colon \mathcal{M}^* \to \mathcal{Z}$ takes as input a sequence of elements of $\mathcal{M}$ (which will be taken to be the output of the shuffler) and outputs an answer in $\mathcal{Z}$ that is taken to be the output of the protocol P.

[10] Therefore, for every $x \in \mathcal{X}$ and any two tuples $z_1, z_2 \in \mathcal{M}^*$ that are equivalent up to a permutation, $R(x)$ outputs them with the same probability.

We will write $P = (R, S, A)$ to denote the protocol whose components are given by R, S, and A. The main distinction between the shuffle and local model is the introduction of the shuffler S between the local randomizer and the analyzer. As in the local model, the analyzer is untrusted in the shuffle model; hence privacy must be guaranteed with respect to the input to the analyzer, i.e., the output of the shuffler. Formally, we have:

▶ **Definition 14** (DP in the Shuffle Model, [24, 12]). *A protocol $P = (R, S, A)$ is (ε, δ)-DP if, for any dataset $X = (x_1, \ldots, x_n)$, the algorithm*

$$(x_1, \ldots, x_n) \mapsto S(R(x_1), \ldots, R(x_n))$$

is (ε, δ)-DP.

Notice that the output of $S(R(x_1), \ldots, R(x_n))$ can be simulated by an algorithm that takes as input the *multiset* consisting of the union of the elements of $R(x_1), \ldots, R(x_n)$ (which we denote as $\bigcup_i R(x_i)$, with a slight abuse of notation) and outputs a uniformly random permutation of them. Thus, by Lemma 13, it can be assumed without loss of generality for privacy analyses that the shuffler simply outputs the multiset $\bigcup_i R(x_i)$. For the purpose of analyzing the accuracy of the protocol $P = (R, S, A)$, we define its *output* on the dataset $X = (x_1, \ldots, x_n)$ to be $P(X) := A(S(R(x_1), \ldots, R(x_n)))$. We also remark that the case of *local DP*, formalized in Definition 15, is a special case of the shuffle model where the shuffler S is replaced by the identity function:

▶ **Definition 15** (Local DP [31]). *A protocol $P = (R, A)$ is (ε, δ)-DP in the local model (or (ε, δ)-locally DP) if the function $x \mapsto R(x)$ is (ε, δ)-DP.*

We say that the *output* of the protocol P on an input dataset $X = (x_1, \ldots, x_n)$ is $P(X) := A(R(x_1), \ldots, R(x_n))$.

We denote DP in the shuffle model by $\mathrm{DP}_{\text{shuffle}}$, and the special case where each user can send at most[11] k messages by $\mathrm{DP}^k_{\text{shuffle}}$. We denote DP in the local model by $\mathrm{DP}_{\text{local}}$.

Public-Coin DP

The default setting for local and shuffle models is private-coin, i.e., there is no randomness shared between the randomizers and the analyzer. We will also study the public-coin variants of the local and shuffle models. In the public-coin setting, each local randomizer also takes a public random string $\alpha \leftarrow \{0, 1\}^*$ as input. The analyzer is also given the public random string α. We use $R_\alpha(x)$ to denote the local randomizer with public random string being fixed to α. At the start of the protocol, all users jointly sample a public random string from a publicly known distribution $\mathcal{D}_{\mathsf{pub}}$.

Now, we say that a protocol $P = (R, A)$ is (ε, δ)-*DP in the public-coin local model*, if the function

$$x \underset{\alpha \leftarrow \mathcal{D}_{\mathsf{pub}}}{\mapsto} (\alpha, R_\alpha(x))$$

is (ε, δ)-DP.

[11] We may assume w.l.o.g. that each user sends *exactly* k messages; otherwise, we may define a new symbol $\perp$ and make each user sends $\perp$ messages so that the number of messages becomes exactly k.

Similarly, we say that a protocol $P = (R, S, A)$ is (ε, δ)*-DP in the public-coin shuffle model*, if for any dataset $X = (x_1, \ldots, x_n)$, the algorithm

$$(x_1, \ldots, x_n) \underset{\alpha \leftarrow \mathcal{D}_{\mathsf{pub}}}{\mapsto} (\alpha, S(R_\alpha(x_1), \ldots, R_\alpha(x_n)))$$

is (ε, δ)-DP.

2.4 Useful Divergences

We will make use of two important divergences between distributions, the KL-divergence and the χ^2-divergence, defined as

$$KL(P||Q) = \mathop{\mathbb{E}}_{z \leftarrow P} \log\left(\frac{P_z}{Q_z}\right) \quad \text{and} \quad \chi^2(P||Q) = \mathop{\mathbb{E}}_{z \leftarrow Q}\left[\frac{P_z - Q_z}{Q_z}\right]^2.$$

We will also use Pinsker's inequality, whereby the total variation distance lower-bounds the KL-divergence:

$$\mathrm{KL}(P||Q) \geq \frac{2}{\ln 2}\|P - Q\|_{TV}^2.$$

2.5 Fourier Analysis

We now review some basic Fourier analysis and then introduce two inequalities that will be heavily used in our proofs. For a function $f \colon \{0,1\}^D \to \mathbb{R}$, its Fourier transform is given by the function $\hat{f}(S) := \mathop{\mathbb{E}}_{x \leftarrow \mathcal{U}_D}[f(x) \cdot (-1)^{\sum_{i \in S} x_i}]$. We also define $\|f\|_2^2 = \mathop{\mathbb{E}}_{x \leftarrow \mathcal{U}_D}[f(x)^2]$. For $k \in \mathbb{N}$, we define the *level-k Fourier weight* as $\mathbf{W}^k[f] := \sum_{S \subseteq [D], |S| = k} \hat{f}(S)^2$. For convenience, for $s \in \{0,1\}^D$, we will also write $\hat{f}(s)$ to denote $f(\chi_s)$, where χ_s is the set $\{i : i \in [D] \wedge s_i = 1\}$. One key technical lemma is the Level-1 Inequality from [37], which was also used in [27].

