Shuffled Model of Federated Learning: Privacy, Accuracy and Communication Trade-offs

Antonious M. Girgis, Deepesh Data, Suhas Diggavi,
Peter Kairouz, and Ananda Theertha Suresh

***Abstract*—We consider a distributed empirical risk minimization (ERM) optimization problem with communication efficiency and privacy requirements, motivated by the federated learning (FL) framework [1]. Unique challenges to the traditional ERM problem in the context of FL include (i) need to provide privacy guarantees on clients' data, (ii) compress the communication between clients and the server, since clients might have low-bandwidth links, (iii) work with a dynamic client population at each round of communication between the server and the clients, as a small fraction of clients are sampled at each round. To address these challenges we develop (optimal) communication-efficient schemes for private mean estimation for several ℓ_p spaces, enabling efficient gradient aggregation for each iteration of the optimization solution of the ERM. We also provide lower and upper bounds for mean estimation with privacy and communication constraints for arbitrary ℓ_p spaces. To get the overall communication, privacy, and optimization performance operation point, we combine this with privacy amplification opportunities inherent to this setup. Our solution takes advantage of the inherent privacy amplification provided by client sampling and data sampling at each client (through Stochastic Gradient Descent) as well as the recently developed privacy framework using anonymization, which effectively presents to the server responses that are randomly shuffled with respect to the clients. Putting these together, we demonstrate that one can get the same privacy, optimization-performance operating point developed in recent methods that use full-precision communication, but at a much lower communication cost, *i.e.,* effectively getting communication efficiency for "free".**

I. Introduction

In this paper we consider a federated learning (FL) framework [1]–[3], where the data is generated across m clients. The server wants to learn a machine learning model that minimizes a certain objective function using the m local datasets, without collecting the data at the central server due to privacy considerations. Specifically, each client i has a local dataset $\mathcal{D}_i = \{d_{i1}, \ldots, d_{ir}\} \subset \mathfrak{S}^r$ comprising r data points, where $\mathfrak{S}$ is the set from which all clients data is from.[1] The server wants to solve the following empirical risk minimization problem:

$$\arg\min_{\theta \in \mathcal{C}} \left(F(\theta) := \frac{1}{m} \sum_{i=1}^{m} F_i(\theta) \right). \tag{1}$$

Here, $\mathcal{C} \subset \mathbb{R}^d$ is a closed convex set, and $F_i(\theta)$ is a local loss function dependent on the local dataset $\mathcal{D}_i$ at client i evaluated at the model parameters $\theta \in \mathbb{R}^d$; see Figure 1 for a pictorial representation of the setting and Section III for more details on the problem setup. In order to generate a learning model using (1), the commonly used mechanism is Stochastic Gradient Descent (SGD) [4]. Federated learning (FL) introduces several unique challenges to this traditional model that cause tension with the objective in (1): (i) we need to provide privacy guarantees on the locally residing data $\mathcal{D}_i$ at client i, as the data not only needs to *remain* at the clients but additionally needs to be kept private according to certain requirements/guarantees; (ii) compress (as efficiently as possible) the communication between clients and the server, since the clients may connect with low-bandwidth (wireless) links; and (iii) work with a dynamic client population in each round of communication between the server and the clients. This happens due to scale (*e.g.,* tens of millions of devices) and only a small fraction of clients are sampled at each communication round depending on their availability.

These requirements make the problem challenging, especially when one wants to give strong privacy guarantees while training models that give good learning performance. Since we need to give privacy to the local data residing at the clients, the traditional framework to give guarantees is through the notion of local differential privacy, where the server is itself untrusted. The challenge is that traditional approaches to learning under local differential privacy (LDP) [5]–[9] are known to give poor learning performance [7], [9], [10].

In recent works, a new privacy framework using anonymization has been proposed in the so-called *shuffling model* [11]–[19]. This model enables significantly better privacy-utility performance by amplifying privacy (scaling with number of clients as $\frac{1}{\sqrt{m}}$ with respect to LDP) through this anonymization, which effectively presents the central server with responses

Antonious M. Girgis, Deepesh Data, and Suhas Diggavi are with the University of California, Los Angeles, USA. Peter Kairouz and Ananda Theertha Suresh are with Google Research, USA.

Email: amgirgis@g.ucla.edu, deepesh.data@gmail.com, suhas@ee.ucla.edu, kairouz@google.com, theertha@google.com.

This was supported by the NSF grant #1740047 and by the UC-NL grant LFR-18-548554. This work was also supported in part through the Google Faculty Research Award.

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org., provided by the author. The material includes Appendices for proofs and discussions referenced in the paper. Contact amgirgis@ucla.edu for further questions about this work.

[1]The data could be images with labels, *e.g.,* 8×8 pixel blocks with labels, where each pixel is represented by 32 bits and each label is represented by an integer from $\{1 \ldots, 10\}$, in which case $\mathfrak{S} = \mathbb{F}^{64} \times \mathbb{G}$, where $\mathbb{F} = \{1, \ldots, 256\}$ and $\mathbb{G} = \{1, \ldots, 10\}$. Another example is the text represented by words, in which case $\mathfrak{S} = \mathcal{W}^*$, where $\mathcal{W}$ is the language alphabet and $\mathfrak{S}$ are strings of letters from the alphabet.

which are randomly shuffled with respect to the clients, providing additional privacy. Another mechanism to amplify privacy is through randomized sampling [9], [20], [21]. This naturally arises in the considered SGD framework, since clients do mini-batch sampling of local data and also there is sampling of clients themselves in each iteration, as in the federated learning framework [1]–[3].

In this paper, we enable privacy amplification for the FL problem using both forms of amplification: shuffling and sampling (data and clients). Note that privacy amplification by subsampling (both data and clients) happens automatically[2], and we quantify that in this paper, while the secure shuffling (anonymization) is performed explicitly which adds an additional layer of privacy that allows transferring the local privacy guarantees to central privacy guarantees.

Another important aspect is that of requiring communication efficiency instantiated through compression of the gradients computed by each active client. There has been significant recent progress in this topic (see [22]–[30] and references therein). However, there has been less work in combining privacy and compression in the optimization/learning framework of (1), with the notable exception of [31], which we will elaborate on soon. One question that arises is whether one pays a price to do compression in terms of the privacy-performance trade-off; a question we address in this paper.

In this paper we (partially) solve the main problem of privately learning a model with compressed communication, with good learning performance while giving strong guarantees on privacy. We believe that this is the first result that analyses the optimization performance with schemes devised using compressed gradient exchange, mini-batch SGD while giving data privacy guarantees for clients using a shuffled framework. Our main contributions are as follows:

• We analyze the convergence-privacy trade-offs of the proposed CLDP-SGD algorithm for Lipschitz convex function under several ℓ_p geometries (See [32, Chapter 4] for the relevance of ℓ_p geometries in optimization)[3]. We prove that one can get communication efficiency "for free" by demonstrating schemes that use $O(\log d)$ bits per client for several cases) to obtain the same privacy-performance operating point achieved by full precision gradient exchange.[4] We do this using the shuffled privacy model and amplification by sampling (client data through mini-batch SGD and clients themselves in federated sampling).

• One ingredient of our main result is showing that we can compose amplification by sampling (client data through mini-batch SGD and clients themselves in federated sampling) along with amplification by shuffling. Note that sampling of clients and data points together give overall non-uniform sampling of data points, so we cannot use the existing results on privacy amplification by subsampling, necessitating our privacy proof, of Lemma 7 in Appendix B, that composes sampling and shuffling techniques.

• At each round of the iterative optimization, one needs to privately aggregate the gradients in a communication efficient manner. To do this, we develop new private, compressed mean estimation techniques in a minimax estimation framework, that are (order optimal) under several ℓ_p geometries for the vectors. We develop both lower bounds and matching schemes for this problem. These results may also be of independent interest (see Section V).

We will put our contributions in context to the existing literature next.

Related Work

Among several main challenges in the recently developed FL framework (see [1] and references therein), we focus in this paper on the combination of privacy and communication efficiency, and examining its impact on model learning. We briefly review some of the main developments in related papers on these topics below.

1) Communication-Privacy Trade-offs: Distributed mean estimation and its use in training learning models has been studied extensively in the literature (see [22], [33]–[35] and references therein). In [33], the authors have proposed a communication efficient scheme for estimating the mean of set a of vectors distributed over multiple clients. In [36], Acharya et. al. studied the discrete distribution estimation under LDP. They proposed a randomized mechanism based on Hadamard coding which is optimal for all privacy regime and requires $\mathcal{O}(\log(d))$ bits per client, where d denotes the support size of the discrete distribution. In [37], the authors consider both private and public coin mechanisms, and show that the Hadamard mechanism is near optimal in terms of communication for both distribution and frequency estimation. Recently, [38] proposed a communication efficient scheme for mean estimation under local differential privacy constraints. This work is is done concurrently and independently of our work. Furthermore, it focuses on mean estimation for bounded ℓ_2-norm vectors, in contrast to our optimization approach, privacy amplification through sampling and shuffling. Also, this work considers the existence of public randomness, while we do not need public randomness.

LDP mechanisms suffer from the utility degradation that motivates other work to find alternative techniques to improve the utility under LDP. One of new developments in privacy is the use of anonymization to amplify the privacy by using secure shuffler. In [17]–[19], the authors studied the mean estimation problem under LDP with secure shuffler, where they show that the shuffling provides better utility than the LDP framework without shuffling.

2) Private Optimization: In [39], Chaudhuri et al. studied *centralized* privacy-preserving machine learning algorithms for convex optimization problem. The authors proposed a new idea of perturbing the objective function to preserve privacy of the

[2]In this paper, we use an abstraction for the federated learning model, where clients are sampled randomly. In practice, there are many more complicated considerations for sampling, including availability, energy usage, time-of-day etc., which we do not model in this work. Also in the terminology of [1], we focus on cross devices, *i.e.,* where we have individual clients and not siloed scenarios where institutions are collaborating.

[3]See also a simple example illustrated in Appendix A-D. Such ℓ_p constraints on the gradient also arise practically when one does gradient clipping according to an ℓ_p geometry.

[4]Our work focuses on symmetric, private-randomness mechanisms. We do not assume the existence of public randomness in this work as we use the shuffling model.

training dataset. In [40], Bassily et al. derived lower bounds on the empirical risk minimization under *central* differential privacy constraints. Furthermore, they proposed a differential privacy SGD algorithm that matches the lower bound for convex functions. In [41], the authors have generalized the private SGD algorithm proposed in [40] for non-convex optimization framework. In addition, the authors have proposed a new analysis technique, called moment accounting, to improve on the strong composition theorems to compute the central differential privacy guarantee for iterative algorithms. However, the works mentioned, [39]–[41], assume that there exists a trusted server that collects the clients' data. This motivates other works to design a distributed SGD algorithms, where each client perturbs her own data without needing a trusted server. For this, the natural privacy framework is *local* differential privacy or LDP (*e.g.,* see [5]–[7], [42]). However, it is well understood that LDP does not give good performance guarantees as it requires significant local randomization to give privacy guarantees [7], [9], [10]. In this paper, we use the strong composition theorem [43, Theorem 3.20] to analyze the differential privacy of the proposed algorithm. The strong composition is a generic composition method and does not take into account the specific noise distribution under consideration to bound the privacy loss. In [41], the authors have proposed a different technique for keeping track of the privacy loss by computing the Renyi differential privacy of the iterative algorithm. This method provides tighter bounds for the case when we add Gaussian noise to gradients for achieving differential privacy [41], [44]. Therefore, analyzing the Renyi differential privacy of the shuffled model can improve the privacy parameters of the iterative algorithms. However, computing (or even bounding non-trivially) the Renyi divergence in the shuffled model appears to be challenging – the different properties of Renyi divergence do not seem to help in bounding it, and if one tries to do a direct calculation, the problem is combinatorial in nature and blows up even with moderate values of the parameters involved. This is an interesting open question of future interest; see also a discussion on this open question in recent parallel independent (unpublished) work [45]. The two most related papers to our work are [31], [46] which we describe below.

In [46], the authors have proposed a distributed local-differential-privacy gradient descent algorithm, where each client has one sample. In their proposed algorithm, each client perturbs the gradient of her sample using an LDP mechanism. To improve upon the LDP performance guarantees, they use the newly proposed anonymization/shuffling framework [18]. Therefore in their work, gradients of all clients are passed through a secure shuffler that eliminates the identities of the clients to amplify the central privacy guarantee. However, their proposed algorithm is not communication efficient, where each client has to send the full-precision gradient without compression. Our work is different from [46], as we propose a communication efficient mechanism for each client that requires $O(\log d)$ bits per client, which can be significant for large d. Furthermore, our algorithm consider multiple data samples at client, which is accessed through a mini-batch random sampling at each iteration of the optimization. This requires a careful combination of compression and privacy analysis in order to preserve the variance reduction of mini-batch as well as privacy.[5] In addition we obtain a gain in privacy by using the fact that (anonymized) clients are sampled (*i.e.,* not all clients are selected at each iteration) as motivated by the federated learning framework.

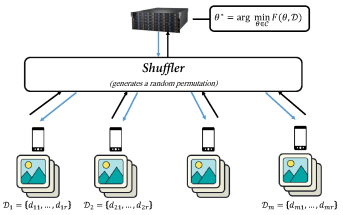

Fig. 1: We have m clients, each having a dataset $\mathcal{D}_i$ of r samples. The clients are connected to a central server to learn a global model θ under privacy and communication constraints.

Agarwal et. al. proposed in [31] a communication-efficient algorithm for learning models with central differential privacy. Let n be the number of clients per round and d be the dimensionality of the parameter space. They proposed cp-sgd, a communication efficient algorithm, where clients need to send $O(\log(1+\frac{d}{n}\epsilon^2) + \log\log\log\frac{nd}{\epsilon\delta})$ bits of communication *per coordinate i.e.,* $O\left(d\left\{\log(1+\frac{d}{n}\epsilon^2) + \log\log\log\frac{nd}{\epsilon\delta}\right\}\right)$ bits per round to achieve the same local differential privacy guarantees of ϵ_0 as the Gaussian mechanism. Their algorithm is based on a Binomial noise addition mechanism and secure aggregation. In contrast, we propose a generic framework to convert any LDP algorithm to a central differential privacy guarantee and further use recent results on amplification by shuffling, that also achieves better compression in terms of number of bits per client.

Paper organization. The paper is organized as follows. In Section II, we set up the notation while giving preliminary background results on composition of differentially-private mechanisms, and privacy amplification through subsampling and shuffling. In Section III, we formally define the problem, describe our algorithm, and give the overview of our approach and the challenges faced. We provide the main results of the paper in Section IV and also give some interpretations. In Section V, we analyze private vector minimax mean estimation for various geometrical constraints, applicable to gradient aggregation for optimization, providing schemes and impossibility results. In Section VI, we examine the communication-privacy and optimization-performance trade-offs of our schemes, putting together the results from Section V to give the proof of the main theorem 1. We conclude with a brief discussion in Section VII.

II. Preliminaries

In this section, we state some preliminary definitions that we use throughout the paper; we give a more detailed exposition of the background in Appendix A of the supplementary material.

Since we are interested in communication constrained privacy of the client, we define a two parameter LDP with privacy and

[5]The naive method of quantizing the aggregated mini-batch gradient will fail to preserve the required variance reduction.

communication budget; generalizing the standard LDP privacy definition (see Definition 3 in Appendix A-A of supplementary material).

Definition 1 (Local Differential Privacy with Communication Budget - CLDP)**.** For $\epsilon_0 \geq 0$ and $b \in \mathbb{N}^+$, a randomized mechanism $\mathcal{R} : \mathcal{X} \to \mathcal{Y}$ is said to be (ϵ_0, b)-communication-limited-local differentially private (in short, (ϵ_0, b)-CLDP), if $\mathcal{R}(\boldsymbol{x})$ can be represented using b bits $\forall \boldsymbol{x}$ and for every pair $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$, we have

$$\Pr[\mathcal{R}(\boldsymbol{x}) = \boldsymbol{y}] \leq \exp(\epsilon_0) \Pr[\mathcal{R}(\boldsymbol{x}') = \boldsymbol{y}], \ \forall \boldsymbol{y} \in \mathcal{Y}. \quad (2)$$

Here, ϵ_0 captures the privacy level, lower the ϵ_0, higher the privacy. When we are not concerned about the communication budget, we succinctly denote the corresponding (ϵ_0, ∞)-CLDP, by its correspondence to the classical LDP as ϵ_0-LDP [9].

We define $\mathcal{D} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ and $\mathcal{D}' = \{\boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_n\}$ as neighboring if they differ in one data point.

Definition 2 (Central Differential Privacy - DP [43])**.** For $\epsilon, \delta \geq 0$, a randomized mechanism $\mathcal{M} : \mathcal{X}^n \to \mathcal{Y}$ is said to be (ϵ, δ)-differentially private (in short, (ϵ, δ)-DP), if for all neighboring datasets $\mathcal{D}, \mathcal{D}' \in \mathcal{X}^n$ and every subset $\mathcal{E} \subseteq \mathcal{Y}$, we have

$$\Pr\left[\mathcal{M}\left(\mathcal{D}\right) \in \mathcal{E}\right] \leq \exp(\epsilon) \Pr\left[\mathcal{M}\left(\mathcal{D}'\right) \in \mathcal{E}\right] + \delta. \quad (3)$$

We will propose an iterative algorithm to solve the optimization problem (1) under privacy and communication constraints. Hence, we need the strong composition theorem [47] (we desribe it in detail in Appendix A-B for completeness) to compute the final privacy guarantees of the proposed algorithm. Furthermore, in order to overcome the poor performance of LDP, we need to use privacy amplification provided by subsampling (data and clients) as well as through the shuffled model; both of which we briefly review next.

Consider a set of m clients, where client $i \in [m]$ has a data $\boldsymbol{x}_i \in \mathcal{X}$. Let $\mathcal{R} : \mathcal{X} \to \mathcal{Y}$ be an ϵ_0-LDP mechanism. The i-th client applies $\mathcal{R}$ on her data $\boldsymbol{x}_i$ to get $\boldsymbol{y}_i = \mathcal{R}(\boldsymbol{x}_i)$. In the shuffled model of privacy, the shuffler $\mathcal{H}_m : \mathcal{Y}^m \to \mathcal{Y}^m$ has m messages $(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m)$ as input and outputs a uniformly random permutation of it. Lemma 12 in Appendix A-C2 in supplementary material states that the shuffling amplifies the privacy of an LDP mechanism by a factor of $\mathcal{O}\left(\frac{1}{\sqrt{m}}\right)$. We review the known results for privacy Amplification by *uniform* subsampling in Appendix A-C1 of the supplementary material.

III. Problem Formulation and Solution Overview

In this section, first we present the problem formulation and describe our algorithm for solving the empirical risk minimization problem under the constraints of privacy, communication, and dynamic client population. Then we give an overview of our approach to analyze this algorithm and briefly describe the challenges faced. One of our main ingredients, in the proposed compressed and private SGD algorithm, is a method of private mean estimation using compressed updates, formulated in Section III-D. We use this formulation to study the problem in the minimax framework and derive upper and lower bounds in a variety of settings. A summary of the notation used throughout the paper is given in Table I.

Symbol	Description
m	Total number of clients in the system
r	Total number of samples per client
k	$(\leq m)$ Number of clients chosen per iteration
s	$(\leq r)$ Number of samples chosen per client per iteration
n	$(= mr)$ Total number of samples in the dataset
q	$(= \frac{ks}{mr})$ Probability of a sample to be chosen at an iteration
$\mathcal{D}_i$	Local dataset of client i for $i \in [m]$
$\mathcal{D}$	$(\bigcup_{i=1}^m \mathcal{D}_i)$ The entire dataset
ϵ_0	Local differential privacy parameter
ϵ	Central differential privacy parameter
θ	$(\in \mathbb{R}^d)$ Model parameter vector
$\mathcal{C}$	$(\subset \mathbb{R}^d)$ convex set of interest
D	$(= \|\mathcal{C}\|_2)$ Diameter of the set $\mathcal{C}$
L	Lipschitz continuous parameter
$\mathcal{B}_p^d(a)$	ℓ_p norm ball of radius a

TABLE I: Notation used throughout the paper

A. Problem Formulation

We have a set of m clients, where each client has a local dataset $\mathcal{D}_i = \{d_{i1}, \ldots, d_{ir}\}$ comprising r data points drawn from a universe $\mathfrak{S}$. Let $\mathcal{D} = \bigcup_{i=1}^m \mathcal{D}_i$ denote the entire dataset and $n = mr$ denote the total number of data points in the system (see Figure 1). The clients are connected to an untrusted server in order to solve the following empirical risk minimization (ERM) problem

$$\min_{\theta \in \mathcal{C}} \left(F(\theta, \mathcal{D}) := \frac{1}{m} \sum_{i=1}^{m} F_i(\theta, \mathcal{D}_i) \right). \quad (4)$$

Here, $\mathcal{C} \subset \mathbb{R}^d$ is a closed convex set and $F_i(\theta, \mathcal{D}_i) = \frac{1}{r} \sum_{j=1}^{r} f\left(\theta, d_{ij}\right)$ is a local loss function dependent on the local dataset $\mathcal{D}_i$ at client i evaluated at the model parameters $\theta \in \mathcal{C}$.

As described in Section I, solving the ERM problem (4) in the FL framework introduces several unique challenges, such as the locally residing data $\{\mathcal{D}_i\}$ at all clients need to kept private, the low-bandwidth links between clients and the server necessitates compressed communication exchange between them, and only a small fraction of clients are sampled in each round of communication. Our goal is to solve (4) while preserving privacy on the training dataset $\mathcal{D}$ and minimizing the total number of bits for communication between clients and the server, while dealing with a dynamic client population in each iteration.

B. Our Algorithm: CLDP-SGD

In order to solve (4) in the presence of the above challenges in the FL setting, we propose CLDP-SGD, a differentially-private SGD algorithm that works with compressed updates and dynamic client population. The procedure is described in Algorithm 1; also see Figure 2 for a pictorial description of our algorithm.

