Shuffled Model of Differential Privacy in Federated Learning

Antonious M. Girgis
UCLA

Deepesh Data
UCLA

Suhas Diggavi
UCLA

Peter Kairouz
Google

Ananda Theertha Suresh
Google

Abstract

We consider a distributed empirical risk minimization (ERM) optimization problem with communication efficiency and privacy requirements, motivated by the federated learning (FL) framework. We propose a distributed communication-efficient and local differentially private stochastic gradient descent (CLDP-SGD) algorithm and analyze its communication, privacy, and convergence trade-offs. Since each iteration of the CLDP-SGD aggregates the client-side local gradients, we develop (optimal) communication-efficient schemes for mean estimation for several ℓ_p spaces under local differential privacy (LDP). To overcome performance limitation of LDP, CLDP-SGD takes advantage of the inherent privacy amplification provided by client subsampling and data subsampling at each selected client (through SGD) as well as the recently developed shuffled model of privacy. For convex loss functions, we prove that the proposed CLDP-SGD algorithm matches the known lower bounds on the *centralized* private ERM while using a finite number of bits per iteration for each client, *i.e.,* effectively getting communication efficiency for "free". We also provide preliminary experimental results supporting the theory.

1 Introduction

We consider a federated learning (FL) framework (*e.g.,* Kairouz et al. [2019]), where data is generated across m *distributed* clients, and the server builds a machine learning model by solving the empirical risk minimization (ERM) problem:

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

where $F_i(\theta)$ is a local loss function dependent on the local dataset $\mathcal{D}_i$ at client i, comprising r data points, and $\mathcal{C} \subset \mathbb{R}^d$ is a closed convex set, where d denotes the model dimension; see Section 3 for details on the setup. The goal is to solve this problem while satisfying the FL requirements: (i) give privacy guarantees on the data $\mathcal{D}_i$ at client i, (ii) compress (as efficiently as possible) the communication between clients and the server, and (iii) work with a dynamic client population in each round of communication between the server and the clients – in FL only a small fraction of clients are sampled at each communication round; see Figure 1.

A challenge is that local differential privacy (LDP),*e.g.,* [Beimel et al., 2008, Warner, 1965], is known to give poor learning performance [Duchi et al., 2013, Kairouz et al., 2016, Kasiviswanathan et al., 2011]. Recently, a new privacy framework, the *shuffled model* [Balle et al., 2019a,b,c, 2020a, Cheu et al., 2019, Erlingsson et al., 2019, Ghazi et al., 2019a,b, 2020], enables significantly better privacy-utility performance by amplifying privacy (scaling with the number of clients as $\frac{1}{\sqrt{m}}$ with respect to LDP) through anonymization. Another technique to amplify privacy is through randomized subsampling [Beimel et al., 2010]. This naturally arises in a stochastic gradient descent (SGD) framework for optimizing (1), since clients do mini-batch sampling of local data; moreover clients themselves are sampled in each iteration motivated by the FL setup.

In this paper, we analyse privacy amplification for the FL problem using both forms of amplification: shuffling and subsampling (data and clients). Note that privacy amplification by subsampling (both data and clients)

Proceedings of the 24[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

happens automatically,[1] while the secure shuffling is performed explicitly adding an additional layer of privacy transferring local privacy guarantees to central privacy guarantees.

Another important aspect is communication efficiency instantiated through compression of the gradients computed by each active client. There has been a significant recent progress on this topic (see [Alistarh et al., 2017, 2018, Basu et al., 2019, Karimireddy et al., 2019, Singh et al., 2019, 2020, Stich et al., 2018] and references therein). However, there has been less work in combining privacy and compression in the optimization/learning framework, with the notable exception of [Agarwal et al., 2018], which we will elaborate on shortly. One question that we address is whether one pays a price to do compression in terms of the privacy-performance trade-off.

In this paper, we solve the main problem of learning a model with communication constraints, with reasonable learning performance while giving strong privacy guarantees. 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 privacy guarantees for clients using a shuffled framework. Here are our main contributions.

• We prove that one can get communication efficiency "for free" by demonstrating schemes that use $O(\log d)$ bits per gradient to obtain the same privacy-utility operating point as full precision gradient exchange.[2]

• 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 a privacy proof 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. For this, we develop new private and compressed vector mean estimation techniques in a minimax estimation framework, that are (order optimal) under several ℓ_p geometries. We develop both lower bounds and matching schemes for this problem. These results may also be of independent interest.

[1] 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.

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

Related work: There has been a lot of work on privacy in the context of FL (see [Kairouz et al., 2019] and references therein) and also on compression for private mean estimation (see [Acharya and Sun, 2019, Balle et al., 2019c, 2020a, Cheu et al., 2019] and references therein). Our focus in this paper is on *distributed* learning with local differential privacy guarantees, which has fewer results, especially in the shuffled privacy framework. We give an extensive account of the related work in Appendix A of the supplementary material, and due to space constraints we will focus on the two most related papers to our work [Agarwal et al., 2018, Erlingsson et al., 2020], which we describe below.