▶ **Lemma 16** (Level-1 Inequality). *Suppose $f \colon \{0,1\}^D \to \mathbb{R}_{\geq 0}$ is a non-negative-valued function with $f(x) \in [0, L]$ for all $x \in \{0,1\}^D$, and $\mathop{\mathbb{E}}_{x \sim \mathcal{U}_D}[f(x)] \leq 1$. Then, $\mathbf{W}^1[f] \leq 6\ln(L+1)$.*

We also need the standard Parseval's identity.

▶ **Lemma 17** (Parseval's Identity). *For all functions $f \colon \{0,1\}^D \to \mathbb{R}$,*

$$\|f\|_2^2 = \sum_{S \subseteq [D]} \hat{f}(S)^2.$$

3 Overview of Techniques

In this section, we describe the main intuition behind our lower bounds. As alluded to in Section 1, we give two different proofs of the lower bounds for CountDistinct in the $\mathrm{DP}_{\mathrm{local}}$ and $\mathrm{DP}_{\mathrm{shuffle}}$ settings, each with its own advantages:

- **Proof via Moment Matching.** Our first proof is technically the hardest in our work. It applies to the much more challenging low-privacy setting (i.e., $(\ln n - O(\ln\ln n), \delta)$-$\mathrm{DP}_{\mathrm{local}}$), and shows an $\Omega(n/\operatorname{polylog}(n))$ lower bound on the additive error (Theorem 2). Together with our new improved connection between $\mathrm{DP}^1_{\mathrm{shuffle}}$ and $\mathrm{DP}_{\mathrm{local}}$ (Lemma 5), it also implies the same lower bound for protocols in the $\mathrm{DP}^1_{\mathrm{shuffle}}$ model. The key ideas behind the first proof will be discussed in Section 3.1.

- **Proof via Dominated Protocols.** Our second proof has the advantage of giving the optimal $\Omega(n)$ lower bound on the additive error (Theorem 1), but only in the constant privacy regime (i.e., $(O(1), \delta)$-$\mathrm{DP}_{\mathrm{local}}$), and it is relatively simple compared to the first proof.
  Moreover, the second proof technique is very general and is a conceptual contribution: it can be applied to show lower bounds for other fundamental problems (i.e., Selection and ParityLearning; Theorems 9 and 10) against multi-message $\mathrm{DP}_{\mathrm{shuffle}}$ protocols. We will highlight the intuition behind the second proof in Section 3.2.

While our lower bounds also work for the public-coin $\mathrm{DP}_{\mathrm{shuffle}}$ models, throughout this section, we focus on private-coin models in order to simplify the presentation. The full proofs extending to public-coin protocols are given in the full version.

3.1 Lower Bounds for CountDistinct via Moment Matching

To clearly illustrate the key ideas behind the first proof, we will focus on the pure-DP case where each user can only send $O(\log n)$ bits. In the full version, we generalize the proof to approximate-DP and remove the restriction on communication complexity.

▶ **Theorem 18** (A Weaker Version of Theorem 2)**.** *For $\varepsilon = \ln(n/\log^7 n)$ and $D = n/\log^5 n$, no ε-$\mathrm{DP}_{\mathrm{local}}$ protocol where each user sends $O(\log n)$ bits can solve $\mathsf{CountDistinct}_{n,D}$ with error $o(D)$.*

Throughout our discussion, we use $R : [D] \to \mathcal{M}$ to denote a $\ln(n/\log^7 n)$-$\mathrm{DP}_{\mathrm{local}}$ randomizer. By the communication complexity condition of Theorem 18, we have that $|\mathcal{M}| \le \mathrm{poly}(n)$.

Our proof is inspired by the lower bounds for estimating distinct elements in the property testing model, e.g., [43, 45]. In particular, we use the so-called *Poissonization* trick. To discuss this trick, we start with some notation. For a vector $\vec{\lambda} \in \mathbb{R}^D$, we use $\vec{\mathsf{Poi}}(\vec{\lambda})$ to denote the joint distribution of D independent Poisson random variables:

$$\vec{\mathsf{Poi}}(\vec{\lambda}) := (\mathsf{Poi}(\vec{\lambda}_1), \mathsf{Poi}(\vec{\lambda}_2), \dots, \mathsf{Poi}(\vec{\lambda}_n)).$$

For a distribution $\vec{U}$ on $\mathbb{R}^D$, we define the corresponding mixture of multi-dimensional Poisson distributions as follows:

$$\mathbb{E}[\vec{\mathsf{Poi}}(\vec{U})] := \mathop{\mathbb{E}}_{\vec{\lambda} \leftarrow \vec{U}} \vec{\mathsf{Poi}}(\vec{\lambda}).$$

For two random variables X and Y supported on $\mathbb{R}^{\mathcal{M}}$, we use $X + Y$ to denote the random variable distributed as a sum of two independent samples from X and Y.

