Private Federated Learning Without a Trusted Server: Optimal Algorithms for Convex Losses

Andrew Lowy
University of Southern California
lowya@usc.edu

Meisam Razaviyayn
University of Southern California
razaviya@usc.edu

Abstract

This paper studies the problem of federated learning (FL) in the absence of a trustworthy server/clients. In this setting, each client needs to ensure the privacy of its own data without relying on the server or other clients. We study local differential privacy (LDP) at the client level and provide *tight upper and lower bounds that establish the minimax optimal rates* (up to logarithms) for LDP convex/strongly convex federated stochastic optimization. Our rates match the optimal statistical rates in certain practical parameter regimes ("privacy for free"). Second, we develop a novel time-varying noisy SGD algorithm, leading to the first non-trivial LDP risk bounds for FL with *non-i.i.d.* clients. Third, we consider the special case where each client's loss function is empirical and develop an *accelerated* LDP FL algorithm to improve communication complexity compared to existing works. We also provide matching lower bounds, establishing the optimality of our algorithm for convex/strongly convex settings. Fourth, with a secure shuffler to anonymize client reports (but without a trusted server), our algorithm attains the optimal central DP rates for stochastic convex/strongly convex optimization, thereby achieving optimality in the local and central models simultaneously. Our upper bounds quantify the role of network communication reliability in performance. We illustrate the practical utility of our algorithm with numerical experiments.

1 Introduction

In federated learning, each "client" (e.g. cell-phone users or organizations such as hospitals) stores its data locally and a central server coordinates updates to achieve the global learning objective [40]. Federated learning (FL) has been deployed across application domains such as the internet of things [61], wireless communication [55], medicine [16], finance [1], and by companies such as Google [63] and Apple [4]. One of the primary reasons for the introduction of FL was to offer greater privacy for sensitive user data [54]. Unfortunately, merely storing data locally is not sufficient to prevent data leakage, as model parameters or updates can still reveal sensitive information [27, 35, 66, 80]. These leaks can occur when clients send updates to the server, which an adversary may be able to access, or (in decentralized/peer-to-peer FL) directly to other clients. Therefore, it is important to develop privacy-preserving mechanisms for FL that do not rely on the server or other clients to implement.

Consider a FL setting with N clients, each containing a local data set with n_i samples: $X_i = (x_{i,1}, \cdots, x_{i,n_i})$ for $i \in [N]$. In each round of communication r, a uniformly random subset S_r of $M_r \in [N]$ clients is able to participate, where $\{M_r\}_{r=1}^R$ are i.i.d. random variables. For all i, let $\mathcal{D}_i$ be a probability distribution on a set $\mathcal{X}_i$ containing data and denote $\mathcal{X} := \bigcup_{i=1}^N \mathcal{X}_i$. Given a convex (in w) loss function $f : \mathcal{W} \times \mathcal{X} \to \mathbb{R}$, define client i's local objective as

$$F_i(w) := \mathbb{E}_{x_i \sim \mathcal{D}_i}[f(w, x_i)], \tag{1}$$

where $\mathcal{W} \subset \mathbb{R}^d$ is closed, convex, and $\|w\|_2 \leqslant D$, $\forall w \in \mathcal{W}$. At times, we may focus on empirical risk minimization (ERM) framework where $\widehat{F}_i(w) := \frac{1}{n_i} \sum_{j=1}^{n_i} f(w, x_{i,j})$. Our goal is to solve

1st NeurIPS Workshop on New Frontiers in Federated Learning (NFFL 2021), Virtual Meeting.

$$\min_{w \in \mathcal{W}} \left\{ F(w) := \sum_{i=1}^{N} p_i F_i(w) \right\}, \tag{2}$$

or, in the ERM case, $\min_{w \in \mathcal{W}} \{\widehat{F}(w) := \sum_{i=1}^{N} p_i \widehat{F}_i(w)\}$, while maintaining the privacy of each client. Here $p_i \geqslant 0$ with $\sum_i p_i = 1$. We assume WLOG (see Appendix B) that $p_i = \frac{1}{N}, \forall i \in [N]$. We say problem (2) is "i.i.d." if $\mathcal{X}_i = \mathcal{X}$ and $\mathcal{D}_i = \mathcal{D} \ \forall i$. When F_i takes the general form (1) (not necessarily ERM), we refer to the problem as "SCO" (stochastic convex optimization) for emphasis.

Popular algorithms for FL include Local SGD/Federated Averaging [55] and Minibatch SGD. Both of these are *fully interactive* algorithms, meaning they can adaptively query each client multiple times. A subset of fully interactive algorithms is the set of *sequentially interactive* algorithms, which can query clients adaptively in sequence, but cannot query any one client more than once. *Non-interactive* algorithms are non-adaptive sequentially interactive algorithms: they query each client once, independently of other clients' reports. See [39] for further discussion.

Notions of Privacy for FL: Given the practical importance of maintaining the privacy of user data during the FL process, numerous different definitions of private FL have been proposed. Some of these have used secure multi-party computation (MPC) [12, 53], but this approach leaves users vulnerable to inference attacks on the trained model. This is in contrast to differential privacy (DP), which by now has been widely accepted as the gold standard of rigorous data privacy notions. DP is defined with respect to a database space $\mathbb{X}$ and a measure of distance $\rho : \mathbb{X}^2 \to [0, \infty)$ between databases. We say two databases $\mathbf{X}, \mathbf{X}' \in \mathbb{X}$ are ρ-adjacent if $\rho(\mathbf{X}, \mathbf{X}') \leqslant 1$.

Definition 1. *(Differential Privacy) Let $\epsilon \geqslant 0$, $\delta \in [0, 1)$. A randomized algorithm $\mathcal{A} : \mathbb{X} \to \mathcal{W}$ is (ϵ, δ)-differentially private (DP) if for all ρ-adjacent data sets $\mathbf{X}, \mathbf{X}' \in \mathbb{X}$ and all measurable subsets $S \subset \mathcal{W}$, we have*

$$\mathbb{P}(\mathcal{A}(\mathbf{X}) \in S) \leqslant e^{\epsilon} \mathbb{P}(\mathcal{A}(\mathbf{X}') \in S) + \delta. \tag{3}$$

If (3) holds for all measurable subsets S, then we denote this property by $\mathcal{A}(\mathbf{X}) \underset{(\epsilon,\delta)}{\simeq} \mathcal{A}(\mathbf{X}')$. For FL, we will fix $\mathbb{X} := \mathcal{X}_1^{n_1} \times \cdots \times \mathcal{X}_N^{n_N}$ so that databases $\mathbf{X}$ in our FL setting consist of N client datasets $\mathbf{X} = (X_1, \cdots X_N)$. Definition 1 says that an algorithm is DP if with high probability, an adversary cannot distinguish between the outputs of the algorithm when it is run on adjacent databases. Depending on the choice of ρ, we can get different variations of DP. For example, in the classical notion of central differential privacy (CDP) (often simply referred to as "differential privacy") [19], $\rho(\mathbf{X}, \mathbf{X}') := \sum_{i=1}^{N} \sum_{j=1}^{n_i} \mathbb{1}_{x_{i,j} \neq x'_{i,j}}$ is the hamming distance and adjacent databases are those differing in a single sample. In the context of FL, a major problem with CDP is that it does not preclude the untrusted server from accessing non-private updates (which may leak clients' data).