In each step of CLDP-SGD, we choose uniformly at random a set $\mathcal{U}_t$ of $k \leq m$ clients out of m clients. Each client $i \in \mathcal{U}_t$ computes the gradient $\nabla_{\theta_t} f\left(\theta_t; d_{ij}\right)$ for a random subset $\mathcal{S}_{it}$ of $s \leq r$ samples. The i'th client clips the ℓ_p-norm of the gradient $\nabla_{\theta_t} f\left(\theta_t; d_{ij}\right)$ for each $j \in \mathcal{S}_{it}$ and applies the LDP-compression mechanism $\mathcal{R}_p$, where $\mathcal{R}_p : \mathcal{B}_p^d \to \{0,1\}^b$ is an (ϵ_0, b)-CLDP mechanism when inputs come from an ℓ_p-norm

Algorithm 1 $\mathcal{A}_{\text{cldp}}$: CLDP-SGD

1: **Inputs:** Datasets $\mathcal{D} = \bigcup_{i\in[m]} \mathcal{D}_i$, where $\mathcal{D}_i = \{d_{i1}, \ldots, d_{ir}\}$ for $i \in [m]$, loss function $F(\theta) = \frac{1}{mr}\sum_{i=1}^{m}\sum_{j=1}^{r} f(\theta; d_{ij})$, LDP privacy parameter ϵ_0, gradient norm bound C, and learning rate schedule $\{\eta_t\}$.
2: **Initialize:** $\theta_0 \in \mathcal{C}$
3: **for** $t \in [T]$ **do**
4: A random set $\mathcal{U}_t$ of k clients is chosen.
5: **for** clients $i \in \mathcal{U}_t$ **do**
6: Client i chooses uniformly at random a set $\mathcal{S}_{it}$ of s samples.
7: **for** Samples $j \in \mathcal{S}_{it}$ **do**
8: $\mathbf{g}_t(d_{ij}) \leftarrow \nabla_{\theta_t} f(\theta_t; d_{ij})$
9: $\tilde{\mathbf{g}}_t(d_{ij}) \leftarrow \mathbf{g}_t(d_{ij}) / \max\left\{1, \frac{\|\mathbf{g}_t(d_{ij})\|_p}{C}\right\}$[6]
10: $\mathbf{q}_t(d_{ij}) \leftarrow \mathcal{R}_p(\tilde{\mathbf{g}}_t(d_{ij}))$
11: Client i sends $\{\mathbf{q}_t(d_{ij}) : j \in \mathcal{S}_{it}\}$ to the shuffler.
12: The shuffler randomly shuffles the elements in $\{\mathbf{q}_t(d_{ij}) : i \in \mathcal{U}_t, j \in \mathcal{S}_{it}\}$ and sends them to the server.
13: $\overline{\mathbf{g}}_t \leftarrow \frac{1}{ks}\sum_{i\in\mathcal{U}_t}\sum_{j\in\mathcal{S}_{it}} \boldsymbol{q}_t(d_{ij})$
14: $\theta_{t+1} \leftarrow \prod_{\mathcal{C}}(\theta_t - \eta_t \overline{\mathbf{g}}_t)$, where $\prod_{\mathcal{C}}$ denotes the projection operator onto the set $\mathcal{C}$.
15: **Output:** The model θ_T and the privacy parameters ϵ, δ

ball; we describe (ϵ_0, b)-CLDP mechanisms $\mathcal{R}_p$ for several values of $p \in [1, \infty]$ in Section V. After that, each client i sends the set of s LDP-compressed gradients $\{\mathcal{R}_p(\mathbf{g}_t(d_{ij}))\}_{j\in\mathcal{S}_{it}}$ in a communication-efficient manner to the secure shuffler. The shuffler randomly shuffles (i.e., outputs a random permutation of) the received ks gradients and sends them to the server. Finally, the server takes the average of the received gradients and updates the parameter vector.

C. Overview of Our Approach for Analyzing CLDP-SGD

CLDP-SGD has the following components, which need to be analyzed together: (i) sampling of clients, necessitated by FL; (ii) sampling of data at each client for mini-batch SGD; (iii) compressing the gradients at each client for communication efficiency; (iv) privatizing the gradients at each client to prevent information leakage – the (compressed) gradients received by the server my leak information about the datasets; and (v) shuffling. The two main technical ingredients needed for the analysis are (a) Privacy analysis of coupled sampling and shuffling (b) Commununication efficient private mean estimatioon.

Privacy of coupled sampling and shuffling: As explained in Section I, client and data sampling as well as shuffling contribute to privacy amplification. However, there are several challenges in analyzing the overall privacy amplification: Firstly, both types of sampling together induce non-uniform sampling

[6]Note that gradient clipping may not preserve unbiasedness of the stochastic gradients. However for the case when the loss function f is L-Lipschitz, this is not necessary for the following reason. If the loss function f is L-Lipschitz (with respect to the model parameters) in the dual norm ℓ_g, where $\frac{1}{p}+\frac{1}{g}=1, p, g \geq 1$, then the norm of the gradients (with respect to some ℓ_p-norm, for $p \geq 1$) is bounded, and hence we do not need to clip it.

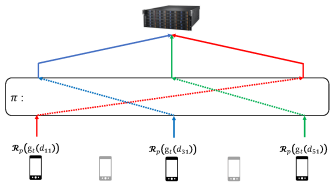

Fig. 2: An example of 5 clients, where each client has a single data point. At the current iteration, 3 clients are chosen at random to send the compressed and private gradients $\{\mathcal{R}_p(g_t(d_{i1}))\}$ to the secure shuffler that permutes the private gradients before sending them to the server.

of data, so we cannot use the existing privacy amplification from subsampling results (see Section A-C1) directly to analyze the privacy gain[7] in CLDP-SGD just by subsampling; and secondly, the privacy amplification by shuffling has not been analyzed together with that by subsampling. In this paper, we give one unifying proof that analyzes the privacy amplification by both types of subsampling (that induces non-uniform sampling of data points) as well as shuffling; see Section VI-A, Lemma 7 and its proof in Appendix B, for more details.

Communication-efficient private mean estimation: For compressing and privatizing the gradients, we design communication-efficient local differentially private mechanisms $\mathcal{R}_p$ for $p \in [0, \infty]$ to estimate the mean of a set of bounded ℓ_p-norm gradients. These mechanisms $\mathcal{R}_p$'s are in fact more generally applicable for private mean estimation of a set of vectors, each having a bounded ℓ_p-norm and coming from a different client in a communication efficient manner. We study the mean estimation problem in the minimax framework and derive matching lower and upper bounds on the minimax risk for several ℓ_p geometries; see Section V. We can further save on communication by having each client communicating the *histogram* of its s compressed gradients (instead of separately sending the s gradients) – in Algorithm 1 (line 11), we write for simplicity that each client i sends $\{\mathbf{q}_t(d_{ij}) : j \in \mathcal{S}_{it}\}$ to the shuffler. In fact, sending the histogram of these $|\mathcal{S}_{it}| = s$ elements will suffice for our purpose, thereby achieving further savings on communication; see Section VI-B for more details. This privacy mechanism is composed with the sampling and shuffling to provide the overall privacy analysis. In the next section, we formulate the compressed and private mean estimation problem as of independent interest.

D. Compressed and Private Mean Estimation via Minimax Risk

In this section, we formulate the generic mimimax estimation framework for mean estimation of a given set of n vectors that preserves privacy and is also communication-efficient. We then apply that method at the server in each SGD iteration for aggregating the gradients. We derive upper and lower bounds for various ℓ_p geometries for $p \geq 1$ including the ℓ_∞-norm.

The setup is illustrated in Figure 3. For any $p \geq 1$ and $d \in \mathbb{N}$, let $\mathcal{B}_p^d(a) = \{\boldsymbol{x} \in \mathbb{R}^d : \|\boldsymbol{x}\|_p \leq a\}$ denote the p-

[7]We use privacy gain interchangeably with privacy amplification.

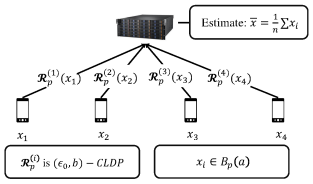

Fig. 3: We have n clients, each observing a bounded ℓ_p-norm vector $\boldsymbol{x}_i \in \mathcal{B}_p(a)$. A compressed and private mechanism $\mathcal{R}_p^{(i)}$ is applied to $\boldsymbol{x}_i$. The server wants to estimate the mean of vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ from the privatized vectors $\{\mathcal{R}_p^{(i)}(\boldsymbol{x}_i)\}$.

norm ball with radius a centered at the origin in $\mathbb{R}^d$,[8] where $\|\boldsymbol{x}\|_p = \left(\sum_{j=1}^d |\boldsymbol{x}_j|^p\right)^{1/p}$. Each client $i \in [n]$ has an input vector $\boldsymbol{x}_i \in \mathcal{B}_p^d(a)$ and the server wants to estimate the mean $\overline{\boldsymbol{x}} := \frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i$. We have two constraints: (i) each client has a communication budget of b bits to transmit the information about its input vector to the server, and (ii) each client wants to keep its input vector private from the server. Hence, client $i \in [n]$ applies a private-quantization mechanism $\mathcal{R}_p^{(i)} : \mathcal{B}_p^d(a) \to \{0,1\}^b$ on her input $\boldsymbol{x}_i$ to obtain a private output $\boldsymbol{y}_i = \mathcal{R}_p^{(i)}(\boldsymbol{x}_i)$ and sends it to the server. Upon receiving $\boldsymbol{y}^n = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n]$, the server applies a decoding function $\widehat{\boldsymbol{x}} : \left(\{0,1\}^b\right)^n \to \mathcal{B}_p^d$ to estimate the mean vector $\overline{\boldsymbol{x}} = \frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i$. Our objective is to design (ϵ_0, b)-CLDP mechanisms $\mathcal{R}_p^{(i)}$ for all $i \in [n]$ (see Definition 4) and also a (stochastic) decoding function $\widehat{\boldsymbol{x}}$ to minimize the worst-case expected error as follows

$$r_{\epsilon_0,b,n}^{p,d}(a) = \inf_{\{\mathcal{R}_p^{(i)} \in \mathcal{Q}_{(\epsilon_0,b)}\}} \inf_{\widehat{\boldsymbol{x}}} \sup_{\{\boldsymbol{x}_i\} \in \mathcal{B}_p^d(a)} \mathbb{E}\left\|\overline{\boldsymbol{x}} - \widehat{\boldsymbol{x}}(\boldsymbol{y}^n)\right\|_2^2, \tag{5}$$

where $\mathcal{Q}_{(\epsilon_0,b)}$ is the set of all (ϵ_0, b)-CLDP mechanisms, and the expectation is taken over the randomness of $\{\mathcal{R}_p^{(i)} : i \in [n]\}$ and the estimator $\widehat{\boldsymbol{x}}$.

Now we extend the formulation in (5) to a probabilistic model. Let $\mathcal{P}_p^d(a)$ denote the set of all probability density functions on $\mathcal{B}_p^d(a)$. For every distribution $\boldsymbol{q} \in \mathcal{P}_p^d(a)$, let $\boldsymbol{\mu_q}$ denote its mean. Since the support of each distribution $\boldsymbol{q} \in \mathcal{P}_p^d$ is $\mathcal{B}_p^d(a)$ and ℓ_p is a norm, we have that $\boldsymbol{\mu_q} \in \mathcal{B}_p^d(a)$. For a given unknown distribution $\boldsymbol{q} \in \mathcal{P}_p^d(a)$, client $i \in [n]$ observes $\boldsymbol{x}_i$, where $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ are i.i.d. according to $\boldsymbol{q}$, and the goal for the server is to estimate $\boldsymbol{\mu_q}$, while satisfying the same two constraints as above, i.e., only b bits of communication is allowed from any client to the server while preserving the privacy of clients' inputs. Analogous to (5), we are interested in characterizing the following quantity.

$$R_{\epsilon_0,b,n}^{p,d}(a) = \inf_{\{\mathcal{R}_p^{(i)} \in \mathcal{Q}_{(\epsilon_0,b)}\}} \inf_{\widehat{\boldsymbol{x}}} \sup_{\boldsymbol{q} \in \mathcal{P}_p^d(a)} \mathbb{E}\left\|\boldsymbol{\mu_q} - \widehat{\boldsymbol{x}}(\boldsymbol{y}^n)\right\|_2^2, \tag{6}$$

where the expectation is taken over the randomness of the output $\boldsymbol{y}^n$ and the estimator $\widehat{\boldsymbol{x}}$.

In this paper, we design private-quantization mechanisms $\{\mathcal{R}_p^{(1)}, \ldots, \mathcal{R}_p^{(n)}\}$ such that they are symmetric (i.e., $\mathcal{R}_p^{(i)}$'s are same for all $i \in [n]$) and any client uses only private source of randomness that is not accessible by any other party in the system.

[8]Assuming that the ball is centered at origin is without loss of generality; otherwise, we can translate the ball to origin and work with that.

IV. Main Technical Results

In this section, we first state our results on convergence, privacy, and communication bits of the proposed CLDP-SGD algorithm. We also discuss their implications. Then, we present the results on compressed and private mean estimation in Section IV-B.

A. Optimization

In Theorem 1, we state the privacy guarantees, the communication cost per client, and the privacy-convergence trade-offs for the CLDP-SGD Algorithm. We assume that the constraint set $\mathcal{C}$ is closed convex set with diameter D,[9]. Furthermore, we assume that the loss function $f(\theta, .)$ is convex and L-Lipschitz continuous with respect to the ℓ_g-norm which is the dual of the ℓ_p-norm[10]. Let $n = mr$ denote the total number of data points in the dataset $\mathcal{D}$. Observe that the probability that an arbitrary data point $d_{ij} \in \mathcal{D}$ is chosen at time $t \in [T]$ is given by $q = \frac{ks}{mr}$.

Theorem 1. *Let the set $\mathcal{C}$ be convex with diameter D and the function $f(\theta; .) : \mathcal{C} \to \mathbb{R}$ be convex and L-Lipschitz continuous with respect to the ℓ_g-norm, which is the dual of the ℓ_p-norm. Let $\theta^* = \arg\min_{\theta \in \mathcal{C}} F(\theta)$ denote the minimizer of the problem* (4)*. For $s = 1$ and $q = \frac{k}{mr}$, where $n = mr$, if we run Algorithm $\mathcal{A}_{cldp}$ over T iterations, then we have*

1) ***Privacy:*** *For $\epsilon_0 = \mathcal{O}(1)$, $\mathcal{A}_{cldp}$ is (ϵ, δ)-DP, where $\delta > 0$ is arbitrary, and*

$$\epsilon = \mathcal{O}\left(\epsilon_0 \sqrt{\frac{qT\log(2qT/\delta)\log(2/\delta)}{n}}\right). \tag{7}$$

2) ***Communication:*** *Our algorithm $\mathcal{A}_{cldp}$ requires $\frac{k}{m}s \times \left(\log(e) + \log\left(\frac{s+2^b-1}{s}\right)\right)$ bits of communication in expectation[11] per client per iteration, where expectation is taken with respect to the sampling of clients. Here, $b = \log(d) + 1$ if $p \in \{1, \infty\}$ and $b = d(\log(e) + 1)$ otherwise.*

3) ***Convergence:*** *If we run $\mathcal{A}_{cldp}$ with learning rate schedule $\eta_t = \frac{D}{G\sqrt{t}}$, where $G^2 = L^2 \max\{d^{1-\frac{2}{p}}, 1\}\left(1 + \frac{cd}{qn}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2\right)$, then*

$$\mathbb{E}[F(\theta_T)] - F(\theta^*) \leq \mathcal{O}\left(\frac{LD\log(T)\max\{d^{\frac{1}{2}-\frac{1}{p}}, 1\}}{\sqrt{T}} \sqrt{\frac{cd}{qn}}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\right). \tag{8}$$

[9]Diameter of a bounded set $\mathcal{C} \subseteq \mathbb{R}^d$ is defined as $\sup_{\boldsymbol{x},\boldsymbol{y} \in \mathcal{C}} \|\boldsymbol{x} - \boldsymbol{y}\|$.

[10]For any data point $d \in \mathfrak{S}$, the function $f : \mathcal{C} \to \mathbb{R}$ is L-Lipschitz continuous w.r.t. ℓ_g-norm if for every $\theta_1, \theta_2 \in \mathcal{C}$, we have $|f(\theta_1; d) - f(\theta_2; d)| \leq L\|\theta_1 - \theta_2\|_g$.

[11]A client communicates in an iteration only when that client is selected (sampled) in that iteration.

	Communication	Minimax risk
ℓ_1-norm	$\log(d)+1$	$\theta\left(\frac{d}{n\epsilon_0^2}\right)$
ℓ_2-norm	$d(\log(e)+1)$	$\theta\left(\frac{d}{n\epsilon_0^2}\right)$
ℓ_∞-norm	$\log(d)+1$	$\theta\left(\frac{d^2}{n\epsilon_0^2}\right)$

TABLE II: Summary of private mean estimation results

where $c=4$ if $p\in\{1,\infty\}$ and $c=14$ otherwise.

We prove Theorem 1 in Section VI. Observe that the privacy results in Theorem 1 is stated for $\epsilon_0=\mathcal{O}(1)$, where the results for general ϵ_0 is presented in Section VI-A.

Remark 1 (Arbitrary SGD mini-batch size s)**.** The communication and convergence results in Theorem 1 are general and hold for any $s\in[r]$; however, the privacy result is stated for $s=1$, i.e., each client only samples a single data point in each SGD iteration. Results for any mini-batch size $s\in[r]$ are provided in Appendix B.

Remark 2 (Recovering the Result [46, ESA])**.** In [46], each client has only one data point and all clients participate in each iteration, and gradients have bounded ℓ_2-norm. If we put $p=2$, $T=n/\log^2(n)$, and $q=1$ in (8), we get the following privacy-accuracy trade-off, which is the same as that in [46, Theorem VI.1].

$$\mathbb{E}\left[F\left(\theta_T\right)\right]-F\left(\theta^*\right)\leq\mathcal{O}\left(\frac{LD\log^2(n)\sqrt{d}}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\right)$$
$$\epsilon=\mathcal{O}\left(\epsilon_0\sqrt{\frac{T\log\left(T/\delta\right)\log\left(1/\delta\right)}{n}}\right) \tag{9}$$

We want to emphasize that the above privacy-accuracy trade-off in [46] is achieved by full-precision gradient exchange, whereas, we can achieve the same trade-off with compressed gradients. Moreover, our results are in more general setting, where clients' local datasets have multiple data-points (no bound on that) and we do two types of sampling, one of clients and other of data for SGD.

Remark 3 (Optimality of CLDP-SGD for ℓ_2-norm case)**.** Suppose that our target is to achieve $\epsilon=\mathcal{O}(1)$ and $\delta\ll 1$. Substituting $\epsilon_0=\epsilon\sqrt{\frac{n}{qT\log(2qT/\delta)\log(2/\delta)}}$, $T=n/q$, and $p=2$ in (8), we get

$$\mathbb{E}\left[F\left(\theta_T\right)\right]-F\left(\theta^*\right)=\mathcal{O}\left(\frac{LD\log^{\frac{3}{2}}\left(\frac{n}{\delta}\right)\sqrt{d\log\left(\frac{1}{\delta}\right)}}{n\epsilon}\right). \tag{10}$$

This matches the optimal excess risk of central differential privacy presented in [40]. Note that the results in [40] are for centralized SGD with full precision gradients, whereas, our results are for federated learning (which is a distributed setup) with compressed gradient exchange.

B. Compressed and Private Mean Estimation

In this subsection, we state our lower and upper bound results on minimax risks both in the worst case model (see (5)) and the probabilistic model (see (6)). For the lower bounds, we state our results when there is no communication constraints, and for clarity, we denote the corresponding minimax risks by $r^{p,d}_{\epsilon_0,\infty,n}(a)$ and $R^{p,d}_{\epsilon_0,\infty,n}(a)$.

Theorem 2. *For any $d,n\geq 1$, $a,\epsilon_0>0$, and $p\in[1,\infty]$, we have the minimax risk in* (6) *satisfies*

$$R^{p,d}_{\epsilon_0,\infty,n}(a) \geq \begin{cases}\Omega\left(a^2\min\left\{1,\frac{d}{n\epsilon_0^2}\right\}\right) & \text{if } 1\leq p\leq 2,\\ \Omega\left(a^2d^{1-\frac{2}{p}}\min\left\{1,\frac{d}{n\min\{\epsilon_0,\epsilon_0^2\}}\right\}\right) & \text{if } p\geq 2.\end{cases}$$

Theorem 3. *For any $d,n\geq 1$, $a,\epsilon_0>0$, and $p\in[1,\infty]$, the minimax risk in* (5) *satisfies*

$$r^{p,d}_{\epsilon_0,\infty,n}(a) \geq \begin{cases}\Omega\left(a^2\min\left\{1,\frac{d}{n\epsilon_0^2}\right\}\right) & \text{if } 1\leq p\leq 2,\\ \Omega\left(a^2d^{1-\frac{2}{p}}\min\left\{1,\frac{d}{n\min\{\epsilon_0,\epsilon_0^2\}}\right\}\right) & \text{if } p\geq 2.\end{cases}$$

Theorem 4. *For any private-randomness, symmetric mechanism $\mathcal{R}$ with communication budget $b<\log(d)$ bits per client, and any decoding function $g:\{0,1\}^b\to\mathbb{R}^d$, when $\widehat{\boldsymbol{x}}=\frac{1}{n}\sum_{i=1}^n g\left(\mathcal{R}\left(\boldsymbol{x}_i\right)\right)$, we have*

$$r^{p,d}_{\epsilon_0,b,n}(a)>a^2\max\left\{1,d^{1-\frac{2}{p}}\right\}. \tag{11}$$

Remark 4. Note that Theorem 4 works only when the estimator $\widehat{\boldsymbol{x}}$ applies the decoding function g on individual responses and then takes the average. We leave its extension for arbitrary decoders as a future work.

We prove Theorem 4 in Section V-C.

Though our lower bound results are for arbitrary estimators $\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)$, for the minimax risk estimation problems (5) and (6), we can show that the optimal estimator $\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)$ is a *deterministic* function of $\boldsymbol{y}^n$. In other words, the randomized decoder does not help in reducing the minimax risk. See Lemma 13 in Appendix C.

Theorem 5 (ℓ_1-norm)**.** *For any $d,n\geq 1$, $a,\epsilon_0>0$, we have*

$$r^{1,d}_{\epsilon_0,b,n}\left(a\right)\leq\frac{a^2d}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2$$
$$R^{1,d}_{\epsilon_0,b,n}\left(a\right)\leq\frac{4a^2d}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2,$$

for $b=\log(d)+1$.

Theorem 6 (ℓ_2-norm)**.** *For any $d,n\geq 1$, $a,\epsilon_0>0$, we have*

$$r^{2,d}_{\epsilon_0,b,n}\left(a\right)\leq\frac{6a^2d}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2$$
$$R^{2,d}_{\epsilon_0,b,n}\left(a\right)\leq\frac{14a^2d}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2,$$

for $b=d\log(e)+1$.

Theorem 7 (ℓ_∞-norm)**.** *For any $d,n\geq 1$, $a,\epsilon_0>0$, we have*

$$r^{\infty,d}_{\epsilon_0,b,n}(a)\leq\frac{a^2d^2}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2$$

$$R^{\infty,d}_{\epsilon_0,b,n}(a) \le \frac{4a^2d^2}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2,$$

for $b = \log(d)+1$.

Note that when $\epsilon_0 = \mathcal{O}(1)$, then the upper and lower bounds on minimax risks match for $p \in [1,2]$. Furthermore, when $\epsilon_0 \le 1$, then they match for all $p \in [1,\infty]$.

Now we give a general achievability result for any ℓ_p-norm ball $\mathcal{B}^d_p(a)$ for any $p \in [1,\infty)$. For this, we use standard inequalities between different norms, and probabilistically use the mechanisms for ℓ_1-norm or ℓ_2-norm with expanded radius of the corresponding ball. We assume that every work can pick any mechanisms with the same probability $\bar{p} \in [0,1]$. This gives the following result, which we prove in Section V-G.

Corollary 1 (General ℓ_p-norm, $p \in [1,\infty)$)**.** Suppose clients pick the mechanism for ℓ_1-norm with probability $\bar{p} \in [0,1]$. Then, for any $d, n \ge 1$, $a, \epsilon_0 > 0$, we have:

$$\begin{aligned} r^{p,d}_{\epsilon_0,b,n}(a) \;&\le\; \bar{p}\, d^{2-\frac{2}{p}} \cdot r^{1,d}_{\epsilon_0,b,n}(a) \\ &\quad + (1-\bar{p}) \max\left\{d^{1-\frac{2}{p}}, 1\right\} \cdot r^{2,d}_{\epsilon_0,b,n}(a), \qquad (12) \\ R^{p,d}_{\epsilon_0,b,n}(a) \;&\le\; \bar{p}\, d^{2-\frac{2}{p}} \cdot R^{1,d}_{\epsilon_0,b,n}(a) \\ &\quad + (1-\bar{p}) \max\left\{d^{1-\frac{2}{p}}, 1\right\} \cdot R^{2,d}_{\epsilon_0,b,n}(a). \qquad (13) \end{aligned}$$

for $b = \bar{p}\log(d) + (1-\bar{p})d\log(e) + 1$. Note that this communication is in expectation, which is taken over the sampling of selecting ℓ_1 or ℓ_2 mechanisms.