Shuffling the Outputs of the Local Protocol. Our first observation is that the analyzer for any local protocol computing CountDistinct should achieve the same accuracy if it only sees the histogram of the randomizers' outputs. This holds because only seeing the histogram of the outputs is equivalent to shuffling the outputs by a uniformly random permutation, which is in turn equivalent to shuffling the users in the dataset uniformly at random. Since shuffling the users in a dataset does not affect the number of distinct elements, it follows that only seeing the histogram does not affect the accuracy. Therefore, we only have to consider the histogram of the outputs of the local protocol computing CountDistinct. For a dataset W, we use $\mathsf{Hist}_R(W)$ to denote the distribution of the histogram with randomizer R.

Poissonization Trick. Given a distribution $\mathcal{D}$ on $\mathcal{M}$, suppose we draw a sample $m \leftarrow \mathsf{Poi}(\lambda)$, and then draw m samples from $\mathcal{D}$. If we use N to denote the random variable corresponding to the histogram of these m samples, it follows that each coordinate of N is independent, and N is distributed as $\vec{\mathsf{Poi}}(\lambda\vec{\mu})$, where $\vec{\mu}_i = \mathcal{D}_i$ for each $i \in \mathcal{M}$.

We can now apply the above trick to the context of local protocols (recall that by our first observation, we can focus on the histogram of the outputs). Suppose we build a dataset by drawing a sample $m \leftarrow \mathsf{Poi}(\lambda)$ and then adding m users with input z. By the above discussion, the corresponding histogram of the outputs with randomizer R is distributed as $\vec{\mathsf{Poi}}(\lambda \cdot R(z))$, where $R(z)$ is treated as an $|\mathcal{M}|$-dimensional vector corresponding to its probability distribution.

Moment-Matching Random Variables. Our next ingredient is the following construction of two moment-matching random variables used in [45]. Let $L \in \mathbb{N}$ and $\Lambda = \Theta(L^2)$. There are two random variables U and V supported on $\{0\} \cup [1, \Lambda]$, such that $\mathbb{E}[U] = \mathbb{E}[V] = 1$ and $\mathbb{E}[U^j] = \mathbb{E}[V^j]$ for every $j \in [L]$. Moreover $U_0 - V_0 > 0.9$. That is, U and V have the same moments up to degree L, while the probabilities of them being zero differs significantly. We will set $L = \log n$ and hence $\Lambda = \Theta(\log^2 n)$.

Construction of Hard Distribution via Signal/Noise Decomposition. Recalling that $D = n/\log^5 n$, we will construct two input distributions for $\mathsf{CountDistinct}_{n,D}$.[12] A sample from both distributions consists of two parts: a signal part with D many users in expectation, and a noise part with $n - D$ many users in expectation.

Formally, for a distribution W over $\mathbb{R}^{\geq 0}$ and a subset $E \subseteq [D]$, the dataset distributions $\mathcal{D}^W_{\mathsf{signal}}$ and $\mathcal{D}^E_{\mathsf{noise}}$ are constructed as follows:

- $\boldsymbol{\mathcal{D}^W_{\mathsf{signal}}}$**:** for each $i \in [D]$, we independently draw $\lambda_i \leftarrow W$, and $n_i \leftarrow \mathsf{Poi}(\lambda_i)$, and add n_i many users with input i.
- $\boldsymbol{\mathcal{D}^E_{\mathsf{noise}}}$**:** for each $i \in E$, we independently draw $n_i \leftarrow \mathsf{Poi}((n - D)/|E|)$, and add n_i many users with input i.

We are going to fix a "good" subset E of $[D]$ such that $|E| \leq 0.02 \cdot D$ (we will later specify the other conditions for being "good"). Therefore, when it is clear from the context, we will use $\mathcal{D}_{\mathsf{noise}}$ instead of $\mathcal{D}^E_{\mathsf{noise}}$.

Our two hard distributions are then constructed as $\mathcal{D}^U := \mathcal{D}^U_{\mathsf{signal}} + \mathcal{D}_{\mathsf{noise}}$ and $\mathcal{D}^V := \mathcal{D}^V_{\mathsf{signal}} + \mathcal{D}_{\mathsf{noise}}$. Using the fact that $\mathbb{E}[U] = \mathbb{E}[V] = 1$, one can verify that there are D users in each of $\mathcal{D}^U_{\mathsf{signal}}$ and $\mathcal{D}^V_{\mathsf{signal}}$ in expectation. Similarly, one can also verify there are $n - D$ users in $\mathcal{D}_{\mathsf{noise}}$ in expectation. Hence, both $\mathcal{D}^U$ and $\mathcal{D}^V$ have n users in expectation. In fact, the number of users from both distributions concentrates around n.

We now justify our naming of the signal/noise distributions. First, note that the number of distinct elements in the signal parts $\mathcal{D}^U_{\mathsf{signal}}$ and $\mathcal{D}^V_{\mathsf{signal}}$ concentrates around $(1 - \mathbb{E}[e^{-U}]) \cdot D$ and $(1 - \mathbb{E}[e^{-V}]) \cdot D$ respectively. By our condition that $U_0 - V_0 > 0.9$, it follows that the signal parts of $\mathcal{D}^U$ and $\mathcal{D}^V$ separates their numbers of distinct elements by at least $0.4D$. Second, note that although $\mathcal{D}_{\mathsf{noise}}$ has $n - D \gg D$ many users in expectation, they are from the subset E of size less than $0.02 \cdot n$. Therefore, these users collectively cannot change the number of distinct elements by more than $0.02 \cdot n$, and the numbers of distinct elements in $\mathcal{D}^U$ and $\mathcal{D}^V$ are still separated by $\Omega(D)$.