Client-level DP (also called user-level DP) has been proposed as an alternative to CDP where a single person may contribute many samples to the database (e.g. language modeling) [55, 29, 38, 28, 72, 79, 47]. It is defined by taking $\rho(\mathbf{X}, \mathbf{X}') = \sum_{i=1}^{N} \mathbb{1}_{X_i \neq X'_i}$. Client-level DP shares the same drawback as CDP: it allows sensitive data to be leaked to the untrusted server. In contrast to the centralized models of CDP and client-level DP, this work imposes the stronger requirement that *client updates be private before they are sent to the untrusted server (or other clients)* for aggregation.

First, we consider local differential privacy (LDP), a generalization of the the classic notion with the same name [44] to FL. An R-round fully interactive randomized algorithm $\mathcal{A} : \mathbb{X} \to \mathcal{Z}^{R \times N}$ for FL is characterized in every round $r \in [R]$ by N local client functions called *randomizers* $\mathcal{R}_r^{(i)} : \mathcal{Z}^{(r-1) \times N} \times \mathcal{X}_i^{n_i} \to \mathcal{Z}$ $(i \in [N])$ and an aggregation mechanism. The randomizers send messages $Z_r^{(i)} := \mathcal{R}_r^{(i)}(\mathbf{Z}_{1:r-1}, X_i)$ (which may depend on client data X_i and the outputs $\mathbf{Z}_{1:r-1} := \{Z_t^{(j)}\}_{j \in [N], t \in [r-1]}$ of clients' randomizers in prior rounds) to the server or (in peer-to-peer FL) other clients. [1] Then, the server (or clients, for peer-to-peer FL) updates the global model. Algorithm $\mathcal{A}$ is $\{(\epsilon_i, \delta_i)\}_{i=1}^{N}$-LDP if for all $i \in [N]$, the full transcript of client i's communications, i.e. the collection

[1] We assume that $\mathcal{R}_r^{(i)}(\mathbf{Z}_{1:r-1}, X_i)$ is conditionally independent of X_j $(j \neq i)$ given $\mathbf{Z}_{1:r-1}$ and X_i. That is, the randomizers of i cannot "eavesdrop" on another client's data (consistent with the local data principle of FL). We allow for $Z_t^{(i)}$ to be empty/zero if client i does not output anything to the server in round t.

of all R messages $\{Z_r^{(i)}\}_{r\in[R]}$, is (ϵ_i, δ_i)-DP, conditional on the messages and data of all other clients. See Fig. 1. [2] More precisely:

Definition 2. *(Local Differential Privacy) A randomized algorithm $\mathcal{A} : \mathcal{X}_1^{n_1} \times \cdots \times \mathcal{X}_N^{n_N} \to \mathcal{Z}^{R\times N}$ is $\{(\epsilon_i, \delta_i)\}_{i=1}^N$-LDP if for all $i \in [N]$ and all ρ_i-adjacent $X_i, X_i' \in \mathcal{X}_i^{n_i}$, we have*

$$(\mathcal{R}_1^{(i)}(X_i), \mathcal{R}_2^{(i)}(\mathbf{Z}_1, X_i), \cdots, \mathcal{R}_R^{(i)}(\mathbf{Z}_{1:R-1}, X_i)) \underset{(\epsilon_i,\delta_i)}{\simeq} (\mathcal{R}_1^{(i)}(X_i'), \mathcal{R}_2^{(i)}(\mathbf{Z}_1', X_i'), \cdots, \mathcal{R}_R^{(i)}(\mathbf{Z}_{1:R-1}', X_i')),$$

where for all r the distribution of $\mathcal{R}_r^{(i)}(\mathbf{Z}_{1:r-1}, X_i)$ is conditional on the transcripts $\mathbf{Z}_{1:r-1}^{(j\neq i)}$ of all other clients in all previous rounds. Here $\rho_i : \mathcal{X}_i^2 \to [0, \infty)$ is given by $\rho_i(X_i, X_i') = \sum_{j=1}^{n_i} \mathbb{1}_{x_{i,j}\neq x_{i,j}'}$.

We sometimes assume for simplicity that privacy parameters are the same across clients and denote these common parameters by (ϵ_0, δ_0). Note that *LDP is stronger than CDP*: (ϵ_0, δ_0)-LDP implies (ϵ_0, δ_0)-CDP but the converse is false. Moreover, *LDP is stronger than client-level DP* in the following sense: (ϵ_0, δ_0)-LDP implies $(n\epsilon, ne^{(n-1)\epsilon}\delta)$ client-level DP, but (ϵ, δ)-client-level DP does not imply (ϵ', δ')-LDP for any ϵ', δ'. See Appendix C for proofs.

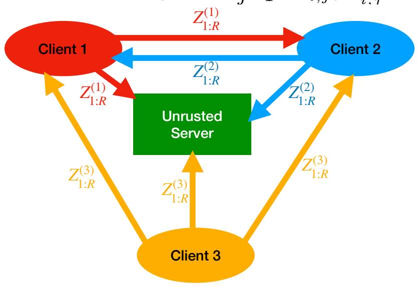

Figure 1: LDP protects the privacy of each client's data regardless of whether the server or other clients are trustworthy and regardless of the network topology (e.g. peer-to-peer or server-orchestrated). The messages $Z_{1:R}^{(i)}$ of client i are DP, ensuring that *client i's data cannot be leaked, even if the server or other clients are curious or leak data themselves.*

It is also illuminating to compare Definition 2 with classical item-level LDP [44, 18], which requires each individual *person* (rather than client) to randomize their own data. When $n = 1$, so that each client has just one person's data, classical LDP is equivalent to Definition 2. But if $n > 1$, then classical LDP would require each person (e.g. patient) to randomize her reports (e.g. medical test results) before sending them to the data collector (doctors/researchers) within the client (hospital). Since we assume in FL that clients can be trusted with their own data, this intra-client randomization is practically unnecessary. Thus, *Definition 2 is more practically relevant for FL than classical item-level LDP*, and it results in higher accuracy models.