Erlingsson et al. [2020] proposed a distributed local differentially private gradient descent algorithm, where all clients participate in each iteration. They use LDP on gradients as well as the shuffled framework [Balle et al., 2019c]. However, their proposed algorithm sends the full-precision gradient without compression. Our work is different from [Erlingsson et al., 2020] in multiple ways: (i) we propose a communication efficient mechanism for each client that requires $O(\log d)$ bits per client, which can be significant for large d; (ii) our algorithm performs data sampling (using SGD at each client) and client sampling *i.e.*, not all clients are selected at each iteration, as motivated by the FL setup. This requires a careful combination of compression and privacy analysis; see Remark 2, where we recover the convergence result of [Erlingsson et al., 2020] as a special case of our general results.

Agarwal et al. [2018] proposed a communication-efficient algorithm for learning models with local differential privacy. They proposed cp-SGD, a communication efficient algorithm, where clients need to send $O(\log(1 + \frac{d}{n}\epsilon_0^2) + \log\log\log\frac{nd}{\epsilon_0\delta})$ bits of communication *per coordinate*, *i.e.,* $O\left(d\left\{\log(1 + \frac{d}{n}\epsilon_0^2) + \log\log\log\frac{nd}{\epsilon_0\delta}\right\}\right)$ bits per gradient to achieve the same local differential privacy guarantees of the Gaussian mechanism. In contrast, we achieve better compression in terms of number of bits per gradient, and our framework converts the LDP algorithm to central differential privacy guarantees.

Paper organization. In Section 2, we establish some background results. In Section 3, we set up the problem including the formulation for private mean-estimation and describe our algorithm. We state our results in Section 4 and also give some interpretations. In Section 5, we provide brief proof outlines for some of the results. Section 6 provides preliminary evaluation of the algorithm in terms of communication-privacy-performance operating points on the MNIST dataset. Many of the proof details as well as some additional results are provided in the supplementary material.

2 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 B of the supplementary material.

Since we are interested in communication constrained privacy of the client, we define a two parameter LDP with privacy and communication budget, generalizing the standard LDP privacy definition (see Definition 3 in Appendix B.1 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 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 [Kasiviswanathan et al., 2011].

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

Definition 2 (Central Differential Privacy - DP [Dwork and Roth, 2014])**.** 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 [Dwork et al., 2010] (we describe it in detail in Appendix B.2 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 these are described in detail in Appendix B.3.

3 Problem Formulation and Solution Overview

In this section, first we present the problem formulation and describe our algorithm for solving the empirical risk minimization (ERM) 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. In the end, we describe one of the main ingredients in our algorithm, which is a method of private mean estimation using compressed updates.

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. The clients are connected to an untrusted server in order to solve the ERM problem described in (1). In (1), $F_i(\theta, \mathcal{D}_i) = \frac{1}{r} \sum_{j=1}^{r} f\left(\theta, d_{ij}\right)$ 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 1, solving the ERM problem (1) 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 (1) 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.

Our algorithm CLDP-SGD: In order to solve (1) 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 1 for a pictorial description of our algorithm. In each step of CLDP-SGD, the secure shuffler chooses 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 ball. In this paper, we describe (ϵ_0, b)-CLDP mechanisms $\mathcal{R}_p$ for several values of $p \in [1, \infty]$; see Section 5. After that, each client i sends the set of s LDP-compressed gradients $\{\mathcal{R}_p\left(\mathbf{g}_t\left(d_{ij}\right)\right)\}_{j \in \mathcal{S}_{it}}$ 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.

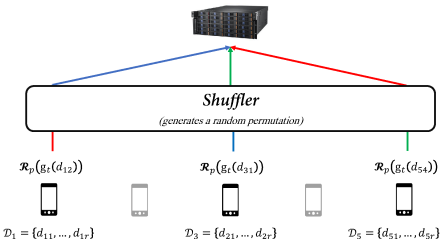

Figure 1: An example of 5 clients, where each client has r data points. At the current iteration, 3 clients are chosen at random. Each client chooses one data point at random to send the compressed and private gradient $\{\mathcal{R}_p(g_t(d_{ij}))\}$ to the secure shuffler that permutes the private gradients before sending them to the server.

Observe that our CLDP-SGD algorithm provides privacy guarantees against any adversary that can observe the output of the secure shuffler including the untrusted server. Furthermore, we assume that the trusted shuffler samples the clients, and this sampled set is unknown to the server; see line 4 in Algorithm 1. For future work, one could enable client self-sampling and self-anonymization. For example, the authors in Balle et al. [2020b] proposed a new sampling scheme called random check-in, in which each client independently chooses which time slot to participate in the training process.

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 may 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 1, 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 of data, so we cannot use the existing privacy amplification from sub-sampling results (see Section B.3.1) directly to analyze the privacy gain 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 F.1 for more details.