[12] In fact, in our presentation the number of inputs in each dataset from our hard distributions will not be exactly n, but only concentrated around n. This issue can be easily resolved by throwing "extra" users in the dataset; we refer the reader to the full version for the details.

Decomposition of Noise Part. To establish the desired lower bound, it now suffices to show for the local randomizer R, it holds that $\mathsf{Hist}_R(\mathcal{D}^U)$ and $\mathsf{Hist}_R(\mathcal{D}^V)$ are very close in statistical distance. For $W \in \{U, V\}$, we can decompose $\mathsf{Hist}_R(\mathcal{D}^W)$ as

$$\mathsf{Hist}_R(\mathcal{D}^W) = \sum_{i \in [D]} \vec{\mathsf{Poi}}(W \cdot R(i)) + \sum_{i \in [E]} \vec{\mathsf{Poi}}((n-D)/|E| \cdot R(i)).$$

By the additive property of Poisson distributions, letting $\vec{\nu} = (n-D)/|E| \cdot \sum_{i \in [E]} R(i)$, we have that $\sum_{i \in [E]} \vec{\mathsf{Poi}}((n-D)/|E| \cdot R(i)) = \vec{\mathsf{Poi}}(\vec{\nu})$.

Our key idea is to decompose $\vec{\nu}$ carefully into $D+1$ nonnegative vectors $\vec{\nu}^{(0)}, \vec{\nu}^{(1)}, \ldots, \vec{\nu}^{(D)}$, such that $\vec{\nu} = \sum_{i=0}^{D} \vec{\nu}^{(i)}$. Then, for $W \in \{U, V\}$, we have

$$\mathsf{Hist}_R(\mathcal{D}^W) = \vec{\mathsf{Poi}}(\vec{\nu}^{(0)}) + \sum_{i \in [D]} \vec{\mathsf{Poi}}(W \cdot R(i) + \vec{\nu}^{(i)}).$$

To show that $\mathsf{Hist}_R(\mathcal{D}^U)$ and $\mathsf{Hist}_R(\mathcal{D}^V)$ are close, it suffices to show that for each $i \in [D]$, it is the case that $\vec{\mathsf{Poi}}(U \cdot R(i) + \vec{\nu}^{(i)})$ and $\vec{\mathsf{Poi}}(V \cdot R(i) + \vec{\nu}^{(i)})$ are close. We show that they are close when $\vec{\nu}^{(i)}$ is sufficiently large on every coordinate compared to $R(i)$.

▶ **Lemma 19.** *For each $i \in [D]$, and every $\vec{\lambda} \in (\mathbb{R}^{\geq 0})^{\mathcal{M}}$, if $\vec{\lambda}_z \geq 2\Lambda^2 \cdot R(i)_z$ for every $z \in \mathcal{M}$, then*[13]

$$\| \mathbb{E}[\vec{\mathsf{Poi}}(U \cdot R(i) + \vec{\lambda})] - \mathbb{E}[\vec{\mathsf{Poi}}(V \cdot R(i) + \vec{\lambda})] \|_{TV} \leq \frac{1}{n^2}.$$

To apply Lemma 19, we simply set $\vec{\nu}^{(i)} = (2\Lambda^2) \cdot R(i)$ and $\vec{\nu}^{(0)} = \vec{\nu} - \sum_{i \in [D]} \vec{\nu}^{(i)}$. Letting $\vec{\mu} = \sum_{i \in [D]} R(i)$, the requirement that $\vec{\nu}^{(0)}$ has to be nonnegative translates to $\vec{\nu}_z \geq 2\Lambda^2 \cdot \vec{\mu}_z$, for each $z \in \mathcal{M}$.

Construction of a Good Subset E. So we want to pick a subset $E \subseteq [D]$ of size at most $0.02 \cdot D$ such that the corresponding $\vec{\nu}^E = (n-D)/|E| \cdot \sum_{i \in [E]} R(i)$ satisfies $\vec{\nu}^E_z \geq 2\Lambda^2 \cdot \vec{\mu}_z$ for each $z \in \mathcal{M}$. We will show that a simple random construction works with high probability: i.e., one can simply add each element of $[D]$ to E independently with probability 0.01.

More specifically, for each $z \in \mathcal{M}$, we will show that with high probability $\vec{\nu}^E_z \geq 2\Lambda^2 \cdot \vec{\mu}_z$. Then the correctness of our construction follows from a union bound (and this step crucially uses the fact that $|\mathcal{M}| \leq \mathrm{poly}(n)$).

Now, let us fix a $z \in \mathcal{M}$. Let $m^* = \max_{i \in [D]} R(i)_z$. Since R is $\ln(n/\log^7 n)$-DP, it follows that $\vec{\nu}_z \geq \frac{n-D}{n/\log^7 n} \cdot m^* \geq \frac{\log^7 n}{2} \cdot m^*$. We consider the following two cases:

1. If $m^* \geq \vec{\mu}_z / \log^2 n$, we immediately get that $\vec{\nu}_z \geq \log^5 n/2 \cdot \vec{\mu}_z \geq 2\Lambda^2 \cdot \vec{\mu}_z$ (which uses the fact that $\Lambda = \Theta(\log^2 n)$).
2. If $m^* < \vec{\mu}_z / \log^2 n$, then in this case, the mass $\vec{\mu}_z$ is distributed over at least $\log^2 n$ many components $R(i)_z$. Applying Hoeffding's inequality shows that with high probability over E, it is the case that $\vec{\nu}^E_z \geq \Theta(n/D) \cdot \vec{\mu}_z \geq \Lambda^2 \cdot \vec{\mu}_z$ (which uses the fact that $D = n/\log^5 n$).