We can recover Theorem 5 by setting $p = 1$ and $\bar{p} = 1$ and Theorem 6 by setting $p = 2$ and $\bar{p} = 0$.

V. Compressed and Private Mean Estimation

In this section, we study the private mean-estimation problem in the minimax framework given in Section III-D. Note that in this section we focus on giving (ϵ_0, b)-CLDP) privacy-communication guarantees for the mean-estimation problem and give the performance of schemes in terms of the associated minimax risk. This framework is applied at each round of the optimization problem, and is then converted to the eventual central DP privacy guarantees using the shuffling framework in Section VI, yielding the main result Theorem 1 stated in Section IV.

This section is divided into six subsections. We prove the lower bound results (Theorems 2, 3) in the first two subsections and the achievable results (Theorems 5, 6, 7, and Corollary 1) in the last four subsections, respectively.

We prove lower bounds for private mechanisms with no communication constraints, and for clarity, we denote such mechanisms by (ϵ_0, ∞)-CLDP mechanisms. Our achievable schemes use finite amount of randomness.

For lower bounds, for simplicity, we assume that the inputs come from an ℓ_p-norm ball of unit radius – the bounds will be scaled by the factor of a^2 if inputs come from an ℓ_p-norm ball of radius a. For convenience, we denote $\mathcal{B}^d_p(1), \mathcal{P}^d_p(1), r^{p,d}_{\epsilon_0,b,n}(1)$, and $R^{p,d}_{\epsilon_0,b,n}(1)$ by $\mathcal{B}^d_p, \mathcal{P}^d_p, r^{p,d}_{\epsilon_0,b,n}$, and $R^{p,d}_{\epsilon_0,b,n}$, respectively.

A. Lower Bound on $R^{p,d}_{\epsilon_0,\infty,n}$: Proof of Theorem 2

Theorem 2 states separate lower bounds on $R^{p,d}_{\epsilon_0,\infty,n}$ depending on whether $p \ge 2$ or $p \le 2$ (at $p = 2$, both bounds coincide), and we prove them below in Section V-A1 and Section V-A2, respectively.

1) Lower bound for $p \in [2,\infty]$: The main idea of the lower bound is to transform the problem to the private mean estimation when the inputs are sampled from Bernoulli distributions. Recall that $\mathcal{P}^d_p$ denote the set of all distributions on the p-norm ball $\mathcal{B}^d_p$. Let $\mathcal{P}^{\text{Bern}}_{p,d}$ denote the set of Bernoulli distributions on $\left\{0, \frac{1}{d^{1/p}}\right\}^d$, i.e., any element of $\mathcal{P}^{\text{Bern}}_{p,d}$ is a product of d independent Bernoulli distributions, one for each coordinate. We first prove a lower bound on $R^{p,d}_{\epsilon_0,\infty,n}$ when the input distribution belongs to $\mathcal{P}^{\text{Bern}}_{p,d}$.

Lemma 1. *For any $p \in [2,\infty]$, we have*

$$\begin{aligned} &\inf_{\{\mathcal{R}^{(i)}_p\} \in \mathcal{Q}_{(\epsilon_0,\infty)}} \inf_{\widehat{\boldsymbol{x}}} \sup_{\boldsymbol{q} \in \mathcal{P}^{Bern}_{p,d}} \mathbb{E}\left\|\boldsymbol{\mu}_{\boldsymbol{q}} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 \\ &\qquad \ge \Omega\left(d^{1-\frac{2}{p}} \min\left\{1, \frac{d}{n\min\{\epsilon_0, \epsilon_0^2\}}\right\}\right). \end{aligned} \qquad (14)$$

Proof. The proof is straightforward from the proof of Duchi and Rogers [48, Corollary 3]. In their setting, $\mathcal{P}^{\text{Bern}}_{p,d}$ is supported on $\{0,1\}^d$, and they proved a lower bound of $\Omega\left(\min\left\{1, \frac{d}{n\min\{\epsilon_0,\epsilon_0^2\}}\right\}\right)$. In our setting, since $\mathcal{P}^{\text{Bern}}_{p,d}$ is supported on $\left\{0, \frac{1}{d^{1/p}}\right\}^d$, we can simply scale the elements in the support of $\mathcal{P}^{\text{Bern}}_{p,d}$ by a factor of $1/d^{1/p}$, which will also scale the mean $\boldsymbol{\mu}_{\boldsymbol{q}}$ by the same factor. Note that the best estimator $\widehat{\boldsymbol{x}}$ will be equal to the scaled version of the best estimator from [48, Corollary 3] with the same value $1/d^{1/p}$. This proves Lemma 1. ∎

In order to use Lemma 1, first observe that for every $\boldsymbol{x} \in \mathcal{P}^{\text{Bern}}_{p,d}$, we have $\|\boldsymbol{x}\|_p \le 1$, which implies that $\boldsymbol{x} \in \mathcal{P}^d_p$. Thus we have $\mathcal{P}^{\text{Bern}}_{p,d} \subset \mathcal{P}^d_p$. Now our bound on $R^{p,d}_{\epsilon_0,\infty,n}$ trivially follows from the following inequalities:

$$\begin{aligned} R^{p,d}_{\epsilon_0,\infty,n} &= \inf_{\{\mathcal{R}^{(i)}_p\} \in \mathcal{Q}_{(\epsilon_0,\infty)}} \inf_{\widehat{\boldsymbol{x}}} \sup_{\boldsymbol{q} \in \mathcal{P}^d_p} \mathbb{E}\left\|\boldsymbol{\mu}_{\boldsymbol{q}} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 \\ &\ge \inf_{\{\mathcal{R}^{(i)}_p\} \in \mathcal{Q}_{(\epsilon_0,\infty)}} \inf_{\widehat{\boldsymbol{x}}} \sup_{\boldsymbol{q} \in \mathcal{P}^{\text{Bern}}_{p,d}} \mathbb{E}\left\|\boldsymbol{\mu}_{\boldsymbol{q}} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 \\ &\ge \Omega\left(d^{1-\frac{2}{p}} \min\left\{1, \frac{d}{n\min\{\epsilon_0,\epsilon_0^2\}}\right\}\right), \end{aligned} \qquad (15)$$

where the last inequality follows from (14).

2) Lower bound for $p \in [1,2]$: Fix an arbitrary $p \in [1,2]$. Note that $\|\boldsymbol{x}\|_p \le \|\boldsymbol{x}\|_1$, which implies that $\mathcal{B}^d_1 \subset \mathcal{B}^d_p$, and therefore, we have $\mathcal{P}^d_1 \subset \mathcal{P}^d_p$. These imply that the lower bound derived for $\mathcal{P}^d_1$ also holds for $\mathcal{P}^d_p$, i.e., $R^{p,d}_{\epsilon_0,\infty,n} \ge R^{1,d}_{\epsilon_0,\infty,n}$ holds for any $p \in [1,2]$. So, in the following, we only lower-bound $R^{1,d}_{\epsilon_0,\infty,n}$. The main idea of the lower bound is to transform the problem to the private discrete distribution estimation when the inputs are sampled from a discrete distribution taken from a simplex in d dimensions. Recall that $\mathcal{P}^d_1$ denotes all probability density functions q over the 1-norm ball $\mathcal{B}^d_1$. Note that q may be a continuous distribution supported over all of $\mathcal{B}^d_1$. Let

$\widehat{\mathcal{P}}_1^d$ denote a set of all discrete distributions $\boldsymbol{q}$ supported over the d standard basis vectors $\boldsymbol{e}_1, \ldots, \boldsymbol{e}_d$, *i.e.,* the distribution has support on $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_d\}$. Since $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_d\} \subset \mathcal{B}_1^d$, we have $\widehat{\mathcal{P}}_1^d \subset \mathcal{P}_1^d$. Moreover, since any $q \in \widehat{\mathcal{P}}_1^d$ is a discrete distribution, by abusing notation, we describe q through a $d-$dimensional vector $\boldsymbol{q}$ of its probability mass function. Note that, for any $\boldsymbol{q} \in \widehat{\mathcal{P}}_1^d$, the average over this distribution is $\boldsymbol{\mu_q} = \mathbb{E}_{\boldsymbol{q}}[\mathbf{U}]$, where $\mathbb{E}_{\boldsymbol{q}}[\cdot]$ denotes the expectation over the distribution $\boldsymbol{q}$ for a discrete random variable $\mathbf{U} \sim \boldsymbol{q}$, where we denote $q_i = \Pr[\mathbf{U} = \boldsymbol{e}_i]$. Therefore we have $\boldsymbol{\mu_q} = \sum_{i=1}^d q_i \boldsymbol{e}_i = (q_1, \ldots, q_d)^T = \boldsymbol{q}$, for every $\boldsymbol{q} \in \widehat{\mathcal{P}}_1^d$. Let Δ_d denote the probability simplex in d dimensions. Since the discrete distribution $q \in \widehat{\mathcal{P}}_1^d$ is representable as $\boldsymbol{q} \in \Delta_d$, we have an isomorphism between Δ_d and $\widehat{\mathcal{P}}_1^d$, *i.e.,* we can equivalently think of $\widehat{\mathcal{P}}_1^d = \Delta_d$. Fix arbitrary (ϵ_0, ∞)-CLDP mechanisms $\{\mathcal{R}_p^{(i)} : i \in [n]\}$ and an estimator $\widehat{\boldsymbol{x}}$. Using the above notations and observations, we have:

$$\begin{aligned}\sup_{\boldsymbol{q} \in \mathcal{P}_1^d} \mathbb{E}\left\|\boldsymbol{\mu_q} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 &\geq \sup_{\boldsymbol{q} \in \widehat{\mathcal{P}}_1^d} \mathbb{E}\left\|\boldsymbol{\mu_q} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 \\ &= \sup_{\boldsymbol{q} \in \widehat{\mathcal{P}}_1^d} \mathbb{E}\left\|\boldsymbol{q} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 .\end{aligned} \tag{16}$$

Using $\widehat{\mathcal{P}}_1^d = \Delta_d$, and taking the infimum in (16) over all (ϵ_0, ∞)-CLDP mechanisms $\{\mathcal{R}_p^{(i)} : i \in [n]\}$ and estimators $\widehat{\boldsymbol{x}}$, we get

$$\begin{aligned}&\inf_{\{\mathcal{R}_p^{(i)} \in \mathcal{Q}_{(\epsilon,\infty)}\}} \inf_{\widehat{\boldsymbol{x}}} \sup_{\boldsymbol{q} \in \mathcal{P}_1^d} \mathbb{E}\left\|\boldsymbol{\mu_q} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 \\ &\quad\geq \inf_{\{\mathcal{R}_p^{(i)} \in \mathcal{Q}_{(\epsilon,\infty)}\}} \inf_{\widehat{\boldsymbol{x}}} \sup_{\boldsymbol{q} \in \Delta_d} \mathbb{E}\left\|\boldsymbol{q} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 .\end{aligned} \tag{17}$$

Girgis et al. [49, Theorem 1] lower-bounded the RHS of (17) in the context of characterizing a privacy-utility-randomness tradeoff in LDP. When specializing to our setting, where we are not concerned about the amount of randomness used, their lower bound result gives $\inf_{\{\mathcal{R}_p^{(i)} \in \mathcal{Q}_{(\epsilon_0,\infty)}\}} \inf_{\widehat{\boldsymbol{x}}} \sup_{\boldsymbol{q} \in \Delta_d} \mathbb{E}\left\|\boldsymbol{q} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 \geq \Omega\left(\min\left\{1, \frac{d}{n\epsilon_0^2}\right\}\right)$. Substituting this in (17) gives

$$R_{\epsilon_0,\infty,n}^{1,d} \geq \Omega\left(\min\left\{1, \frac{d}{n\epsilon_0^2}\right\}\right). \tag{18}$$

B. Lower Bound on $r_{\epsilon_0,\infty,n}^{p,d}$: Proof of Theorem 3

Similar to Section V-A, we prove the lower bound on $r_{\epsilon_0,\infty,n}^{p,d}$ separately depending on whether $p \geq 2$ or $p \leq 2$ (at $p = 2$, both bounds coincide) below in Section V-B1 and Section V-B2, respectively. In both the proofs, the main idea is to transform the worst-case lower bound to the average case lower bound and then use relation between different norms.

1) Lower bound for $p \in [2, \infty]$: Fix arbitrary (ϵ_0, ∞)-CLDP mechanisms $\{\mathcal{R}_p^{(i)} : i \in [n]\}$ and an estimator $\widehat{\boldsymbol{x}}$. It follows from (15) that there exists a distribution $\boldsymbol{q} \in \mathcal{P}_p^d$, such that if we sample $\boldsymbol{x}_i^{(q)} \sim \boldsymbol{q}$, i.i.d. for all $i \in [n]$ and letting $\boldsymbol{y}_i = \mathcal{R}_p^{(i)}(\boldsymbol{x}_i^{(q)})$, we would have $\mathbb{E}\left\|\boldsymbol{\mu_q} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 \geq \Omega\left(d^{1-\frac{2}{p}} \min\left\{1, \frac{d}{n \min\{\epsilon_0, \epsilon_0^2\}}\right\}\right)$. We have

$$\begin{aligned}\sup_{\{\boldsymbol{x}_i\} \in \mathcal{B}_p^d} \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 &\overset{(a)}{\geq} \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i^{(q)} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 \\ &\overset{(b)}{\geq} \frac{1}{2}\mathbb{E}\left\|\boldsymbol{\mu_q} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 - \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i^{(q)} - \boldsymbol{\mu_q}\right\|_2^2 \qquad (19) \\ &\overset{(c)}{\geq} \Omega\left(d^{1-\frac{2}{p}} \min\left\{1, \frac{d}{n \min\{\epsilon_0, \epsilon_0^2\}}\right\}\right) - \frac{d^{1-\frac{2}{p}}}{n} \\ &\overset{(d)}{\geq} \Omega\left(d^{1-\frac{2}{p}} \min\left\{1, \frac{d}{n \min\{\epsilon_0, \epsilon_0^2\}}\right\}\right) \qquad (20)\end{aligned}$$

In the LHS of (a), the expectation is taken over the randomness of the mechanisms $\{\mathcal{R}_p^{(i)}\}$ and the estimator $\widehat{\boldsymbol{x}}$; whereas, in the RHS of (a), in addition, the expectation is also taken over sampling $\boldsymbol{x}_i$'s from the distribution $\boldsymbol{q}$. Moreover (a) holds since the LHS is supremum $\{\boldsymbol{x}_i\} \in \mathcal{B}_p^d$ and the RHS of (a) takes expectation w.r.t. a distribution over $\mathcal{B}_p^d$ and hence lower-bounds the LHS. The inequality (b) follows from the Jensen's inequality $2\|\mathbf{u}\|_2^2 + 2\|\mathbf{v}\|_2^2 \geq \|\boldsymbol{u} + \mathbf{v}\|_2^2$ by setting $\boldsymbol{u} = \frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i^{(q)} - \widehat{\boldsymbol{x}}(\boldsymbol{y}^n)$ and $\mathbf{v} = \boldsymbol{\mu_q} - \frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i^{(q)}$. In (c) we used $\mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i^{(q)} - \boldsymbol{\mu_q}\right\|_2^2 \leq \frac{d^{1-\frac{2}{p}}}{n}$, which we show below. In (d), we assume $\min\{\epsilon_0, \epsilon_0^2\} \leq \mathcal{O}(d)$.

Note that for any vector $\boldsymbol{u} \in \mathbb{R}^d$, we have $\|\boldsymbol{u}\|_2 \leq d^{\frac{1}{2}-\frac{1}{p}}\|\boldsymbol{u}\|_p$, for any $p \geq 2$. Since each $\boldsymbol{x}_i^{(q)} \in \mathcal{B}_p^d$, which implies $\|\boldsymbol{x}_i^{(q)}\|_p \leq 1$, we have that $\|\boldsymbol{x}_i^{(q)}\|_2 \leq d^{\frac{1}{2}-\frac{1}{p}}$. Hence, $\mathbb{E}\|\boldsymbol{x}_i^{(q)}\|_2^2 \leq d^{1-\frac{2}{p}}$ holds for all $i \in [n]$. Now, since $\boldsymbol{x}_i$'s are i.i.d. with $\mathbb{E}[\boldsymbol{x}_i^{(q)}] = \boldsymbol{\mu_q}$, we have

$$\begin{aligned}\mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i^{(q)} - \boldsymbol{\mu_q}\right\|_2^2 &= \frac{1}{n^2}\sum_{i=1}^n \mathbb{E}\left\|\boldsymbol{x}_i^{(q)} - \boldsymbol{\mu_q}\right\|_2^2 \\ &\overset{(a)}{\leq} \frac{1}{n^2}\sum_{i=1}^n \mathbb{E}\left\|\boldsymbol{x}_i^{(q)}\right\|_2^2 \leq \frac{1}{n^2}\sum_{i=1}^n d^{1-\frac{2}{p}} = \frac{d^{1-\frac{2}{p}}}{n},\end{aligned} \tag{21}$$

where (a) uses $\mathbb{E}\|\boldsymbol{x} - \mathbb{E}[\boldsymbol{x}]\|_2^2 \leq \mathbb{E}\|\boldsymbol{x}\|_2^2$, which holds for any random vector $\boldsymbol{x}$.

Taking supremum in (20) over all (ϵ_0, ∞)-CLDP mechanisms $\{\mathcal{R}_p^{(i)} : i \in [n]\}$ and estimators $\widehat{\boldsymbol{x}}$, we get

$$r_{\epsilon_0,\infty,n}^{p,d} \geq \Omega\left(d^{1-\frac{2}{p}} \min\left\{1, \frac{d}{n \min\{\epsilon_0, \epsilon_0^2\}}\right\}\right). \tag{22}$$

2) Lower bound for $p \in [1, 2]$: Similar to the argument given in Section V-A2, since $r_{\epsilon_0,\infty,n}^{p,d} \geq r_{\epsilon_0,\infty,n}^{1,d}$ holds for any $p \in [1, 2]$, it suffices to lower-bound $r_{\epsilon_0,\infty,n}^{1,d}$.

Fix arbitrary (ϵ_0, ∞)-CLDP mechanisms $\{\mathcal{R}_p^{(i)} : i \in [n]\}$ and an estimator $\widehat{\boldsymbol{x}}$. It follows from (18) that there exists a distribution $\boldsymbol{q} \in \mathcal{P}_p^d$, such that if we sample $\boldsymbol{x}_i^{(q)} \sim \boldsymbol{q}$, i.i.d. for all $i \in [n]$ and letting $\boldsymbol{y}_i = \mathcal{R}_p^{(i)}(\boldsymbol{x}_i^{(q)})$, we would have $\mathbb{E}\left\|\boldsymbol{\mu_q} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 \geq \Omega\left(\min\left\{1, \frac{d}{n\epsilon_0^2}\right\}\right)$. Now, by the same reasoning using which we obtained (19), we have

$$\begin{aligned}&\sup_{\{\boldsymbol{x}_i\} \in \mathcal{B}_p^d} \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 \\ &\geq \frac{1}{2}\mathbb{E}\left\|\boldsymbol{\mu_q} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 - \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i^{(q)} - \boldsymbol{\mu_q}\right\|_2^2 \qquad (23) \\ &\overset{(a)}{\geq} \Omega\left(\min\left\{1, \frac{d}{n\epsilon_0^2}\right\}\right) - \frac{1}{n} \overset{(b)}{\geq} \Omega\left(\min\left\{1, \frac{d}{n\epsilon_0^2}\right\}\right)\end{aligned}$$

In (a) we used

$$\mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i^{(q)}-\boldsymbol{\mu}_{\boldsymbol{q}}\right\|_2^2\leq\frac{1}{n}, \tag{24}$$

which can be obtained by first noting that for any $\boldsymbol{u}\in\mathbb{R}^d$, we have $\|\boldsymbol{u}\|_2\leq\|\boldsymbol{u}\|_p$ for $p\in[1,2]$, and then using this in the set of inequalities which give (21). In (b), we assume $\epsilon_0\leq\mathcal{O}(\sqrt{d})$. Taking supremum in (20) over all (ϵ_0,∞)-CLDP mechanisms $\{\mathcal{R}_p^{(i)}:i\in[n]\}$ and estimators $\widehat{\boldsymbol{x}}$, we get $r^{1,d}_{\epsilon_0,\infty,n}\geq\Omega\left(\min\left\{1,\frac{d}{n\epsilon_0^2}\right\}\right)$.

C. Lower Bound on $r^{p,d}_{\epsilon_0,b,n}$: Proof of Theorem 4

Let $M=2^b<d$ be the total number of possible outputs of the mechanism $\mathcal{R}$. Let $\{o_1,o_2,\ldots,o_M\}$ be the set of M possible outputs of $\mathcal{R}$. For every $i\in[M]$, let $q_i=g(o_i)$. We can write the M possible outputs of $\mathcal{R}$ as columns of a $d\times M$ matrix $Q=[q_1,\ldots,q_M]$. Since $M<d$, the rank of the matrix Q is at most M. Let $\boldsymbol{x}\in\mathbb{R}^d$ be a vector in the null space of the matrix Q, i.e., $\boldsymbol{x}^Tq_j=0$ for all $j\in[M]$. Then, we set the sample of each client by $\boldsymbol{x}_i=\overline{\boldsymbol{x}}=\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|_p}$ for all $i\in[n]$, and hence, $\boldsymbol{x}_i\in\mathcal{B}_p^d$. Observe that the estimator $\hat{\boldsymbol{x}}=\frac{1}{n}\sum_{i=1}^n g\left(\mathcal{R}\left(\boldsymbol{x}_i\right)\right)$ is in the column space of the matrix Q. Thus, we get

$$\begin{aligned}r^{p,d}_{\epsilon_0,b,n}&\geq\mathbb{E}\left\|\overline{\boldsymbol{x}}-\frac{1}{n}\sum_{i=1}^{n}g\left(\mathcal{R}\left(\boldsymbol{x}_i\right)\right)\right\|_2^2\\&\overset{(a)}{=}\|\overline{\boldsymbol{x}}\|_2^2+\mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^{n}g\left(\mathcal{R}\left(\boldsymbol{x}_i\right)\right)\right\|_2^2\geq\max\left\{1,d^{1-\frac{2}{p}}\right\}\end{aligned}$$

where step (a) follows from the fact that $\overline{\boldsymbol{x}}$ is in the null space of Q, while the estimator $\hat{\boldsymbol{x}}$ is in the column space of Q. This completes the proof of Theorem 4.