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

1: **Inputs:** Datasets $\mathcal{D} = \bigcup_{i\in[m]} \mathcal{D}_i$, $\mathcal{D}_i = \{d_{i1}, \ldots, d_{ir}\}$, 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 η_t.
2: **Initialize:** $\theta_0 \in \mathcal{C}$
3: **for** $t \in [T]$ **do**
4: **Sampling of clients:** The secure shuffler chooeses a random set $\mathcal{U}_t$ of k clients.
5: **for** clients $i \in \mathcal{U}_t$ **do**
6: **Sampling of data:** 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: $\tilde{\mathbf{g}}_t(d_{ij}) \leftarrow \mathbf{g}_t(d_{ij}) \,/\, \max\left\{1, \frac{\|\mathbf{g}_t(d_{ij})\|_p}{C}\right\}$[3]
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: **Shuffling:** The shuffler randomly shuffles the elements in $\{\boldsymbol{q}_t(d_{ij}) : i \in \mathcal{U}_t,\ j \in \mathcal{S}_{it}\}$ and sends them to the server.
13: **Aggregate:** $\overline{\mathbf{g}}_t \leftarrow \frac{1}{ks}\sum_{i\in\mathcal{U}_t,\ j\in\mathcal{S}_{it}} \boldsymbol{q}_t(d_{ij})$.
14: **Gradient Descent** $\theta_{t+1} \leftarrow \prod_{\mathcal{C}}(\theta_t - \eta_t \overline{\mathbf{g}}_t)$
15: **Output:** The final model parameters θ_T.

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. This privacy mechanism is composed with the sampling and shuffling to provide the overall privacy analysis. Next, we formulate the compressed and private mean estimation problem as of independent interest.

Compressed and private mean estimation via minimax risk: Now we formulate the generic minimax estimation framework for mean estimation of a

[3] Let ℓ_g denote the dual norm of ℓ_p norm, where $\frac{1}{p}+\frac{1}{g} = 1$ and $p, g \geq 1$. Thus, when the loss function $f(\theta, d_{ij})$ is convex and L-Lipschitz continuous with respect to ℓ_g-norm, then the gradient $\nabla_\theta f(\theta; .)$ has a bounded ℓ_p norm [Shalev-Shwartz et al., 2012, Lemma 2.6]. In this case, we do not need the clipping step.

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 as follows. 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-norm ball with radius a centered at the origin in $\mathbb{R}^d$, 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. Our objective is to design private-quantization mechanisms $\mathcal{R}_i : \mathcal{B}_p^d(a) \to \{0,1\}^b$ for all $i \in [n]$ and also a (stochastic) decoding function $\widehat{\boldsymbol{x}} : \left(\{0,1\}^b\right)^n \to \mathcal{B}_p^d$ that minimizes the worst-case expected error $\sup_{\{\boldsymbol{x}_i\} \in \mathcal{B}_p^d} \mathbb{E}\|\overline{\boldsymbol{x}} - \widehat{\boldsymbol{x}}(\boldsymbol{y}^n)\|^2$ and characterize the following.

$$r^{p,d}_{\epsilon_0,b,n}(a) = \inf_{\{\mathcal{R}_i \in \mathcal{Q}_{(\epsilon_0,b)}\}} \inf_{\widehat{\boldsymbol{x}}} \sup_{\{\boldsymbol{x}_i\} \in \mathcal{B}_p^d(a)} \mathbb{E}\left\|\overline{\boldsymbol{x}} - \widehat{\boldsymbol{x}}(\boldsymbol{y}^n)\right\|_2^2, \tag{4}$$

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 $\{\mathcal{R}_i : i \in [n]\}$ and the estimator $\widehat{\boldsymbol{x}}$. Note that in this setup we do not consider any probabilistic assumptions on the vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$. We also provide additional results when the inputs are sampled from a distribution in Appendix B.4 in the supplementary material.

4 Main 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 4.2.

4.1 Optimization

In the next theorem, we state the privacy guarantees, the communication cost per client, and the privacy-convergence trade-offs for the CLDP-SGD Algorithm. 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,[4] 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.[5] For $s = 1$ and $q = \frac{k}{mr}$, if we run Algorithm $\mathcal{A}_{cldp}$, then we have*

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

$$\epsilon = \mathcal{O}\left(\epsilon_0 \sqrt{\frac{qT\log(2qT/\delta)\log(2/\delta)}{n}}\right). \tag{5}$$

2. ***Communication:*** *Our algorithm $\mathcal{A}_{cldp}$ requires $\frac{k}{m} \times b$ bits of communication in expectation[6] 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*

$$\mathbb{E}[F(\theta_T)] - F(\theta^*) \leq \mathcal{O}\left(\frac{LD\log(T)\max\{d^{\frac{1}{2}-\frac{1}{p}}, 1\}}{\sqrt{T}}\sqrt{\frac{cd}{qn}}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\right). \tag{6}$$

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

We prove Theorem 1 in Section 5.3. Note that the privacy bound (49) holds when $\epsilon_0 = \mathcal{O}(1)$; the general result for $\epsilon_0 = \mathcal{O}\left(\log\left(\frac{qn}{\log(T/\delta)}\right)\right)$ is presented in Section 5.3.1.