See the full version for a formal argument and how to get rid of the assumption that $|\mathcal{M}| \leq \mathrm{poly}(n)$.

[13] We use $\|\mathcal{D}_1 - \mathcal{D}_2\|_{TV}$ to denote the total variation (aka statistical) distance between two distributions $\mathcal{D}_1, \mathcal{D}_2$.

The Lower Bound. From the above discussions, we get that

$$\|\mathsf{Hist}_R(\mathcal{D}^U)-\mathsf{Hist}_R(\mathcal{D}^V)\|_{TV} \le \sum_{i=1}^{D} \| \mathbb{E}[\vec{\mathsf{Poi}}(U\cdot R(i)+\vec{\nu}^{(i)})]-\mathbb{E}[\vec{\mathsf{Poi}}(V\cdot R(i)+\vec{\nu}^{(i)})]\|_{TV} \le 1/n.$$

Hence, the analyzer of the local protocol with randomizer R cannot distinguish $\mathcal{D}^U$ and $\mathcal{D}^V$, and thus it cannot solve $\mathsf{CountDistinct}_{n,D}$ with error $o(D)$ and 0.99 probability. See the full version for a formal argument and how to deal with the fact that there may not be exactly n users in dataset from $\mathcal{D}^U$ or $\mathcal{D}^V$.

Single-Message $\mathrm{DP}_{\mathsf{shuffle}}$ Lower Bound. To apply the above lower bound to $\mathrm{DP}^1_{\mathrm{shuffle}}$ protocols, the natural idea is to resort to the connection between the $\mathrm{DP}^1_{\mathrm{shuffle}}$ and $\mathrm{DP}_{\mathrm{local}}$ models. In particular, [12] showed that (ε,δ)-$\mathrm{DP}^1_{\mathrm{shuffle}}$ protocols are also $(\varepsilon+\ln n,\delta)$-$\mathrm{DP}_{\mathrm{local}}$.

It may seem that the $\ln n$ privacy guarantee is very close to the $\ln n - O(\ln\ln n)$ one in Theorem 2. But surprisingly, it turns out (as was stated in Theorem 3) that there is a $(\ln n + O(1))$-$\mathrm{DP}_{\mathrm{local}}$ protocol solving $\mathsf{CountDistinct}_{n,n}$ (hence also $\mathsf{CountDistinct}_{n,D}$) with error $O(\sqrt{n})$. Hence, to establish the $\mathrm{DP}^1_{\mathrm{shuffle}}$ lower bound (Theorem 4), we rely on the following stronger connection between $\mathrm{DP}^1_{\mathrm{shuffle}}$ and $\mathrm{DP}_{\mathrm{local}}$ protocols.

▶ **Lemma 20** (Simplification of Lemma 5). *For every $\delta \le 1/n^{\omega(1)}$, if the randomizer R is $(O(1),\delta)$-$\mathrm{DP}^1_{\mathrm{shuffle}}$ on n users, then R is $\left(\ln(n\log^2 n/\log\delta^{-1}), n^{-\omega(1)}\right)$-$\mathrm{DP}_{\mathrm{local}}$.*

Setting $\delta = 2^{-\log^k n}$ for a sufficiently large k and combining Lemma 20 and Theorem 2 gives us the desired lower bound against $\mathrm{DP}^1_{\mathrm{shuffle}}$ protocols.

3.2 Lower Bounds for CountDistinct and Selection via Dominated Protocols

We will first describe the proof ideas behind Theorem 1, which is restated below. Then, we apply the same proof technique to obtain lower bounds for $\mathsf{Selection}$ (the lower bound for $\mathsf{ParityLearning}$ is established similarly; see the full version for details).

▶ **Lemma 21** (Detailed Version of Theorem 1). *For $\varepsilon = o(\ln n)$, no $(\varepsilon, o(1/n))$-dominated protocol can solve $\mathsf{CountDistinct}$ with error $o(n/e^{\varepsilon})$.*

Hard Distributions for $\mathsf{CountDistinct}_{n,n}$. We now construct our hard instances for $\mathsf{CountDistinct}_{n,n}$. For simplicity, we assume $n = 2^D$ for an integer D, and identify the input space $[n]$ with $\{0,1\}^D$ by a fixed bijection. Let $\mathcal{U}_D$ be the the uniform distribution over $\{0,1\}^D$. For $(\ell,s) \in [2]\times\{0,1\}^D$, we let $\mathcal{D}_{\ell,s}$ be the uniform distribution on $\{x \in \{0,1\}^D : \langle x,s\rangle = \ell\}$.

We also use $\mathcal{D}^{\alpha}_{\ell,s}$ to denote the mixture of $\mathcal{D}_{\ell,s}$ and $\mathcal{U}_D$ which outputs a sample from $\mathcal{D}_{\ell,s}$ with probability α and a sample from $\mathcal{U}_D$ with probability $1-\alpha$.

For a parameter $\alpha > 0$, we consider the following two dataset distributions with n users:

- $\mathcal{W}^{\mathsf{uniform}}$: each user gets an i.i.d. input from $\mathcal{U}_D$. That is, $\mathcal{W}^{\mathsf{uniform}} := \mathcal{U}_D^{\otimes n}$.
- $\mathcal{W}^{\alpha}$: to sample a dataset from $\mathcal{W}^{\alpha}$, we first draw (ℓ,s) from $[2]\times\{0,1\}^D$ uniformly at random, then each user gets an i.i.d. input from $\mathcal{D}^{\alpha}_{\ell,s}$. Formally, $\mathcal{W}^{\alpha} := \mathbb{E}_{(\ell,s)\leftarrow[2]\times\{0,1\}^D}(\mathcal{D}^{\alpha}_{\ell,s})^{\otimes n}$.

