Private Counting from Anonymous Messages: Near-Optimal Accuracy with Vanishing Communication Overhead

Badih Ghazi [1] **Ravi Kumar** [1] **Pasin Manurangsi** [1] **Rasmus Pagh** [2 1]

Abstract

Differential privacy (DP) is a formal notion for quantifying the privacy loss of algorithms. Algorithms in the central model of DP achieve high accuracy but make the strongest trust assumptions whereas those in the local DP model make the weakest trust assumptions but incur substantial accuracy loss. The shuffled DP model (Bittau et al., 2017; Erlingsson et al., 2019; Cheu et al., 2019) has recently emerged as a feasible middle ground between the central and local models, providing stronger trust assumptions than the former while promising higher accuracies than the latter. In this paper, we obtain practical communication-efficient algorithms in the shuffled DP model for two basic aggregation primitives used in machine learning: 1) binary summation, and 2) histograms over a moderate number of buckets. Our algorithms achieve accuracy that is arbitrarily close to that of central DP algorithms with an expected communication per user essentially matching what is needed without any privacy constraints! We demonstrate the practicality of our algorithms by experimentally comparing their performance to several widely-used protocols such as Randomized Response (Warner, 1965) and RAPPOR (Erlingsson et al., 2014).

1. Introduction

Motivated by the need for scalable, distributed privacy-preserving machine learning, there has been an intense interest, both in academia and industry, on designing algorithms with low communication overhead and high accuracy while protecting potentially sensitive, user-specific information. While many notions of privacy have been proposed, *differential privacy* (DP) (Dwork et al., 2006b;a) has become by far the most popular and well-studied candidate, leading to several real-world deployments at companies such as Google (Erlingsson et al., 2014; Shankland, 2014), Apple (Greenberg, 2016; Apple Differential Privacy Team, 2017), and Microsoft (Ding et al., 2017), and in government agencies such as the U.S. Census Bureau (Abowd, 2018). Most research has focused on the *central* model of DP where a curator, who sees the raw user data, is required to release a private data structure. While many accurate DP algorithms have been discovered in this framework, the requirement that the curator observes the raw data constitutes a significant obstacle to deployment in many industrial settings where the users do not necessarily trust the central authority. To circumvent this limitation, several works have studied the *local* model of DP (Kasiviswanathan et al., 2008) (also (Warner, 1965)), which enforces the more stringent constraint that each message sent from a user device to the server is private. While requiring near-minimal trust assumptions, the local model turns out to inherently suffer from large estimation errors. For numerous basic tasks, including binary summation and histograms that we study in this work, errors are at least on the order of $\sqrt{n}$, where n is the number of users (Beimel et al., 2008; Chan et al., 2012).

Shuffled Privacy Model. The shuffled (aka. anonymous) model of privacy has recently generated significant interest as a potential compromise between the central and local frameworks: having trust assumptions better than the former but enabling estimation accuracies higher than the latter. While the shuffled model was originally studied in the field of cryptography by Ishai et al. (2006) in their work on cryptography from anonymity, it was first suggested as a framework for privacy-preserving computations by Bittau et al. (2017) in their Encode-Shuffle-Analyze architecture. This setting only requires the multiset of *anonymized* messages that are transmitted by the different users to be private. Equivalently, this corresponds to the setup where a trusted shuffler *randomly permutes* all incoming messages from the users before passing them to the analyzer. This is illustrated in Figure 1. We point out that several efficient cryptographic implementations of the shuffler have been considered includ-

[*]Equal contribution [1]Google Research, Mountain View [2]IT University of Copenhagen. Correspondence to: Badih Ghazi <badihghazi@gmail.com>, Ravi Kumar <ravi.k53@gmail.com>, Pasin Manurangsi <pasin@google.com>, Rasmus Pagh <pagh@itu.dk>.

Proceedings of the 37^{th} International Conference on Machine Learning, Vienna, Austria, PMLR 119, 2020.

ing mixnets, onion routing, secure hardware, and third-party servers (see, e.g., (Ishai et al., 2006; Bittau et al., 2017) for more details). As in all previous work on the shuffled model, we treat the shuffler as a black box.

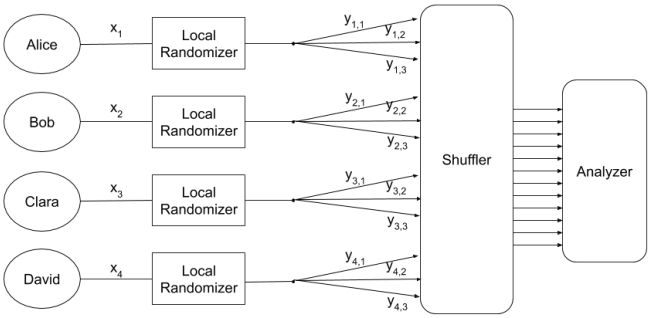

Figure 1. The Shuffled Model.

DP in the shuffled model was first formally investigated, independently, by Erlingsson et al. (2019) and Cheu et al. (2019). Several recent works have aimed to determine the optimal trade-offs between communication, accuracy, and privacy in this model for various algorithmic tasks (Balle et al., 2019; Ghazi et al., 2019b;a; 2020b; Balle et al., 2020; Balcer & Cheu, 2020).

Summation. One of the most basic distributed computation problems is *summation* (aka. aggregation) where the goal of the analyzer is to estimate the sum of the user inputs. In machine learning, and specifically in the nascent field of federated learning (Konečnỳ et al., 2016) (see, e.g., (Kairouz et al., 2019) for a recent survey), private summation enables private Stochastic Gradient Descent (SGD), which in turn allows the private training of deep neural networks that are guaranteed not to overfit to any user-specific information. Moreover, summation is perhaps the most primitive functionality in database systems in general, and in private implementations in particular (see, e.g., (Kotsogiannis et al., 2019; Wilson et al., 2019; Suresh et al., 2017)).

A notable special case is *binary* summation (aka. *counting query*) where each user holds a bit as an input and the goal of the analyzer is to estimate the number of users whose input equals 1. The vector version of this problem captures, e.g., the case where gradients have been quantized to bits in order to reduce the communication cost (e.g., the 1-bit SGD of Seide et al. (2014)). As observed in Blum et al. (2005), binary summation is of particular interest in ML since it is sufficient for implementing any learning algorithm based on *statistical queries* (Kearns, 1998), which includes most of the known PAC-learning algorithms.

Several recent works have studied private summation in the shuffled model (Cheu et al., 2019; Balle et al., 2019; Ghazi et al., 2019b; Balle et al., 2020; Ghazi et al., 2020b;a). These results achieve DP with parameters ε, δ (defined in Section 2). Cheu et al. (2019) showed that the standard Randomized Response (which goes back to Warner (1965) in the local DP case) is (ε, δ)-DP and incurs a squared error of $O(\frac{1}{\varepsilon^2} \cdot \log \frac{1}{\delta})$ with high probability. All mentioned works have also studied *real* summation, culminating in an (ε, δ)-DP protocol in the shuffled model with error arbitrarily close to a discrete Laplace random variable with parameter $1/\varepsilon$, and where each user sends $O(1 + \frac{\log(1/\delta)}{\log n})$ messages of $O(\log n)$ bits each (Ghazi et al., 2020b; Balle et al., 2020).