D. Achievability for ℓ_1-norm Ball: Proof of Theorem 5

In this section, we propose an ϵ_0-LDP mechanism that requires $\mathcal{O}\left(\log(d)\right)$-bits of communication per client using private randomness and 1-bit of communication per client using public randomness. In other words we can guarantee $(\epsilon_0,\mathcal{O}\left(\log(d)\right))$-CLDP with private randomness and $(\epsilon_0,1)$-CLDP using public randomness. The proposed mechanism is based on the Hadamard matrix and is inspired from the Hadamard mechanism proposed by Acharya et al. [36]. We assume that d is a power of 2. Let $\mathbf{H}_d$ denote the Hadamard matrix of order d, which can be constructed by the following recursive mechanism:

$$\mathbf{H}_d=\begin{bmatrix}\mathbf{H}_{d/2}&\mathbf{H}_{d/2}\\\mathbf{H}_{d/2}&-\mathbf{H}_{d/2}\end{bmatrix}\qquad\mathbf{H}_1=\begin{bmatrix}1\end{bmatrix}$$

Client i has an input $\boldsymbol{x}_i\in\mathcal{B}_1^d\left(a\right)$. It computes $\boldsymbol{y}_i=\frac{1}{\sqrt{d}}\mathbf{H}_d\boldsymbol{x}_i$. Note that each coordinate of $\boldsymbol{y}_i$ lies in the interval $\left[-a/\sqrt{d},a/\sqrt{d}\right]$. Client i selects $j\sim\mathsf{Unif}\left[d\right]$ and quantize $y_{i,j}$ privately according to (25) and obtains $\boldsymbol{z}_i\in\left\{\pm a\mathbf{H}_d(j)\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\right\}$, which can be represented using only 1-bit. Here, $\mathbf{H}_d(j)$ denotes the j-th column of the Hadamard matrix $\mathbf{H}_d$. Server receives the n messages $\{\boldsymbol{z}_1,\ldots,\boldsymbol{z}_n\}$ from the clients and outputs their average $\frac{1}{n}\sum_{i=1}^n\boldsymbol{z}_i$. We present this mechanism in Algorithm 2 – we only present the client-side part of the algorithm, as server only averages the messages received from the clients.

Algorithm 2 ℓ_1-MEAN-EST ($\mathcal{R}_1$: the client-side algorithm)

1: **Input:** Vector $\boldsymbol{x}\in\mathcal{B}_1^d\left(a\right)$, and local privacy level $\epsilon_0>0$.
2: Construct $\boldsymbol{y}=\frac{1}{\sqrt{d}}\mathbf{H}_d\boldsymbol{x}$
3: Sample $j\sim\mathsf{Unif}[d]$ and quantize y_j as follows:

$$\boldsymbol{z}=\begin{cases}+a\mathbf{H}_d\left(j\right)\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)&\text{w.p. }\frac{1}{2}+\frac{\sqrt{d}y_j}{2a}\frac{e^{\epsilon_0}-1}{e^{\epsilon_0}+1}\\-a\mathbf{H}_d\left(j\right)\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)&\text{w.p. }\frac{1}{2}-\frac{\sqrt{d}y_j}{2a}\frac{e^{\epsilon_0}-1}{e^{\epsilon_0}+1}\end{cases} \tag{25}$$

4: Return $\boldsymbol{z}$.

Lemma 2. *The mechanism $\mathcal{R}_1$ presented in Algorithm 2 satisfies the following properties, where $\epsilon_0>0$:*

1) *$\mathcal{R}_1$ is $(\epsilon_0,\log\left(d\right)+1)$-CLDP and requires only 1-bit of communication using public randomness.*
2) *$\mathcal{R}_1$ is unbiased and has bounded variance, i.e., for every $\boldsymbol{x}\in\mathcal{B}_1^d\left(a\right)$, we have $\mathbb{E}\left[\mathcal{R}_1\left(\boldsymbol{x}\right)\right]=\boldsymbol{x}$ and*

$$\mathbb{E}\|\mathcal{R}_1\left(\boldsymbol{x}\right)-\boldsymbol{x}\|_2^2\leq a^2d\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2.$$

We prove Lemma 2 in Appendix D-A. Now we are ready to prove Theorem 5. Let $\mathcal{R}_1(\boldsymbol{x})$ denote the output of Algorithm 2 on input $\boldsymbol{x}$. As mentioned above, the server employs a simple estimator that simply averages the n received messages, i.e., the server outputs $\widehat{\boldsymbol{x}}(\boldsymbol{z}^n)=\frac{1}{n}\sum_{i=1}^n\boldsymbol{z}_i=\frac{1}{n}\sum_{i=1}^n\mathcal{R}_1(\boldsymbol{x}_i)$. In the following, first we show the bound on $r^{1,d}_{\epsilon_0,b,n}\left(a\right)$ and then on $R^{1,d}_{\epsilon_0,b,n}\left(a\right)$ for $b=\log(d)+1$. For $r^{1,d}_{\epsilon_0,b,n}\left(a\right)$:

$$\begin{aligned}&\sup_{\{\boldsymbol{x}_i\}\in\mathcal{B}_1^d(a)}\mathbb{E}\left\|\overline{\boldsymbol{x}}-\widehat{\boldsymbol{x}}(\boldsymbol{z}^n)\right\|_2^2\\&=\sup_{\{\boldsymbol{x}_i\}\in\mathcal{B}_1^d(a)}\mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^{n}\left(\boldsymbol{x}_i-\mathcal{R}_1(\boldsymbol{x}_i)\right)\right\|_2^2\\&\overset{(a)}{=}\sup_{\{\boldsymbol{x}_i\}\in\mathcal{B}_1^d(a)}\frac{1}{n^2}\sum_{i=1}^{n}\mathbb{E}\left\|\boldsymbol{x}_i-\mathcal{R}_1(\boldsymbol{x}_i)\right\|_2^2\overset{(b)}{\leq}\frac{a^2d}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2,\end{aligned} \tag{26}$$

where (a) uses the fact that all clients use independent private randomness (which makes the random variables $\boldsymbol{x}_i-\mathcal{R}_1(\boldsymbol{x}_i)$ independent for different i's and also that $\mathcal{R}_1$ is unbiased. (b) uses that $\mathcal{R}_1$ has bounded variance. Taking infimum in (26) over all (ϵ_0,b)-CLDP mechanisms (where $b=\log(d)+1$) and estimators $\widehat{\boldsymbol{x}}$, we have that $r^{1,d}_{\epsilon_0,b,n}\left(a\right)\leq\frac{a^2d}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2$, which is $\mathcal{O}\left(\frac{a^2d}{n\epsilon_0^2}\right)$ when $\epsilon_0=\mathcal{O}(1)$. For $R^{1,d}_{\epsilon_0,b,n}\left(a\right)$:

$$\begin{aligned}&\sup_{\boldsymbol{q}\in\mathcal{P}_1^d(a)}\mathbb{E}\left\|\boldsymbol{\mu}_{\boldsymbol{q}}-\widehat{\boldsymbol{x}}(\boldsymbol{z}^n)\right\|_2^2\\&\overset{(c)}{\leq}\sup_{\boldsymbol{q}\in\mathcal{P}_1^d(a)}\left[2\mathbb{E}\left\|\boldsymbol{\mu}_{\boldsymbol{q}}-\overline{\boldsymbol{x}}\right\|_2^2+2\mathbb{E}\left\|\overline{\boldsymbol{x}}-\widehat{\boldsymbol{x}}(\boldsymbol{z}^n)\right\|_2^2\right]\\&\overset{(d)}{\leq}\frac{2a^2}{n}+\frac{2a^2d}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2\end{aligned} \tag{27}$$

In the LHS of (c), for any $\boldsymbol{q} \in \mathcal{P}_1^d(a)$, first we generate n i.i.d. samples $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ and then compute $\boldsymbol{z}_i = \mathcal{R}_1(\boldsymbol{x}_i)$ for all $i \in [n]$. We use the Jensen's inequality in (c). We used $\mathbb{E}\left\|\boldsymbol{\mu}_{\boldsymbol{q}} - \overline{\boldsymbol{x}}\right\|_2^2 \leq \frac{a^2}{n}$ (see (24)) in (d). Taking infimum in (27) over all (ϵ_0, b)-CLDP mechanisms (where $b = \log(d)+1$) and estimators $\widehat{\boldsymbol{x}}$, we have that $R^{1,d}_{\epsilon_0,b,n}(a) \leq \frac{2a^2}{n} + \frac{2a^2 d}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2$, which is $\mathcal{O}\left(\frac{a^2 d}{n\epsilon_0^2}\right)$ when $\epsilon_0 = \mathcal{O}(1)$. This completes the proof of Theorem 5.

E. Achievability for ℓ_2-norm Ball: Proof of Theorem 6

In this section, we propose an ϵ_0-LDP mechanism that requires $\mathcal{O}(d)$-bits of communication per client using private randomness. Our proposed mechanism is a combination of the private-mechanism Priv of Duchi et al. [50, Section 4.2.3] and the non-private quantization mechanism Quan of Mayekar and Tyagi [35, Section 4.2]. For completeness, we describe both these mechanisms in Algorithm 4 and Algorithm 5, respectively, and our proposed mechanism in Algorithm 3. Each client i first privatize its input $\boldsymbol{x}_i \in \mathcal{B}_2^d(a)$ using Priv and then quantize the privatized result using Quan and sends the final result $\boldsymbol{z}_i = \mathsf{Quan}(\mathsf{Priv}(\boldsymbol{x}_i))$ to the server, which outputs the average of all the received n messages. Since the server is only taking an average of the received messages, we only present the client side of our mechanism in Algorithm 3.

Algorithm 3 ℓ_2-MEAN-EST ($\mathcal{R}_2$: the client-side algorithm)

1: **Input:** Vector $\boldsymbol{x} \in \mathcal{B}_2^d(a)$, and local privacy level $\epsilon_0 > 0$.
2: Apply the randomized mechanism $\boldsymbol{y} = \mathsf{Priv}(\boldsymbol{x})$.
3: Return $\boldsymbol{z} = \mathsf{Quan}(\boldsymbol{y})$.

Algorithm 4 Priv (a private mechanism from [50])

1: **Input:** Vector $\boldsymbol{x} \in \mathcal{B}_2^d(a)$, and local privacy level $\epsilon_0 > 0$.
2: Compute $\widetilde{\boldsymbol{x}} = \begin{cases} +a\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|_2} & \text{w.p. } \frac{1}{2} + \frac{\|\boldsymbol{x}\|_2}{2a} \\ -a\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|_2} & \text{w.p. } \frac{1}{2} - \frac{\|\boldsymbol{x}\|_2}{2a} \end{cases}$
3: Sample $U \sim \text{Bernoulli}\left(\frac{e^{\epsilon_0}}{e^{\epsilon_0}+1}\right)$
4: $M \triangleq a\frac{\sqrt{\pi}}{2}\frac{\Gamma\left(\frac{d-1}{2}+1\right)}{\Gamma\left(\frac{d}{2}+1\right)}\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}$
5: $\boldsymbol{z} = \begin{cases} \mathsf{Unif}\left(\boldsymbol{y} : \boldsymbol{y}^T\widetilde{\boldsymbol{x}} > 0, \|\boldsymbol{y}\|_2 = M\right) & \text{if } U = 1 \\ \mathsf{Unif}\left(\boldsymbol{y} : \boldsymbol{y}^T\widetilde{\boldsymbol{x}} \leq 0, \|\boldsymbol{y}\|_2 = M\right) & \text{if } U = 0 \end{cases}$
6: Return $\boldsymbol{z}$.

Lemma 3 ([50, Appendix I.2]). *The mechanism* Priv *presented in Algorithm 4 is unbiased and outputs a bounded length vector, i.e., for every $\boldsymbol{x} \in \mathcal{B}_2^d(a)$, we have $\mathbb{E}[\mathsf{Priv}(\boldsymbol{x})] = \boldsymbol{x}$ and*

$$\|\mathsf{Priv}(\boldsymbol{x})\|_2^2 = M^2 \leq a^2 d\left(\frac{3\sqrt{\pi}}{4}\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2.$$

Lemma 4 ([35, Theorem 4.2]). *The mechanism* Quan *presented in Algorithm 5 is unbiased and has bounded variance, i.e., for every $\boldsymbol{x} \in \mathcal{B}_2^d(a)$, we have*

$$\mathbb{E}[\mathsf{Quan}(\boldsymbol{x})] = \boldsymbol{x} \quad \textit{and} \quad \mathbb{E}\|\mathsf{Quan}(\boldsymbol{x}) - \boldsymbol{x}\|_2^2 \leq 2\|\boldsymbol{x}\|^2 \leq 2a^2.$$

Algorithm 5 Quan (a quantization mechanism from [35])

1: **Input:** Vector $\boldsymbol{x} \in \mathcal{B}_2^d(a)$, where a is the radius of the ball.
2: Compute $\widetilde{\boldsymbol{x}} = \begin{cases} \frac{\boldsymbol{x}}{\|\boldsymbol{x}\|_1} & \text{w.p. } \frac{1}{2} + \frac{\|\boldsymbol{x}\|_1}{2a\sqrt{d}} \\ -\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|_1} & \text{w.p. } \frac{1}{2} - \frac{\|\boldsymbol{x}\|_1}{2a\sqrt{d}} \end{cases}$
3: Generate a discrete distribution $\boldsymbol{\mu} = (|\widetilde{x}_1|, \ldots, |\widetilde{x}_d|)$ where $\Pr[\boldsymbol{\mu} = i] = |\widetilde{x}_i|$.
4: Construct a d-dimensional vector $\boldsymbol{y}$ by sampling $y_j \sim \boldsymbol{\mu}$ for $j \in [d]$
5: Return $\boldsymbol{z} = \frac{1}{d}\sum_{j=1}^d \left(a\sqrt{d} \cdot \text{sgn}(\widetilde{x}_{y_j}) \cdot \boldsymbol{e}_{y_j}\right)$.

Furthermore, it requires $d(\log(e)+1)$-bits to represent its output.

Note that the radius a in Lemma 4 is equal to the length of any output of Priv, which is M (see line 4 of Algorithm 4).

Lemma 5. *The mechanism $\mathcal{R}_2$ presented in Algorithm 3 satisfies the following properties, where $\epsilon_0 > 0$:*

1) *$\mathcal{R}_2$ is $(\epsilon_0, d(\log(e)+1))$-CLDP.*
2) *$\mathcal{R}_2$ is unbiased and has bounded variance, i.e., for every $\boldsymbol{x} \in \mathcal{B}_2^d(a)$, we have $\mathbb{E}[\mathcal{R}_2(\boldsymbol{x})] = \boldsymbol{x}$ and*

$$\mathbb{E}\|\mathcal{R}_2(\boldsymbol{x}) - \boldsymbol{x}\|_2^2 \leq 6a^2 d\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2.$$

We prove Lemma 5 in Appendix D-B. Now we are ready to prove Theorem 6. In order to bound $r^{2,d}_{\epsilon_0,b,n}(a)$ for $b = d(\log(e)+1)$, we follow exactly the same steps that we used to bound $r^{1,d}_{\epsilon_0,b,n}(a)$ and arrived at (26). This would give $r^{2,d}_{\epsilon_0,b,n}(a) \leq \frac{6a^2 d}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2$, which is $\mathcal{O}\left(\frac{a^2 d}{n\epsilon_0^2}\right)$ when $\epsilon_0 = \mathcal{O}(1)$. To bound $R^{2,d}_{\epsilon_0,b,n}(a)$, first note that when $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{B}_2^d(a)$, then we have from (24) that $\mathbb{E}\left\|\boldsymbol{\mu}_{\boldsymbol{q}} - \overline{\boldsymbol{x}}\right\|_2^2 \leq \frac{a^2}{n}$. Here $\boldsymbol{q} \in \mathcal{P}_2^d(a)$ and $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ are sampled from $\boldsymbol{q}$ i.i.d. Now, following exactly the same steps that we used to bound $R^{1,d}_{\epsilon_0,b,n}(a)$ and arrived at (27). This would give $R^{2,d}_{\epsilon_0,b,n}(a) \leq \frac{2a^2}{n} + \frac{12a^2 d}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2$ for $b = d(\log(e)+1)$. Note that $R^{2,d}_{\epsilon_0,b,n}(a) = \mathcal{O}\left(\frac{a^2 d}{n\epsilon_0^2}\right)$ when $\epsilon_0 = \mathcal{O}(1)$. This completes the proof of Theorem 6.

F. Achievability for ℓ_∞-norm Ball: Proof of Theorem 7

In this section, we propose an ϵ_0-LDP mechanism that requires $\mathcal{O}(\log(d))$-bits per client using private randomness and 1-bit of communication per client using public randomness. Each client i has an input $\boldsymbol{x}_i \in \mathcal{B}_\infty^d(a)$. It selects $j \sim \mathsf{Unif}[d]$ and quantize $x_{i,j}$ according to (28) and obtains $\boldsymbol{z}_i \in \left\{ \pm ad\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\boldsymbol{e}_j\right\}$, which can be represented using only 1 bit, where $\boldsymbol{e}_j$ is the j'th standard basis vector in $\mathbb{R}^d$. Client i sends $\boldsymbol{z}_i$ to the server. Server receives the n messages $\{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n\}$ from the clients and outputs their average $\frac{1}{n}\sum_{i=1}^n \boldsymbol{z}_i$. We present this mechanism in Algorithm 6 – we only present the client-side part of the algorithm, as server only averages the messages received from the clients.

Lemma 6. *The mechanism $\mathcal{R}_\infty$ presented in Algorithm 6 satisfies the following properties, where $\epsilon_0 > 0$:*

Algorithm 6 ℓ_∞-MEAN-EST ($\mathcal{R}_\infty$: the client-side algorithm)

1: **Input:** Vector $\boldsymbol{x} \in \mathcal{B}_\infty^d(a)$, and local privacy level $\epsilon_0 > 0$.
2: Sample $j \sim \mathsf{Unif}[d]$ and quantize x_j as follows:

$$\boldsymbol{z} = \begin{cases} +ad\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\boldsymbol{e}_j & \text{w.p. } \frac{1}{2} + \frac{x_j}{2a}\frac{e^{\epsilon_0}-1}{e^{\epsilon_0}+1} \\ -ad\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\boldsymbol{e}_j & \text{w.p. } \frac{1}{2} - \frac{x_j}{2a}\frac{e^{\epsilon_0}-1}{e^{\epsilon_0}+1} \end{cases} \tag{28}$$

where $\boldsymbol{e}_j$ is the j'th standard basis vector in $\mathbb{R}^d$
3: Return $\boldsymbol{z}$.

1) *$\mathcal{R}_\infty$ is $(\epsilon_0, \log(d)+1)$-CLDP and requires only 1-bit of communication using public randomness.*
2) *$\mathcal{R}_\infty$ is unbiased and has bounded variance, i.e., for every $\boldsymbol{x} \in \mathcal{B}_\infty^d(a)$, we have $\mathbb{E}[\mathcal{R}_\infty(\boldsymbol{x})] = \boldsymbol{x}$ and*

$$\mathbb{E}\|\mathcal{R}_\infty(\boldsymbol{x}) - \boldsymbol{x}\|_2^2 \le a^2 d^2 \left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2.$$

We prove Lemma 6 in Appendix D-C. Now we are ready to prove Theorem 7. In order to bound $r^{\infty,d}_{\epsilon_0,b,n}(a)$ for $b = \log(d)+1$, we follow exactly the same steps that we used to bound $r^{1,d}_{\epsilon_0,b,n}(a)$ and arrived at (26). This would give $r^{\infty,d}_{\epsilon_0,b,n}(a) \le \frac{a^2d^2}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2$, which is $\mathcal{O}\left(\frac{a^2d^2}{n\epsilon_0^2}\right)$ when $\epsilon_0 = \mathcal{O}(1)$. To bound $R^{\infty,d}_{\epsilon_0,b,n}(a)$, first note that when $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n \in \mathcal{B}_\infty^d(a)$, then we have from (21) (by substituting $p = \infty$) that $\mathbb{E}\left\|\boldsymbol{\mu_q} - \overline{\boldsymbol{x}}\right\|_2^2 \le \frac{a^2d}{n}$. Here $\boldsymbol{q} \in \mathcal{P}_\infty^d(a)$ and $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n$ are sampled from $\boldsymbol{q}$ i.i.d. Now, following exactly the same steps that we used to bound $R^{1,d}_{\epsilon_0,b,n}(a)$ and arrived at (27). This would give $R^{\infty,d}_{\epsilon_0,b,n}(a) \le \frac{2a^2d}{n} + \frac{2a^2d^2}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^{2,d}$ for $b = \log(d)+1$. Note that $R^{\infty,d}_{\epsilon_0,b,n}(a) = \mathcal{O}\left(\frac{a^2d^2}{n\epsilon_0^2}\right)$ when $\epsilon_0 = \mathcal{O}(1)$. This completes the proof of Theorem 7.

G. Achievability for ℓ_p-norm Ball for $p \in [1,\infty)$: Proof of Corollary 1

In this section, first we propose two ϵ_0-LDP mechanisms for ℓ_p-norm ball $\mathcal{B}_p^d(a)$ for $p \in [1,\infty)$ based on the inequalities between different norms, and our final mechanism will be chosen probabilistically from these two. The first mechanism, which we denote by $\mathcal{R}_p^{(1)}$, is based on the private mechanism $\mathcal{R}_1$ (presented in Algorithm 2) that requires $\mathcal{O}(\log(d))$ bits per client. The second mechanism, which we denote by $\mathcal{R}_p^{(2)}$ is based on the private mechanism $\mathcal{R}_2$ (presented in Algorithm 3) that requires $\mathcal{O}(d)$ bits per client. Observe that for any $1 \le p \le q \le \infty$, using the relation between different norms ($\|\boldsymbol{u}\|_q \le \|\boldsymbol{u}\|_p \le d^{\frac{1}{p}-\frac{1}{q}}\|\boldsymbol{u}\|_q$), we have

$$\mathcal{B}_q^d(a) \subseteq \mathcal{B}_p^d(a) \subseteq \mathcal{B}_q^d\left(ad^{\frac{1}{p}-\frac{1}{q}}\right). \tag{29}$$

1) *Description of the private mechanism $\mathcal{R}_p^{(1)}$:* Each client has a vector $\boldsymbol{x}_i \in \mathcal{B}_p^d(a) \subseteq \mathcal{B}_1^d\left(ad^{1-\frac{1}{p}}\right)$. Thus, each client runs the private mechanism $\mathcal{R}_1(\boldsymbol{x}_i)$ presented in Algorithm 2 with radius $ad^{1-\frac{1}{p}}$. Thus, the mechanism $\mathcal{R}_p^{(1)}$ for $p \in [1,\infty)$ satisfies the following properties, where $\epsilon_0 > 0$:

- $\mathcal{R}_p^{(1)}$ is $(\epsilon_0, \log(d)+1)$-CLDP and requires only 1-bit of communication using public randomness.
- $\mathcal{R}_p^{(1)}$ is unbiased and has bounded variance, i.e., for every $\boldsymbol{x} \in \mathcal{B}_p^d(a)$, we have $\mathbb{E}\left[\mathcal{R}_p^{(1)}(\boldsymbol{x})\right] = \boldsymbol{x}$ and

$$\mathbb{E}\|\mathcal{R}_p^{(1)}(\boldsymbol{x}) - \boldsymbol{x}\|_2^2 \le a^2 d^{3-\frac{2}{p}}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2.$$

2) *Description of the private mechanism $\mathcal{R}_p^{(2)}$:* Each client has a vector $\boldsymbol{x}_i \in \mathcal{B}_p^d(a) \subseteq \mathcal{B}_2^d\left(a\max\{d^{\frac{1}{2}-\frac{1}{p}}, 1\}\right)$. Thus, each client runs the private mechanism $\mathcal{R}_2(\boldsymbol{x}_i)$ presented in Algorithm 3 with radius $a\max\{d^{\frac{1}{2}-\frac{1}{p}}, 1\}$. Thus, the mechanism $\mathcal{R}_p^{(2)}$ for $p \in [1,\infty)$ satisfies the following properties, where $\epsilon_0 > 0$:

- $\mathcal{R}_p^{(2)}$ is $(\epsilon_0, d(\log(e)+1))$-CLDP.
- $\mathcal{R}_p^{(2)}$ is unbiased and has bounded variance, i.e., for every $\boldsymbol{x} \in \mathcal{B}_p^d(a)$, we have $\mathbb{E}\left[\mathcal{R}_p^{(2)}(\boldsymbol{x})\right] = \boldsymbol{x}$ and

$$\mathbb{E}\|\mathcal{R}_p^{(2)}(\boldsymbol{x}) - \boldsymbol{x}\|_2^2 \le 6a^2\max\{d^{2-\frac{2}{p}}, d\}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2.$$

Note that $\mathcal{R}_p^{(1)}$ requires low communication and has high variance, whereas, $\mathcal{R}_p^{(2)}$ requires high communication and has low variance: $\mathcal{R}_p^{(2)}$ requires exponentially more communication than $\mathcal{R}_p^{(1)}$, whereas, $\mathcal{R}_p^{(1)}$ has a factor of d more variance than $\mathcal{R}_p^{(2)}$. To define our final mechanism $\mathcal{R}_p$ for any norm $p \in [1,\infty)$, we choose $\mathcal{R}_p^{(1)}$ with probability $\bar{p}$ and $\mathcal{R}_p^{(2)}$ with probability $(1-\bar{p})$, where $\bar{p}$ is any number in $[0,1]$. Note that $\mathcal{R}_p$ is ϵ_0-LDP and requires $\bar{p}\log(d) + (1-\bar{p})d\log(e) + 1$ expected communication, where expectation is taken over the sampling of choosing $\mathcal{R}_p^{(1)}$ or $\mathcal{R}_p^{(2)}$. We have the following bounds on $r^{p,d}_{\epsilon_0,b,n}(a)$ and $R^{p,d}_{\epsilon_0,b,n}(a)$:

$$\begin{aligned} &r^{p,d}_{\epsilon_0,b,n}(a) \\ &\quad\le \bar{p}\, d^{2-\frac{2}{p}} r^{1,d}_{\epsilon_0,b,n}(a) + (1-\bar{p})\max\{d^{1-\frac{2}{p}}, 1\} r^{2,d}_{\epsilon_0,b,n}(a) \\ &R^{p,d}_{\epsilon_0,b,n}(a) \\ &\quad\le \bar{p}\, d^{2-\frac{2}{p}} R^{1,d}_{\epsilon_0,b,n}(a) + (1-\bar{p})\max\{d^{1-\frac{2}{p}}, 1\} R^{2,d}_{\epsilon_0,b,n}(a) \end{aligned}$$

This completes the proof of Corollary 1.