Since for every ℓ, s, it holds that $|\mathrm{supp}(\mathcal{D}^1_{\ell,s})| \le n/2$, the number of distinct elements from any dataset in $\mathcal{W}^1$ is at most $n/2$. On the other hand, since $\mathcal{U}_D$ is a uniform distribution over n elements, a random dataset from $\mathcal{W}^{\mathsf{uniform}} = \mathcal{W}^0$ has roughly $(1-e^{-1})\cdot n > n/2$ distinct elements with high probability. Hence, the expected number of distinct elements of datasets from $\mathcal{W}^\alpha$ is controlled by the parameter α. A simple but tedious calculation shows that it is approximately $(1 - e^{-1} \cdot \cosh(\alpha)) \cdot n$, which can be approximated by $(1 - e^{-1} \cdot (1 + \alpha^2)) \cdot n$ for $n^{-0.1} < \alpha < 0.01$. Hence, any protocol solving CountDistinct with error $o(\alpha^2 n)$ should be able to distinguish between the above two distributions. Our goal is to show that this is impossible for $(\varepsilon, o(1/n))$-dominated protocols.

Bounding KL Divergence for Dominated Protocols. Our next step is to upper-bound the statistical distance $\|\mathsf{Hist}_R(\mathcal{W}^{\mathsf{uniform}}) - \mathsf{Hist}_R(\mathcal{W}^\alpha)\|_{TV}$. As in previous work [41, 27, 23], we may upper-bound the KL divergence instead. By the convexity and chain-rule properties of KL divergence, it follows that

$$\begin{aligned}\mathrm{KL}(\mathsf{Hist}_R(\mathcal{W}^\alpha)||\mathsf{Hist}_R(\mathcal{W}^{\mathsf{uniform}})) &\le \mathop{\mathbb{E}}_{(\ell,s)\leftarrow[2]\times\{0,1\}^D} \mathrm{KL}(R(\mathcal{D}^\alpha_{\ell,s})^{\otimes n}||R(\mathcal{U}_D)^{\otimes n}) \\ &= n \cdot \mathop{\mathbb{E}}_{(\ell,s)\leftarrow[2]\times\{0,1\}^D} \mathrm{KL}(R(\mathcal{D}^\alpha_{\ell,s})||R(\mathcal{U}_D)). \end{aligned} \tag{1}$$

Bounding the Average KL Divergence between a Family and a Single Distribution. We are now ready to introduce our general tool for bounding average KL divergence quantities like (1). We first set up some notation. Let $\mathcal{I}$ be an index set and $\{\lambda_v\}_{v\in\mathcal{I}}$ be a family of distributions on $\mathcal{X}$, let π be a distribution on $\mathcal{I}$, and μ be a distribution on $\mathcal{X}$. For simplicity, we assume that for every $x \in \mathcal{X}$ and $v \in \mathcal{I}$, it holds that $(\lambda_v)_x \le 2 \cdot \mu_x$ (which is true for $\{\mathcal{D}^\alpha_{\ell,s}\}_{(\ell,s)\in[2]\times\{0,1\}^D}$ and $\mathcal{U}_D$).

▶ **Theorem 22.** *Let $W : \mathbb{R} \to \mathbb{R}$ be a concave function such that for all functions $\psi \colon \mathcal{X} \to \mathbb{R}^{\ge 0}$ satisfying $\psi(\mu) \le 1$, it holds that*

$$\mathop{\mathbb{E}}_{v\leftarrow\pi} \left[(\psi(\lambda_v) - \psi(\mu))^2\right] \le W(\|\psi\|_\infty).$$

Then for an (ε, δ)-dominated randomizer R, it follows that

$$\mathop{\mathbb{E}}_{v\leftarrow\pi} [\mathrm{KL}(R(\lambda_v)||R(\mu))] \le O\left(W(2e^\varepsilon) + \delta\right).$$

Similar theorems are proved in the previous work [17, 18, 41, 23] but only for locally private randomizers. Theorem 22 can be seen as a generalization of these previous results to dominated protocols.

Bounding (1) via Fourier Analysis. To apply Theorem 22, for $f \colon \mathcal{X} \to \mathbb{R}^{\ge 0}$ with $f(\mathcal{U}_D) = \mathbb{E}_{x\in\{0,1\}^D}[f(x)] \le 1$, we want to bound

$$\mathop{\mathbb{E}}_{(\ell,s)\leftarrow[2]\times\{0,1\}^D} [(f(\mathcal{D}^\alpha_{\ell,s}) - f(\mathcal{U}_D))^2] = \mathop{\mathbb{E}}_{s\in\{0,1\}^D} \alpha^2 \cdot \hat{f}(s)^2.$$

By Parseval's Identity (see Lemma 17),

$$\sum_{s\in\{0,1\}^D} \hat{f}(s)^2 = \mathop{\mathbb{E}}_{x\in\{0,1\}^D} f(x)^2 \le f(\mathcal{U}_D) \cdot \|f\|_\infty \le \|f\|_\infty.$$

Therefore, we can set $W(L) := \alpha^2 \cdot \frac{L}{2^D}$, and apply Theorem 22 to obtain

$$\mathop{\mathbb{E}}_{(\ell,s)\leftarrow[2]\times\{0,1\}^D} \mathrm{KL}(R(\mathcal{D}^\alpha_{\ell,s})||R(\mathcal{U}_D)) \le O(\alpha^2 \cdot e^\varepsilon/n + \delta).$$