An intermediate trust model between the low-trust local model and the high-trust central/client-level model is the shuffle model [10, 14, 21, 22, 25, 50, 32] where clients have access to a secure shuffler (also known as a mixnet) that receives randomized reports from the clients and randomly permutes them (effectively anonymizing them), before the reports are sent to the untrusted server/other clients.[3]

Definition 3. *(Shuffle Differential Privacy) A randomized algorithm $\mathcal{A} : \mathcal{X}_1^{n_1} \times \cdots \times \mathcal{X}_N^{n_N} \to \mathcal{Z}^{N\times R}$ is (ϵ, δ)-shuffle DP (SDP) if for all ρ-adjacent databases $\mathbf{X}, \mathbf{X}' \in \mathcal{X}_1^{n_1} \times \cdots \times \mathcal{X}_N^{n_N}$ and all measurable subsets S, the collection of all uniformly randomly permuted messages that are sent by the shuffler satisfies (3), with $\rho(\mathbf{X}, \mathbf{X}') := \sum_{i=1}^N \sum_{j=1}^{n_i} \mathbb{1}_{x_{i,j}\neq x_{i,j}'}$ (same ρ as CDP).*

That is, SDP prohibits the server from viewing non-private functions of clients' data, instead restricting clients to randomize their own data and use shuffling. Since (ϵ_0, δ_0)-LDP implies (ϵ_0, δ_0)-CDP and shuffling can be seen as post-processing, it follows that (ϵ_0, δ_0)-*LDP implies* (ϵ_0, δ_0)-*SDP*.

Notation and Assumptions: Denote by $\|\cdot\|$ the Euclidean norm. A differentiable function $g : \mathcal{W} \to \mathbb{R}$ ($\mathcal{W} \subseteq \mathbb{R}^d$) is *$\mu$-strongly convex* ($\mu \geqslant 0$) if $g(w) \geqslant g(w') + \langle \nabla g(w'), w - w' \rangle + \frac{\mu}{2}\|w - w'\|^2 \ \forall \ w, w' \in \mathcal{W}$. If $\mu = 0$, we say g is *convex*. A function $h : \mathcal{W} \to \mathbb{R}^m$ is *L-Lipschitz* if $\|h(w) - h(w')\| \leqslant L\|w - w'\|$ for all $w, w' \in \mathcal{W}$. h is *β-smooth* if its derivative ∇h is β-Lipschitz. Denote $w^* \in \text{argmin}_{w\in\mathcal{W}} F(w)$. We write $a \lesssim b$ if $\exists C > 0$ such that $a \leqslant Cb$. We write $a = \widetilde{O}(b)$ if $a \lesssim \log(\theta)b$ for some parameters θ. We assume the following throughout this work:

Assumption 1. *$f(\cdot, x)$ is L_i-Lipschitz and μ-strongly convex (with $\mu = 0$ for convex) $\forall x \in \mathcal{X}_i$.*

Assumption 2. *In each round r, a uniformly random subset S_r of $M_r \in [N]$ distinct clients can communicate with the server, where $\{M_r\}_{r\geqslant 0}$ are i.i.d. random variables with $\frac{1}{M} := \mathbb{E}(\frac{1}{M_r})$.*

Assumption 2 is more realistic and general than existing FL works, which usually assume $M_r = M$ is deterministic [40]. M_r is determined by the network and is not a design parameter: M_r is the number

[2] Ultimately, the algorithm $\mathcal{A}$ may output some $\widehat{w} \in \mathcal{W}$ that is a function of the client transcripts $(\mathbf{Z}_1, \cdots, \mathbf{Z}_R)$. By the post-processing property of DP [20, Proposition 2.1], the privacy of $\widehat{w}$ will be guaranteed if the client transcripts are DP. Thus, here we simply consider the output of $\mathcal{A}$ to be the client transcripts.

[3] Assume that client reports can be decrypted by the server, but not by the shuffler [21, 25].

of clients that are able to contribute to global updates in each round, which is not the same as the number of clients that a given algorithm queries in each round. For example, the one-pass sequential algorithms of [18, 65] query just $O(1)$ clients in each round, but require $M_r = N$ to implement since they dictate exactly which client(s) must communicate with the server in every round and do not allow any client to contribute more than one report.

Related Work and Our Contributions: Below we discuss the most relevant related work and describe our main contributions. See Appendix A for additional discussion of related work.

1. Tight minimax risk bounds for LDP Federated SCO with i.i.d. clients and reliable communication (Theorem D.1, Theorem 2.1, and Theorem 3.1): [18] studied a special case of the i.i.d. ($\mathcal{D}_i = \mathcal{D}$) FL problem with $n = 1$ and $M = N$, establishing minimax risk bounds for the class of *sequentially interactive* ϵ_0-LDP algorithms and convex, Lipschitz loss functions. We consider the (ϵ_0, δ_0)-LDP ($\delta_0 > 0$) i.i.d. FL problem in a more general setting where *each client has an arbitrary number of samples ($n \geqslant 1$)* and establish tight minimax risk bounds for two Lipschitz function classes–*strongly convex and convex*. Crucially, our minmax risk bounds hold with respect to a wider class of algorithms than that considered by [18]; in addition to sequentially interactive algorithms, our algorithm class also includes a broad subset of *fully interactive algorithms*. These risk bounds match (up to logarithms) the respective non-private rates if $\frac{d}{\epsilon_0^2} \lesssim n$ ("privacy for free").

2. The first non-trivial upper bound for LDP SCO with non-i.i.d. clients (Theorem 2.2): For the challenging problem of LDP FL (SCO) with *non-i.i.d.* client distributions, we develop an *accelerated* distributed noisy SGD algorithm based on [30], which runs in linear time and obtains the first non-trivial risk bound for smooth convex/strongly convex loss. Unsurprisingly, our non-i.i.d. bound does not match the i.i.d. minimax rate, leading to the open question of what the minimax rate is for non-i.i.d. LDP FL.

3. Improved communication complexity for LDP Federated ERM, plus matching lower bounds for a subset of LDP algorithms (Theorem D.2, Theorem 2.3, Section 3 and Theorem E.4): The special case of problem (2), which we refer to as LDP Federated ERM, has been extensively studied in recent years [70, 37, 36, 76, 73, 17, 78, 5, 64, 32], but only one of these works, [32], provides an algorithm that achieves a tight upper bound. [32] focuses primarily on the shuffled model of DP, but we observe that their algorithm also yields an (ϵ_0, δ_0)-LDP empirical risk bound for convex loss. We employ a variation of our accelerated LDP algorithm–combined with Nesterov smoothing [60] in the non-smooth case–to achieve the same risk bound as [32] in fewer rounds of server communication. We also address strongly convex ERM. Further, we provide matching lower bounds, implying that our algorithm is optimal among a subclass of fully interactive LDP algorithms.

4. Achieving the optimal CDP i.i.d. SCO rates without a trusted curator (Theorem 4.1): [32, 21] showed that the optimal CDP convex federated ERM rate [9] can be attained in the lower trust (relative to the central model) shuffle model of DP. The concurrent work [13, Theorem 4.9] shows that when all $M = N$ clients can communicate in every round and each client has just $n = 1$ sample, the optimal CDP SCO rate can be attained with an SDP algorithm. We show that our algorithm (with shuffling) attains the optimal CDP SCO rates for clients with any number $n \geqslant 1$ of samples. In particular, with shuffling, *our algorithm is simultaneously optimal in both the local and central models of DP for i.i.d. FL.* We also provide upper bounds for $M < N$.