VI. Optimization: Privacy, Communication, and Convergence Analyses

In this section, we establish the privacy, communication, and convergence guarantees of Algorithm 1 and prove Theorem 1. We show these three results on privacy, communication, and convergence separately in the next three subsections.

A. Proof of Theorem 1: Privacy

Recall from Algorithm 1 that each client applies the compressed LDP mechanism $\mathcal{R}_p$ (hereafter denoted by $\mathcal{R}$, for simplicity) with privacy parameter ϵ_0 on each gradient. This implies that the mechanism $\mathcal{A}_{cldp}$ guarantees local differential privacy ϵ_0 for each sample d_{ij} per epoch. Thus, it remains to analyze the central DP of the mechanism $\mathcal{A}_{cldp}$.

Fix an iteration number $t \in [T]$. Let $\mathcal{M}_t(\theta_t, \mathcal{D})$ denote the private mechanism at time t that takes the dataset $\mathcal{D}$ and

an auxiliary input θ_t (which is the parameter vector at the t'th iteration) and generates the parameter θ_{t+1} as an output. Recall that the input dataset at client $i \in [m]$ is denoted by $\mathcal{D}_i = \{d_{i1}, d_{i2}, \ldots, d_{ir}\} \in \mathfrak{S}^r$ and $\mathcal{D} = \bigcup_{i=1}^m \mathcal{D}_i$ denotes the entire dataset. Thus, the mechanism $\mathcal{M}_t$ on any input dataset $\mathcal{D} = \bigcup_{i=1}^m \mathcal{D}_i \in \mathfrak{S}^n$ can be defined as:

$$\mathcal{M}_t(\theta_t; \mathcal{D}) = \mathcal{H}_{ks} \circ \mathrm{samp}_{m,k}\left(\mathcal{G}_1, \ldots, \mathcal{G}_m\right), \qquad (30)$$

where $\mathcal{G}_i = \mathrm{samp}_{r,s}\left(\mathcal{R}(\boldsymbol{x}_{i1}^t), \ldots, \mathcal{R}(\boldsymbol{x}_{ir}^t)\right)$ and $\boldsymbol{x}_{ij}^t = \nabla_{\theta_t} f(\theta_t; d_{ij}), \forall i \in [m], j \in [r]$. Here, $\mathcal{H}_{ks}$ denotes the shuffling operation on ks elements and $\mathrm{samp}_{m,k}$ denotes the sampling operation for choosing a random subset of k elements from a set of m elements.

For convenience, in the rest of the proof, we suppress the auxiliary input θ_t and simply denote $\mathcal{M}_t(\theta_t; \mathcal{D})$ by $\mathcal{M}_t(\mathcal{D})$. We can do this because θ_t only affects the gradients, and the analysis in this section is for an arbitrary set of gradients.

In the following lemma, we state the privacy guarantee of the mechanism $\mathcal{M}_t$ for each $t \in [T]$.

Lemma 7. *Let $s = 1$ and $q = \frac{k}{mr}$. Suppose $\mathcal{R}$ is an ϵ_0-LDP mechanism, where $\epsilon_0 \leq \frac{\log\left(qn/\log\left(1/\tilde{\delta}\right)\right)}{2}$ and $\tilde{\delta} > 0$ is arbitrary. Then, for any $t \in [T]$, the mechanism $\mathcal{M}_t$ is $\left(\bar{\epsilon}, \bar{\delta}\right)$-DP, where $\bar{\epsilon} = \ln(1 + q(e^{\tilde{\epsilon}} - 1)), \bar{\delta} = q\tilde{\delta}$ with $\tilde{\epsilon} = \mathcal{O}\left(\min\{\epsilon_0, 1\} e^{\epsilon_0} \sqrt{\frac{\log\left(1/\tilde{\delta}\right)}{qn}}\right)$. In particular, if $\epsilon_0 = \mathcal{O}(1)$, we get $\bar{\epsilon} = \mathcal{O}\left(\epsilon_0 \sqrt{\frac{q \log\left(1/\tilde{\delta}\right)}{n}}\right)$.*

We prove Lemma 7 in Appendix B. In the statement of Lemma 7, we are amplifying the privacy by using the subsampling as well as shuffling ideas.

Observe that the shuffler first chooses uniformly at random k clients of the available m clients. Then, each client samples her local dataset $\mathcal{D}_i$ by choosing uniformly at random $s = 1$ data points out of the available r data points. This two-steps sampling procedure is not the same as choosing uniformly at random ks data points from the entire dataset $\mathcal{D}$[12]. So, we cannot directly apply the amplification by subsampling result stated in Lemma 11. Thus, we derive a new privacy proof to compute the privacy parameters of the mechanism $\mathcal{M}_t$ under non-uniform sampling. Conisder two neighboring datasets $\mathcal{D} = \bigcup_{i=1}^m \mathcal{D}_i$, $\mathcal{D}' = \mathcal{D}_1' \bigcup \bigcup_{i=2}^m \mathcal{D}_i$ that are differerent only in the first data point at the first client d_{11}. The main idea of the proof is to split the probability distribution of the output of the mechaism $\mathcal{M}_t$ into a summation of four conditional probabilities dependeing on the event whether the first client is picked or not and the first client pick the first data point or not (Please, see (41)). We use the biparite graph to get the relation between these events, where each vertix corresponds to one of the possible outputs of the sampling procedure, and each edge connects two neighboring vertices. See Appendix B for more details.

[12]For example, when $s = 1$, the probability to observe two data points from the same cleint is zero in our sampling procedure, while observing these two data pointw has non-zero probability in the uniform sampling of the entire dataset $\mathcal{D}$.

Note that the Algorithm $\mathcal{A}_{cldp}$ is a sequence of T adaptive mechanisms $\mathcal{M}_1, \ldots, \mathcal{M}_T$, where each $\mathcal{M}_t$ for $t \in [T]$ satisfies the privacy guarantee stated in Lemma 7. Now, we invoke the strong composition stated in Lemma 10 to obtain the privacy guarantee of the algorithm $\mathcal{A}_{cldp}$. We can conclude that for any $\delta' > 0$, $\mathcal{A}_{cldp}$ is (ϵ, δ)-DP for

$$\epsilon = \sqrt{2T \log\left(1/\delta'\right)}\bar{\epsilon} + T\bar{\epsilon}\left(e^{\bar{\epsilon}} - 1\right), \quad \delta = qT\tilde{\delta} + \delta',$$

where $\bar{\epsilon}$ is from Lemma 7. We have from Lemma 10 that if $\bar{\epsilon} = \mathcal{O}\left(\sqrt{\frac{\log(1/\delta')}{T}}\right)$, then $\epsilon = \mathcal{O}\left(\bar{\epsilon}\sqrt{T \log\left(1/\delta'\right)}\right)$. If $\epsilon_0 = \mathcal{O}(1)$, then we can satisfy this condition on $\bar{\epsilon}$ by choosing $\epsilon_0 = \mathcal{O}\left(\sqrt{\frac{n \log(1/\delta')}{qT \log(1/\tilde{\delta})}}\right)$. By substituting the bound on $\bar{\epsilon} = \mathcal{O}\left(\epsilon_0 \sqrt{\frac{q \log\left(1/\tilde{\delta}\right)}{n}}\right)$ from Lemma 7, we have $\epsilon = \mathcal{O}\left(\epsilon_0 \sqrt{\frac{qT \log\left(1/\tilde{\delta}\right) \log(1/\delta')}{n}}\right)$. By setting $\tilde{\delta} = \frac{\delta}{2qT}$ and $\delta' = \frac{\delta}{2}$, we get $\epsilon_0 = \mathcal{O}\left(\sqrt{\frac{n \log(2/\delta)}{qT \log(2qT/\delta)}}\right)$ and $\epsilon = \mathcal{O}\left(\epsilon_0 \sqrt{\frac{qT \log(2qT/\delta) \log(2/\delta)}{n}}\right)$. This completes the proof of the privacy part of Theorem 1.

B. Proof of Theorem 1: Communication

The (ϵ_0, b)-CLDP mechanism $\mathcal{R}_p : \mathcal{X} \to \mathcal{Y}$ used in Algorithm 1 has output alphabet $\mathcal{Y} = \{1, 2, \ldots, B = 2^b\}$. So, the output of $\mathcal{R}_p$ on any input can be represented by b bits. Therefore, the naïve scheme for any client to send the s compressed and private gradients requires sb bits per iteration. We can reduce this communication cost by using the histogram trick from [35] which was applied in the context of non-private quantization. The idea is as follows. Since any client applies the *same* randomized mechanism $\mathcal{R}_p$ to the s gradients, the output of these s identical mechanisms can be represented accurately using the histogram of the s outputs, which takes value from the set $\mathcal{A}_B^s = \{(n_1, \ldots, n_B) : \sum_{j=1}^B n_j = s \text{ and } n_j \geq 0, \forall j \in [B]\}$. Since the cardinality of this set is $\binom{s+B-1}{s} \leq \left(\frac{e(s+B-1)}{s}\right)^s$, it requires at most $s\left(\log(e) + \log\left(\frac{s+B-1}{s}\right)\right)$ bits to send the s compressed gradients. Since the probability that the client is chosen at any time $t \in [T]$ is given by $\frac{k}{m}$, the expected number of bits per client in Algorithm $\mathcal{A}_{cldp}$ is given by $\frac{k}{m} \times T \times s\left(\log(e) + \log\left(\frac{s+B-1}{s}\right)\right)$ bits, where expectation is taken over the sampling of k out of m clients in all T iterations.

This completes the proof of the second part of Theorem 1.

C. Proof of Theorem 1 : Convergence

At iteration $t \in [T]$ of Algorithm 1, server averages the ks received compressed and privatized gradients and obtains $\overline{\mathbf{g}}_t = \frac{1}{ks} \sum_{i \in \mathcal{U}_t} \sum_{j \in \mathcal{S}_{it}} \mathbf{q}_t(d_{ij})$ (line 12 of Algorithm 1) and then updates the parameter vector as $\theta_{t+1} \leftarrow \prod_{\mathcal{C}}\left(\theta_t - \eta_t \overline{\mathbf{g}}_t\right)$. Here, $\mathbf{q}_t(d_{ij}) = \mathcal{R}_p\left(\nabla_{\theta_t} f(\theta_t; d_{ij})\right)$. Since the randomized mechanism $\mathcal{R}_p$ is unbiased, the average gradient $\overline{\mathbf{g}}_t$ is also unbiased, i.e., we have $\mathbb{E}\left[\overline{\mathbf{g}}_t\right] = \nabla_{\theta_t} F\left(\theta_t\right)$, where expectation is taken with respect to the random sampling of clients and the data points as well as the randomness of the mechanism $\mathcal{R}_p$. Now we show that $\overline{\mathbf{g}}_t$ has a bounded second moment.

Lemma 8. *For any $d \in \mathfrak{S}$, if the function $f(\theta;.) : \mathcal{C} \to \mathbb{R}$ is convex and L-Lipschitz continuous with respect to the ℓ_g-norm, which is the dual of ℓ_p-norm, then we have*

$$\mathbb{E}\|\overline{\mathbf{g}}_t\|_2^2 \leq L^2 \max\{d^{1-\frac{2}{p}}, 1\} \left(1 + \frac{cd}{qn}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2\right), \quad (31)$$

where c is a global constant: $c = 4$ if $p \in \{1, \infty\}$ and $c = 14$ if $p \notin \{1, \infty\}$.

Proof. Under the conditions of the lemma, we have from [32, Lemma 2.6] that $\|\nabla_\theta f(\theta; d)\| \leq L$ for all $d \in \mathfrak{S}$, which implies that $\nabla_\theta F(\theta) \leq L$. Thus, we have

$$\begin{aligned}
\mathbb{E}\|\overline{\mathbf{g}}_t\|_2^2 &= \|\mathbb{E}[\overline{\mathbf{g}}_t]\|_2^2 + \mathbb{E}\|\overline{\mathbf{g}}_t - \mathbb{E}[\overline{\mathbf{g}}_t]\|_2^2 \\
&\overset{(a)}{\leq} \max\{d^{1-\frac{2}{p}}, 1\}L^2 + \mathbb{E}\|\overline{\mathbf{g}}_t - \mathbb{E}[\overline{\mathbf{g}}_t]\|_2^2 \\
&\overset{(b)}{\leq} \max\{d^{1-\frac{2}{p}}, 1\}L^2 + \frac{cL^2 \max\{d^{2-\frac{2}{p}}, d\}}{ks}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2 \\
&\overset{(c)}{=} \max\{d^{1-\frac{2}{p}}, 1\}L^2 + \frac{cL^2 \max\{d^{2-\frac{2}{p}}, d\}}{qn}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2,
\end{aligned}$$

where c is a global constant, and $c = 4$ if $p \in \{1, \infty\}$ and $c = 14$ if $p \notin \{1, \infty\}$. Step (a) follows from the fact that $\|\nabla_{\theta_t} F(\theta_t)\| \leq L$ together with the norm inequality $\|\boldsymbol{u}\|_q \leq \|\boldsymbol{u}\|_p \leq d^{\frac{1}{p}-\frac{1}{q}}\|\boldsymbol{u}\|_q$ for $1 \leq p \leq q \leq \infty$. Step (b) follows from Corollary 1 with $\overline{p} = 1$, i.e., for any p-norm, we use the mechanism for ℓ_2-norm ball only (together with norm inequality) which gives the smallest variance. Step (c) uses $q = \frac{ks}{n}$. ∎

Now, we can use standard SGD convergence results for convex functions. In particular, we use the following result from [51].

Lemma 9 (SGD Convergence [51])**.** *Let $F(\theta)$ be a convex function, and the set $\mathcal{C}$ has diameter D. Consider a stochastic gradient descent algorithm $\theta_{t+1} \leftarrow \prod_{\mathcal{C}}(\theta_t - \eta_t \mathbf{g}_t)$, where $\mathbf{g}_t$ satisfies $\mathbb{E}[\mathbf{g}_t] = \nabla_{\theta_t} F(\theta_t)$ and $\mathbb{E}\|\mathbf{g}_t\|_2^2 \leq G^2$. By setting $\eta_t = \frac{D}{G\sqrt{t}}$, we get*

$$\mathbb{E}[F(\theta_T)] - F(\theta^*) \leq 2DG\frac{2+\log(T)}{\sqrt{T}} = \mathcal{O}\left(DG\frac{\log(T)}{\sqrt{T}}\right). \quad (32)$$

As shown in Lemma 8 and above that Algorithm 1 satisfies the premise of Lemma 9. Now, using the bound on G^2 from Lemma 8, we have that the output θ_T of Algorithm 1 satisfies

$$\begin{aligned}
\mathbb{E}[F(\theta_T)] - F(\theta^*) \leq \mathcal{O}\Bigg(&\frac{LD\log(T)\max\{d^{\frac{1}{2}-\frac{1}{p}}, 1\}}{\sqrt{T}} \\
&\left(1 + \sqrt{\frac{cd}{qn}}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\right)\Bigg),
\end{aligned} \quad (33)$$

where we used the inequality $\sqrt{1 + \frac{cd}{qn}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2} \leq \left(1 + \sqrt{\frac{cd}{qn}}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\right)$.

Note that if $\sqrt{\frac{cd}{qn}}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right) \leq \mathcal{O}(1)$, then we recover the convergence rate of vanilla SGD without privacy. So, the interesting case is when $\sqrt{\frac{cd}{qn}}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right) \geq \Omega(1)$, which gives

$$\begin{aligned}
\mathbb{E}[F(\theta_T)] - F(\theta^*) \leq \mathcal{O}\Bigg(&\frac{LD\log(T)\max\{d^{\frac{1}{2}-\frac{1}{p}}, 1\}}{\sqrt{T}} \\
&\sqrt{\frac{cd}{qn}}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\Bigg).
\end{aligned}$$

This completes the proof of the third part of Theorem 1.

VII. Discussion

In this paper we have developed a compressed, private optimization solution for a problem motivated by federated learning, where distributed clients jointly build a common learning model. The main technical contributions were developing order-optimal schemes for private mean-estimation and combining them with privacy amplification by sampling (of data and clients) as well as shuffling. We demonstrated that iterative application of this enables us to get the same privacy, optimization performance operating point as reported in [46], while obtaining order-wise improvement in the number of bits required, per iteration, thereby getting these communication gains for "free". Moreover, when the functions are L-Lipschitz with respect to the ℓ_2-norm, our scheme obtains the optimal excess risk of the central differential privacy obtained in [40], while operating in a distributed manner.

There are several open questions which are part of ongoing investigations. These include sharper privacy analyses for these schemes, which can improve the constants associated with the performance parameters. For example, suppose that we train a machine learning model on a dataset having 60000 clients, where each client has a single sample. After running our CLDP-SGD algorithm over $T = 1000$ iteration with $\epsilon_0 = 1$, $\delta = 10^{-5}$, and sampling $k = 5000$ clients at each iteration, we get a privacy parameter $\epsilon \approx 2$. We believe that the privacy parameters can be improved by analyzing the Renyi differential privacy of the shuffled model, which is an important open question of ongoing investigation. Extending these ideas to non-convex functions and examining their numerical performance for large-scale neural network models, is also of future interest.

References

[1] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *arXiv preprint arXiv:1912.04977*, 2019.

[2] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtarik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," in *NIPS Workshop on Private Multi-Party Machine Learning*, 2016. [Online]. Available: https://arxiv.org/abs/1610.05492

[3] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, pp. 1–19, 2019.

[4] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010*. Springer, 2010, pp. 177–186.

[5] S. L. Warner, "Randomized response: A survey technique for eliminating evasive answer bias," *Journal of the American Statistical Association*, vol. 60, no. 309, pp. 63–69, 1965.

[6] A. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke, "Privacy preserving mining of association rules," *Information Systems*, vol. 29, no. 4, pp. 343–364, 2004.

[7] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Local privacy and statistical minimax rates," in *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*. IEEE, 2013, pp. 429–438.

[8] A. Beimel, K. Nissim, and E. Omri, "Distributed private data analysis: Simultaneously solving how and what," in *Annual International Cryptology Conference*. Springer, 2008, pp. 451–468.

[9] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith, "What can we learn privately?" *SIAM Journal on Computing*, vol. 40, no. 3, pp. 793–826, 2011.

[10] P. Kairouz, K. Bonawitz, and D. Ramage, "Discrete distribution estimation under local privacy," in *International Conference on Machine Learning, ICML*, 2016, pp. 2436–2444.

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

[12] B. Ghazi, N. Golowich, R. Kumar, R. Pagh, and A. Velingker, "On the power of multiple anonymous messages." *IACR Cryptol. ePrint Arch.*, vol. 2019, p. 1382, 2019.

[13] B. Balle, J. Bell, A. Gascón, and K. Nissim, "Improved summation from shuffling," *arXiv preprint arXiv:1909.11225*, 2019.

[14] B. Ghazi, R. Pagh, and A. Velingker, "Scalable and differentially private distributed aggregation in the shuffled model," *arXiv preprint arXiv:1906.08320*, 2019.

[15] B. Balle, J. Bell, A. Gascon, and K. Nissim, "Differentially private summation with multi-message shuffling," *arXiv preprint arXiv:1906.09116*, 2019.

[16] B. Ghazi, R. Kumar, P. Manurangsi, and R. Pagh, "Private counting from anonymous messages: Near-optimal accuracy with vanishing communication overhead."

[17] A. Cheu, A. D. Smith, J. Ullman, D. Zeber, and M. Zhilyaev, "Distributed differential privacy via shuffling," in *Advances in Cryptology - EUROCRYPT 2019*, vol. 11476. Springer, 2019, pp. 375–403.

[18] B. Balle, J. Bell, A. Gascón, and K. Nissim, "The privacy blanket of the shuffle model," in *Annual International Cryptology Conference*. Springer, 2019, pp. 638–667.

[19] B. Balle, J. Bell, A. Gascon, and K. Nissim, "Private summation in the multi-message shuffle model," *arXiv preprint arXiv:2002.00817*, 2020.

[20] A. Beimel, S. P. Kasiviswanathan, and K. Nissim, "Bounds on the sample complexity for private learning and private data release," in *Theory of Cryptography Conference*. Springer, 2010, pp. 437–454.

[21] J. Ullman, "Cs7880. rigorous approaches to data privacy," 2017. [Online]. Available: http://www.ccs.neu.edu/home/jullman/cs7880s17/HW1sol.pdf

[22] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "Qsgd: Communication-efficient sgd via gradient quantization and encoding," in *Advances in Neural Information Processing Systems*, 2017, pp. 1709–1720.

[23] S. P. Karimireddy, Q. Rebjock, S. Stich, and M. Jaggi, "Error feedback fixes SignSGD and other gradient compression schemes," in *ICML*, 2019, pp. 3252–3261.

[24] W. Wen, C. Xu, F. Yan, C. Wu, Y. Wang, Y. Chen, and H. Li, "Terngrad: Ternary gradients to reduce communication in distributed deep learning," in *Advances in neural information processing systems*, 2017, pp. 1509–1519.