We set α such that $\alpha^2 = c/e^\varepsilon$ for a sufficiently small constant c and note that $\delta = o(1/n)$. It follows that

$$\mathrm{KL}(\mathsf{Hist}_R(\mathcal{W}^\alpha)||\mathsf{Hist}_R(\mathcal{W}^{\mathsf{uniform}})) \le 0.01,$$

and therefore

$$\|\mathsf{Hist}_R(\mathcal{W}^\alpha) - \mathsf{Hist}_R(\mathcal{W}^{\mathsf{uniform}})\|_{TV} \le 0.1$$

by Pinsker's inequality. Hence, we conclude that $(\varepsilon, o(1/n))$-dominated protocols cannot solve $\mathsf{CountDistinct}_{n,n}$ with error $o(n/e^\varepsilon)$, completing the proof of Lemma 21. Now Theorem 1 follows from Lemma 21 and the fact that (ε,δ)-$\mathrm{DP}_{\mathrm{local}}$ protocols are also (ε,δ)-dominated.

Lower Bounds for Selection against Multi-Message $\mathbf{DP_{shuffle}}$ Protocols. Now we show how to apply Theorem 22 and Lemma 20 to prove lower bounds for $\mathsf{Selection}$. For $(\ell, j) \in [2] \times [D]$, let $\mathcal{D}_{\ell,j}$ be the uniform distribution on all length-D binary strings with jth bit being ℓ. Recall that $\mathcal{U}_D$ is the uniform distribution on $\{0,1\}^D$. Again we aim to upper-bound the average-case KL divergence $\mathbb{E}_{(\ell,j)\leftarrow[2]\times[D]}\, \mathrm{KL}(R(\mathcal{D}_{\ell,j})||R(\mathcal{U}_D))$.

To apply Theorem 22, for $f\colon \mathcal{X} \to \mathbb{R}^{\ge 0}$ with $f(\mathcal{U}_D) = \mathbb{E}_{x\in\{0,1\}^D}[f(x)] \le 1$, we want to bound

$$\mathop{\mathbb{E}}_{(\ell,j)\leftarrow[2]\times[D]}[(f(\mathcal{D}^\alpha_{\ell,j}) - f(\mathcal{U}_D))^2] = \mathop{\mathbb{E}}_{j\in[D]} \hat{f}(\{j\})^2.$$

By Lemma 16, it is the case that

$$\sum_{j\in[D]} \hat{f}(\{j\})^2 \le O(\log\|f\|_\infty).$$

Therefore, we can set $W(L) := c_1 \cdot \frac{\log L}{D}$ for an appropriate constant c_1, and apply Theorem 22 to obtain

$$\mathop{\mathbb{E}}_{(\ell,j)\leftarrow[2]\times[D]} \mathrm{KL}(R(\mathcal{D}_{\ell,j})||R(\mathcal{U}_D)) \le O\left(\frac{\varepsilon}{D} + \delta\right).$$

Combining this with Lemma 20 completes the proof (see the full version for the details).

References

1 John M Abowd. The US Census Bureau adopts differential privacy. In *KDD*, pages 2867–2867, 2018.

2 Kareem Amin, Matthew Joseph, and Jieming Mao. Pan-private uniformity testing. In *COLT*, pages 183–218, 2020.

3 Apple Differential Privacy Team. Learning with privacy at scale. *Apple Machine Learning Journal*, 2017.

4 Victor Balcer and Albert Cheu. Separating local & shuffled differential privacy via histograms. In *ITC*, pages 1:1–1:14, 2020.

5 Victor Balcer, Albert Cheu, Matthew Joseph, and Jieming Mao. Connecting robust shuffle privacy and pan-privacy. In *SODA*, 2021. `arXiv:2004.09481`.

6 Borja Balle, James Bell, Adrià Gascón, and Kobbi Nissim. The privacy blanket of the shuffle model. In *CRYPTO*, pages 638–667, 2019.

7 Amos Beimel, Iftach Haitner, Kobbi Nissim, and Uri Stemmer. On the round complexity of the shuffle model. In *TCC*, 2020.

8 Andrea Bittau, Úlfar Erlingsson, Petros Maniatis, Ilya Mironov, Ananth Raghunathan, David Lie, Mitch Rudominer, Ushasree Kode, Julien Tinnes, and Bernhard Seefeld. Prochlo: Strong privacy for analytics in the crowd. In *SOSP*, pages 441–459, 2017.

9 Joshua Brody, Amit Chakrabarti, Ranganath Kondapally, David P Woodruff, and Grigory Yaroslavtsev. Beyond set disjointness: the communication complexity of finding the intersection. In *PODC*, pages 106–113, 2014.

10 Mark Bun, Jelani Nelson, and Uri Stemmer. Heavy hitters and the structure of local privacy. *TALG*, 15(4):1–40, 2019.

11 TH Hubert Chan, Elaine Shi, and Dawn Song. Optimal lower bound for differentially private multi-party aggregation. In *ESA*, pages 277–288, 2012.

12 Albert Cheu, Adam D. Smith, Jonathan Ullman, David Zeber, and Maxim Zhilyaev. Distributed differential privacy via shuffling. In *EUROCRYPT*, pages 375–403, 2019. `arXiv:1808.01394`.

13 Albert Cheu and Jonathan Ullman. The limits of pan privacy and shuffle privacy for learning and estimation. *arXiv*, 2020. `arXiv:2009.08000`.