Histograms. A generalization of the binary summation problem is that of computing *histograms* (aka. *frequency oracles* or *frequency estimation*), where each user holds an element from some finite set $[B] := \{1, \ldots, B\}$ and the goal of the analyzer is to estimate for all $j \in [B]$, the number of users holding element j as input. Computing histograms is fundamental in data analytics and is well-studied in DP (e.g., (Kairouz et al., 2016; Acharya & Sun, 2019; Suresh, 2019)), as private histogram procedures can be used as a black-box to solve important algorithmic problems such as *heavy hitters* (e.g., (Bassily et al., 2017)) as well as unsupervised machine learning tasks such as *clustering* (e.g., (Stemmer, 2020)); furthermore, computing histogram is intimately related to *distribution estimation* (e.g., (Kairouz et al., 2016; Acharya & Sun, 2019)). In central DP, the smallest possible estimation error for histograms is known to be $\Theta(\min(\frac{\log(1/\delta)}{\varepsilon}, \frac{\log B}{\varepsilon}, n))$ (e.g., Section 7.1 in Vadhan (2017)). On the other hand, the smallest possible error in local DP is $\Theta(\min(\frac{\sqrt{n \log B}}{\varepsilon}, n))$ provided $\delta < 1/n$ (Bassily & Smith, 2015). In the shuffled DP setting and for ε a constant and δ inverse-polynomial in n, the tight estimation error for single-message protocols (where each user sends a single message) is $\widetilde{\Theta}(\min\{n^{1/4}, \sqrt{B}\})$, whereas multi-message protocols with both error and per-user communication that are logarithmic in B and n are known (Ghazi et al., 2019a; Erlingsson et al., 2020). Recently, Balcer & Cheu (2020) obtained a protocol with error independent of B but logarithmic in n, albeit with a per-user communication of $O(B)$ messages each consisting of $O(\log B)$ bits.

Two recent works (Wang et al., 2019; Erlingsson et al., 2020) studied private histograms in extensions of the shuffled model to multiple shufflers. Wang et al. (2019) uses Randomized Response whereas Erlingsson et al. (2020) uses a fragmented version of RAPPOR (Erlingsson et al., 2014).

In this work we focus on the regime where $B \ll n$, which captures numerous practical scenarios since the number of buckets is typically small compared to the population size.

1.1. Main Results

For the binary summation problem, we give the first private protocol in the shuffled model achieving mean squared er-

ror (MSE) arbitrarily close to the central performance of the Discrete Laplace mechanism while having an *expected* communication per user of $1 + o(1)$ messages of 1 bit each.

Theorem 1 (Binary Summation Protocol). *For every $\varepsilon \leq O(1)$ and every $\delta, \gamma \in (0, 1/2)$, there is an (ε, δ)-DP protocol for binary summation in the multi-message shuffled model with error equal to a Discrete Laplace random variable with parameter $(1-\gamma)\varepsilon$ and with an expected communication per user of $1 + O\left(\frac{\log^2(1/\delta)}{\gamma\varepsilon^2 n}\right)$ bits.*

We extend Theorem 1 to a protocol for histograms that, with a moderate number of buckets, has error arbitrarily close to the central DP performance of the Discrete Laplace mechanism while using essentially minimal communication.

Corollary 2 (Histogram Protocol). *For every $\varepsilon \leq O(1)$ and every $\delta, \gamma \in (0, 1/2)$, there is an (ε, δ)-DP protocol for histograms on sets of size B in the multi-message shuffled model, with error equal to a vector of independent Discrete Laplace random variables each with parameter $\frac{(1-\gamma)\varepsilon}{2}$ and with an expected number of messages sent per user equal to $1 + O\left(\frac{B\log^2(1/\delta)}{\gamma\varepsilon^2 n}\right)$, each consisting of $\lceil \log B \rceil + 1$ bits.*

For the standard setting of constant ε and δ inverse-polynomial in n and for an arbitrarily small positive constant γ, the expected communication per user in Theorem 1 is $1 + o(1)$ bits. Note that 1 bit of communication per user is required for accurate estimation of the binary summation even in the absence of any privacy constraints. Likewise, the expected communication per user in Corollary 2 is $\lceil \log B \rceil + 1 + o(1)$ bits. Here again, $\log B$ bits of communication per user is required for accurate estimation of the histogram even in the absence of any privacy constraints.

A natural question in the context of Theorem 1 and Corollary 2 is whether the same accuracy and communication can be achieved by a *single-message* protocol in the shuffled model. For histograms, this is impossible given the $\tilde{\Omega}(\min\{n^{1/4}, \sqrt{B}\})$ lower bound of Ghazi et al. (2019a) on the ℓ_∞-error of any single-message protocol whereas the expected ℓ_∞ error in Corollary 2 is at most $O(\frac{\log B}{\varepsilon})$. We prove that this is also impossible for binary summation:

Theorem 3 (Binary Summation Lower Bound). *Let $\delta = 1/n^{\Omega(1)}$ and $\varepsilon \leq O(1)$. Then, any (ε, δ)-DP protocol for Binary Summation in the single-message shuffled model should incur an expected squared error of at least $\Omega(\log n)$.*

In light of the lower bound in Theorem 3 and the aforementioned lower bound of Ghazi et al. (2019a), it is striking that the protocols in Theorem 1 and Corollary 2 can get arbitrarily close to the central performance of the Discrete Laplace mechanism while being *almost* single-message: the vast majority of users send a single message (consisting of a single bit in the binary summation protocol and $\lceil \log B \rceil + 1$ bits in the histogram protocol) while only a random $o(1)$ fraction of users sends more than one message!

We point out that, as in previous work in the shuffled and local models of DP, the communication costs in Theorem 1 and Corollary 2 exclude the encryption costs. However, as different messages sent by the users have to be encrypted separately, the encryption overhead increases with the number of messages, and hence our almost single-message protocols would be even more appealing compared to other multi-message procedures as in Ghazi et al. (2020b); Balle et al. (2020); Ghazi et al. (2019a) when the encryption costs are taken into account.

Experimental Evaluation. We implement our algorithms and compare their performance to several alternatives proposed in the literature, both for binary summation and histogram. For the latter, we evaluate the algorithms on public 1940 US Census IPUMS dataset, considering both categorical and numerical features. Our experiments support our theoretical analysis: for a broad setting of n, ε, δ, and B, we incur small communication overhead while achieving near-central errors that are noticeably smaller than previous protocols. The experimental results are presented in the supplementary material.

Remark 4. *We note that our algorithms in Theorem 1 and Corollary 2 can be used to learn the empirical distribution of the users' data up a small error. In light of the near-optimality properties of the Discrete Laplace distribution in the central DP model (Ghosh et al., 2012), our algorithms are also close to optimal. This holds for general error measures including the ℓ_1, ℓ_2, and ℓ_∞ norms, which are well-studied in the literature on distribution estimation and learning (e.g., (Kairouz et al., 2016; Acharya & Sun, 2019)).*

1.2. Overview of Techniques

Before outlining the proof of Theorem 1, we first note that the Discrete Laplace mechanism in the central model incurs only a *constant* MSE. To get a similar bound in the shuffled model, any single-message protocol—in particular, Randomized Response and RAPPOR (Erlingsson et al., 2014) (which for binary summation coincides with Randomized Response)—is ruled out by Theorem 3. Furthermore, the recent histogram protocols of Ghazi et al. (2019a); Erlingsson et al. (2020), are also not applicable since they all incur an MSE of $\Omega(\log n)$. Theorem 1 also guarantees vanishing communication overhead: this rules out the split-and-mix protocol in Ghazi et al. (2020b); Balle et al. (2020) and a recent protocol of Ghazi et al. (2020a).

