Published as a conference paper at ICLR 2020

ON THE CONVERGENCE OF FEDAVG ON NON-IID DATA

Xiang Li*
School of Mathematical Sciences
Peking University
Beijing, 100871, China
smslixiang@pku.edu.cn

Kaixuan Huang*
School of Mathematical Sciences
Peking University
Beijing, 100871, China
hackyhuang@pku.edu.cn

Wenhao Yang*
Center for Data Science
Peking University
Beijing, 100871, China
yangwenhaosms@pku.edu.cn

Shusen Wang
Department of Computer Science
Stevens Institute of Technology
Hoboken, NJ 07030, USA
shusen.wang@stevens.edu

Zhihua Zhang
School of Mathematical Sciences
Peking University
Beijing, 100871, China
zhzhang@math.pku.edu.cn

ABSTRACT

Federated learning enables a large amount of edge computing devices to jointly learn a model without data sharing. As a leading algorithm in this setting, Federated Averaging (`FedAvg`) runs Stochastic Gradient Descent (SGD) in parallel on a small subset of the total devices and averages the sequences only once in a while. Despite its simplicity, it lacks theoretical guarantees under realistic settings. In this paper, we analyze the convergence of `FedAvg` on non-iid data and establish a convergence rate of $\mathcal{O}(\frac{1}{T})$ for strongly convex and smooth problems, where T is the number of SGDs. Importantly, our bound demonstrates a trade-off between communication-efficiency and convergence rate. As user devices may be disconnected from the server, we relax the assumption of full device participation to partial device participation and study different averaging schemes; low device participation rate can be achieved without severely slowing down the learning. Our results indicates that heterogeneity of data slows down the convergence, which matches empirical observations. Furthermore, we provide a necessary condition for `FedAvg` on non-iid data: the learning rate η must decay, even if full-gradient is used; otherwise, the solution will be $\Omega(\eta)$ away from the optimal.

1 INTRODUCTION

Federated Learning (FL), also known as federated optimization, allows multiple parties to collaboratively train a model without data sharing (Konevcnỳ et al., 2015; Shokri and Shmatikov, 2015; McMahan et al., 2017; Konevcnỳ, 2017; Sahu et al., 2018; Zhuo et al., 2019). Similar to the centralized parallel optimization (Jakovetic, 2013; Li et al., 2014a;b; Shamir et al., 2014; Zhang and Lin, 2015; Meng et al., 2016; Reddi et al., 2016; Richtárik and Takác, 2016; Smith et al., 2016; Zheng et al., 2016; Shusen Wang et al., 2018), FL let the user devices (aka worker nodes) perform most of the computation and a central parameter server update the model parameters using the descending directions returned by the user devices. Nevertheless, FL has three unique characters that distinguish it from the standard parallel optimization Li et al. (2019).

*Equal contribution.

First, the training data are massively distributed over an incredibly large number of devices, and the connection between the central server and a device is slow. A direct consequence is the slow communication, which motivated communication-efficient FL algorithms (McMahan et al., 2017; Smith et al., 2017; Sahu et al., 2018; Sattler et al., 2019). Federated averaging (`FedAvg`) is the first and perhaps the most widely used FL algorithm. It runs E steps of SGD in parallel on a small sampled subset of devices and then averages the resulting model updates via a central server once in a while.[1] In comparison with SGD and its variants, `FedAvg` performs more local computation and less communication.

Second, unlike the traditional distributed learning systems, the FL system does not have control over users' devices. For example, when a mobile phone is turned off or WiFi access is unavailable, the central server will lose connection to this device. When this happens during training, such a non-responding/inactive device, which is called a straggler, appears tremendously slower than the other devices. Unfortunately, since it has no control over the devices, the system can do nothing but waiting or ignoring the stragglers. Waiting for all the devices' response is obviously infeasible; it is thus impractical to require all the devices be active.

Third, the training data are non-iid[2], that is, a device's local data cannot be regarded as samples drawn from the overall distribution. The data available locally fail to represent the overall distribution. This does not only bring challenges to algorithm design but also make theoretical analysis much harder. While `FedAvg` actually works when the data are non-iid McMahan et al. (2017), `FedAvg` on non-iid data lacks theoretical guarantee even in convex optimization setting.