In non-private distributed optimization, [51, 68, 58] provide convergence results with random connectivity graphs. Our upper bounds describe the effect of the mean/variance of $1/M_r$ on DP FL. Tight lower bounds for DP FL when $M < N$ is an open problem stemming from our work.

Finally, numerical experiments demonstrate the practical performance of our algorithm.

2 Upper Bounds for LDP FL

To simplify the presentation of our results, we will assume that $n_i = n$, $\epsilon_i = \epsilon_0$, $\delta_i = \delta_0$, and $L_i = L$ for all i. Appendix D contains the general versions of these upper bounds and their proofs.

Noisy Minibatch SGD for i.i.d. Clients: Consider the case of i.i.d. clients: $\mathcal{X}_i = \mathcal{X}$, $\mathcal{D}_i = \mathcal{D}$ for all i. We derive tight loss bounds via Algorithm 1 (Appendix D.2), an LDP version of distributed minibatch SGD (MB-SGD). In each round r, all M_r available clients send noisy stochastic gradients to the server: $\widetilde{g}_r^i := \frac{1}{K_i} \sum_{j=1}^{K_i} \nabla f(w_r, x_{i,j}^r) + u_i$, where $u_i \sim N(0, \sigma_i^2 \mathbf{I}_d)$ and $\{x_{i,j}\}_{j \in [K_i]}$ are drawn uniformly from X_i (and then replaced). The server averages these M_r reports, updates

$w_{r+1} := \Pi_{\mathcal{W}}[w_r - \frac{\eta_r}{M_r}\sum_{i \in S_r} \widetilde{g}_r^i]$ and reports w_{r+1} to all N clients. Algorithm 1 can also be seen as a distributed version of [8, Algorithm 1]. We now give privacy and excess population loss guarantees for Algorithm 1, run with $\sigma_i^2 := \frac{256L^2 R \ln(2.5R/\delta_0)\ln(2/\delta_0)}{n^2\epsilon_0^2}$:

Theorem 2.1. [Informal] *Assume $\epsilon_0 \leqslant \ln(2/\delta_0)$ and choose $K \geqslant \frac{\epsilon_0 n}{4\sqrt{2R\ln(2/\delta_0)}}$. Then Algorithm 1 is (ϵ_0, δ_0)-LDP. Further, there exists $\beta > 0$ such that running Algorithm 1 on $f_\beta(w) := \min_{v \in \mathcal{W}}\left(f(v) + \frac{\beta}{2}\|w - v\|^2\right)$ yields:*

1. (Convex) Setting $R = M\min\left\{n, \frac{\epsilon_0^2 n^2}{d}\right\}$ yields

$$\mathbb{E}F(\widehat{w}_R) - F(w^*) = \widetilde{O}\left(\frac{LD}{\sqrt{M}}\left(\frac{1}{\sqrt{n}} + \frac{\sqrt{d\ln(2/\delta_0)}}{\epsilon_0 n}\right)\right). \tag{4}$$

2. (μ-strongly convex) Setting $R = M\min\left\{n, \frac{\epsilon_0^2 n^2}{d}\right\}\ln\left(\frac{D^2\mu^2 M^2\epsilon_0^2 n^3}{dL^2}\right)$ yields

$$\mathbb{E}F(\widehat{w}_R) - F(w^*) = \widetilde{O}\left(\frac{L^2}{\mu M}\left(\frac{1}{n} + \frac{d\ln(2/\delta_0)}{\epsilon_0^2 n^2}\right)\right). \tag{5}$$

We will see (Section 3) that these upper bounds are tight up to logarithmic factors when $M = N$, implying that Algorithm 1 is optimal among a large class of fully interactive LDP algorithms. Further, if $M = N$ and $\frac{d}{\epsilon_0^2} \lesssim n$, then both of these upper bounds match (up to logarithms) the respective non-private lower bounds for SCO ("privacy for free") [59, 3, 34]. The proof of Theorem 2.1 uses the corresponding smooth result Theorem D.1 in the Appendix and Nesterov smoothing [60]. Theorem D.1 is proved by bounding the empirical loss and using a *uniform stability* [11] argument, similar to how [8] proceeded for the case $N = 1$.

One-pass Accelerated Noisy Distributed SGD for Non-i.i.d. Clients: Consider the general **non-i.i.d.** FL problem, where $F_i(w)$ takes the general form (1) for some unknown distributions $\mathcal{D}_i$ on $\mathcal{X}_i$ ($i \in [N]$). The uniform stability approach that we used to obtain our i.i.d. upper bounds does not work in this setting. [4] Instead, we directly minimize F by modifying Algorithm 1 as follows:
1. We draw $K := 1$ local samples *without replacement* and set $R := n$. Thus, each sample is used at most one time during the algorithm, so that the bounds we obtain apply to F.
2. We use *acceleration* to increase the convergence rate.
3. To provide LDP, we use Gaussian *noise with larger variance* $\sigma^2 = \frac{8L^2\ln(1.25/\delta_0)}{\epsilon_0^2}$.
We call this algorithm **One-pass Accelerated Noisy Distributed SGD**. It is an instantiation of Accelerated Noisy MB-SGD, described in Algorithm 2 (Appendix D.4). Algorithm 2 is a noisy LDP version of the accelerated MB-SGD of [30], which was analyzed in the distributed setting by [75].

Theorem 2.2. [Informal] *Let $f(\cdot, x)$ be β-smooth for all $x \in \mathcal{X}$. Assume $\epsilon_0 \leqslant 1$. Then One-pass Accelerated Noisy Distributed SGD is (ϵ_0, δ_0)-LDP. Moreover:*
1. If $f(\cdot, x)$ is convex, then

$$\mathbb{E}F(\widehat{w}_R) - F(w^*) = O\left(\frac{\beta D^2}{n^2} + \frac{LD\sqrt{d\ln(1/\delta_0)}}{\epsilon_0\sqrt{Mn}}\right). \tag{6}$$

2. If $f(\cdot, x)$ is μ-strongly convex, then

$$\mathbb{E}F(\widehat{w}_R) - F(w^*) = O\left(LD\exp\left(-\sqrt{\frac{\mu}{\beta}}n\right) + \frac{L^2}{\mu}\frac{d\ln(1/\delta_0)}{\epsilon_0^2 Mn}\right). \tag{7}$$

This upper bound is looser than the optimal bounds of Theorem D.1 (yet the tightest known bound for non-i.i.d. LDP FL), leaving open the question of what the optimal rate is for non-i.i.d. LDP FL.