We next recall the prototypical private binary summation procedures in the central setup. If user i's input is x_i, then the analyzer simply computes the correct sum $\sum_{i\in[n]} x_i$ and then adds to it a random variable sampled from some

probability distribution $\mathcal{D}$. A common choice of $\mathcal{D}$ is the Discrete Laplace distribution with parameter ε, which yields an $(\varepsilon, 0)$-DP protocol for binary summation with an asymptotically tight MSE of $O(1/\varepsilon^2)$; this is known to be optimal in the central DP model (Ghosh et al., 2012).

Using Infinitely Divisible Distributions. In order to emulate the prototypical central model mechanism in the shuffled model, we need to distribute both the signal and the noise over the n users. Distributing the signal can be naturally done by having each user i merely send their true input bit x_i. Distributing the noise is significantly more challenging since the shuffled model is symmetric and does not allow coordination of noise across users. Consider the framework, captured in Algorithms 1 and 2 on page 5, where (a) each user sends (possibly several) bits to the shuffler and (b) the analyzer counts the number of 1s received from the shuffler and outputs it as a proxy for the true sum (possibly after subtracting a fixed bias term). In this case, we would need to decompose the noise random variable into n i.i.d. non-negative components, and have each user sample and transmit one component in unary. Distributions that are decomposable into the sum of n i.i.d. (not necessarily non-negative) samples for any positive integer n are well-studied in probability theory and are known as *infinitely divisible*. In DP, the Discrete Laplace distribution was observed to be infinitely divisible by Goryczka & Xiong (2015), and this property was used by Balle et al. (2020) for real summation in the shuffled model, albeit with several messages per user, each consisting of $\Omega(\log n)$ bits. However, decomposing the Discrete Laplace distribution into a sum of i.i.d. *non-negative* samples—as required by our template above—is clearly impossible since its support contains negative values.

One basic discrete non-negative infinitely divisible distribution is the *Poisson* distribution with parameter λ, which can be sampled by summing n i.i.d. samples from a Poisson distribution with parameter λ/n, for any positive integer n. The resulting *Poisson mechanism* can thus be used as a candidate binary summation procedure in the shuffled model. It turns out that this mechanism is (ε, δ)-DP if we set λ to $O\left(\frac{\log(1/\delta)}{\varepsilon^2}\right)$ (see Theorem 11). In this case, the expected communication cost of transmitting the per-user noise is equal to the expectation λ/n, which is much smaller than 1. We note that, for a reason explained in Section 3.2, we can further reduce the communication by considering the Negative Binomial distribution $\mathrm{NB}(r, p)$. This distribution is infinitely divisible as a random sample from $\mathrm{NB}(r, p)$ can be generated by summing n i.i.d. samples from $\mathrm{NB}(r/n, p)$, for any positive integer n.

Unfortunately, it turns out we cannot hope to achieve near-central accuracy using any *non-negative* infinitely divisible distribution. Specifically, we prove in Section 3.3 that for every such noise distribution, the incurred MSE will grow asymptotically with $\log(1/\delta)$. This is in sharp contrast with the error in the central model, which is independent of δ.

Unary Encoding and Correlated Noise. Instead, the support of our noise distribution has to also contain negative values. To allow this, a natural extension of the above template algorithm is to let each message consists of either an increment (e.g., $+1$) value or a decrement (e.g., -1) value. This template leads to a distributed noise strategy that can achieve near-central accuracy, described next. We know from Goryczka & Xiong (2015) that the Discrete Laplace distribution with parameter ε is the same as the distribution of the difference of two independent $\mathrm{NB}(1, e^{-\varepsilon})$ random variables, and is thus infinitely divisible. This noise can be distributed in the shuffled model by letting each user sample two independent random variables Z^1 and Z^2 from $\mathrm{NB}(1/n, e^{-\varepsilon})$, and send Z^1 increment messages and Z^2 decrement messages to the shuffler. This mechanism would achieve the same error as the central Discrete Laplace mechanism. However, since the analyzer can still see the number of increment messages, this scheme is no more private than the (non-negative) mechanism with noise distribution $\mathrm{NB}(1, e^{-\varepsilon})$, and thus cannot be (ε, δ)-DP by virtue of the lower bound (Section 3.3).

To leverage the power of sending both positive and negative messages, we correlate the input-dependent and noise components sent by the users so that the analyzer is unable to extract much information about the user inputs from one type of messages. We do so by employing a *unary version* of the split-and-mix procedure of Ishai et al. (2006); Ghazi et al. (2020b); Balle et al. (2020). Namely, in addition to the aforementioned random variables Z^1 and Z^2, each user will independently sample a third random variable Z^3 from another infinitely divisible distribution, and will send $Z^1 + Z^3$ increment messages and $Z^2 + Z^3$ decrement messages (see Algorithm 3 on page 7). Note that in this case, when Z^3 is sufficiently "spread out", the analyzer cannot extract much information from counting the number of increments alone, since the noise from Z^3 already overwhelms the user inputs. We formalize this intuition by proving that, for carefully selected infinitely divisible noise distributions, the resulting mechanism is (ε, δ)-DP and incurs an error that can be made arbitrarily close to that of the central Discrete Laplace mechanism, while incurring an expected communication overhead per user that goes to 0 with as n increases.

To prove Corollary 2, we run the binary summation protocol in parallel on all B buckets, and instead of sending ± 1 valued messages, we concatenate each with the length $\lceil \log B \rceil$ binary expansion of the index of the bucket being incremented/decremented. While a straightforward implementation of the randomizer has a running time of $\Omega(B)$, we show, using the characterization of infinitely divisible distributions in terms of Discrete Compound Poisson (DCP) distributions,

that the expected running time can be significantly reduced to the order of the expected per-user communication cost.

Size-Freeness. Infinitely divisible noise mechanisms are much easier to deploy in practice compared to general schemes. This is because for fixed (ε, δ), the "noise parameter" of infinitely divisible mechanisms is independent of the number of users n: e.g., in the case of the Poisson Mechanism, the parameter λ only depends on ε and δ. In contrast, computing near-optimal parameters in the shuffled model of the noise parameters for non-infinitely divisible schemes such as Randomized Response (Warner, 1965) and RAPPOR (Erlingsson et al., 2014) requires re-running a time-expensive algorithm for each new value of n (see the supplementary material for more details). This can be undesirable in practice, especially for real-time applications.

1.3. Organization

We provide some background and notation in Section 2. In Sections 3 and 4 we prove Theorem 1. Our experimental setup and results are presented in Section 5. We conclude with some open questions in Section 6. Corollary 2 and Theorem 3 are proved in the supplementary material.