[25] S. U. Stich, J.-B. Cordonnier, and M. Jaggi, "Sparsified sgd with memory," in *Advances in Neural Information Processing Systems*, 2018, pp. 4447–4458.

[26] D. Alistarh, T. Hoefler, M. Johansson, N. Konstantinov, S. Khirirat, and C. Renggli, "The convergence of sparsified gradient methods," in *Advances in Neural Information Processing Systems*, 2018, pp. 5973–5983.

[27] A. Koloskova, T. Lin, S. U. Stich, and M. Jaggi, "Decentralized deep learning with arbitrary communication compression," in *International Conference on Learning Representations*, 2019.

[28] D. Basu, D. Data, C. Karakus, and S. Diggavi, "Qsparse-local-sgd: Distributed sgd with quantization, sparsification and local computations," in *Advances in Neural Information Processing Systems*, 2019, pp. 14 695–14 706.

[29] N. Singh, D. Data, J. George, and S. Diggavi, "Sparq-sgd: Event-triggered and compressed communication in decentralized stochastic optimization," *arXiv preprint arXiv:1910.14280*, 2019.

[30] ——, "Squarm-sgd: Communication-efficient momentum sgd for decentralized optimization," *arXiv preprint arXiv:2005.07041*, 2020.

[31] N. Agarwal, A. T. Suresh, F. X. X. Yu, S. Kumar, and B. McMahan, "cpsgd: Communication-efficient and differentially-private distributed sgd," in *Advances in Neural Information Processing Systems*, 2018, pp. 7564–7575.

[32] S. Shalev-Shwartz *et al.*, "Online learning and online convex optimization," *Foundations and Trends® in Machine Learning*, vol. 4, no. 2, pp. 107–194, 2012.

[33] A. T. Suresh, F. X. Yu, S. Kumar, and H. B. McMahan, "Distributed mean estimation with limited communication," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 3329–3337.

[34] V. Gandikota, D. Kane, R. K. Maity, and A. Mazumdar, "vqsgd: Vector quantized stochastic gradient descent," *arXiv preprint arXiv:1911.07971*, 2019.

[35] P. Mayekar and H. Tyagi, "Limits on gradient compression for stochastic optimization," *IEEE International Symposium on Information Theory (ISIT)*, 2020.

[36] J. Acharya, Z. Sun, and H. Zhang, "Hadamard response: Estimating distributions privately, efficiently, and with little communication," in *The 22nd International Conference on Artificial Intelligence and Statistics*, 2019, pp. 1120–1129.

[37] J. Acharya and Z. Sun, "Communication complexity in locally private distribution estimation and heavy hitters," in *Proceedings of the 36th International Conference on Machine Learning*, vol. 97. PMLR, 2019.

[38] W.-N. Chen, P. Kairouz, and A. Ozgur, "Breaking the communication-privacy-accuracy trilemma," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[39] K. Chaudhuri, C. Monteleoni, and A. D. Sarwate, "Differentially private empirical risk minimization." *Journal of Machine Learning Research*, vol. 12, no. 3, 2011.

[40] R. Bassily, A. Smith, and A. Thakurta, "Private empirical risk minimization: Efficient algorithms and tight error bounds," in *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*. IEEE, 2014, pp. 464–473.

[41] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proceedings of ACM CCS*, 2016, pp. 308–318.

[42] A. Bhowmick, J. Duchi, J. Freudiger, G. Kapoor, and R. Rogers, "Protection against reconstruction and its applications in private federated learning," *arXiv preprint arXiv:1812.00984*, 2018.

[43] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211–407, 2014.

[44] Y.-X. Wang, B. Balle, and S. P. Kasiviswanathan, "Subsampled rényi differential privacy and analytical moments accountant," in *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, 2019, pp. 1226–1235.

[45] V. Feldman, A. McMillan, and K. Talwar, "Hiding among the clones: A simple and nearly optimal analysis of privacy amplification by shuffling," *arXiv preprint arXiv:2012.12803*, December 2020.

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

[47] C. Dwork, G. N. Rothblum, and S. P. Vadhan, "Boosting and differential privacy," in *51th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2010, October 23-26, 2010, Las Vegas, Nevada, USA*, 2010, pp. 51–60.

[48] J. C. Duchi and R. Rogers, "Lower bounds for locally private estimation via communication complexity," in *Conference on Learning Theory (COLT)*, 2019, pp. 1161–1191.

[49] A. M. Girgis, D. Data, K. Chaudhuri, C. Fragouli, and S. Diggavi, "Successive refinement of privacy," *IEEE Journal on Selected Areas in Information Theory*, 2020.

[50] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Minimax optimal procedures for locally private estimation," *Journal of the American Statistical Association*, vol. 113, no. 521, pp. 182–201, 2018.

[51] O. Shamir and T. Zhang, "Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes," in *International conference on machine learning*, 2013, pp. 71–79.

[52] C. Dwork, F. McSherry, K. Nissim, and A. D. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography Conference (TCC)*, 2006, pp. 265–284.

[53] P. Kairouz, S. Oh, and P. Viswanath, "The composition theorem for differential privacy," in *International conference on machine learning*. PMLR, 2015, pp. 1376–1385.

Supplementary Material

APPENDIX A
BACKGROUND TOOLS

In this section, we state some preliminary definitions that we use throughout the paper and also state some results from literature. We state the formal definitions of (local) differential privacy (DP) in Section A-A and strong composition theorem for DP in Section A-B. As mentioned in Section I, we use subsampling and shuffling techniques for privacy amplification and we describe them in Section A-C.

A. Differential Privacy

In this section, we formally define local differential privacy (LDP) and (central) differential privacy (DP). First we recall the standard definition of LDP [9].

Definition 3 (Local Differential Privacy - LDP [9])**.** For $\epsilon_0 \geq 0$, a randomized mechanism $\mathcal{R} : \mathfrak{S} \to \mathcal{Y}$ is said to be ϵ_0-local differentially private (in short, ϵ_0-LDP), if for every pair of inputs $\boldsymbol{x}, \boldsymbol{x}' \in \mathfrak{S}$, we have

$$\Pr[\mathcal{R}(\boldsymbol{x}) \in \mathcal{S}] \leq \exp(\epsilon_0) \Pr[\mathcal{R}(\boldsymbol{x}') \in \mathcal{S}], \qquad \forall \mathcal{S} \subset \mathcal{Y}. \tag{34}$$

In our problem formulation, since each client has a communication budget on what it can send in each SGD iteration while keeping its data private, it would be convenient for us to define two parameter LDP with privacy and communication budget.

Definition 4 (Local Differential Privacy with Communication Budget - CLDP)**.** For $\epsilon_0 \geq 0$ and $b \in \mathbb{N}^+ := \{1, 2, 3, \ldots\}$, a randomized mechanism $\mathcal{R} : \mathfrak{S} \to \mathcal{Y}$ is said to be (ϵ_0, b)-communication-limited-local differentially private (in short, (ϵ_0, b)-CLDP), if for every pair of inputs $\boldsymbol{x}, \boldsymbol{x}' \in \mathfrak{S}$, we have

$$\Pr[\mathcal{R}(\boldsymbol{x}) = \boldsymbol{y}] \leq \exp(\epsilon_0) \Pr[\mathcal{R}(\boldsymbol{x}') = \boldsymbol{y}], \qquad \forall \boldsymbol{y} \in \mathcal{Y}. \tag{35}$$

Furthermore, the output of $\mathcal{R}$ can be represented using b bits.

Here, ϵ_0 captures the privacy level, lower the ϵ_0, higher the privacy. When we are not concerned about the communication budget, we succinctly denote the corresponding (ϵ_0, ∞)-CLDP, by its correspondence to the classical LDP as ϵ_0-LDP [9].

Let $\mathcal{D} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ denote a dataset comprising n points from $\mathfrak{S}$. We say that two datasets $\mathcal{D} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ and $\mathcal{D}' = \{\boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_n\}$ are neighboring if they differ in one data point. In other words, $\mathcal{D}$ and $\mathcal{D}'$ are neighboring if there exists an index $i \in [n]$ such that $\boldsymbol{x}_i \neq \boldsymbol{x}'_i$ and $\boldsymbol{x}_j = \boldsymbol{x}'_j$ for all $j \neq i$.

Definition 5 (Central Differential Privacy - DP [43], [52])**.** For $\epsilon, \delta \geq 0$, a randomized mechanism $\mathcal{M} : \mathfrak{S}^n \to \mathcal{Y}$ is said to be (ϵ, δ)-differentially private (in short, (ϵ, δ)-DP), if for all neighboring datasets $\mathcal{D}, \mathcal{D}' \in \mathfrak{S}^n$ and every subset $\mathcal{E} \subseteq \mathcal{Y}$, we have

$$\Pr\left[\mathcal{M}\left(\mathcal{D}\right) \in \mathcal{E}\right] \leq \exp(\epsilon) \Pr\left[\mathcal{M}\left(\mathcal{D}'\right) \in \mathcal{E}\right] + \delta. \tag{36}$$

Remark 5. For any ϵ_0-LDP mechanism $\mathcal{R} : \mathfrak{S} \to \mathcal{Y}$, it is easy to verify that the randomized mechanism $\mathcal{M} : \mathfrak{S}^n \to \mathcal{Y}$ defined by $\mathcal{M}\left(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\right) := \left(\mathcal{R}\left(\boldsymbol{x}_1\right), \ldots, \mathcal{R}\left(\boldsymbol{x}_n\right)\right)$ is $(\epsilon_0, 0)$-DP.

Remark 6. Note that in this paper we make a clear distinction between the notation used for central differential privacy, denoted by (ϵ, δ)-DP (see Definition 5), local differential privacy ϵ_0-LDP (see definition 3) and communication limited local differential privacy, denoted by (ϵ_0, b)-CLDP (see Definition 4).

The main objective of this paper is to make SGD differentially private and communication-efficient, suitable for federated learning. For that we compress and privatize gradients in each SGD iteration. Since the parameter vectors in any iteration depend on the previous iterations, so do the gradients, which makes this procedure a sequence of many adaptive DP mechanisms. We can calculate the final privacy guarantees achieved at the end of this procedure by using composition theorems.

B. Strong Composition [47]

Let $\mathcal{M}_1\left(\mathcal{I}_1, \mathcal{D}\right), \ldots, \mathcal{M}_T\left(\mathcal{I}_T, \mathcal{D}\right)$ be a sequence of T adaptive DP mechanisms, where $\mathcal{I}_i$ denotes the auxiliary input to the ith mechanism, which may depend on the previous mechanisms' outputs and the auxiliary inputs $\{(\mathcal{I}_j, \mathcal{M}_j(\mathcal{I}_j, \mathcal{D})) : j < i\}$. There are different composition theorems in literature to analyze the privacy guarantees of the composed mechanism $\mathcal{M}(\mathcal{D}) = \left(\mathcal{M}_1\left(\mathcal{I}_1, \mathcal{D}\right), \ldots, \mathcal{M}_T\left(\mathcal{I}_T, \mathcal{D}\right)\right)$ (see [47], [53] and references therein).

Dwork et al. [47] provided a strong composition theorem (which is stronger than the basic composition theorem in which the privacy parameters scale linearly with T) where the privacy parameter of the composition mechanism scales as $\sqrt{T}$ with some loss in δ. Below, we provide a formal statement of that result from [43].

Lemma 10 (Strong Composition [43, Theorem 3.20])**.** *Let $\mathcal{M}_1, \ldots, \mathcal{M}_T$ be T adaptive $(\overline{\epsilon}, \overline{\delta})$-DP mechanisms, where $\overline{\epsilon}, \overline{\delta} \geq 0$. Then, for any $\delta' > 0$, the composed mechanism $\mathcal{M} = (\mathcal{M}_1, \ldots, \mathcal{M}_T)$ is (ϵ, δ)-DP, where*

$$\epsilon = \sqrt{2T \log(1/\delta')}\overline{\epsilon} + T\overline{\epsilon}\left(e^{\overline{\epsilon}} - 1\right), \quad \delta = T\overline{\delta} + \delta'.$$

In particular, when $\overline{\epsilon} = \mathcal{O}\left(\sqrt{\frac{\log(1/\delta')}{T}}\right)$, we have $\epsilon = \mathcal{O}\left(\overline{\epsilon}\sqrt{T\log(1/\delta')}\right)$.

Note that training large-scale machine learning models (e.g., in deep learning) typically requires running SGD for millions of iterations, as the dimension of the model parameter is quite large. We can make it differentially private by adding noise to the gradients in each iteration, and appeal to the strong composition theorem to bound the privacy loss of the entire process (which in turn dictates the amount of noise to be added in each iteration).

C. Privacy Amplification

In this section, we describe the techniques that can be used for privacy amplification. The first one amplifies privacy by subsampling the data (to compute stochastic gradients) as well as the clients (as in FL), and the other one amplifies privacy by shuffling.

1) Privacy Amplification by Subsampling: Suppose we have a dataset $\mathcal{D}' = \{U_1, \ldots, U_{r_1}\} \in \mathfrak{S}^{r_1}$ consisting of r_1 elements from a universe $\mathfrak{S}$. A subsampling procedure takes a dataset $\mathcal{D}' \in \mathfrak{S}^{r_1}$ and subsamples without replacement a subset from it as formally defined below.

Definition 6 (Subsampling)**.** The subsampling operation $\mathrm{samp}_{r_1,r_2} : \mathfrak{S}^{r_1} \to \mathfrak{S}^{r_2}$ takes a dataset $\mathcal{D}' \in \mathfrak{S}^{r_1}$ as input and selects uniformly at random a subset $\mathcal{D}''$ of $r_2 \leq r_1$ elements from $\mathcal{D}'$. Note that each element of $\mathcal{D}'$ appears in $\mathcal{D}''$ with probability $q = \frac{r_2}{r_1}$.

The following result states that the above subsampling procedure amplifies the privacy guarantees of a DP mechanism.

Lemma 11 (Amplification by Subsampling [9])**.** *Let $\mathcal{M} : \mathfrak{S}^{r_2} \to \mathcal{V}$ be an (ϵ, δ)-DP mechanism. Then, the mechanism $\mathcal{M}' : \mathfrak{S}^{r_1} \to \mathcal{V}$ defined by $\mathcal{M}' = \mathcal{M} \circ \mathrm{samp}_{r_1,r_2}$ is (ϵ', δ')-DP, where $\epsilon' = \log(1 + q(e^{\epsilon} - 1))$ and $\delta' = q\delta$ with $q = \frac{r_2}{r_1}$. In particular, when $\epsilon < 1$, $\mathcal{M}'$ is $(\mathcal{O}(q\epsilon), q\delta)$-DP.*

Note that in the case of subsampling the data for computing stochastic gradients, where client i selects a mini-batch of size s from its local dataset $\mathcal{D}_i$ that has r data points, we take $\mathcal{D}' = \mathcal{D}_i$, $r_1 = r$, and $r_2 = s$. In the case of subsampling the clients, k clients are randomly selected from the m clients, we take $\mathcal{D}' = \{1, 2, \ldots, m\}$, $r_1 = m$, and $r_2 = k$. An important point is that such a sub-sampling is not uniform overall (i.e., this does not imply that any subset of ks data points is chosen with equal probability) and we cannot directly apply the above result. We need to revisit the proof of Lemma 11 to adapt it to our case, and we do it in Lemma 7, which is proved in Appendix B. In fact, the proof of Lemma 7 is more general than just adapting the amplification by subsampling to our setting, it also incorporates the amplification by shuffling, which is crucial for obtaining strong privacy guarantees. We describe it next.

2) Privacy Amplification by Shuffling: Consider a set of m clients, where client $i \in [m]$ has a data $\boldsymbol{x}_i \in \mathfrak{S}$. Let $\mathcal{R} : \mathfrak{S} \to \mathcal{Y}$ be an ϵ_0-LDP mechanism. The i-th client applies $\mathcal{R}$ on her data $\boldsymbol{x}_i$ to get a private message $\boldsymbol{y}_i = \mathcal{R}(\boldsymbol{x}_i)$. There is a secure shuffler $\mathcal{H}_m : \mathcal{Y}^m \to \mathcal{Y}^m$ that receives the set of m messages $(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m)$ and generates the same set of messages in a uniformly random order.

The following lemma states that the shuffling amplifies the privacy of an LDP mechanism by a factor of $\frac{1}{\sqrt{m}}$.

Lemma 12 (Amplification by Shuffling)**.** *Let $\mathcal{R}$ be an ϵ_0-LDP mechanism. Then, the mechanism $\mathcal{M}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m) := \mathcal{H}_m \circ (\mathcal{R}(\boldsymbol{x}_1), \ldots, \mathcal{R}(\boldsymbol{x}_m))$ satisfies (ϵ, δ)-differential privacy, where*

1) [18, Corollary 5.3.1]. *If $\epsilon_0 \leq \frac{\log(m/\log(1/\delta))}{2}$, then for any $\delta > 0$, we have $\epsilon = \mathcal{O}\left(\min\{\epsilon_0, 1\} e^{\epsilon_0} \sqrt{\frac{\log(1/\delta)}{m}}\right)$.*
2) [11, Corollary 9]. *If $\epsilon_0 < \frac{1}{2}$, then for any $\delta \in (0, \frac{1}{100})$ and $m \geq 1000$, we have $\epsilon = 12\epsilon_0 \sqrt{\frac{\log(1/\delta)}{m}}$.*

In our proposed algorithm, only $k \leq m$ clients send messages and each client sends a mini-batch of s gradients. So, in total, shuffler applies the shuffling operation on ks gradients. In our algorithm, though sampling and shuffling are applied one after another (first k clients are sampled, then each client samples s data points, and then shuffling of these ks data points is performed), we analyze the privacy amplification we get using both of these techniques by analyzing them together; see Lemma 7 proved in Appendix B.

D. ℓ_p Geometry in Optimization

In this section, we give an example showing that why it is important to analyse the convergence of the SGD algorithm for Lipschitz convex function under several ℓ_p geometries.

Example 1. Let $z = (x, y)$ be a data point and $\theta \in \mathbb{R}^{d+1}$ represent the model parameters to be discovered in the learning process. We can define a mapping function of the data point by $\phi(z) = [x, y]$, where $x \in \mathbb{R}^d$ is a feature vector and y is a scalar, resulting in a feature map in the dimension of $\mathbb{R}^{d+1}$. For a linear model, $\theta^T\phi(z)$, we can observe that $|\theta^T\phi(z)| \leq \|\theta\|_q \|\phi(z)\|_p$ from Holder's inequality, where ℓ_p is the dual norm of ℓ_q, i.e., $\frac{1}{q} + \frac{1}{p} = 1$. Now suppose, as an example, the dataset has bounded ℓ_p-norm, i.e., $\max_z \|\phi(z)\|_p \leq L_1$. Thus, the function $g(\theta, z) = \theta^T\phi(z)$ is L_1-Lipschitz continuous with respect to ℓ_q-norm. To show this, observe that

$$|g(\theta_1, z) - g(\theta_2, z)| = |(\theta_1 - \theta_2)^T\phi(z)| \leq \|\phi(z)\|_p \|\theta_1 - \theta_2\|_q \leq L_1\|\theta_1 - \theta_2\|_q$$

Suppose our loss function is in the form $f(\theta, z) = h(g(\theta, z)) = h(\theta^T\phi(z))$, where the function $h : \mathbb{R} \to \mathbb{R}$ is L_2-Lipschitz continuous. Thus, the loss function $f(\theta, z)$ is L_2L_1-Lipschitz continuous with respect to ℓ_q-norm. For example, if the dataset has bounded ℓ_∞ norm, then the loss function will be Lipschitz continuous with respect to ℓ_1 norm, and hence, the gradient of the loss function has bounded ℓ_∞ norm. Observe that the class of the functions of the form $f(\theta, z) = h(g(\theta, z))$ contains the soft-max loss and hinge loss function. Thus, it is relevant to work with the general ℓ_p spaces.

APPENDIX B
PROOF OF LEMMA 7

Recall that the input dataset at client $i \in [m]$ is denoted by $\mathcal{D}_i = \{d_{i1}, d_{i2}, \ldots, d_{ir}\} \in \mathfrak{S}^r$ and $\mathcal{D} = \bigcup_{i=1}^m \mathcal{D}_i$ denotes the entire dataset. Recall from (30) that the mechanism $\mathcal{M}_t$ on input dataset $\mathcal{D}$ can be defined as:

$$\mathcal{M}_t(\mathcal{D}) = \mathcal{H}_{ks} \circ \mathrm{samp}_{m,k}\left(\mathcal{G}_1, \ldots, \mathcal{G}_m\right), \tag{37}$$

where $\mathcal{G}_i = \mathrm{samp}_{r,s}\left(\mathcal{R}(\boldsymbol{x}_{i1}^t), \ldots, \mathcal{R}(\boldsymbol{x}_{ir}^t)\right)$ and $\boldsymbol{x}_{ij}^t = \nabla_{\theta_t} f(\theta_t; d_{ij}), \forall i \in [m], j \in [r]$. We define a mechanism $\mathcal{Z}\left(\mathcal{D}^{(t)}\right) = \mathcal{H}_{ks}\left(\mathcal{R}\left(\boldsymbol{x}_1^t\right), \ldots, \mathcal{R}\left(\boldsymbol{x}_{ks}^t\right)\right)$ which is a shuffling of ks outputs of local mechanism $\mathcal{R}$, where $\mathcal{D}^{(t)}$ denotes an arbitrary set of ks data points and we index $\boldsymbol{x}_i^t$'s from $i = 1$ to ks just for convenience. From the amplification by shuffling result [18, Corollary 5.3.1] (also see Lemma 12), the mechanism $\mathcal{Z}$ is $(\tilde{\epsilon}, \tilde{\delta})$-DP, where $\tilde{\delta} > 0$ is arbitrary, and, if $\epsilon_0 \leq \frac{\log\left(ks/\log\left(1/\tilde{\delta}\right)\right)}{2}$, then

$$\tilde{\epsilon} = \mathcal{O}\left(\min\{\epsilon_0, 1\}e^{\epsilon_0}\sqrt{\frac{\log\left(1/\tilde{\delta}\right)}{ks}}\right). \tag{38}$$

Furthermore, when $\epsilon_0 = \mathcal{O}(1)$, we get $\tilde{\epsilon} = \mathcal{O}\left(\epsilon_0\sqrt{\frac{\log\left(1/\tilde{\delta}\right)}{ks}}\right)$.

Let $\mathcal{T} \subseteq \{1, \ldots, m\}$ denote the identities of the k clients chosen at iteration t, and for $i \in \mathcal{T}$, let $\mathcal{T}_i \subseteq \{1, \ldots, r\}$ denote the identities of the s data points chosen at client i at iteration t.[13] For any $\mathcal{T} \in \binom{[m]}{k}$ and $\mathcal{T}_i \in \binom{[r]}{s}, i \in \mathcal{T}$, define $\overline{\mathcal{T}} = (\mathcal{T}, \mathcal{T}_i, i \in \mathcal{T})$, $\mathcal{D}^{\mathcal{T}_i} = \{d_j : j \in \mathcal{T}_i\}$ for $i \in \mathcal{T}$, and $\mathcal{D}^{\overline{\mathcal{T}}} = \{\mathcal{D}^{\mathcal{T}_i} : i \in \mathcal{T}\}$. Note that $\mathcal{T}$ and $\mathcal{T}_i, i \in \mathcal{T}$ are random sets, where randomness is due to the sampling of clients and of data points, respectively. The mechanism $\mathcal{M}_t$ can be equivalently written as $\mathcal{M}_t = \mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}})$.