Accelerated Noisy MB-SGD for Federated ERM: With non-random $M_r = M$, [32] provides an upper bound for convex LDP ERM that nearly matches the one we provide below.[5] We use an accelerated LDP algorithm, Algorithm 2 (Appendix D.4), which achieves the upper bounds for convex and strongly convex loss in fewer rounds of communications than [32]. Unlike the one-pass version of Algorithm 2 used for non-i.i.d. FL, here we sample local minibatches from each client *with replacement* to get tighter (in fact, optimal) bounds. Here we present just the non-smooth result:

[4] Specifically, Lemma D.1 in the Appendix does not apply without the i.i.d. assumption.

[5] The bound in [32] is looser than the bound in (35) by logarithmic factor.

Theorem 2.3. [Informal] *Assume $\epsilon_0 \leqslant \ln(2/\delta_0)$ and choose $K \geqslant \frac{\epsilon_0 n}{4\sqrt{2R\ln(2/\delta_0)}}$. Then Algorithm 2 is (ϵ_0, δ_0)-LDP. Further, there is $\beta > 0$ such that running Algorithm 2 on $f_\beta(w) := \min_{v \in \mathcal{W}} \left(f(v) + \frac{\beta}{2}\|w - v\|^2\right)$ yields the following bounds on the excess empirical loss (w.r.t. f):*

1. (Convex) Setting $R = \max\left(\frac{\sqrt{M}\epsilon_0 n}{\sqrt{d}}, \frac{\epsilon_0^2 n^2}{d} \begin{cases} \frac{1}{K} & \textit{if } M = N \\ 1 & \textit{otherwise} \end{cases}\right)$ yields

$$\mathbb{E}\widehat{F}(\widehat{w}_R) - \widehat{F}(w^*) = \widetilde{O}\left(LD\left(\frac{\sqrt{d\ln(2/\delta_0)}}{\epsilon_0 n\sqrt{M}}\right)\right). \tag{8}$$

2. (μ-strongly convex) Setting $R = \max\left(\frac{\sqrt{M}\epsilon_0 n}{\sqrt{d}} \ln\left(\frac{D\mu M\epsilon_0^2 n^2}{Ld}\right), \frac{\epsilon_0^2 n^2}{d} \begin{cases} \frac{1}{K} & \textit{if } M = N \\ 1 & \textit{otherwise} \end{cases}\right)$ yields

$$\mathbb{E}\widehat{F}(\widehat{w}_R) - \widehat{F}(w^*) = \widetilde{O}\left(\frac{L^2}{\mu}\left(\frac{d\ln(2/\delta_0)}{\epsilon_0^2 n^2 M}\right)\right). \tag{9}$$

The upper bounds in Theorem 2.3 match (up to logarithms) the lower bounds in Theorem E.4 when $M = N$, establishing the optimality of Algorithm 2 for convex/strongly convex Federated ERM among a wide subclass of fully interactive algorithms. The algorithm in [32] was not analyzed for random M_r. For fixed $M_r = M$, their algorithm requires $R = \widetilde{\Omega}(\epsilon_0^2 n^2 M/d)$ communication rounds to attain (8), making Algorithm 2 faster by a factor of $\min(\sqrt{M}\epsilon_0 n/\sqrt{d}, M)$.

3 Lower Bounds for LDP FL

We provide tight lower bounds on the excess population/empirical loss of LDP algorithms for the federated SCO/ERM problems when $M = N$. As a consequence, Algorithm 1 and Algorithm 2 are minimax optimal for i.i.d. SCO and ERM respectively, for two function classes: $\mathcal{F}_{L,D} := \{f : \mathcal{W} \times \mathcal{X} \to \mathbb{R}^d \mid \forall x \in \mathcal{X}\ f(\cdot, x)$ is convex, L-Lipschitz, and $\mathcal{W} \subseteq B_2(0, D)\}$; and $\mathcal{G}_{\mu,L,D} := \{f \in \mathcal{F}_{L,D} \mid \forall x \in \mathcal{X}\ f(\cdot, x)$ is μ-strongly convex$\}$. For SCO, the (ϵ_0, δ_0)-LDP algorithm class $\mathbb{A}_{(\epsilon_0,\delta_0)} = \mathbb{A}$ that we consider contains all sequentially interactive algorithms, as well as fully interactive algorithms that are compositional (c.f. [39]):

Definition 4. *Let $\mathcal{A}$ be an R-round (ϵ_0, δ_0)-LDP FL algorithm with data domain $\mathcal{X}$. Let $\{(\epsilon_r^0, \delta_r^0)\}_{r=1}^R$ denote the minimal (non-negative) parameters of the local randomizers $\mathcal{R}_r^{(i)}$ selected at round r ($r \in [R]$) such that $\mathcal{R}_r^{(i)}(\mathbf{Z}_{(1:r-1)}, \cdot) : \mathcal{X}^n \to \mathcal{Z}$ is $(\epsilon_r^0, \delta_r^0)$-DP for all $i \in [N]$ and all $\mathbf{Z}_{(1:r-1)} \in \mathcal{Z}^{(r-1)N}$. For an absolute constant $C > 0$, we say that $\mathcal{A}$ is C-compositional if $\sqrt{\sum_{r\in[R]}(\epsilon_0^r)^2} \leqslant C\epsilon_0$. If such a C exists, we simply say $\mathcal{A}$ is compositional.*

Any (ϵ_0, δ_0)-LDP $\mathcal{A}$ has $\epsilon_0^r \leqslant \epsilon_0$. The vast majority of (ϵ_0, δ_0)-LDP algorithms studied in the literature satisfy Definition 4. For example, any algorithm that uses the strong composition theorems of [20, Thm. 3.20] or [41] for its privacy analysis is 1-compositional. In particular, the three algorithms presented in Section 2 are 1-compositional, hence they are in $\mathbb{A}$. See also Appendix E.1.

Theorem 3.1. *Let $n, d, N, R \in \mathbb{N}$, $\epsilon_0 \in (0, \sqrt{N}]$, $\delta_0 \in (0, 1)$ and $\mathcal{A} \in \mathbb{A}_{(\epsilon_0,\delta_0)}$ such that in every round $r \in [R]$, the local randomizers $\mathcal{R}_r^{(i)}(\mathbf{Z}_{(1:r-1)}, \cdot) : \mathcal{X}^n \to \mathcal{Z}$ are $(\epsilon_0^r, \delta_0^r)$-DP for all $i \in [N]$, $\mathbf{Z}_{(1:r-1)} \in \mathcal{Z}^{r-1\times N}$, with $\epsilon_0^r \leqslant \frac{1}{n}$, and $N \geqslant 16\ln(2/\delta_0^{\min} n)$, where $\delta_0^{\min} := \min_r \delta_0^r$. If $\mathcal{A}$ is if $\mathcal{A}$ is sequentially interactive, assume $\delta_0 = o(1/n^2N^2)$; if $\mathcal{A}$ is compositional, assume $\sum_r \delta_0^r = o(1/n^2N^2)$ instead. Then there exists a β-smooth ($\forall\beta \geqslant 0$) loss $f \in \mathcal{F}_{L,D}$ and a distribution $\mathcal{D}$ on a set $\mathcal{X}$ such that if the local data sets are drawn i.i.d. $X_i \sim \mathcal{D}^n$, then:*

$$\mathbb{E}F(\mathcal{A}(\mathbf{X})) - F(w^*) = \widetilde{\Omega}\left(LD\left(\frac{1}{\sqrt{Nn}} + \min\left\{1, \frac{\sqrt{d}}{\epsilon_0 n\sqrt{N}}\right\}\right)\right). \tag{10}$$