14 Seung Geol Choi, Dana Dachman-Soled, Mukul Kulkarni, and Arkady Yerukhimovich. Differentially-private multi-party sketching for large-scale statistics. *PoPETs*, 3:153–174, 2020.

15 Damien Desfontaines, Andreas Lochbihler, and David Basin. Cardinality estimators do not preserve privacy. *PoPETs*, 2019(2):26–46, 2019.

16 Bolin Ding, Janardhan Kulkarni, and Sergey Yekhanin. Collecting telemetry data privately. In *NIPS*, pages 3571–3580, 2017.

17 John C Duchi, Michael I Jordan, and Martin J Wainwright. Local privacy and statistical minimax rates. In *FOCS*, pages 429–438, 2013.

18 John C. Duchi and Feng Ruan. The right complexity measure in locally private estimation: It is not the Fisher information. *arXiv*, 2018. `arXiv:1806.05756`.

19 Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *EUROCRYPT*, pages 486–503, 2006. `doi:10.1007/11761679_29`.

20 Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam D. Smith. Calibrating noise to sensitivity in private data analysis. In *TCC*, pages 265–284, 2006. `doi:10.1007/11681878_14`.

21 Cynthia Dwork, Moni Naor, Toniann Pitassi, Guy N Rothblum, and Sergey Yekhanin. Pan-private streaming algorithms. In *ICS*, pages 66–80, 2010.

22 Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3-4):211–407, 2014. `doi:10.1561/0400000042`.

23 Alexander Edmonds, Aleksandar Nikolov, and Jonathan Ullman. The power of factorization mechanisms in local and central differential privacy. In *STOC*, pages 425–438, 2020. `doi:10.1145/3357713.3384297`.

24 Úlfar Erlingsson, Vitaly Feldman, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Abhradeep Thakurta. Amplification by shuffling: From local to central differential privacy via anonymity. In *SODA*, pages 2468–2479, 2019.

25 Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. RAPPOR: Randomized aggregatable privacy-preserving ordinal response. In *CCS*, pages 1054–1067, 2014.

26 Badih Ghazi, Noah Golowich, Ravi Kumar, Pasin Manurangsi, Rasmus Pagh, and Ameya Velingker. Pure differentially private summation from anonymous messages. In *ITC*, pages 15:1–15:23, 2020.

27 Badih Ghazi, Noah Golowich, Ravi Kumar, Rasmus Pagh, and Ameya Velingker. On the power of multiple anonymous messages. *IACR Cryptol. ePrint Arch.*, 2019:1382, 2019. URL: `https://eprint.iacr.org/2019/1382`.

28 Andy Greenberg. Apple's "differential privacy" is about collecting your data – but not your data. *Wired, June*, 13, 2016.

29 Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. Cryptography from anonymity. In *FOCS*, pages 239–248, 2006.

30 Daniel M Kane, Jelani Nelson, and David P Woodruff. An optimal algorithm for the distinct elements problem. In *PODS*, pages 41–52, 2010.

31 Shiva Prasad Kasiviswanathan, Homin K Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What can we learn privately? *SICOMP*, 40(3):793–826, 2011.

32 Michael Kearns. Efficient noise-tolerant learning from statistical queries. *JACM*, 45(6):983–1006, 1998.

33 Andrew McGregor, Ilya Mironov, Toniann Pitassi, Omer Reingold, Kunal Talwar, and Salil Vadhan. The limits of two-party differential privacy. In *FOCS*, pages 81–90, 2010.

34 Andrew McGregor, Ilya Mironov, Toniann Pitassi, Omer Reingold, Kunal Talwar, and Salil P. Vadhan. The limits of two-party differential privacy. *ECCC*, 18:106, 2011. URL: `http://eccc.hpi-web.de/report/2011/106`.

35 Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *FOCS*, pages 94–103, 2007.

36 Darakhshan Mir, Shan Muthukrishnan, Aleksandar Nikolov, and Rebecca N Wright. Pan-private algorithms via statistics on sketches. In *PODS*, pages 37–48, 2011.

37 Ryan O'Donnell. *Analysis of Boolean Functions*. Cambridge University Press, 2014. URL: `http://www.cambridge.org/de/academic/subjects/computer-science/algorithmics-complexity-computer-algebra-and-computational-g/analysis-boolean-functions`.

38 Rasmus Pagh and Nina Mesing Stausholm. Efficient differentially private f_0 linear sketching. *arXiv*, 2020. `arXiv:2001.11932`.

39 Stephen Shankland. How Google tricks itself to protect Chrome user privacy. *CNET, October*, 2014.

40 Thomas Steinke and Jonathan Ullman. Tight lower bounds for differentially private selection. In *FOCS*, pages 552–563, 2017.

41 Jonathan Ullman. Tight lower bounds for locally differentially private selection. *arXiv*, 2018. `arXiv:1802.02638`.

42 Salil Vadhan. The complexity of differential privacy. In *Tutorials on the Foundations of Cryptography*, pages 347–450. Springer, 2017.

43 Gregory Valiant and Paul Valiant. Estimating the unseen: Improved estimators for entropy and other properties. *JACM*, 64(6):37:1–37:41, 2017. `doi:10.1145/3125643`.

44 Stanley L Warner. Randomized response: A survey technique for eliminating evasive answer bias. *JASA*, 60(309):63–69, 1965.

45 Yihong Wu and Pengkun Yang. Chebyshev polynomials, moment matching, and optimal estimation of the unseen. *The Annals of Statistics*, 47(2):857–883, 2019.