Remark 1. Using a slightly different sampling procedure, the result in Theorem 1 holds for arbitrary s. We can achieve the same central privacy bound ϵ as stated in Theorem 1 with $q = \frac{ks}{mr}$ (instead of $q = \frac{k}{mr}$) using the following sampling: all clients send s compressed and private gradients corresponding to a uniformly random subset of s data points from their dataset; shuffler selects a uniformly random subset of ks gradients from them and then sends the shuffled output to the server. We analyze the privacy guarantees of our algorithm with this new sampling procedure in Appendix F.4. Note that, in this new sampling procedure, each data point has a probability $q = \frac{ks}{mr}$ of being picked, and we pick $\frac{ks}{m}$ data points (in expectation) from each clients. Note that even for this sampling (which does not yield uniform sampling of ks points from mr points), the privacy amplification of this sampling mechanism does not directly follow from existing results. We provide a

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

[5] 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 $|f(\theta_1; d) - f(\theta_2; d)| \leq L\|\theta_1 - \theta_2\|_g$.

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

proof of this in the supplementary material, along with a discussion on other sampling procedures.

Remark 2 (Recovering the Result [Erlingsson et al., 2020, ESA])**.** In Erlingsson et al. [2020], 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 (6) and $q = 1$ in (49), we recover the convergence and the privacy bound in [Erlingsson et al., 2020, Theorem VI.1].

We want to emphasize that the above privacy-accuracy trade-off in Erlingsson et al. [2020] 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 our privacy amplification is *effectively* due to two types of sampling, one of data, and the other of clients.

Remark 3 (Optimality of CLDP-SGD for ℓ_2-norm case)**.** Suppose $\epsilon = \mathcal{O}(1)$. Substituting $\epsilon_0 = \epsilon\sqrt{\frac{n}{qT\log(2qT/\delta)\log(2/\delta)}}$, $T = n/q$, and $p = 2$ in (6), gives the *optimal* excess risk of central differential privacy, as shown in Bassily et al. [2014]. Note that the results in Bassily et al. [2014] are for centralized SGD with full precision gradients, whereas, our results are for federated learning (which is a distributed setup) with compressed gradient exchange.

4.2 Compressed & Private Mean Estimation

In this subsection, we state our lower and upper bounds on the minimax risk $r^{p,d}_{\epsilon_0,b,n}(a)$ for all $p \in [1,\infty]$. For the lower bounds, we state our results when there is no communication constraints, and for clarity, we denote the corresponding minimax risk by $r^{p,d}_{\epsilon_0,\infty,n}(a)$. Furthermore, we prove that any symmetric private mechanism requires at least $b \geq \log(d)$ bits of communication.

Theorem 2. *For any $d, n \geq 1$, $a, \epsilon_0 > 0$, and $p \in [1,\infty]$, the minimax risk in (4) satisfies*

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

Theorem 3. *For any 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 $\widehat{\boldsymbol{x}} = \frac{1}{n}\sum_{i=1}^n g(\mathcal{R}(\boldsymbol{x}_i))$, we have*[7]

$$r^{p,d}_{\epsilon,b,n}(a) > a^2 \max\left\{1, d^{1-\frac{2}{p}}\right\}. \tag{7}$$

[7]Note that Theorem 3 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.

Theorem 3 shows that it is required at least $\log(d)$ bits per client to design a non-trivial private mechanism $\mathcal{R}$. Though our lower bound results are for arbitrary estimators $\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)$, we can show that the optimal estimator $\widehat{\boldsymbol{x}}(\boldsymbol{y}^n)$ is a *deterministic* function of $\boldsymbol{y}^n$; see Lemma 15 in Appendix E.

Theorem 4. *For any $d, n \geq 1$, $a, \epsilon_0 > 0$, we have*

$$\ell_1 : r^{1,d}_{\epsilon_0,b,n}(a) \leq \frac{a^2 d}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2, \text{for } b = \log(d)+1,$$

$$\ell_2 : r^{2,d}_{\epsilon_0,b,n}(a) \leq \frac{6a^2 d}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2, \text{for } b = d\log(e)+1,$$

$$\ell_\infty : r^{\infty,d}_{\epsilon_0,b,n}(a) \leq \frac{a^2 d^2}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2, \text{for } b = \log(d)+1.$$

Note that when $\epsilon_0 = \mathcal{O}(1)$, then the upper and lower bounds on minimax risks match for all $p \in [1,\infty]$. We can give general achievability results for any ℓ_p-norm ball $\mathcal{B}_p^d(a)$ 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. The main results for this are stated in Appendix D.1.

All the above results are for $r^{p,d}_{\epsilon_0,b,n}(a)$, which is defined for worst cast inputs. Essentially the same results hold when the inputs are sampled from a distribution, and we provide those results in Appendix D.1.