Observe that our sampling strategy is different from subsampling of choosing a uniformly random subset of ks data points from the entire dataset $\mathcal{D}$. Thus, we revisit the proof of privacy amplification by subsampling (see, for example, [21]) – which is for uniform sampling – to compute the privacy parameters of the mechanism $\mathcal{M}_t$, where sampling is non-uniform. Define a dataset $\mathcal{D}' = (\mathcal{D}_1')\bigcup(\cup_{i=2}^m \mathcal{D}_i) \in \mathfrak{S}^n$, where $\mathcal{D}_1' = \{d_{11}', d_{12}, \ldots, d_{1r}\}$ is different from the dataset $\mathcal{D}_1$ in the first data point d_{11}. Note that $\mathcal{D}$ and $\mathcal{D}'$ are neighboring datasets – where, we assume, without loss of generality, that the differing elements are d_{11} and d_{11}'.

In order to show that $\mathcal{M}_t$ is $(\bar{\epsilon}, \bar{\delta})$-DP, we need show that for an arbitrary subset $\mathcal{S}$ of the range of $\mathcal{M}_t$, we have

$$\Pr\left[\mathcal{M}_t\left(\mathcal{D}\right) \in \mathcal{S}\right] \leq e^{\bar{\epsilon}}\Pr\left[\mathcal{M}_t\left(\mathcal{D}'\right) \in \mathcal{S}\right] + \bar{\delta} \tag{39}$$

$$\Pr\left[\mathcal{M}_t\left(\mathcal{D}'\right) \in \mathcal{S}\right] \leq e^{\bar{\epsilon}}\Pr\left[\mathcal{M}_t\left(\mathcal{D}\right) \in \mathcal{S}\right] + \bar{\delta} \tag{40}$$

Note that both (39) and (40) are symmetric, so it suffices to prove only one of them. We prove (39) below.

Let $q = \frac{ks}{mr}$. We define conditional probabilities as follows:

$$A_{11} = \Pr\left[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}}) \in \mathcal{S} | 1 \in \mathcal{T} \text{ and } 1 \in \mathcal{T}_1\right]$$

[13] Though $\mathcal{T}$ and $\mathcal{T}_i, i \in \mathcal{T}$ may be different at different iteration t, for notational convenience, we suppress the dependence on t here.

$$A'_{11} = \Pr\left[\mathcal{Z}(\mathcal{D}'^{\overline{\mathcal{T}}}) \in \mathcal{S} | 1 \in \mathcal{T} \text{ and } 1 \in \mathcal{T}_1\right] \tag{41}$$
$$A_{10} = \Pr\left[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}}) \in \mathcal{S} | 1 \in \mathcal{T} \text{ and } 1 \notin \mathcal{T}_1\right] = \Pr\left[\mathcal{Z}(\mathcal{D}'^{\overline{\mathcal{T}}}) \in \mathcal{S} | 1 \in \mathcal{T} \text{ and } 1 \notin \mathcal{T}_1\right]$$
$$A_0 = \Pr\left[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}}) \in \mathcal{S} | 1 \notin \mathcal{T}\right] = \Pr\left[\mathcal{Z}(\mathcal{D}'^{\overline{\mathcal{T}}}) \in \mathcal{S} | 1 \notin \mathcal{T}\right]$$

Let $q_1 = \frac{k}{m}$ and $q_2 = \frac{s}{r}$, and hence $q = q_1 q_2$. Thus, we have

$$\Pr\left[\mathcal{M}_t\left(\mathcal{D}\right) \in \mathcal{S}\right] = qA_{11} + q_1\left(1 - q_2\right)A_{10} + \left(1 - q_1\right)A_0$$
$$\Pr\left[\mathcal{M}_t\left(\mathcal{D}'\right) \in \mathcal{S}\right] = qA'_{11} + q_1\left(1 - q_2\right)A_{10} + \left(1 - q_1\right)A_0$$

Note that the mechanism $\mathcal{Z}$ is $(\tilde{\epsilon}, \tilde{\delta})$-DP. Therefore, we have

$$A_{11} \leq e^{\tilde{\epsilon}} A'_{11} + \tilde{\delta} \tag{42}$$
$$A_{11} \leq e^{\tilde{\epsilon}} A_{10} + \tilde{\delta} \tag{43}$$

Here (42) is straightforward, but proving (43) requires a combinatorial argument, which we give at the end of this proof.

We prove (39) separately for two cases, first when $s = 1$ and other when $s > 1$; k is arbitrary in both cases.

A. *For $s = 1$ and arbitrary $k \in [m]$*

Since the mechanism $\mathcal{Z}$ is $(\tilde{\epsilon}, \tilde{\delta})$-DP, in addition to (42)-(43), since $s = 1$, we also have the following inequality:

$$A_{11} \leq e^{\tilde{\epsilon}} A_0 + \tilde{\delta} \tag{44}$$

Similar to (43), proving (44) requires a combinatorial argument, which we will give at the end of this proof. Note that (44) only holds for $s = 1$ and may not hold for arbitrary s.

Inequalities (42)-(44) together imply $A_{11} \leq e^{\tilde{\epsilon}} \min\{A'_{11}, A_{10}, A_0\} + \tilde{\delta}$. Now we prove (39) for $\overline{\epsilon} = \ln(1 + q(e^{\tilde{\epsilon}} - 1)$ and $\overline{\delta} = q\tilde{\delta}$. Note that when $s = 1$, we have $q_1 = \frac{k}{m}$, $q_2 = \frac{1}{r}$, and $q = \frac{k}{mr}$.

$$\begin{aligned}
&\Pr\left[\mathcal{M}_t\left(\mathcal{D}\right) \in \mathcal{S}\right] = qA_{11} + q_1\left(1 - q_2\right)A_{10} + \left(1 - q_1\right)A_0 \\
&\leq q\left(e^{\tilde{\epsilon}} \min\{A'_{11}, A_{10}, A_0\} + \tilde{\delta}\right) + q_1\left(1 - q_2\right)A_{10} + \left(1 - q_1\right)A_0 \\
&= q\left(\left(e^{\tilde{\epsilon}} - 1\right)\min\{A'_{11}, A_{10}, A_0\} + \min\{A'_{11}, A_{10}, A_0\}\right) + q_1\left(1 - q_2\right)A_{10} + \left(1 - q_1\right)A_0 + q\tilde{\delta} \\
&\overset{(a)}{\leq} q(e^{\tilde{\epsilon}} - 1)\min\{A'_{11}, A_{10}, A_0\} + qA'_{11} + q_1\left(1 - q_2\right)A_{10} + \left(1 - q_1\right)A_0 + q\tilde{\delta} \\
&\overset{(b)}{\leq} q(e^{\tilde{\epsilon}} - 1)\left(qA'_{11} + q_1(1 - q_2)A_{10} + (1 - q_1)A_0\right)) + \left(qA'_{11} + q_1\left(1 - q_2\right)A_{10} + \left(1 - q_1\right)A_0\right) + q\tilde{\delta} \\
&= \left(1 + q\left(e^{\tilde{\epsilon}} - 1\right)\right)\left(qA'_{11} + q_1\left(1 - q_2\right)A_{10} + \left(1 - q_1\right)A_0\right) + q\tilde{\delta} \\
&= e^{\ln(1+q(e^{\tilde{\epsilon}}-1))}\Pr\left[\mathcal{M}_t\left(\mathcal{D}'\right) \in \mathcal{S}\right] + q\tilde{\delta}.
\end{aligned}$$

Here, (a) follows from $\min\{A'_{11}, A_{10}, A_0\} \leq A'_{11}$, and (b) follows from the fact that minimum is upper-bounded by the convex combination. By substituting the value of $\tilde{\epsilon}$ from (38) and using $ks = qn$, we get that for $\epsilon_0 = \mathcal{O}\left(1\right)$, we have $\overline{\epsilon} = \mathcal{O}\left(\epsilon_0\sqrt{\frac{q\log\left(1/\tilde{\delta}\right)}{n}}\right)$.

B. *For $s > 1$ and arbitrary $k \in [m]$*

Note that (42)-(43) together imply $A_{11} \leq e^{\tilde{\epsilon}} \min\{A'_{11}, A_{10}\} + \tilde{\delta}$. Now we prove (39) for $\overline{\epsilon} = \ln(1 + q_2(e^{\tilde{\epsilon}} - 1))$ and $\overline{\delta} = q\tilde{\delta}$.

$$\begin{aligned}
&\Pr\left[\mathcal{M}_t\left(\mathcal{D}\right) \in \mathcal{S}\right] = qA_{11} + q_1(1 - q_2)A_{10} + (1 - q_1)A_0 \\
&\leq q\left(e^{\tilde{\epsilon}} \min\{A'_{11}, A_{10}\} + \tilde{\delta}\right) + q_1(1 - q_2)A_{10} + (1 - q_1)A_0 \\
&= q\left(\left(e^{\tilde{\epsilon}} - 1\right)\min\{A'_{11}, A_{10}\} + \min\{A'_{11}, A_{10}\}\right) + q_1(1 - q_2)A_{10} + (1 - q_1)A_0 + q\tilde{\delta} \\
&\overset{(a)}{\leq} q\left(e^{\tilde{\epsilon}} - 1\right)\min\{A'_{11}, A_{10}\}\big) + qA'_{11} + q_1(1 - q_2)A_{10} + (1 - q_1)A_0 + q\tilde{\delta} \\
&\overset{(b)}{\leq} q\left((e^{\tilde{\epsilon}} - 1)(q_2 A'_{11} + (1 - q_2)A_{10})\right) + (qA'_{11} + q_1(1 - q_2)A_{10} + (1 - q_1)A_0) + q\tilde{\delta} \\
&= q_2\left((e^{\tilde{\epsilon}} - 1)(q_1 q_2 A'_{11} + q_1(1 - q_2)A_{10})\right) + (qA'_{11} + q_1(1 - q_2)A_{10} + (1 - q_1)A_0) + q\tilde{\delta} \\
&\overset{(c)}{\leq} q_2\left((e^{\tilde{\epsilon}} - 1)(qA'_{11} + q_1(1 - q_2)A_{10}) + (1 - q_1)A_0\right) + (qA'_{11} + q_1(1 - q_2)A_{10} + (1 - q_1)A_0) + q\tilde{\delta} \\
&= \left(1 + q_2\left((e^{\tilde{\epsilon}} - 1)\right)\right)(qA'_{11} + q_1(1 - q_2)A_{10}) + (1 - q_1)A_0\big) + q\tilde{\delta}
\end{aligned}$$

$$= e^{\ln(1+q_2(e^{\tilde{\epsilon}}-1))} \Pr\left[\mathcal{M}_t\left(\mathcal{D}'\right) \in \mathcal{S}\right] + q\tilde{\delta}$$

Here, (a) follows from $\min\{A'_{11}, A_{10}\} \leq A'_{11}$, (b) follows from the fact that minimum is upper-bounded by the convex combination, and (c) holds because $(1-q_1)A_0 \geq 0$. By substituting the value of $\tilde{\epsilon}$ from (38) and using $ks = qn$, we get that for $\epsilon_0 = \mathcal{O}(1)$, we have $\overline{\epsilon} = \mathcal{O}\left(\epsilon_0 \sqrt{\frac{q_2 \log\left(1/\tilde{\delta}\right)}{q_1 n}}\right)$. Note that when $q_1 = 1$ (i.e., we select all the clients in each iteration), then this gives the desired privacy amplification of $q = q_2$.

The proof of Lemma 7 is complete, except for that we have to prove (43) and (44). Before proving (43) and (44), we state an important remark about the privacy amplification in both the cases.

Remark 7. Note that when $s = 1$ and $\epsilon_0 = \mathcal{O}(1)$, we have $\overline{\epsilon} = \ln(1 + q(e^{\tilde{\epsilon}} - 1)) = \mathcal{O}(q\tilde{\epsilon})$. So we get a privacy amplification by a factor of $q = \frac{ks}{mr}$ – the sampling probability of each data point from the entire dataset. Here, we get a privacy amplification from both types of sampling, of clients as well of data points.

On the other hand, when $s > 1$ and $\epsilon_0 = \mathcal{O}(1)$, we have $\overline{\epsilon} = \ln(1 + q_2(e^{\tilde{\epsilon}} - 1)) = \mathcal{O}(q_2\tilde{\epsilon})$, which, unlike the case of $s = 1$, only gives the privacy amplification by a factor of $q_2 = \frac{s}{r}$ – the sampling probability of each data point from a client. So, unlike the case of $s = 1$, here we only get a privacy amplification from sampling of data points, not from sampling of clients. Note that when $k = m$ and any $s \in [r]$ (which implies $q_1 = 1$ and $q = q_2$), we have $\overline{\epsilon} = \mathcal{O}\left(\epsilon_0 \sqrt{\frac{q_2 \log\left(1/\tilde{\delta}\right)}{n}}\right)$, which gives the desired amplification when we select all the clients in each iteration.

Proof of (43)**.** First note that the number of subsets $\mathcal{T}_1 \subset [r]$ such that $|\mathcal{T}_1| = s, 1 \in \mathcal{T}_1$ is equal to $\binom{r-1}{s-1}$ and the number of subsets $\mathcal{T}_1 \subset [r]$ such that $|\mathcal{T}_1| = s, 1 \notin \mathcal{T}_1$ is equal to $\binom{r-1}{s}$. It is easy to verify that $(r-s)\binom{r-1}{s-1} = s\binom{r-1}{s}$.

Consider the following bipartite graph $G = (V_1 \cup V_2, E)$, where the left vertex set V_1 has $\binom{r-1}{s-1}$ vertices, one for each configuration of $\mathcal{T}_1 \subset [r]$ such that $|\mathcal{T}_1| = s, 1 \in \mathcal{T}_1$, the right vertex set V_2 has $\binom{r-1}{s}$ vertices, one for each configuration of $\mathcal{T}_1 \subset [r]$ such that $|\mathcal{T}_1| = s, 1 \notin \mathcal{T}_1$, and the edge set E contains all the edges between neighboring vertices, i.e., if $(\boldsymbol{u}, \boldsymbol{v}) \in V_1 \times V_2$ is such that $\boldsymbol{u}$ and $\boldsymbol{v}$ differ in only one element, then $(\boldsymbol{u}, \boldsymbol{v}) \in E$. Observe that each vertex of V_1 has $(r-s)$ neighbors in V_2 – the neighbors of $\mathcal{T}_1 \in V_1$ will be $\{(\mathcal{T}_1 \setminus \{1\}) \cup \{i\} : i \in [m] \setminus \mathcal{T}_1\} \subset V_2$. Similarly, each vertex of V_2 has s neighbors in V_1 – the neighbors of $\mathcal{T}_1 \in V_2$ will be $\{(\mathcal{T}_1 \setminus \{i\}) \cup \{1\} : i \in \mathcal{T}_1\} \subset V_1$.

Now, fix any $\mathcal{T} \in \binom{[m]}{k}$ s.t. $1 \in \mathcal{T}$, and for $i \in \mathcal{T} \setminus \{1\}$, fix any $\mathcal{T}_i \in \binom{[r]}{s}$, and consider an arbitrary $(\boldsymbol{u}, \boldsymbol{v}) \in E$. Since the mechanism $\mathcal{Z}$ is $(\tilde{\epsilon}, \tilde{\delta})$-DP, we have

$$\Pr\left[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}}) \in \mathcal{S} | 1 \in \mathcal{T}, \mathcal{T}_1 = \boldsymbol{u}, \mathcal{T}_i, i \in \mathcal{T} \setminus \{1\}\right] \leq e^{\tilde{\epsilon}} \Pr\left[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}}) \in \mathcal{S} | 1 \in \mathcal{T}, \mathcal{T}_1 = \boldsymbol{v}, \mathcal{T}_i, i \in \mathcal{T} \setminus \{1\}\right] + \tilde{\delta}. \tag{45}$$

Now we are ready to prove (43).

$$\begin{aligned}
A_{11} &= \Pr\left[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}}) \in \mathcal{S} | 1 \in \mathcal{T} \text{ and } 1 \in \mathcal{T}_1\right] \\
&= \sum_{\substack{\mathcal{T} \in \binom{[m]}{k}: 1 \in \mathcal{T} \\ \mathcal{T}_1 \in \binom{[r]}{s}: 1 \in \mathcal{T}_1 \\ \mathcal{T}_i \in \binom{[r]}{s} \text{ for } i \in \mathcal{T} \setminus \{1\}}} \Pr[\mathcal{T}, \mathcal{T}_i, i \in \mathcal{T} | 1 \in \mathcal{T} \text{ and } 1 \in \mathcal{T}_1] \Pr[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}}) \in \mathcal{S} | \mathcal{T}, \mathcal{T}_1, \ldots, \mathcal{T}_m] \\
&\overset{(a)}{=} \sum_{\substack{\mathcal{T} \in \binom{[m]}{k}: 1 \in \mathcal{T} \\ \mathcal{T}_i \in \binom{[r]}{s} \text{ for } i \in \mathcal{T} \setminus \{1\}}} \Pr[\mathcal{T}, \mathcal{T}_i, i \in \mathcal{T} \setminus \{1\} | 1 \in \mathcal{T}] \sum_{\mathcal{T}_1 \in \binom{[r]}{s}: 1 \in \mathcal{T}_1} \Pr[\mathcal{T}_1 | 1 \in \mathcal{T}_1] \Pr[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}}) \in \mathcal{S} | \mathcal{T}, \mathcal{T}_1, \ldots, \mathcal{T}_m] \\
&= \sum_{\substack{\mathcal{T} \in \binom{[m]}{k}: 1 \in \mathcal{T} \\ \mathcal{T}_i \in \binom{[r]}{s} \text{ for } i \in \mathcal{T} \setminus \{1\}}} \Pr[\mathcal{T}, \mathcal{T}_i, i \in \mathcal{T} \setminus \{1\} | 1 \in \mathcal{T}] \frac{1}{(r-s)\binom{r-1}{s-1}} \sum_{\mathcal{T}_1 \in \binom{[r]}{s}: 1 \in \mathcal{T}_1} (r-s) \Pr[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}}) \in \mathcal{S} | \mathcal{T}, \mathcal{T}_1, \ldots, \mathcal{T}_m] \\
&= \sum_{\substack{\mathcal{T} \in \binom{[m]}{k}: 1 \in \mathcal{T} \\ \mathcal{T}_i \in \binom{[r]}{s} \text{ for } i \in \mathcal{T} \setminus \{1\}}} \Pr[\mathcal{T}, \mathcal{T}_i, i \in \mathcal{T} \setminus \{1\} | 1 \in \mathcal{T}] \frac{1}{s\binom{r-1}{s}} \sum_{\mathcal{T}_1 \in \binom{[r]}{s}: 1 \in \mathcal{T}_1} (r-s) \Pr[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}}) \in \mathcal{S} | \mathcal{T}, \mathcal{T}_1, \ldots, \mathcal{T}_m] \\
&\overset{(b)}{\leq} \sum_{\substack{\mathcal{T} \in \binom{[m]}{k}: 1 \in \mathcal{T} \\ \mathcal{T}_i \in \binom{[r]}{s} \text{ for } i \in \mathcal{T} \setminus \{1\}}} \Pr[\mathcal{T}, \mathcal{T}_i, i \in \mathcal{T} \setminus \{1\} | 1 \in \mathcal{T}] \frac{1}{s\binom{r-1}{s}} \sum_{\mathcal{T}_1 \in \binom{[r]}{s}: 1 \notin \mathcal{T}_1} s\left(e^{\tilde{\epsilon}} \Pr[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}}) \in \mathcal{S} | \mathcal{T}, \mathcal{T}_1, \ldots, \mathcal{T}_m] + \tilde{\delta}\right) \\
&= \sum_{\substack{\mathcal{T} \in \binom{[m]}{k}: 1 \in \mathcal{T} \\ \mathcal{T}_i \in \binom{[r]}{s} \text{ for } i \in \mathcal{T} \setminus \{1\}}} \Pr[\mathcal{T}, \mathcal{T}_i, i \in \mathcal{T} \setminus \{1\} | 1 \in \mathcal{T}] \sum_{\mathcal{T}_1 \in \binom{[r]}{s}: 1 \notin \mathcal{T}_1} \Pr[\mathcal{T}_1 | 1 \notin \mathcal{T}_1] \left(e^{\tilde{\epsilon}} \Pr[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}}) \in \mathcal{S} | \mathcal{T}, \mathcal{T}_1, \ldots, \mathcal{T}_m] + \tilde{\delta}\right)
\end{aligned}$$

$$\begin{aligned}
&\overset{(c)}{=} \sum_{\substack{\mathcal{T}\in\binom{[m]}{k}:1\in\mathcal{T}\\ \mathcal{T}_1\in\binom{[r]}{s}:1\notin\mathcal{T}_1\\ \mathcal{T}_i\in\binom{[r]}{s}\text{ for }i\in\mathcal{T}\setminus\{1\}}} \Pr[\mathcal{T},\mathcal{T}_i,i\in\mathcal{T}|1\in\mathcal{T}\text{ and }1\notin\mathcal{T}_1]\left(e^{\tilde{\epsilon}}\Pr[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}})\in\mathcal{S}|\mathcal{T},\mathcal{T}_1,\ldots,\mathcal{T}_m]+\tilde{\delta}\right)\\
&\le e^{\tilde{\epsilon}}\Pr\left[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}})\in\mathcal{S}|1\in\mathcal{T}\text{ and }1\notin\mathcal{T}_1\right]+\tilde{\delta}\\
&= e^{\tilde{\epsilon}}A_{10}+\tilde{\delta}.
\end{aligned}$$

Here, (a) and (c) follow from the fact that clients sample the data points independent of each other, and (b) follows from (45) together with the fact that there are $(r-s)\binom{r-1}{s-1}=s\binom{r-1}{s}$ edges in the bipartite graph $G=(V_1\cup V_2,E)$, where degree of vertices in V_1 is $(r-s)$ and degree of vertices in V_2 is s.

Proof of (44)**.** First note that the number of subsets $\mathcal{T}\in[m]$ such that $|\mathcal{T}|=k,1\in\mathcal{T}$ is equal to $\binom{m-1}{k-1}$ and the number of subsets $\mathcal{T}\subset[m]$ such that $|\mathcal{T}|=k,1\notin\mathcal{T}$ is equal to $\binom{m-1}{k}$. It is easy to verify that $(m-k)\binom{m-1}{k-1}=k\binom{m-1}{k}$.

Consider the following bipartite graph $G=(V_1\cup V_2,E)$, where the left vertex set V_1 has $\binom{m-1}{k-1}r^{k-1}$ vertices, one for each configuration of $(\mathcal{T},\mathcal{T}_i:i\in\mathcal{T})$ such that $\mathcal{T}\subset[m]$, $|\mathcal{T}|=k,1\in\mathcal{T}$ and $\mathcal{T}_1=1$, the right vertex set V_2 has $\binom{m-1}{k}r^k$ vertices, one for each configuration of $(\mathcal{T},\mathcal{T}_i:i\in\mathcal{T})$ such that $\mathcal{T}\subset[m]$, $|\mathcal{T}|=k,1\notin\mathcal{T}$, and the edge set E contains all the edges between neighboring vertices, i.e., if $(\boldsymbol{u},\boldsymbol{v})\in V_1\times V_2$ is such that $\boldsymbol{u}$ and $\boldsymbol{v}$ differ in only one element, then $(\boldsymbol{u},\boldsymbol{v})\in E$. Observe that each vertex of V_1 has $r(m-k)$ neighbors in V_2. Similarly, each vertex of V_2 has k neighbors in V_1.