2. Preliminaries

Let n be the number of users and $[n] := \{1, \ldots, n\}$. In this work, we only consider discrete probability distributions that are supported on (possibly negative) integers. We write $X \sim \mathcal{D}$ to denote a random variable X sampled according to $\mathcal{D}$. We let $\mathrm{supp}(\mathcal{D})$ denote the support of $\mathcal{D}$, $\mathbb{E}[\mathcal{D}]$ its mean, and $\mathrm{Var}(\mathcal{D})$ its variance. For two distributions $\mathcal{D}$ and $\mathcal{D}'$, we denote by $\mathcal{D} + \mathcal{D}'$ the distribution of $X + X'$ where X and X' are independently sampled from $\mathcal{D}$ and $\mathcal{D}'$ respectively; $\mathcal{D} - \mathcal{D}'$ is defined similarly. For integer k, we denote by $k + \mathcal{D}$ the distribution of $k + X$ when $X \sim \mathcal{D}$. For any $x \in \mathbb{Z}$ we let $\mathcal{D}(x)$ denote $\Pr_{Z \sim \mathcal{D}}[Z = x]$. We denote by $\mathbf{x} \sim \mathbf{x}'$ two datasets (i.e., n-dimensional vectors) $\mathbf{x}$ and $\mathbf{x}'$ differing on a single user's data (i.e., a single coordinate).

Definition 5 (Differential Privacy (Dwork et al., 2006a;b))**.** *For any parameters $\varepsilon \geq 0$ and $\delta \in [0, 1]$, a randomized mechanism $\mathcal{M}$ is* (ε, δ)-differentially private (DP) *if for every pair of $\mathbf{x} \sim \mathbf{x}'$ and for every subset $\mathcal{S}$ of transcripts of $\mathcal{M}$, it holds that $\Pr[\mathcal{M}(\mathbf{x}) \in \mathcal{S}] \leq e^{\varepsilon} \cdot \Pr[\mathcal{M}(\mathbf{x}') \in \mathcal{S}] + \delta$, where the probabilities are over the randomness in $\mathcal{M}$.*

Shuffled Model. A protocol in the shuffled privacy model consists of three procedures: a *local randomizer* that sends one or several messages depending on its input, a *shuffler* that randomly permutes all incoming messages, and an *analyzer* that takes in the output of the shuffler and returns the final output of the protocol. Privacy is required to be guaranteed with respect to the output of the shuffler.

Algorithm 1 $\mathcal{D}$-Distributed Randomizer.

procedure $\textsc{Randomizer}_{\mathcal{D},n}(x)$
  Sample $Z \sim \mathcal{D}_{/n}$
  Send $x + Z$ messages, where each message is 1

3. The $\mathcal{D}$-Distributed Mechanisms

In this section, we propose and study a family of simple mechanisms in the shuffled model for the binary summation problem. While the mechanisms in this section do not achieve the accuracy promised in Theorem 1, they will serve as an important building block to our eventual algorithm in Section 4. In fact, we will need the privacy guarantee of these mechanisms against a generalization of bit summation called Δ-summation defined as follows. For $\Delta \in \mathbb{N}$, in the Δ-*summation* task, the input to each user is a number x_i in $\{0, \ldots, \Delta\}$ and the goal is to compute $\sum_{i \in [n]} x_i$. When $\Delta = 1$, this task is the same as binary summation.

To define our protocol, we first recall that a standard strategy for achieving DP in the central model is to simply add noise to the correct answer. We will refer to such a mechanism the $\mathcal{D}$ *Mechanism* when the noise distribution is $\mathcal{D}$.

Definition 6. *For any distribution $\mathcal{D}$, the* $\mathcal{D}$ Mechanism *for computing a function $f : \mathcal{X}^n \to \mathbb{Z}^d$ is defined as the mechanism that, on input $\mathbf{x} \in \mathcal{X}^n$, outputs $f(\mathbf{x}) + (Y_1, \ldots, Y_d)$ where $Y_i \sim \mathcal{D}, i \in [d]$ are independent.*

The definition of the $\mathcal{D}$ Mechanism applies even when $\mathrm{supp}(\mathcal{D})$ contains negative integers, but in this section we focus on distributions $\mathcal{D}$ that are supported on non-negative integers and that are infinitely divisible as defined next.

Definition 7 (Infinite Divisibility)**.** *A distribution $\mathcal{D}$ is said to be* infinitely divisible *(abbreviated* ∞-div*) if for every $n \in \mathbb{N}$, there exists a distribution $\mathcal{D}_{/n}$ such that $(X_1 + \cdots + X_n) \sim \mathcal{D}$ where $X_i \sim \mathcal{D}_{/n}, i \in [d]$ are independent.*

For an ∞-div $\mathcal{D}$ on non-negative integers, we define the $\mathcal{D}$*-Distributed Mechanism* in the shuffled model as follows:

Algorithm 2 $\mathcal{D}$-Distributed Analyzer.

procedure $\textsc{Analyzer}_{\mathcal{D}}$
  $U \leftarrow$ number of messages received
  return $U - \mathbb{E}[\mathcal{D}]$

Privacy. Observe that, from the analyzer's perspective, it only sees U (because all the messages are identical) and U is distributed exactly as $\sum_{i \in [n]} x_i + \mathcal{D}$ by ∞-div of $\mathcal{D}$. From this, we immediately get that the privacy guarantee of the $\mathcal{D}$-Distributed Mechanism in the *shuffled* model is the same as that of the $\mathcal{D}$ Mechanism in the *central* model.

Observation 8. *For any $\varepsilon > 0$ and $\delta \in (0, 1)$, the $\mathcal{D}$-Distributed Mechanism is (ε, δ)-DP in the shuffled model*

for Δ-summation if and only if the $\mathcal{D}$ Mechanism is (ε, δ)-DP in the central model for Δ-summation.

Accuracy. As discussed above, we are guaranteed in Algorithm 2 that $U = \sum_{i\in[n]} x_i + \mathcal{D}$, which gives:

Observation 9. *The error of the $\mathcal{D}$-distributed Mechanism is distributed as $\mathcal{D} - \mathbb{E}[\mathcal{D}]$.*

Expected Communication. The expected number of messages sent by each user in Algorithm 1 is $x + \mathbb{E}[\mathcal{D}_{/n}] = x + \frac{\mathbb{E}[\mathcal{D}]}{n}$. Using the fact that $x \in \{0, \ldots, \Delta\}$, we get:

Observation 10. *The expected number of messages sent by a user in the $\mathcal{D}$-Distributed Mechanism is at most $\Delta + \frac{\mathbb{E}[\mathcal{D}]}{n}$.*

3.1. Example I: The Poisson Mechanism

Arguably, the simplest protocol in the family of $\mathcal{D}$-Distributed Mechanisms is the *Poisson Mechanism* that uses the Poisson distribution[1], which is ∞-div. In this protocol, $\mathcal{D}$ is $\mathrm{Poi}(\lambda)$ for some $\lambda \in \mathbb{R}^+$ and $\mathcal{D}_{/n}$ is simply $\mathrm{Poi}(\lambda/n)$. We now compute its privacy guarantee.

Theorem 11. *For any $\varepsilon > 0$, $\delta \in (0,1)$, and $\Delta \in \mathbb{N}$, the $\mathrm{Poi}(\lambda)$ Mechanism with $\lambda = \frac{16\log(10/\delta)}{(1-e^{-\varepsilon/\Delta})^2} + \frac{2\Delta}{1-e^{-\varepsilon/\Delta}}$, is (ε, δ)-DP in the central model for Δ-summation.*

By setting $\Delta = 1$ and using Observations 8, 9, and 10, we get the following for binary summation.