Furthermore, there exists another (μ-smooth) $f \in \mathcal{G}_{\mu,L,D}$ and distribution $\mathcal{D}$ such that

$$\mathbb{E}F(\mathcal{A}(\mathbf{X})) - F(w^*) = \widetilde{\Omega}\left(\frac{L^2}{\mu nN} + LD\min\left\{1, \frac{d}{\epsilon_0^2 n^2 N}\right\}\right). \tag{11}$$

These lower bounds are essentially tight [6] by Theorem 2.1. The first term in each of the lower bounds is the optimal non-private rate; the second parts of the bounds are what we prove in Appendix E.2.

[6] Up to logarithms, and for strongly convex case–a factor of $\mu D/L$. If $d > \epsilon_0^2 n^2 N$, then the trivial algorithm attains the matching upper bound $O(LD)$.

Theorem 3.1 is more generally applicable than the lower bound in [18, Proposition 3] (for the L_2 setting), which only applies to sequentially interactive algorithms and databases with $n = 1$ sample per client. As in [18], our lower bounds hold for sufficiently private LDP algorithms–those for which the privacy loss on each client in each round $\epsilon_0^r \lesssim 1/n$ [7]. For $n = 1$, $\epsilon_0^r \lesssim 1/n = 1$ is also required in [18, Proposition 3] since they consider sequential algorithms, where $\epsilon_0^r = \epsilon_0$. Also, the assumptions on δ_0, δ_0^r are not very restrictive in practice; see Remark E.1. For federated ERM, we prove a tight lower bound in Theorem E.4, establishing the optimality of Algorithm 2 for a class $\mathbb{B} \subset \mathbb{A}$.

4 Optimal Algorithm for Shuffle DP FL

Assume access to a secure shuffler and fix $M_r = M \in [N]$. In each round r, the shuffler receives the reports $(Z_r^{(1)}, \cdots Z_r^{(M)})$ from active clients (we assume $S_r = [M]$ here for concreteness), draws a uniformly random permutation of $[M]$, π, and then sends $(Z_r^{(\pi(1))}, \cdots, Z_r^{(\pi(M))})$ to the server for aggregation. We show that this "shuffled version" of Algorithm 1 achieves the optimal convex/strongly convex CDP bounds for i.i.d. SCO when $M = N$, even with the weaker trust assumptions of the shuffle model:

Theorem 4.1. *Let $f : \mathcal{W} \times \mathcal{X} \to \mathbb{R}^d$ be β-smooth, $\epsilon \leqslant \ln(2/\delta)$, $\delta \in (0, 1)$, and $M \geqslant 16\ln(18RM^2/N\delta)$ for R specified below. Then $\exists C > 0$ such that $\sigma_i^2 := \frac{CL^2RM\ln(RM^2/N\delta)\ln(R/\delta)\ln(1/\delta)}{n^2N^2\epsilon^2}$ ensures that the shuffled Algorithm 1 is (ϵ, δ)-CDP $\forall K \in [n]$. Further:*

1. (Convex) Setting $R := \max\left(\frac{n^2N^2\epsilon^2}{M}, \frac{N}{M}, \min\left\{n, \frac{\epsilon^2n^2N^2}{dM}\right\}, \frac{\beta D}{L}\min\left\{\sqrt{nM}, \frac{\epsilon nN}{\sqrt{d}}\right\}\right)$ yields

$$\mathbb{E}F(\widehat{w}_R) - F(w^*) = O\left(LD\left(\frac{1}{\sqrt{nM}} + \frac{\sqrt{d\ln(1/\delta)}}{\epsilon nN}\right)\right). \tag{12}$$

2. (Strongly convex) $R := \max\left(\frac{n^2N^2\epsilon^2}{M}, \frac{N}{M}, \frac{8\beta}{\mu}\ln\left(\frac{\beta D^2\mu\epsilon^2n^2N^2}{dL^2}\right), \min\left\{n, \frac{\epsilon^2n^2N^2}{dM}\right\}\right)$ yields

$$\mathbb{E}F(\widehat{w}_R) - F(w^*) = \widetilde{O}\left(\frac{L^2}{\mu}\left(\frac{1}{nM} + \frac{d\ln(1/\delta)}{\epsilon^2n^2N^2}\right)\right). \tag{13}$$

Together with our LDP results, Theorem 4.1 implies that, with shuffling, *Algorithm 1 is simultaneously optimal for i.i.d. FL in the local and central models of DP* if $M = N$. Nesterov smoothing can be used to obtain the same bounds without the β-smoothness assumption, similar to how we proceeded for Theorem 2.1. We omit the details here.

5 Numerical Experiments

Linear Regression with Health Insurance Data: We divide the data set [15] into N heterogeneous groups based on the level of the target (medical charges). See Appendix G.1 for additional details. Fig. 2 shows that *LDP MB-SGD outperforms LDP Local SGD across all privacy levels*. Also, as $\epsilon \uparrow 10$, the price of LDP shrinks towards 0. Here, υ_*^2 describes the amount of heterogeneity between clients' data (increasing with heterogeneity); see Appendix D.1 for the precise definition.

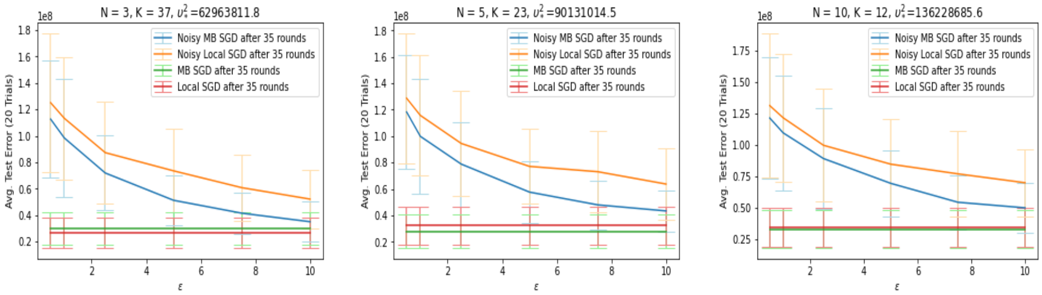

Figure 2: Test error vs. ϵ for linear regression on heterogeneous health insurance data. We display 90% error bars over the 20 trials (train/test splits). $\delta = 1/n^2$; $n = 1070/N$ is the number of training examples per client.