5 Proofs

In this section, first we prove the compressed and private mean estimation results and then prove Theorem 1. Due to the lack of space, we only prove mean estimation results for ℓ_∞-norm case and provide the proofs for other norms in Appendix D.

5.1 Proof of Theorem 2: For ℓ_∞-Norm

The lower bound result presented in this section in fact hold for any ℓ_p-norm 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. Let $\mathcal{P}^{\text{Bern}}_{p,d}$ denote the set of Bernoulli distributions on $\{0, 1/d^{1/p}\}^d$, i.e., any element of $\mathcal{P}^{\text{Bern}}_{p,d}$ is a product of d independent Bernoulli distributions, one for each coordinate. For any $\boldsymbol{q} \in \mathcal{P}^{\text{Bern}}_{p,d}$, let $\boldsymbol{\mu}_{\boldsymbol{q}}$ denote the mean of $\boldsymbol{q}$.

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

$$\inf_{\{\mathcal{M}_i\}\in\mathcal{Q}_{(\epsilon_0,\infty)}} \inf_{\widehat{\boldsymbol{x}}} \sup_{\boldsymbol{q}\in\mathcal{P}^{Bern}_{p,d}} \mathbb{E}\left\|\boldsymbol{\mu}_{\boldsymbol{q}} - \widehat{\boldsymbol{x}}(\boldsymbol{y}^n)\right\|_2^2 \geq \Omega\left(d^{1-\frac{2}{p}} \min\left\{1, \frac{d}{n\min\{\epsilon_0,\epsilon_0^2\}}\right\}\right). \tag{8}$$

The proof is straightforward adaptation of the proof of [Duchi and Rogers, 2019, Corollary 3] to our setting; see Appendix 5.1 for more details.

Let $\mathcal{P}_p^d$ denote the set of all distributions on the ℓ_p-norm ball, implying that $\mathcal{P}_{p,d}^{\text{Bern}} \subset \mathcal{P}_p^d$. This together with (8), implies that for every set of private mechanisms $\{\mathcal{M}_i\} \in \mathcal{Q}_{(\epsilon_0,\infty)}$ and estimator $\widehat{\boldsymbol{x}}$, we have

$$\sup_{\boldsymbol{q}\in\mathcal{P}_p^d} \mathbb{E}\left\|\boldsymbol{\mu_q} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 \geq \sup_{\boldsymbol{q}\in\mathcal{P}_{p,d}^{\text{Bern}}} \mathbb{E}\left\|\boldsymbol{\mu_q} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 \geq \Omega\left(d^{1-\frac{2}{p}} \min\left\{1, \frac{d}{n\min\{\epsilon_0,\epsilon_0^2\}}\right\}\right), \tag{9}$$

We can now obtain a lower bound on $r_{\epsilon_0,\infty,n}^{p,d}$ by transforming the worst-case lower bound to the average case lower bound as follows. Fix arbitrary private mechanisms $\{\mathcal{M}_1,\ldots,\mathcal{M}_n\}$ and an estimator $\widehat{\boldsymbol{x}}$. It follows from (9) 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 $\boldsymbol{y}_i = \mathcal{M}_i(\boldsymbol{x}_i^{(q)})$, 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

$$\begin{aligned}
&\sup_{\{\boldsymbol{x}_i\}\in\mathcal{B}_p^d} \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 \\
&\overset{(a)}{\geq} \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i^{(q)} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 \\
&\overset{(b)}{\geq} \frac{1}{2}\mathbb{E}\left\|\boldsymbol{\mu_q} - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 - \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i^{(q)} - \boldsymbol{\mu_q}\right\|_2^2 \\
&\overset{(c)}{\geq} \Omega\left(d^{1-\frac{2}{p}} \min\left\{1, \frac{d}{n\min\{\epsilon_0,\epsilon_0^2\}}\right\}\right) - \frac{d^{1-\frac{2}{p}}}{n} \\
&\overset{(d)}{\geq} \Omega\left(d^{1-\frac{2}{p}} \min\left\{1, \frac{d}{n\min\{\epsilon_0,\epsilon_0^2\}}\right\}\right)
\end{aligned} \tag{10}$$

Step (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. $\{\boldsymbol{x}_i^{(q)}\}$ in $\mathcal{B}_p^d$ and hence lower-bounds the LHS. The inequality (b) follows from the Jensen's inequality. Step (c) follows from $\mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i^{(q)} - \boldsymbol{\mu_q}\right\|_2^2 \leq \frac{d^{1-\frac{2}{p}}}{n}$, which we show below. Step (d) assumes $\min\{\epsilon_0,\epsilon_0^2\} \leq \mathcal{O}(d)$.