Corollary 12. *For any $0 < \varepsilon \le O(1)$ and $\delta \in (0,1)$, let λ be as in Theorem 11 with $\Delta = 1$. The $\mathrm{Poi}(\lambda)$-Distributed Mechanism is (ε, δ)-DP for binary summation in the shuffled model, each user sends at most $1 + O\left(\frac{\log(1/\delta)}{\varepsilon^2 n}\right)$ one-bit messages in expectation, and the MSE is $O\left(\frac{\log(1/\delta)}{\varepsilon^2}\right)$.*

3.2. Example II: The Negative Binomial Mechanism

A disadvantage of the Poisson Mechanism is that, in the most important regime where $\frac{\varepsilon}{\Delta} \ll 1$, the expected number of messages sent is $1 + O\left(\frac{\log(1/\delta)}{n} \cdot \left(\frac{\Delta}{\varepsilon}\right)^2\right)$. In this subsection, we show how to reduce the dependency on $\frac{\Delta}{\varepsilon}$ from $(\frac{\Delta}{\varepsilon})^2$ to $\frac{\Delta}{\varepsilon}$, while retaining a similar error bound. (As we will see in Section 4, this dependency will also permeate to our eventual algorithm in the proof of Theorem 1.) Before doing so, we note that the $(\frac{\Delta}{\varepsilon})^2$ dependency is necessary for the Poisson Mechanism since it is well-known[2] that the MSE of any central $(\varepsilon, o(1))$-DP protocol for Δ-summation has to be at least $\Omega((\frac{\Delta}{\varepsilon})^2)$ and since the MSE of the $\mathrm{Poi}(\lambda)$ Mechanism is exactly λ, we must hence set $\lambda \ge (\frac{\Delta}{\varepsilon})^2$. Thus, the expected number of messages sent per user in the $\mathrm{Poi}(\lambda)$ Mechanism must be $1 + \frac{\lambda}{n} \ge 1 + \Omega\left(\frac{1}{n} \cdot \left(\frac{\Delta}{\varepsilon}\right)^2\right)$.

To circumvent this, we first observe that the above argument holds only because the parameter λ of the Poisson Mechanism governs both the MSE (i.e., the variance of the distribution) and the number of messages sent (i.e., the expectation of the distribution). This motivates us to seek an ∞-div distribution on the negative integers whose variance and mean can be very different. We consider the negative binomial distribution[3] $\mathrm{NB}(r,p)$ which is ∞-div as $\mathrm{NB}(r,p) = \sum_{i=1}^{n} \mathrm{NB}(\frac{r}{n}, p)$ for every $n \in \mathbb{N}$. Moreover, $\mathbb{E}[\mathrm{NB}(r,p)] = \frac{pr}{(1-p)}$ and $\mathrm{Var}[\mathrm{NB}(r,p)] = \frac{pr}{(1-p)^2}$, which can be very different when p is close to 1. We next show a DP guarantee for this Negative Binomial Mechanism.

Theorem 13. *For any $\varepsilon > 0, \delta \in [0,1)$, and $\Delta \in \mathbb{N}$, let $p = e^{-0.1\varepsilon/\Delta}$ and $r = 50 \cdot e^{\varepsilon/\Delta} \cdot \log\left(\frac{1}{\delta}\right)$. The $\mathrm{NB}(r,p)$ Mechanism is (ε, δ)-DP in the central model.*

Plugging $\Delta = 1$, we get the following corollary. When compared to Poisson Mechanism (Corollary 12), we achieve the same error bound but with a $\frac{1}{\varepsilon}$ instead of $\frac{1}{\varepsilon^2}$ multiplicative term in the expected number of additional messages sent.

Corollary 14. *For any $\varepsilon, \delta > 0$ with $\varepsilon \le O(1)$, let p, r be as in Theorem 13 with $\Delta = 1$. The $\mathrm{NB}(r,p)$-Distributed Mechanism is (ε, δ)-DP for binary summation in the shuffled model, each user sends at most $1 + O\left(\frac{\log(1/\delta)}{\varepsilon n}\right)$ one-bit messages in expectation, and the MSE is $O\left(\frac{\log(1/\delta)}{\varepsilon^2}\right)$.*

3.3. A Lower Bound for $\mathcal{D}$-Distributed Mechanisms

The downside of the Poisson and Negative Binomial Mechanisms is that they suffer from an MSE of $O_\varepsilon(\log(\frac{1}{\delta}))$ instead of the $O(\frac{1}{\varepsilon^2})$ MSE, independent of δ, of the central Discrete Laplace Mechanism. It turns out that this dependency on $\log(\frac{1}{\delta})$ is necessary for every $\mathcal{D}$-Distributed Mechanism:

Lemma 15. *For any infinitely divisible distribution $\mathcal{D}$ on non-negative integers, if $\mathcal{D}$-Distributed Mechanism is (ε, δ)-DP in the shuffled model for binary summation, then the MSE of the mechanism is $\Omega_\varepsilon(\log(1/\delta))$.*

In other words, $\mathcal{D}$-Distributed Mechanisms do not suffice for the goal of achieving near-central error guarantees.

4. Correlated Distributed Mechanisms

We next present a family of protocols in the shuffled model that will overcome the lower bound barrier of Lemma 15 and achieve an accuracy/privacy trade-off arbitrarily close to the central model. We start by outlining the intuition behind the protocol. First, suppose hypothetically that we could somehow implement the $\mathcal{D}$ Mechanism in the shuffled

[1] $\mathrm{Poi}(\lambda)$ is defined as $\mathrm{Poi}(k;\lambda) = \lambda^k e^{-k}/k!$.

[2] This follows from the sensitivity of Δ-summation; see, e.g.. (Vadhan, 2017).

[3] $\mathrm{NB}(r,p)$ is defined as $\mathrm{NB}(k;r,p) = \binom{k+r-1}{k}(1-p)^r p^k$. We remark that the negative binomial distribution may be viewed as a generalization of the Poisson distribution: when taking $r \to \infty$ and letting $p = \lambda/r$, $\mathrm{NB}(r,p)$ point-wise converges to $\mathrm{Poi}(\lambda)$.

model when $\mathrm{supp}(\mathcal{D})$ can contain negative integers. Then, we would actually be done! This is because the Discrete Laplace distribution[4] is ∞-div as $\mathrm{DLap}(\varepsilon) = \mathrm{NB}(1, e^{-\varepsilon}) - \mathrm{NB}(1, e^{-\varepsilon})$ (see, e.g., (Kotz et al., 2001)), and $\mathrm{NB}(1, e^{-\varepsilon})$ is ∞-div, and is in fact the geometric distribution. Of course, the problem is that if we sample the number of messages from $\mathrm{DLap}(\varepsilon)_{/n}$ we will often try to send a *negative* number of messages, which is meaningless!