Logistic Regression with MNIST: See Appendix G for experiments with $M < N$. Algorithm 1 still uniformly outperform LDP Local SGD and even outperforms *non-private* Local SGD for $\epsilon \geqslant 12.5$.

[7] If there are $N = O(1)$ clients, then existing CDP lower bounds [8] apply and match our LDP upper bounds as long as $\epsilon_0 \lesssim 1$; thus, this restriction on ϵ_0^r disappears if $N = O(1)$.

Acknowledgements

We would like to thank Tianjian Huang, Dmitrii Ostrovskii, and Adam Smith for helpful comments and conversations.

References

[1] Webank and swiss re signed cooperation mou. *Fed AI Ecosystem*, Nov 2019.

[2] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang. Deep learning with differential privacy. *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, Oct 2016.

[3] A. Agarwal, P. L. Bartlett, P. Ravikumar, and M. J. Wainwright. Information-theoretic lower bounds on the oracle complexity of stochastic convex optimization. *IEEE Transactions on Information Theory*, 58(5):3235–3249, 2012.

[4] Apple. Private federated learning. *NeurIPS 2019 Expo Talk Abstract*, 2019.

[5] P. C. M. Arachchige, P. Bertok, I. Khalil, D. Liu, S. Camtepe, and M. Atiquzzaman. Local differential privacy for deep learning. *IEEE Internet of Things Journal*, 7(7):5827–5842, 2019.

[6] 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, 2019.

[7] B. Balle, P. Kairouz, B. McMahan, O. D. Thakkar, and A. Thakurta. Privacy amplification via random check-ins. 33, 2020.

[8] R. Bassily, V. Feldman, K. Talwar, and A. Thakurta. Private stochastic convex optimization with optimal rates. In *Advances in Neural Information Processing Systems*, 2019.

[9] 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.

[10] A. Bittau, U. Erlingsson, P. Maniatis, I. Mironov, A. Raghunathan, D. Lie, M. Rudominer, U. Kode, J. Tinnes, and B. Seefeld. Prochlo: Strong privacy for analytics in the crowd. In *Proceedings of the Symposium on Operating Systems Principles (SOSP)*, pages 441–459, 2017.

[11] O. Bousquet and A. Elisseeff. Stability and generalization. *The Journal of Machine Learning Research*, 2:499–526, 2002.

[12] Y.-R. Chen, A. Rezapour, and W.-G. Tzeng. Privacy-preserving ridge regression on distributed data. *Information Sciences*, 451:34–49, 2018.

[13] A. Cheu, M. Joseph, J. Mao, and B. Peng. Shuffle private stochastic convex optimization. *arXiv preprint arXiv:2106.09805*, 2021.

[14] A. Cheu, A. Smith, J. Ullman, D. Zeber, and M. Zhilyaev. Distributed differential privacy via shuffling. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 375–403. Springer, 2019.

[15] M. Choi. Medical insurance charges data. 2018. `https://www.kaggle.com/mirichoi0218/insurance`.

[16] P. Courtiol, C. Maussion, M. Moarii, E. Pronier, S. Pilcer, M. Sefta, P. Manceron, S. Toldo, M. Zaslavskiy, and N. Le Stang. Deep learning-based classification of mesothelioma improves prediction of patient outcome. *Nature Medicine*, page 1–7, 2019.

[17] R. Dobbe, Y. Pu, J. Zhu, K. Ramchandran, and C. Tomlin. Customized local differential privacy for multi-agent distributed optimization, 2020.

[18] 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, 2013.

[19] C. Dwork. Differential privacy. In *International Colloquium on Automata, Languages, and Programming*, pages 1–12. Springer, 2006.

[20] C. Dwork and A. Roth. *The Algorithmic Foundations of Differential Privacy*. 2014.

[21] 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.

[22] U. Erlingsson, V. Feldman, I. Mironov, A. Raghunathan, K. Talwar, and A. Thakurta. Amplification by shuffling: From local to central differential privacy via anonymity, 2020.

[23] M. S. Exchange. Total variation distance of two random vectors whose components are independent. `https://math.stackexchange.com/questions/1558845/total-variation-distance-of-two-random-vectors-whose-components-are-independent`.

[24] V. Feldman, T. Koren, and K. Talwar. Private stochastic convex optimization: optimal rates in linear time. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pages 439–449, 2020.

[25] V. Feldman, A. McMillan, and K. Talwar. Hiding among the clones: A simple and nearly optimal analysis of privacy amplification by shuffling, 2020.

[26] V. Feldman and J. Vondrak. High probability generalization bounds for uniformly stable algorithms with nearly optimal rate. In A. Beygelzimer and D. Hsu, editors, *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pages 1270–1279, Phoenix, USA, 25–28 Jun 2019. PMLR.

[27] M. Fredrikson, S. Jha, and T. Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1322–1333, 2015.

[28] S. Gade and N. H. Vaidya. Privacy-preserving distributed learning via obfuscated stochastic gradients. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 184–191. IEEE, 2018.

[29] R. C. Geyer, T. Klein, and M. Nabi. Differentially private federated learning: A client level perspective. *CoRR*, abs/1712.07557, 2017.

[30] S. Ghadimi and G. Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization i: A generic algorithmic framework. *SIAM Journal on Optimization*, 22(4):1469–1492, 2012.

[31] S. Ghadimi and G. Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, ii: Shrinking procedures and optimal algorithms. *SIAM Journal on Optimization*, 23(4):2061–2089, 2013.

[32] A. Girgis, D. Data, S. Diggavi, P. Kairouz, and A. Theertha Suresh. Shuffled model of differential privacy in federated learning. In A. Banerjee and K. Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 2521–2529. PMLR, 13–15 Apr 2021.

[33] M. Hardt, B. Recht, and Y. Singer. Train faster, generalize better: Stability of stochastic gradient descent. In M. F. Balcan and K. Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1225–1234, New York, New York, USA, 20–22 Jun 2016. PMLR.

[34] E. Hazan and S. Kale. Beyond the regret minimization barrier: optimal algorithms for stochastic strongly-convex optimization. *The Journal of Machine Learning Research*, 15(1):2489–2512, 2014.

[35] Z. He, T. Zhang, and R. B. Lee. Model inversion attacks against collaborative inference. In *Proceedings of the 35th Annual Computer Security Applications Conference*, pages 148–162, 2019.

[36] Z. Huang and Y. Gong. Differentially private ADMM for convex distributed learning: Improved accuracy via multi-step approximation. *arXiv preprint:2005.07890*, 2020.