Note that for any vector $\boldsymbol{u} \in \mathbb{R}^d$, we have $\|\boldsymbol{u}\|_2 \leq d^{1/2-1/p}\|\boldsymbol{u}\|_p$, 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

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

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 infimum in (10) over all ϵ-LDP mechanisms $\{\mathcal{M}_i : i \in [n]\}$ and estimators $\widehat{\boldsymbol{x}}$, we get

$$\begin{aligned}
&r_{\epsilon_0,\infty,n}^{p,d} = \\
&\inf_{\{\mathcal{M}_i\in\mathcal{Q}_{(\epsilon_0,\infty)}\}} \inf_{\widehat{\boldsymbol{x}}} \sup_{\{\boldsymbol{x}_i\}\in\mathcal{B}_p^d} \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i - \widehat{\boldsymbol{x}}\left(\boldsymbol{y}^n\right)\right\|_2^2 \\
&\geq \Omega\left(d^{1-\frac{2}{p}} \min\left\{1, \frac{d}{n\min\{\epsilon_0,\epsilon_0^2\}}\right\}\right).
\end{aligned}$$

5.2 Proof of Theorem 4: For ℓ_∞-Norm

In this section, we propose an (ϵ_0, b)-CLDP mechanism for ℓ_∞-norm ball that requires $b = \mathcal{O}(\log(d))$-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 (11) 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 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 the client-side mechanism in Algorithm 2 and state its properties below, which we show in Appendix D.7.

Algorithm 2 ℓ_∞-MEAN-EST ($\mathcal{R}_\infty$: the client-side algorithm)

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

$$\boldsymbol{z} = \begin{cases} +ad\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\boldsymbol{e}_j & \text{w.p. } \frac{1}{2} + \frac{x_j}{2a}\frac{e^{\epsilon_0}-1}{e^{\epsilon_0}+1} \\ -ad\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)\boldsymbol{e}_j & \text{w.p. } \frac{1}{2} - \frac{x_j}{2a}\frac{e^{\epsilon_0}-1}{e^{\epsilon_0}+1} \end{cases} \tag{11}$$

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

Lemma 2. *The mechanism $\mathcal{R}_\infty$ presented in Algorithm 2 satisfies the following properties, where $\epsilon_0 > 0$:* (i) *$\mathcal{R}_\infty$ is $(\epsilon_0, \log(d)+1)$-CLDP and requires only 1-bit of communication using public randomness.* (ii) *$\mathcal{R}_\infty$ 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 $\mathbb{E}\|\mathcal{R}_\infty(\boldsymbol{x}) - \boldsymbol{x}\|_2^2 \leq a^2d^2\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2$.*

Since the server averages the n received messages, we can easily verify that $r_{\epsilon_0,b,n}^{\infty,d}(a) \leq \frac{a^2d^2}{n}\left(\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\right)^2$.

5.3 Proof of Theorem 1

We show our results on privacy, communication, and convergence separately in the next three subsections.

5.3.1 Privacy

In Algorithm 1, 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, which ensures that the mechanism $\mathcal{A}_{cldp}$ guarantees local differential privacy ϵ_0 for each sample d_{ij} per iteration. 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. Thus, the mechanism $\mathcal{M}_t$ on an input dataset $\mathcal{D} = \bigcup_{i=1}^{m} \mathcal{D}_i \in \mathfrak{S}^n$ can be defined as:

$$\mathcal{M}_t(\theta_t; \mathcal{D}) = \mathcal{H}_{ks} \circ \text{samp}_{m,k}\left(\mathcal{G}_1, \ldots, \mathcal{G}_m\right), \qquad (12)$$

where $\mathcal{G}_i = \text{samp}_{r,s}\left(\mathcal{R}(\boldsymbol{x}_{i1}^t), \ldots, \mathcal{R}(\boldsymbol{x}_{ir}^t)\right)$ 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 $\text{samp}_{a,b}$ denotes the sampling operation for choosing a random subset of b elements from a set of a elements.

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

Lemma 3. *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 3 in Appendix C. In the statement of Lemma 3, we are amplifying the privacy by using the subsampling as well as shuffling ideas. For subsampling, note that we do not pick a uniformly random subset of size ks from $n = mr$ points. So, we cannot directly apply the amplification by subsampling result of Kasiviswanathan et al. [2011] (stated in Lemma 7 in Appendix B.3.1). However, as it turns out that the only property we will need for privacy amplification by subsampling is that each data point is picked with probability $q = \frac{ks}{mr}$, which holds true in our setting.

Consider 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 different 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 mechanism $\mathcal{M}_t$ into a summation of four conditional probabilities depending on the event whether the first client is picked or not and the first client pick the first data point or not. We use bipartite graphs to get the relation between these events, where each vertex corresponds to one of the possible outputs of the sampling procedure, and each edge connects two neighboring vertices. See Appendix C for more details.

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 3. Now, we invoke the strong composition theorem from [Dwork and Roth, 2014, Theorem 3.20] (stated in Lemma 6 in Appendix B.2) to obtain the privacy guarantee of the algorithm $\mathcal{A}_{cldp}$ as stated in Theorem 1. We provide the details in Appendix F.

5.3.2 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 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 [Mayekar and Tyagi, 2020] which was applied in the context of non-private quantization. The idea is as follows. Since all clients apply 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 $|\mathcal{A}_B^s| = \binom{s+B-1}{s} \leq \left(\frac{e(s+B-1)}{s}\right)^s$, 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 a client is chosen with probability $\frac{k}{m}$ at any time $t \in [T]$, 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 clients.