This leads us to using two types of messages: one for increments (denoted $+1$) and one for decrements (denoted -1). The $+1$ messages are sampled as in the $\mathrm{NB}(1, e^{-\varepsilon})$ Mechanism, and so are the -1 messages except that each user pretends that their input is 0 in this case. The analyzer's answer is the difference between the number of $+1$ messages and the number of -1 messages it receives. The aforementioned fact that the difference of two $\mathrm{NB}(1, e^{-\varepsilon})$ random variables is $\mathrm{DLap}(\varepsilon)$ implies that this protocol has the same accuracy as the central Discrete Laplace Mechanism, as desired. However, this protocol is not (ε, δ)-DP in the shuffled model. To see this, note that the analyzer sees the number of $+1$ messages received, and hence the protocol is no more private than the $\mathrm{NB}(1, e^{-\varepsilon})$ Mechanism, which is not (ε, δ)-DP as explained by Lemma 15. To overcome this, we "mask" the numbers of $+1$ and -1 messages by sampling a random variable Z from $\mathcal{D}_{/n}$ for some other ∞-div $\mathcal{D}$ over non-negative integers, and additionally send Z increments (i.e., $+1$) and Z decrements (i.e., -1). Clearly, this does not affect the accuracy of the analyzer, but we will show that it improves privacy. For every choice of ∞-div $\mathcal{D}^1, \mathcal{D}^2, \mathcal{D}^3$ on non-negative integers, we define the $(\mathcal{D}^1, \mathcal{D}^2, \mathcal{D}^3)$-*Correlated Distributed Mechanism*:

Algorithm 3 $(\mathcal{D}^1, \mathcal{D}^2, \mathcal{D}^3)$-Correlated Distributed Randomizer

procedure $\textsc{Randomizer}_{\mathcal{D}^1, \mathcal{D}^2, \mathcal{D}^3, n}(x)$
  Sample $Z^1 \sim \mathcal{D}^1_{/n}$
  Sample $Z^2 \sim \mathcal{D}^2_{/n}$
  Sample $Z^3 \sim \mathcal{D}^3_{/n}$
  Send $x + Z^1 + Z^3$ many $+1$ messages.
  Send $Z^2 + Z^3$ many -1 messages.

Algorithm 4 $(\mathcal{D}^1, \mathcal{D}^2, \mathcal{D}^3)$-Correlated Distributed Analyzer

procedure $\textsc{Analyzer}_{\mathcal{D}^1, \mathcal{D}^2}$
  $U_{+1} \leftarrow$ number of $+1$ messages received.
  $U_{-1} \leftarrow$ number of -1 messages received.
  return $U_{+1} - U_{-1} - \mathbb{E}[\mathcal{D}^1 - \mathcal{D}^2]$

Accuracy and Communication Complexity. The accuracy and communication complexity of the protocol can be derived as in the $\mathcal{D}$-Distributed Mechanism from Section 3:

Observation 16. *The error of $(\mathcal{D}^1, \mathcal{D}^2, \mathcal{D}^3)$-Distributed Mechanism is distributed as $(\mathcal{D}^1 - \mathcal{D}^2) - \mathbb{E}[\mathcal{D}^1 - \mathcal{D}^2]$.*

Observation 17. *The expected number of messages sent by each user in the $(\mathcal{D}^1, \mathcal{D}^2, \mathcal{D}^3)$-Distributed Mechanism is at most $1 + \frac{\mathbb{E}[\mathcal{D}^1] + \mathbb{E}[\mathcal{D}^2] + 2 \cdot \mathbb{E}[\mathcal{D}^3]}{n}$.*

Privacy. The crux of our DP proof for the $(\mathcal{D}^1, \mathcal{D}^2, \mathcal{D}^3)$-Correlated Distributed Mechanism is the following theorem:

Theorem 18. *Let $\Delta > 0$ and $\mathcal{D}^1, \mathcal{D}^2, \mathcal{D}^3$ satisfy:*

- *(Privacy of True Noise) The $(\mathcal{D}^1 - \mathcal{D}^2)$ Mechanism is ε_1-DP for binary summation in the central model.*
- *(Privacy of Correlated Noise) The $\mathcal{D}^3$ Mechanism is $(\varepsilon_2, \delta_2)$-DP for Δ-summation in the central model.*
- *(Concentration of Neg. Noise) $\Pr_{Y \sim \mathcal{D}^2}[Y > \Delta] \le \delta_3$.*

Then, the $(\mathcal{D}^1, \mathcal{D}^2, \mathcal{D}^3)$-Correlated Distributed Mechanism is $(\varepsilon_1 + \varepsilon_2, e^{\varepsilon_1} \cdot \delta_2 + 2e^{2\varepsilon_1} \cdot \delta_3)$-DP in the shuffled model.

Proof Overview. All the analyzer sees is $(U_{+1}, U_{-1}) = (\sum_{i \in [n]} x_i + \hat{Z}^1 + \hat{Z}^3, \hat{Z}^2 + \hat{Z}^3)$ where for $j \in [3]$, $\hat{Z}^j \sim \mathcal{D}^j$. Since there is bijection between (U_{+1}, U_{-1}) and $(U_{+1} - U_{-1}, U_{-1}) = (\sum_{i \in [n]} x_i + \hat{Z}^1 - \hat{Z}^2, \hat{Z}^2 + \hat{Z}^3)$, we may consider the distribution on the latter. The first coordinate is the same as the analyzer's view from the $(\mathcal{D}^1 - \mathcal{D}^2)$ Mechanism, which is ε_1-DP by the first assumption. Once we condition on $U_{+1} - U_{-1}$ being equal to some value, we are left to consider a distribution on U_{-1}. This distribution is not the same as the original one (before conditioning) since U_{-1} and $U_{+1} - U_{-1}$ are correlated. But this correlation only comes via $\hat{Z}^2$, as $\hat{Z}^3$ does not appear in $U_{+1} - U_{-1}$. By the concentration of $\mathcal{D}^2$ (the third assumption), $\hat{Z}^2$ is rarely larger than Δ. When $\hat{Z}^2 \le \Delta$, we can use the $(\varepsilon_2, \delta_2)$-DP of the $\mathcal{D}^3$ Mechanism for Δ-summation to argue the privacy of the protocol. The proof follows this intuition while carefully tracking the privacy loss in each step.

4.1. Near-Central Accuracy with Shuffled Mechanisms

We use Theorem 18 to derive our "near-central" protocol (Theorem 1). Specifically, we set $\mathcal{D}^1, \mathcal{D}^2$ so that $\mathcal{D}^1 - \mathcal{D}^2 = \mathrm{DLap}(0.99\varepsilon)$, which implies that the $(\mathcal{D}^1 - \mathcal{D}^2)$ Mechanism is 0.99ε-DP in the central model. The remaining privacy budget of 0.01ε is allocated to the $\mathcal{D}^3$ Mechanism, which we set to be the $\mathrm{NB}(\cdot, \cdot)$ Mechanism with appropriate parameters. We use the Negative Binomial rather than the Poisson distribution as the former (Theorem 13) has a smaller communication cost than the latter (Theorem 11).

[4] $\mathrm{DLap}(s)$ is defined as $\mathrm{DLap}(k; s) = \frac{1}{C(s)} \cdot e^{-|k| \cdot s}$, where $C(s) = \sum_{k=-\infty}^{\infty} e^{-|k| \cdot s}$ is the normalization constant.