[37] Z. Huang, R. Hu, Y. Guo, E. Chan-Tin, and Y. Gong. DP-ADMM: ADMM-based distributed learning with differential privacy. *IEEE Transactions on Information Forensics and Security*, 15:1002–1012, 2020.

[38] B. Jayaraman and L. Wang. Distributed learning without distress: Privacy-preserving empirical risk minimization. *Advances in Neural Information Processing Systems*, 2018.

[39] M. Joseph, J. Mao, S. Neel, and A. Roth. The role of interactivity in local differential privacy. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 94–105. IEEE, 2019.

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

[41] P. Kairouz, S. Oh, and P. Viswanath. The composition theorem for differential privacy, 2015.

[42] G. Kamath. Cs 860: Algorithms for private data analysis, 2020. `http://www.gautamkamath.com/CS860notes/lec5.pdf`.

[43] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh. SCAFFOLD: Stochastic controlled averaging for federated learning. In H. D. III and A. Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5132–5143. PMLR, 13–18 Jul 2020.

[44] 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.

[45] A. Khaled, K. Mishchenko, and P. Richtárik. Better communication complexity for local SGD. *arXiv preprint*, 2019.

[46] A. Koloskova, N. Loizou, S. Boreiri, M. Jaggi, and S. Stich. A unified theory of decentralized SGD with changing topology and local updates. In H. D. III and A. Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5381–5393. PMLR, 13–18 Jul 2020.

[47] D. Levy, Z. Sun, K. Amin, S. Kale, A. Kulesza, M. Mohri, and A. T. Suresh. Learning with user-level privacy. *arXiv preprint arXiv:2102.11845*, 2021.

[48] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang. On the convergence of FedAvg on non-iid data. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, 2020.

[49] J. Liu and K. Talwar. Private selection from private candidates. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 298–309, 2019.

[50] R. Liu, Y. Cao, H. Chen, R. Guo, and M. Yoshikawa. Flame: Differentially private federated learning in the shuffle model. In *AAAI*, 2020.

[51] I. Lobel and A. Ozdaglar. Distributed subgradient methods for convex optimization over random networks. *IEEE Transactions on Automatic Control*, 56(6):1291–1306, 2010.

[52] A. Lowy and M. Razaviyayn. Output perturbation for differentially private convex optimization with improved population loss bounds, runtimes and applications to private adversarial training. *arXiv preprint:2102.04704*, 2021.

[53] X. Ma, F. Zhang, X. Chen, and J. Shen. Privacy preserving multi-party computation delegation for deep learning in cloud computing. *Information Sciences*, 459:103–116, 2018.

[54] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, 2017.

[55] B. McMahan, D. Ramage, K. Talwar, and L. Zhang. Learning differentially private recurrent language models. In *International Conference on Learning Representations (ICLR)*, 2018.

[56] F. D. McSherry. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, pages 19–30, 2009.

[57] J. Murtagh and S. Vadhan. The complexity of computing the optimal composition of differential privacy. In *Theory of Cryptography Conference*, pages 157–175. Springer, 2016.

[58] A. Nedic, A. Olshevsky, and W. Shi. Achieving geometric convergence for distributed optimization over time-varying graphs. *SIAM Journal on Optimization*, 27(4):2597–2633, 2017.

[59] A. S. Nemirovskii and D. B. Yudin. *Problem Complexity and Method Efficiency in Optimization*. 1983.

[60] Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical programming*, 103(1):127–152, 2005.

[61] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor. Federated learning for internet of things: A comprehensive survey. *IEEE Communications Surveys and Tutorials*, page 1–1, 2021.

[62] N. Papernot and T. Steinke. Hyperparameter tuning with renyi differential privacy, 2021.

[63] S. Pichai. Google's Sundar Pichai: Privacy should not be a luxury good. *The New York Times*, May 2019.

[64] M. Seif, R. Tandon, and M. Li. Wireless federated learning with local differential privacy. In *2020 IEEE International Symposium on Information Theory (ISIT)*, pages 2604–2609, 2020.

[65] A. Smith, A. Thakurta, and J. Upadhyay. Is interaction necessary for distributed private learning? In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 58–77, 2017.

[66] M. Song, Z. Wang, Z. Zhang, Y. Song, Q. Wang, J. Ren, and H. Qi. Analyzing user-level privacy attack against federated learning. *IEEE Journal on Selected Areas in Communications*, 38(10):2430–2444, 2020.

[67] S. U. Stich. Unified optimal analysis of the (stochastic) gradient method. *arXiv preprint:1907.04232*, 2019.

[68] B. Touri and B. Gharesifard. Continuous-time distributed convex optimization on time-varying directed networks. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 724–729, 2015.

[69] S. Truex, N. Baracaldo, A. Anwar, T. Steinke, H. Ludwig, R. Zhang, and Y. Zhou. A hybrid approach to privacy-preserving federated learning. In *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security*, pages 1–11, 2019.

[70] S. Truex, L. Liu, K.-H. Chow, M. E. Gursoy, and W. Wei. LDP-Fed: federated learning with local differential privacy. In *Proceedings of the Third ACM International Workshop on Edge Systems, Analytics and Networking*, page 61–66. Association for Computing Machinery, 2020.

[71] J. Ullman. CS7880: rigorous approaches to data privacy, 2017. `http://www.ccs.neu.edu/home/jullman/cs7880s17/HW1sol.pdf`.

[72] K. Wei, J. Li, M. Ding, C. Ma, H. Su, B. Zhang, and H. V. Poor. User-level privacy-preserving federated learning: Analysis and performance optimization. *arXiv preprint:2003.00229*, 2020.

[73] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. Quek, and H. V. Poor. Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security*, 15:3454–3469, 2020.

[74] B. Woodworth, K. K. Patel, S. Stich, Z. Dai, B. Bullins, B. Mcmahan, O. Shamir, and N. Srebro. Is local SGD better than minibatch SGD? In *International Conference on Machine Learning*, pages 10334–10343. PMLR, 2020.

[75] B. E. Woodworth, K. K. Patel, and N. Srebro. Minibatch vs local sgd for heterogeneous distributed learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6281–6292. Curran Associates, Inc., 2020.

[76] N. Wu, F. Farokhi, D. Smith, and M. A. Kaafar. The value of collaboration in convex machine learning with differential privacy, 2019.

[77] H. Yuan and T. Ma. Federated accelerated stochastic gradient descent. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 5332–5344. Curran Associates, Inc., 2020.

[78] Y. Zhao, J. Zhao, M. Yang, T. Wang, N. Wang, L. Lyu, D. Niyato, and K.-Y. Lam. Local differential privacy based federated learning for internet of things. *IEEE Internet of Things Journal*, 2020.

[79] Y. Zhou and S. Tang. Differentially private distributed learning. *INFORMS Journal on Computing*, 32(3):779–789, 2020.

[80] L. Zhu and S. Han. Deep leakage from gradients. In *Federated learning*, pages 17–31. Springer, 2020.