5.3.3 Convergence

At iteration $t \in [T]$ of Algorithm 1, server averages the received ks 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 13 of Algorithm 1) and then updates the parameter vector as $\theta_{t+1} \leftarrow \prod_{\mathcal{C}}(\theta_t - \eta_t \overline{\mathbf{g}}_t)$. 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 $\mathbb{E}\left[\overline{\mathbf{g}}_t\right] = \nabla_{\theta_t} F(\theta_t)$, where expectation is taken w.r.t. the 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 4. *For any $d \in \mathfrak{S}$, if the function $f(\theta; .)$:*

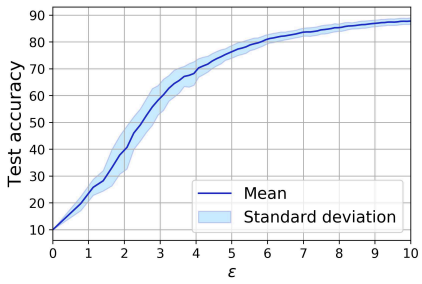

Figure 2: Privacy-Utility trade-offs on the MNIST dataset with ℓ_∞-norm clipping.

Layer	Parameters
Convolution	16 filters of 8×8, Stride 2
Max-Pooling	2×2
Convolution	32 filters of 4×4, Stride 2
Max-Pooling	2×2
Fully connected	32 unites
Softmax	10 unites

Table 1: Model Architecture for MNIST

$\mathcal{C} \to \mathbb{R}$ *is convex and L-Lipschitz continuous w.r.t. the* ℓ_g*-norm, which is the dual of* ℓ_p*-norm, then we have*

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

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

Lemma 4 is proved in Appendix F.3.

Now, using the bound on G^2 from Lemma 4 in the following standard SGD convergence results for convex functions proves the third part of Theorem 1; see Appendix F.3 for more details.

Lemma 5 (SGD Convergence [Shamir and Zhang, 2013])**.** *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* $\mathbb{E}[F(\theta_T)] - F(\theta^*) \leq 2DG\left(\frac{2+\log(T)}{\sqrt{T}}\right)$.

6 Numerical Results

In this section, we present our numerical results to evaluate the proposed CLDP-SGD algorithm for training machine learning models with privacy and communication constraints. We consider the standard MNIST handwritten digit dataset that has $60,000$ training images and $10,000$ test images. We train a simple neural network that was also used in [Erlingsson et al., 2020, Papernot et al., 2020] and described in Table 1. This model has a total number of $d = 13,170$ parameters and achieves an accuracy of 99% for non-private, uncompressed vanilla SGD. In our results, we assume that we have $60,000$ clients, where each client has one sample, i.e., $m = n = 60,000$ and $r = 1$. We present our results for ℓ_∞-norm clipping. At each step of the CLDP-SGD, we choose at random $10,000$ clients. Each client clips the ℓ_∞-norm of the gradient $\nabla_{\theta_t} f(\theta_t; d_i)$ with clipping parameter $C = 1/100$. After that, the client applies the LDP-compression mechanism $\mathcal{R}_\infty$ (presented in Algorithm 2) to the clipped gradient. We run our algorithm for 80 epochs, where we set the learning rate at 0.3 for the first 70 epochs and decrease it to 0.18 in the remaining epochs. We set the local privacy parameters $\epsilon_0 = 2$ and $\delta = 10^{-5}$, while the centralized privacy parameter ϵ is computed numerically from Theorem 1 as follows. We first compute the privacy amplification by shuffling numerically using the expression in [Balle et al., 2019c, Theorem 5.3]. Then, we compute the privacy amplification via subsampling presented in Lemma 3; and finally we use the strong composition stated in Lemma 6 in Appendix B.2 to obtain the central privacy parameter ϵ.

Figure 2 demonstrates the mean and the standard deviation of privacy-accuracy plot averaged over 10 runs. It shows that we can achieve an accuracy 76.7% (± 2) for total privacy $\epsilon = 5$ and an accuracy 87.9% (± 1) for total privacy $\epsilon = 10$. Furthermore, observe that our proposed CLDP-SGD algorithm preserves a local privacy of $\epsilon_0 = 2$ per sample per epoch. In addition, the private mechanism $\mathcal{R}_\infty$ requires only $\lceil \log(d) \rceil + 1$ bits per gradient, while the full precision gradient requires $32 \times d$ bits per gradient. Thus, the proposed private mechanism saves in communication bits a factor of $28096\times$ in comparison with the full precision gradient.