Consider an arbitrary edge $(\boldsymbol{u},\boldsymbol{v})\in E$. By construction, there exists $\mathcal{T}\in\binom{[m]}{k}$ with $1\in\mathcal{T}$ and $\mathcal{T}_i\in[r],i\in\mathcal{T}$ such that $\boldsymbol{u}=(\mathcal{T},\mathcal{T}_i:i\in\mathcal{T})$ and $\mathcal{T}'\in\binom{[m]}{k}$ with $1\notin\mathcal{T}'$ and $\mathcal{T}'_i\in[r],i\in\mathcal{T}'$ such that $\boldsymbol{v}=(\mathcal{T}',\mathcal{T}'_i:i\in\mathcal{T}')$. Note that, since $(\boldsymbol{u},\boldsymbol{v})\in E$, $(\mathcal{T}_i:i\in\mathcal{T})$ and $(\mathcal{T}'_i:i\in\mathcal{T}')$ have $k-1$ elements common. Now, since the mechanism $\mathcal{Z}$ is $(\tilde{\epsilon},\tilde{\delta})$-DP, we have

$$\Pr\left[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}})\in\mathcal{S}|\mathcal{T},\mathcal{T}_i,i\in\mathcal{T}\right]\le e^{\tilde{\epsilon}}\Pr\left[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}'}})\in\mathcal{S}|\mathcal{T}',\mathcal{T}'_i,i\in\mathcal{T}'\right]+\tilde{\delta}. \tag{46}$$

Now we are ready to prove (44).

$$\begin{aligned}
A_{11} &= \Pr\left[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}})\in\mathcal{S}|1\in\mathcal{T}\text{ and }\mathcal{T}_1=1\right]\\
&= \sum_{\substack{\mathcal{T}\in\binom{[m]}{k}:1\in\mathcal{T}\\ \mathcal{T}_i\in[r]\text{ for }i\in\mathcal{T}:\mathcal{T}_1=1}} \Pr[\mathcal{T},\mathcal{T}_i,i\in\mathcal{T}|1\in\mathcal{T}\text{ and }\mathcal{T}_1=1]\Pr[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}})\in\mathcal{S}|\mathcal{T},\mathcal{T}_i,i\in\mathcal{T}]\\
&= \frac{1}{\binom{m-1}{k-1}r^{k-1}} \sum_{\substack{\mathcal{T}\in\binom{[m]}{k}:1\in\mathcal{T}\\ \mathcal{T}_i\in[r]\text{ for }i\in\mathcal{T}:\mathcal{T}_1=1}} \Pr[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}})\in\mathcal{S}|\mathcal{T},\mathcal{T}_i,i\in\mathcal{T}]\\
&= \frac{1}{(m-k)\binom{m-1}{k-1}r^{k}} \sum_{\substack{\mathcal{T}\in\binom{[m]}{k}:1\in\mathcal{T}\\ \mathcal{T}_i\in[r]\text{ for }i\in\mathcal{T}:\mathcal{T}_1=1}} r(m-k)\Pr[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}})\in\mathcal{S}|\mathcal{T},\mathcal{T}_i,i\in\mathcal{T}]\\
&\overset{(a)}{=} \frac{1}{k\binom{m-1}{k}r^{k}} \sum_{\substack{\mathcal{T}\in\binom{[m]}{k}:1\in\mathcal{T}\\ \mathcal{T}_i\in[r]\text{ for }i\in\mathcal{T}:\mathcal{T}_1=1}} r(m-k)\Pr[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}})\in\mathcal{S}|\mathcal{T},\mathcal{T}_i,i\in\mathcal{T}]\\
&\overset{(b)}{\le} \frac{1}{k\binom{m-1}{k}r^{k}} \sum_{\substack{\mathcal{T}\in\binom{[m]}{k}:1\notin\mathcal{T}\\ \mathcal{T}_i\in[r]\text{ for }i\in\mathcal{T}}} k\left(e^{\epsilon}\Pr[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}})\in\mathcal{S}|\mathcal{T},\mathcal{T}_i,i\in\mathcal{T}]+\tilde{\delta}\right)\\
&= \frac{1}{\binom{m-1}{k}r^{k}} \sum_{\substack{\mathcal{T}\in\binom{[m]}{k}:1\notin\mathcal{T}\\ \mathcal{T}_i\in[r]\text{ for }i\in\mathcal{T}}} \left(e^{\epsilon}\Pr[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}})\in\mathcal{S}|\mathcal{T},\mathcal{T}_i,i\in\mathcal{T}]+\tilde{\delta}\right)\\
&= \sum_{\substack{\mathcal{T}\in\binom{[m]}{k}:1\notin\mathcal{T}\\ \mathcal{T}_i\in[r]\text{ for }i\in\mathcal{T}}} \Pr[\mathcal{T},\mathcal{T}_i,i\in\mathcal{T}|1\notin\mathcal{T}]\left(e^{\epsilon}\Pr[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}})\in\mathcal{S}|\mathcal{T},\mathcal{T}_i,i\in\mathcal{T}]+\tilde{\delta}\right)\\
&= e^{\tilde{\epsilon}}\Pr\left[\mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}})\in\mathcal{S}|1\notin\mathcal{T}\right]+\tilde{\delta}\\
&= e^{\tilde{\epsilon}}A_0+\tilde{\delta}
\end{aligned}$$

Here, (a) uses $(m-k)\binom{m-1}{k-1}=k\binom{m-1}{k}$, and (b) follows from (46) together with the fact that there are $r(m-k)\binom{m-1}{k-1}r^{k-1}=k\binom{m-1}{k}r^k$ edges in the bipartite graph $G=(V_1\cup V_2,E)$, where degree of vertices in V_1 is $r(m-k)$ and degree of vertices in V_2 is k.

This completes the proof of Lemma 7.

APPENDIX C
MINIMAX RISK ESTIMATION

Lemma 13. *For the minimax problems* (5) *and* (6)*, the optimal estimator $\widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)$ is a deterministic function. In other words, the randomized decoder does not help in reducing the minimax risk.*

Proof. Towards a contradiction, suppose that the optimal estimator $\widehat{\boldsymbol{x}}$ is a randomized decoder defined as follows. For given clients' responses $\boldsymbol{y}^n$, let the probabilistic estimator generate an estimate $\widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)$ whose mean and trace of the covariance matrix are given by $\boldsymbol{\mu}_{\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)}=\mathbb{E}\left[\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)\right]$ and $\sigma^2_{\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)}=\mathbb{E}\left[\left\|\widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)-\mu_{\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)}\right\|_2^2\left|Y^n\right.\right]$, respectively, where expectation is taken with respect to the randomization of the decoder, conditioned of Y^n.

$$\begin{aligned}
\mathbb{E}\left[\left\|\overline{\boldsymbol{x}}-\widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2\left|\boldsymbol{y}^n\right.\right] &= \mathbb{E}\left[\left\|\overline{\boldsymbol{x}}-\boldsymbol{\mu}_{\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)}+\boldsymbol{\mu}_{\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)}-\widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2\left|\boldsymbol{y}^n\right.\right] \\
&= \mathbb{E}\left[\left\|\overline{\boldsymbol{x}}-\boldsymbol{\mu}_{\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)}\right\|_2^2\left|\boldsymbol{y}^n\right.\right]+\mathbb{E}\left[\left\|\boldsymbol{\mu}_{\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)}-\widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2\left|\boldsymbol{y}^n\right.\right] \\
&\quad +2\mathbb{E}\left\langle\overline{\boldsymbol{x}}-\boldsymbol{\mu}_{\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)},\boldsymbol{\mu}_{\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)}-\widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\left|\boldsymbol{y}^n\right.\right\rangle \\
&\overset{\text{(a)}}{=} \mathbb{E}\left[\left\|\overline{\boldsymbol{x}}-\boldsymbol{\mu}_{\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)}\right\|_2^2\left|\boldsymbol{y}^n\right.\right]+\sigma^2_{\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)} \\
&> \mathbb{E}\left[\left\|\overline{\boldsymbol{x}}-\boldsymbol{\mu}_{\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)}\right\|_2^2\left|\boldsymbol{y}^n\right.\right]
\end{aligned}$$

In (a), we used that $\boldsymbol{\mu}_{\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)}=\mathbb{E}\left[\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)\right]$ to eliminate the last term. Similarly, we can prove that $\mathbb{E}\left[\left\|\boldsymbol{\mu}_{\mathbf{q}}-\widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2\middle|\boldsymbol{y}^n\right]>\mathbb{E}\left[\left\|\boldsymbol{\mu}_{\mathbf{q}}-\boldsymbol{\mu}_{\boldsymbol{y}^n}\right\|_2^2\middle|\boldsymbol{y}^n\right]$. Hence, the deterministic estimator $\widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)=\mu_{\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)}$ has a lower minimax risk than the probabilistic estimator. ∎

APPENDIX D
COMPRESSED AND PRIVATE MEAN ESTIMATION

A. Achievability for ℓ_1-norm Ball: Proof of Theorem 5

Lemma (Restating Lemma 2)**.** *The mechanism $\mathcal{R}_1$ presented in Algorithm 2 satisfies the following properties:*

1) *$\mathcal{R}_1$ is $(\epsilon_0, \log(d)+1)$-LDP and requires only 1-bit of communication using public-randomness.*
2) *$\mathcal{R}_1$ is unbiased and has bounded variance, i.e., for every $\boldsymbol{x}\in\mathcal{B}_1^d(a)$, we have*

$$\mathbb{E}\left[\mathcal{R}_1\left(\boldsymbol{x}\right)\right]=\boldsymbol{x} \quad \textit{and} \quad \mathbb{E}\|\mathcal{R}_1\left(\boldsymbol{x}\right)-\boldsymbol{x}\|_2^2\leq d\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2.$$

Proof. We show these properties one-by-one below.

1) Observe that the output of the mechanism $\mathcal{R}_1$ can be represented using the index $j\in[d]$ and one bit of the sign of $\{\pm a\mathbf{H}_d\left(j\right)\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\}$. Hence, it requires only $\log\left(d\right)+1$ bits for communication. Furthermore, the randomness $j\sim\mathsf{Unif}\left[d\right]$ is independent of the input $\boldsymbol{x}$. Thus, if the client has access to a public randomness j, then the client needs only to send one bit to represent its sign. Now, we show that the mechanism $\mathcal{R}_1$ is ϵ_0-LDP. Let $\mathcal{Z}=\left\{\pm a\mathbf{H}_d(j)\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right):j=1,2,\ldots,d\right\}$ denote all possible $2d$ outputs of the mechanism $\mathcal{R}_1$. We get

$$\begin{aligned}
\sup_{\boldsymbol{x},\boldsymbol{x}'\in\mathcal{B}_1^d(a)}\sup_{\boldsymbol{z}\in\mathcal{Z}}\frac{\Pr[\mathcal{R}_1(\boldsymbol{x})=\boldsymbol{z}]}{\Pr[\mathcal{R}_1(\boldsymbol{x}')=\boldsymbol{z}]} &\leq \sup_{\boldsymbol{x},\boldsymbol{x}'\in\mathcal{B}_1^d(a)}\frac{\frac{1}{d}\sum_{j=1}^d\left(\frac{1}{2}+\frac{\sqrt{d}|y_j|}{2a}\frac{e^{\epsilon_0}-1}{e^{\epsilon_0}+1}\right)}{\frac{1}{d}\sum_{j=1}^d\left(\frac{1}{2}-\frac{\sqrt{d}|y'_j|}{2a}\frac{e^{\epsilon_0}-1}{e^{\epsilon_0}+1}\right)} \\
&= \sup_{\boldsymbol{x},\boldsymbol{x}'\in\mathcal{B}_1^d(a)}\frac{\frac{1}{d}\sum_{j=1}^d\left(a(e^{\epsilon_0}+1)+\sqrt{d}|y_j|(e^{\epsilon_0}-1)\right)}{\frac{1}{d}\sum_{j=1}^d\left(a(e^{\epsilon_0}+1)-\sqrt{d}|y'_j|(e^{\epsilon_0}-1)\right)} \\
&\overset{\text{(a)}}{\leq} \frac{2ae^{\epsilon_0}}{2a}=e^{\epsilon_0},
\end{aligned}$$

   where (a) uses the fact that for every $j\in[d]$, we have $|y_j|\leq a/\sqrt{d}$ and $|y'_j|\leq a/\sqrt{d}$.
2) Fix an arbitrary $\boldsymbol{x}\in\mathcal{B}_1^d\left(a\right)$.

$$\text{Unbiasedness:}\quad \mathbb{E}\left[\mathcal{R}_1\left(\boldsymbol{x}\right)\right]=\frac{1}{d}\sum_{j=1}^d a\mathbf{H}_d\left(j\right)\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\left(\frac{\sqrt{d}y_j}{a}\frac{e^{\epsilon_0}-1}{e^{\epsilon_0}+1}\right)$$

$$= \frac{1}{d}\sum_{j=1}^{d} \mathbf{H}_d\left(j\right)\sqrt{d}y_j \overset{\text{(b)}}{=} \frac{1}{d}\sum_{j=1}^{d} \mathbf{H}_d\left(j\right)\mathbf{H}_d^T(j)\boldsymbol{x} \overset{\text{(c)}}{=} \boldsymbol{x}$$

where (b) uses $\boldsymbol{y} = \frac{1}{\sqrt{d}}\mathbf{H}_d\boldsymbol{x}$ and (c) uses $\sum_{j=1}^{d}\mathbf{H}_d(j)\mathbf{H}_d^T(j) = \mathbf{H}_d\mathbf{H}_d^T = d\mathbf{I}_d$.

$$\begin{aligned}\text{Bounded variance:}\quad \mathbb{E}\|\mathcal{R}_1\left(\boldsymbol{x}\right) - \boldsymbol{x}\|_2^2 &\le \mathbb{E}\|\mathcal{R}_1(\boldsymbol{x})\|^2 = \mathbb{E}[\mathcal{R}_1(\boldsymbol{x})^T\mathcal{R}_1(\boldsymbol{x})] \\ &= \frac{1}{d}\sum_{j=1}^{d} a^2\mathbf{H}_d(j)^T\mathbf{H}_d(j)\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2 \\ &= a^2 d\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2 \qquad \text{(Since } \mathbf{H}_d(j)^T\mathbf{H}_d(j) = d, \forall j \in [d]\text{)}\end{aligned}$$

This completes the proof of Lemma 2. ■

B. Achievability for ℓ_2-norm Ball: Proof of Theorem 6

Lemma (Restating Lemma 5)**.** *The mechanism $\mathcal{R}_2$ presented in Algorithm 3 satisfies the following properties, where $\epsilon_0 > 0$:*

1) *$\mathcal{R}_2$ is $(\epsilon_0, d(\log(e)+1))$-LDP.*
2) *$\mathcal{R}_2$ is unbiased and has bounded variance, i.e., for every $\boldsymbol{x} \in \mathcal{B}_2^d\left(a\right)$, we have*

$$\mathbb{E}\left[\mathcal{R}_2\left(\boldsymbol{x}\right)\right] = \boldsymbol{x} \quad \textit{and} \quad \mathbb{E}\|\mathcal{R}_2\left(\boldsymbol{x}\right) - \boldsymbol{x}\|_2^2 \le 6a^2 d\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2.$$

Proof. We prove these properties one-by-one below.

1) It was shown by Duchi et al. [50, Section 4.2.3] that Priv is an ϵ_0-LDP mechanism. Now, since $\mathcal{R}_2 = \mathsf{Quan} \circ \mathsf{Priv}$ is a post-processing of a differentially-private mechanism Priv and post-processing preserves differential privacy, we have that $\mathcal{R}_2$ is also ϵ_0-LDP. The claim that $\mathcal{R}_2$ uses $d(\log(e)+1)$ bits of communication follows because $\mathcal{R}_2$ outputs the result of Quan, which produces an output which can be represented using $d(\log(e)+1)$ bits; see [35].
2) Unbiasedness of $\mathcal{R}_2$ follows because $\mathcal{R}_2 = \mathsf{Quan} \circ \mathsf{Priv}$ and both Priv and Quan are unbiased. To prove that variance is bounded, fix an $\boldsymbol{x} \in \mathcal{B}_2^d\left(a\right)$.

$$\begin{aligned}\mathbb{E}\|\mathcal{R}_2(\boldsymbol{x}) - \boldsymbol{x}\|_2^2 &= \mathbb{E}\|\mathsf{Quan}\left(\mathsf{Priv}(\boldsymbol{x})\right) - \boldsymbol{x}\|_2^2 \\ &= \mathbb{E}\|\mathsf{Quan}\left(\mathsf{Priv}(\boldsymbol{x})\right) - \mathsf{Priv}(\boldsymbol{x}) + \mathsf{Priv}(\boldsymbol{x}) - \boldsymbol{x}\|_2^2 \\ &\overset{\text{(a)}}{=} \mathbb{E}\|\mathsf{Quan}\left(\mathsf{Priv}(\boldsymbol{x})\right) - \mathsf{Priv}(\boldsymbol{x})\|_2^2 + \mathbb{E}\|\mathsf{Priv}(\boldsymbol{x}) - \boldsymbol{x}\|_2^2 \\ &\overset{\text{(b)}}{\le} 2\|\mathsf{Priv}(\boldsymbol{x})\|^2 + \mathbb{E}\|\mathsf{Priv}(\boldsymbol{x})\|^2 \\ &\overset{\text{(c)}}{\le} 3\|\mathsf{Priv}(\boldsymbol{x})\|^2 \overset{\text{(d)}}{\le} 6d\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2.\end{aligned}$$

In (a) we used the fact that Quan and Priv are unbiased, which implies that the cross multiplication term is zero. In (b) we used Lemma 4 to write $\mathbb{E}\|\mathsf{Quan}\left(\mathsf{Priv}(\boldsymbol{x})\right) - \mathsf{Priv}(\boldsymbol{x})\|_2^2 \le 2\|\mathsf{Priv}(\boldsymbol{x})\|^2$ and used the unbiasedness of Priv together with the fact that variance is bounded by the second moment to write $\mathbb{E}\|\mathsf{Priv}(\boldsymbol{x}) - \boldsymbol{x}\|_2^2 \le \mathbb{E}\|\mathsf{Priv}(\boldsymbol{x})\|_2^2$. In (c) we used that the length of Priv on any input remains fixed, i.e., $\mathbb{E}\|\mathsf{Priv}(\boldsymbol{x})\|^2 = \|\mathsf{Priv}(\boldsymbol{x})\|^2 = M^2$ (where M is from the line 4 of Algorithm 4) holds for any $\boldsymbol{x} \in \mathcal{B}_2^d(a)$. In (d) we used the bound on $\|\mathsf{Priv}(\boldsymbol{x})\|_2^2$ from Lemma 3.

This completes the proof of Lemma 5. ■

C. Achievability for ℓ_∞-norm Ball: Proof of Theorem 7

Lemma (Restating Lemma 6)**.** *The mechanism $\mathcal{R}_\infty$ presented in Algorithm 6 satisfies the following properties:*

1) *$\mathcal{R}_\infty$ is $(\epsilon_0, \log\left(d\right)+1)$-LDP and requires only 1-bit of communication using public-randomness.*
2) *$\mathcal{R}_\infty$ is unbiased and has bounded variance, i.e., for every $\boldsymbol{x} \in \mathcal{B}_\infty^d\left(a\right)$, we have*

$$\mathbb{E}\left[\mathcal{R}_\infty\left(\boldsymbol{x}\right)\right] = \boldsymbol{x} \quad \textit{and} \quad \mathbb{E}\|\mathcal{R}_\infty\left(\boldsymbol{x}\right) - \boldsymbol{x}\|_2^2 \le a^2 d^2\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2.$$

Proof. We prove these properties one-by-one below.

1) Observe that the output of the mechanism $\mathcal{R}_\infty$ can be represented using the index $j \in [d]$ and one bit for the sign of $\left\{\pm ad\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\boldsymbol{e}_j\right\}$. Hence, it requires only $\log\left(d\right)+1$ bits for communication. Furthermore, the randomness $j \sim \mathsf{Unif}\left[d\right]$ is independent of the input $\boldsymbol{x}$. Thus, if the client has access to a public randomness j, then the client needs only to send

one bit for its sign. Now, we show that the mechanism $\mathcal{R}_\infty$ is ϵ_0-LDP. Let $\mathcal{Z} = \left\{ \pm ad\big(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\big)\boldsymbol{e}_j : j = 1, 2, \ldots, d\right\}$ denote all possible $2d$ outputs of the mechanism $\mathcal{R}_\infty$. We get

$$\sup_{\boldsymbol{x},\boldsymbol{x}'\in\mathcal{B}_\infty^d(a)} \sup_{\boldsymbol{z}\in\mathcal{Z}} \frac{\Pr\left[\mathcal{R}_\infty\left(\boldsymbol{x}\right)=\boldsymbol{z}\right]}{\Pr\left[\mathcal{R}_\infty\left(\boldsymbol{x}\right)=\boldsymbol{z}\right]} \leq \sup_{\boldsymbol{x},\boldsymbol{x}'\in\mathcal{B}_\infty^d(a)} \frac{\frac{1}{d}\sum_{i=1}^d \left(\frac{1}{2} + \frac{|x_j|}{2a}\frac{e^{\epsilon_0}-1}{e^{\epsilon_0}+1}\right)}{\frac{1}{d}\sum_{i=1}^d \left(\frac{1}{2} - \frac{|x'_j|}{2a}\frac{e^{\epsilon_0}-1}{e^{\epsilon_0}+1}\right)} \tag{47}$$

$$= \sup_{\boldsymbol{x},\boldsymbol{x}'\in\mathcal{B}_\infty^d} \frac{\frac{1}{d}\sum_{i=1}^d \left(a(e^{\epsilon_0}+1) + |x_j|(e^{\epsilon_0}-1)\right)}{\frac{1}{d}\sum_{i=1}^d \left(a(e^{\epsilon_0}+1) - |x'_j|(e^{\epsilon_0}-1)\right)} \tag{48}$$

$$\overset{\text{(a)}}{\leq} \frac{2ae^{\epsilon_0}}{2a} = e^{\epsilon_0}, \tag{49}$$

where in (a) we used the fact that for every $j \in [d]$, we have $|x_j| \leq a$ and $|x'_j| \leq a$.

2) Fix an arbitrary $\boldsymbol{x} \in \mathcal{B}_\infty^d$.

$$\text{Unbiasedness:} \quad \mathbb{E}\left[\mathcal{R}_\infty\left(\boldsymbol{x}\right)\right] = \frac{1}{d}\sum_{j=1}^d \boldsymbol{e}_j ad\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\left(\frac{x_j}{a}\frac{e^{\epsilon_0}-1}{e^{\epsilon_0}+1}\right)$$
$$= \sum_{j=1}^d \boldsymbol{e}_j x_j$$
$$= \boldsymbol{x}$$

$$\text{Bounded variance:} \quad \mathbb{E}\|\mathcal{R}_\infty(\boldsymbol{x}) - \boldsymbol{x}\|_2^2 \leq \mathbb{E}\|\mathcal{R}_\infty(\boldsymbol{x})\|^2 = \mathbb{E}[\mathcal{R}_\infty(\boldsymbol{x})^T\mathcal{R}_\infty(\boldsymbol{x})]$$
$$= \frac{1}{d}\sum_{j=1}^d a^2d^2\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2$$
$$= a^2d^2\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2$$

This completes the proof of Lemma 6. ■