5. Experimental Evaluation and Results

5.1. Binary Summation

In this section, we evaluate our protocols. Specifically, we consider the Poisson Distributed Mechanism (Theorem 11) and the Correlated Distributed Noise Mechanism (Theorems 1 and 18). We consider the root mean square error (RMSE), which is independent of the input data for all methods considered, and for this reason there is no need to consider performance on particular data sets. For each setting of ε, δ, our parameters are selected in an accurate manner; this is explained in detail in the supplementary material. We only note here that for the Correlated Distributed Mechanism, we set $\mathcal{D}^1 = \mathcal{D}^2 = \mathrm{NB}(1, e^{-\varepsilon_1})$ with ε_1 such that the RMSE of the protocol is 20% more than that of the (central) $\mathrm{DLap}(\varepsilon)$ Mechanism. We compare our algorithms against the classic *Randomized Response (RR)* algorithm, where a user with input x sends x w.p. p, and $1-x$ w.p. $1-p$, for some parameter $p \in [0, 1/2]$.

Error. We compute the errors as ε or δ varies. The corresponding plots are shown in Figure 2; we include the error of the (central) Discrete Laplace Mechanism for comparison (but of course this is not directly implementable in the shuffled model). We remark that the RMSEs of our protocols are independent of the number of users n, and only the RMSE of RR depends on n, which we choose to be 10,000. While the Correlated Distributed protocol has a constant RMSE as we vary δ, both Poisson and RR incur larger RMSEs. In particular, when $\delta = 10^{-6}, \varepsilon = 1$, the RMSE of the Correlated Distributed protocol is 3.5 times less than that of Poisson and RR (which essentially coincide). The fact that the Poisson Mechanism and RR have essentially the same RMSE should come as no surprise, since the binomial distribution $\mathrm{Bin}(n, p)$ converges (in the distributional sense) to the Poisson distribution $\mathrm{Poi}(np)$ as $n \to \infty$ and p is kept constant. This means that we would prefer RR in this case, since it always sends one message.

Communication Complexity. Figure 3 shows the plots of the expected number of additional messages sent by each user in our Poisson and Correlated Distributed Mechanisms. Here we let $n = 10,000$. In the reasonable setting where $\delta = 10^{-6}$ and $\varepsilon = 1$, the expected number of additional messages sent in the Correlated Distributed Mechanism is only 0.04, whereas in the Poisson Mechanism, it is even smaller at 0.0003. Even in the more extreme case of $\varepsilon = 0.1$, the former is still 0.278 and the latter is only 0.141.

5.2. Histograms

We have also performed experiments on the histogram versions of our Correlated Distributed and Poisson Mechanisms, and compare them against three algorithms from the literature: B-Randomized Response (B-RR), RAPPOR (Erlingsson et al., 2014), and Fragmented RAPPOR (Erlingsson et al., 2020). Each of these three can be viewed as a B-ary generalization of the binary RR. We ran these algorithms on two IPUMS datasets (Ruggles et al., 2019). Due to space constraints, the full description of the experiments and results are deferred to the supplementary material. Here we just summarize our findings: For most parameters, RAPPOR incurs significantly larger errors than B-RR. Moreover, similarly to how RR mirrors the Poisson Mechanism in the binary case, Fragmented RAPPOR gives almost the same results as Poisson for histograms. In terms of RMSE, our correlated mechanism is significantly closer to the central model error than its competitors. The plots are in fact very similar to those of binary summation for the Correlated Distributed and Poisson Mechanisms, with RR doing 25-50% worse than Poisson. In terms of ℓ_∞, RR incurs more than $3\times$ ℓ_∞ errors compared to our algorithms. Furthermore, as corroborated by theory (Ghazi et al., 2019a), the error of RR grows quickly with the number B of buckets, while those of our mechanisms grow very slowly.

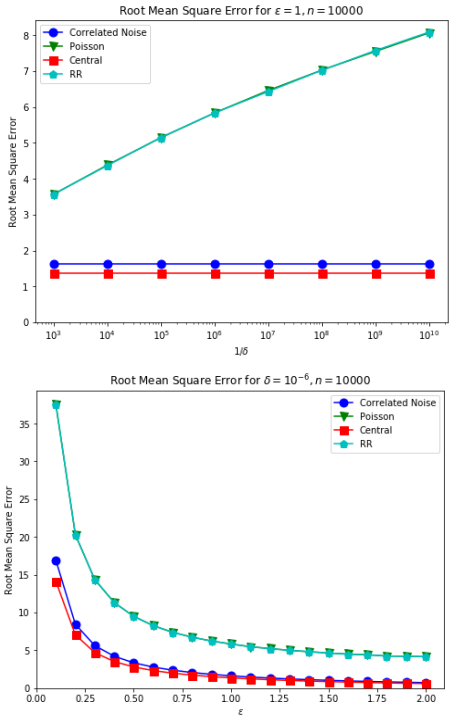

Figure 2. RMSE of the protocols for binary summation. Note that the RMSEs of RR and Poisson are essentially the same.

6. Conclusions and Open Questions

We proposed DP algorithms in the shuffled model for binary summation and histograms with accuracy arbitrarily close to the central model and a vanishing communication overhead. There are several questions left open by our work. One

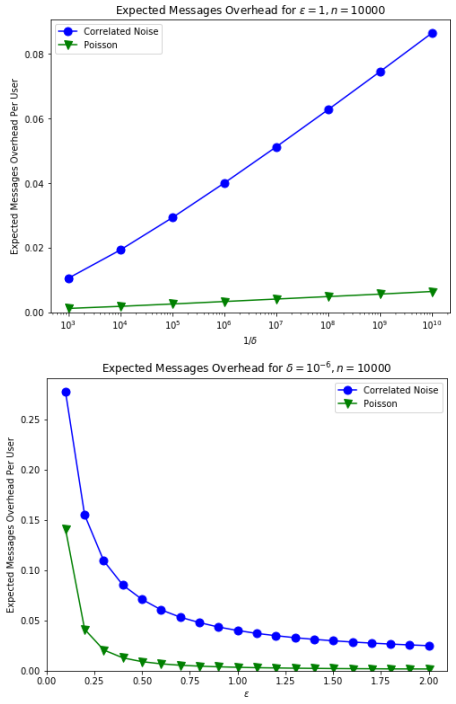

Figure 3. Expected additional number of messages sent by each user for binary summation. While we use $n = 10^4$, this expectation scales linearly in $1/n$. E.g., if $n = 10^5$, the plots will look the same, but with the y-axis scaled down by a factor of 10.

is to obtain algorithms achieving near-central performance with negligible communication overhead for the problems of real summation, vector summation, and histograms over a large number $B \geq \Omega(n)$ of buckets. Another question is to prove a lower bound against *expected single-message* protocols (that can send 0, 1 or more messages in the worst-case) as our lower bound in Theorem 3 does not hold in this case. A very interesting related direction is to build on our algorithms to improve the communication complexity of DP stochastic gradient descent in a federated learning setup.

Acknowledgements

We would like to thank Noah Golowich, Laura Peskin and Alex Zani for insightful discussions and feedback.

References

Abowd, J. M. The US Census Bureau adopts differential privacy. In *KDD*, pp. 2867–2867, 2018.

Acharya, J. and Sun, Z. Communication complexity in locally private distribution estimation and heavy hitters. In *ICML*, pp. 51–60, 2019.

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

Balcer, V. and Cheu, A. Separating local & shuffled differential privacy via histograms. In *ITC*, pp. 1:1–1:14, 2020.