In [Papernot et al., 2020], the authors achieve a test accuracy of 98% on MNIST with central privacy parameters $\epsilon = 3$ and $\delta = 10^{-5}$ using a DP centralized algorithm by adding Gaussian noise to the aggregated gradients in each iteration. However, Papernot et al. [2020] do not offer any local differential privacy guarantees, which can be thought of as $\epsilon_0 = \infty$. Although, Theorem 1 and Remark 3 show that our proposed algorithm matches theoretically the results of the centralized SGD with full precision gradients, the numerical results show that there is a gap between the accuracy of our algorithm and the test accuracy of the centralized algorithm in Papernot et al. [2020]. We believe that the privacy parameters of our algorithm can be improved by analyzing the Renyi differential privacy of the shuffled model, which is an important open question of the ongoing investigation.

Acknowledgements

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.

References

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*, pages 308–318, 2016.

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*, volume 97. PMLR, 2019.

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*, pages 1120–1129, 2019.

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*, pages 7564–7575, 2018.

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*, pages 1709–1720, 2017.

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*, pages 5973–5983, 2018.

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

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

B. Balle, J. Bell, A. Gascón, and K. Nissim. The privacy blanket of the shuffle model. In *Annual International Cryptology Conference*, pages 638–667. Springer, 2019c.

B. Balle, J. Bell, A. Gascón, and K. Nissim. Private summation in the multi-message shuffle model. In *CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, USA, November 9-13, 2020*, pages 657–676. ACM, 2020a. doi: 10.1145/3372297.3417242.

B. Balle, P. Kairouz, B. McMahan, O. D. Thakkar, and A. Thakurta. Privacy amplification via random check-ins. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020b.

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*, pages 464–473. IEEE, 2014.

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*, pages 14695–14706, 2019.

A. Beimel, K. Nissim, and E. Omri. Distributed private data analysis: Simultaneously solving how and what. In *Annual International Cryptology Conference*, pages 451–468. Springer, 2008.

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*, pages 437–454. Springer, 2010.

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.

K. Chaudhuri, C. Monteleoni, and A. D. Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(3), 2011.

W. Chen, P. Kairouz, and A. Özgür. Breaking the communication-privacy-accuracy trilemma. In *Advances in Neural Information Processing Systems*, 2020.

A. Cheu, A. D. Smith, J. Ullman, D. Zeber, and M. Zhilyaev. Distributed differential privacy via shuffling. In *Advances in Cryptology - EUROCRYPT 2019*, volume 11476, pages 375–403. Springer, 2019.

J. C. Duchi and R. Rogers. Lower bounds for locally private estimation via communication complexity. In *Conference on Learning Theory (COLT)*, pages 1161–1191, 2019.

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*, pages 429–438. IEEE, 2013.

J. C. Duchi, M. I. Jordan, and M. J. Wainwright. Minimax optimal procedures for locally private estima-

tion. *Journal of the American Statistical Association*, 113(521):182–201, 2018.

C. Dwork and A. Roth. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

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

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*, pages 51–60, 2010.

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*, pages 2468–2479. SIAM, 2019.

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.

A. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke. Privacy preserving mining of association rules. *Information Systems*, 29(4):343–364, 2004.

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

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

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

B. Ghazi, R. Kumar, P. Manurangsi, and R. Pagh. Private counting from anonymous messages: Near-optimal accuracy with vanishing communication overhead. In *International Conference on Machine Learning (ICML)*, pages 3505–3514, 2020.

A. M. Girgis, D. Data, K. Chaudhuri, C. Fragouli, and S. N. Diggavi. Successive refinement of privacy. *IEEE Journal on Selected Areas in Information Theory*, 1(3):745–759, 2020. doi: 10.1109/JSAIT.2020.3040403.

P. Kairouz, K. Bonawitz, and D. Ramage. Discrete distribution estimation under local privacy. In *International Conference on Machine Learning, ICML*, pages 2436–2444, 2016.

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.

S. P. Karimireddy, Q. Rebjock, S. Stich, and M. Jaggi. Error feedback fixes SignSGD and other gradient compression schemes. In *ICML*, pages 3252–3261, 2019.

S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith. What can we learn privately? *SIAM Journal on Computing*, 40 (3):793–826, 2011.

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

N. Papernot, A. Thakurta, S. Song, S. Chien, and Ú. Erlingsson. Tempered sigmoid activations for deep learning with differential privacy. *arXiv preprint arXiv:2007.14191*, 2020.

S. Shalev-Shwartz et al. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2012.

O. Shamir and T. Zhang. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. In *International conference on machine learning*, pages 71–79, 2013.

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.

N. Singh, D. Data, J. George, and S. Diggavi. Squarm-sgd: Communication-efficient momentum sgd for decentralized optimization. *arXiv preprint arXiv:2005.07041*, 2020.

S. U. Stich, J.-B. Cordonnier, and M. Jaggi. Sparsified sgd with memory. In *Advances in Neural Information Processing Systems*, pages 4447–4458, 2018.

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*, pages 3329–3337. JMLR. org, 2017.

J. Ullman. Cs7880. rigorous approaches to data privacy. 2017. URL `http://www.ccs.neu.edu/home/jullman/cs7880s17/HW1sol.pdf`.

S. L. Warner. Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 60(309):63–69, 1965.