Balle, B., Bell, J., Gascón, A., and Nissim, K. The privacy blanket of the shuffle model. In *CRYPTO*, pp. 638–667, 2019.

Balle, B., Bell, J., Gascón, A., and Nissim, K. Private summation in the multi-message shuffle model. *arXiv: 2002.00817*, 2020.

Bassily, R. and Smith, A. Local, private, efficient protocols for succinct histograms. In *STOC*, pp. 127–135, 2015.

Bassily, R., Nissim, K., Stemmer, U., and Thakurta, A. G. Practical locally private heavy hitters. In *NIPS*, pp. 2288–2296, 2017.

Beimel, A., Nissim, K., and Omri, E. Distributed private data analysis: Simultaneously solving how and what. In *CRYPTO*, pp. 451–468, 2008.

Bittau, A., Erlingsson, Ú., Maniatis, P., Mironov, I., Raghunathan, A., Lie, D., Rudominer, M., Kode, U., Tinnés, J., and Seefeld, B. Prochlo: Strong privacy for analytics in the crowd. In *SOSP*, pp. 441–459, 2017.

Blum, A., Dwork, C., McSherry, F., and Nissim, K. Practical privacy: the SuLQ framework. In *PODS*, pp. 128–138, 2005.

Chan, T. H., Shi, E., and Song, D. Optimal lower bound for differentially private multi-party aggregation. In *ESA*, pp. 277–288, 2012.

Cheu, A., Smith, A. D., Ullman, J., Zeber, D., and Zhilyaev, M. Distributed differential privacy via shuffling. In *EUROCRYPT*, pp. 375–403, 2019.

Ding, B., Kulkarni, J., and Yekhanin, S. Collecting telemetry data privately. In *NIPS*, pp. 3571–3580, 2017.

Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., and Naor, M. Our data, ourselves: Privacy via distributed noise generation. In *EUROCRYPT*, pp. 486–503, 2006a.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. Calibrating noise to sensitivity in private data analysis. In *TCC*, pp. 265–284, 2006b.

Erlingsson, Ú., Pihur, V., and Korolova, A. RAPPOR: Randomized aggregatable privacy-preserving ordinal response. In *CCS*, pp. 1054–1067, 2014.

Erlingsson, Ú., Feldman, V., Mironov, I., Raghunathan, A., Talwar, K., and Thakurta, A. Amplification by shuffling: From local to central differential privacy via anonymity. In *SODA*, pp. 2468–2479, 2019.

Erlingsson, Ú., Feldman, V., Mironov, I., Raghunathan, A., Song, S., Talwar, K., and Thakurta, A. Encode, shuffle, analyze privacy revisited: Formalizations and empirical evaluation. *arXiv:2001.03618*, 2020.

Ghazi, B., Golowich, N., Kumar, R., Pagh, R., and Velingker, A. On the power of multiple anonymous messages. Cryptology ePrint Archive:1382, 2019a.

Ghazi, B., Pagh, R., and Velingker, A. Scalable and differentially private distributed aggregation in the shuffled model. *arXiv: 1906.08320*, 2019b.

Ghazi, B., Kumar, R., Manurangsi, P., Pagh, R., and Velingker, A. Pure differentially private summation from anonymous messages. In *ITC*, pp. 15:1–15:23, 2020a.

Ghazi, B., Manurangsi, P., Pagh, R., and Velingker, A. Private aggregation from fewer anonymous messages. In *EUROCRYPT*, pp. 798–827, 2020b.

Ghosh, A., Roughgarden, T., and Sundararajan, M. Universally utility-maximizing privacy mechanisms. *SICOMP*, 41(6):1673–1693, 2012.

Goryczka, S. and Xiong, L. A comprehensive comparison of multiparty secure additions with differential privacy. *IEEE TDSC*, 14(5):463–477, 2015.

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

Ishai, Y., Kushilevitz, E., Ostrovsky, R., and Sahai, A. Cryptography from anonymity. In *FOCS*, pp. 239–248, 2006.

Kairouz, P., Bonawitz, K., and Ramage, D. Discrete distribution estimation under local privacy. In *ICML*, pp. 2436–2444, 2016.

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., D'Oliveira, R. G. L., Rouayheb, S. E., Evans, D., Gardner, J., Garrett, Z., Gascón, A., Ghazi, B., Gibbons, P. B., Gruteser, M., Harchaoui, Z., He, C., He, L., Huo, Z., Hutchinson, B., Hsu, J., Jaggi, M., Javidi, T., Joshi, G., Khodak, M., Konečný, J., Korolova, A., Koushanfar, F., Koyejo, S., Lepoint, T., Liu, Y., Mittal, P., Mohri, M., Nock, R., Özgür, A., Pagh, R., Raykova, M., Qi, H., Ramage, D., Raskar, R., Song, D., Song, W., Stich, S. U., Sun, Z., Suresh, A. T., Tramèr, F., Vepakomma, P., Wang, J., Xiong, L., Xu, Z., Yang, Q., Yu, F. X., Yu, H., and Zhao, S. Advances and open problems in federated learning. *arXiv: 1912.04977*, 2019.

Kasiviswanathan, S. P., Lee, H. K., Nissim, K., Rashkodnikova, S., and Smith, A. What can we learn privately? In *FOCS*, pp. 531–540, 2008.

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

Konečnỳ, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., and Bacon, D. Federated learning: Strategies for improving communication efficiency. *arXiv: 1610.05492*, 2016.

Kotsogiannis, I., Tao, Y., He, X., Fanaeepour, M., Machanavajjhala, A., Hay, M., and Miklau, G. PrivateSQL: a differentially private SQL query engine. *VLDB*, 12(11):1371–1384, 2019.

Kotz, S., Kozubowski, T., and Podgorski, K. *The Laplace Distribution and Generalizations: A Revisit with Applications to Communications, Economics, Engineering, and Finance.* Progress in Mathematics Series, 2001.

Ruggles, S., Flood, S., Goeken, R., Grover, J., Meyer, E., Pacas, J., and Sobek, M. Integrated public use microdata series (IPUMS) USA: Version 9.0 [dataset]. *Minneapolis, MN*, 2019. URL https://doi.org/10.18128/D010.V9.0.

Seide, F., Fu, H., Droppo, J., Li, G., and Yu, D. 1-bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs. In *INTERSPEECH*, pp. 1058–1062, 2014.

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

Stemmer, U. Locally private k-means clustering. In *SODA*, pp. 548–559, 2020.

Suresh, A. T. Differentially private anonymized histograms. In *NeurIPS*, pp. 7969–7979, 2019.

Suresh, A. T., Yu, F. X., Kumar, S., and McMahan, H. B. Distributed mean estimation with limited communication. In *ICML*, pp. 3329–3337, 2017.

Vadhan, S. *The Complexity of Differential Privacy*, pp. 347–450. Springer International Publishing, 2017.

Wang, T., Xu, M., Ding, B., Zhou, J., Li, N., and Jha, S. MURS: Practical and robust privacy amplification with multi-party differential privacy. *arXiv:1908.11515*, 2019.

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

Wilson, R. J., Zhang, C. Y., Lam, W., Desfontaines, D., Simmons-Marengo, D., and Gipson, B. Differentially private SQL with bounded user contribution. *arXiv:1909.01917*, 2019.