There have been much efforts developing convergence guarantees for FL algorithm based on the assumptions that (1) the data are iid and (2) all the devices are active. Khaled et al. (2019); Yu et al. (2019); Wang et al. (2019) made the latter assumption, while Zhou and Cong (2017); Stich (2018); Wang and Joshi (2018); Woodworth et al. (2018) made both assumptions. The two assumptions violates the second and third characters of FL. Previous algorithm `Fedprox` Sahu et al. (2018) doesn't require the two mentioned assumptions and incorporates `FedAvg` as a special case when the added proximal term vanishes. However, their theory fails to cover `FedAvg`.

Notation. Let N be the total number of user devices and K ($\leq N$) be the maximal number of devices that participate in every round's communication. Let T be the total number of every device's SGDs, E be the number of local iterations performed in a device between two communications, and thus $\frac{T}{E}$ is the number of communications.

Contributions. For strongly convex and smooth problems, we establish a convergence guarantee for `FedAvg` without making the two impractical assumptions: (1) the data are iid, and (2) all the devices are active. To the best of our knowledge, this work is the first to show the convergence rate of `FedAvg` without making the two assumptions.

We show in Theorem 1, 2, and 3 that `FedAvg` has $\mathcal{O}(\frac{1}{T})$ convergence rate. In particular, Theorem 3 shows that to attain a fixed precision ϵ, the number of communications is

$$\frac{T}{E} = \mathcal{O}\left[\frac{1}{\epsilon}\left(\left(1+\frac{1}{K}\right)EG^2 + \frac{\sum_{k=1}^N p_k^2\sigma_k^2 + \Gamma + G^2}{E} + G^2\right)\right]. \tag{1}$$

Here, G, Γ, p_k, and σ_k are problem-related constants defined in Section 3.1. The most interesting insight is that E is a knob controlling the convergence rate: neither setting E over-small ($E = 1$ makes `FedAvg` equivalent to SGD) nor setting E over-large is good for the convergence.

This work also makes algorithmic contributions. We summarize the existing sampling[3] and averaging schemes for `FedAvg` (which do not have convergence bounds before this work) and propose a new scheme (see Table 1). We point out that a suitable sampling and averaging scheme is crucial for the convergence of `FedAvg`. To the best of our knowledge, we are the first to theoretically demonstrate

[1] In original paper (McMahan et al., 2017), E epochs of SGD are performed in parallel. For theoretical analyses, we denote by E the times of updates rather than epochs.

[2] Throughout this paper, "non-iid" means data are not identically distributed. More precisely, the data distributions in the k-th and l-th devices, denote D_k and D_l, can be different.

[3] Throughout this paper, "sampling" refers to how the server chooses K user devices and use their outputs for updating the model parameters. "Sampling" does not mean how a device randomly selects training samples.

Table 1: Sampling and averaging schemes for FedAvg. $S_t \sim \mathcal{U}(N, K)$ means S_t is a size-K subset uniformly sampled **without replacement** from $[N]$. $S_t \sim \mathcal{W}(N, K, \mathbf{p})$ means S_t contains K elements that are iid sampled **with replacement** from $[N]$ with probabilities $\{p_k\}$. In the latter scheme, S_t is not a set.

Paper	Sampling	Averaging	Convergence rate
McMahan et al. (2017)	$S_t \sim \mathcal{U}(N, K)$	$\sum_{k \notin S_t} p_k \mathbf{w}_t + \sum_{k \in S_t} p_k \mathbf{w}_t^k$	-
Sahu et al. (2018)	$S_t \sim \mathcal{W}(N, K, \mathbf{p})$	$\frac{1}{K} \sum_{k \in S_t} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})^5$
Ours	$S_t \sim \mathcal{U}(N, K)$	$\sum_{k \in S_t} p_k \frac{N}{K} \mathbf{w}_t^k$	$\mathcal{O}(\frac{1}{T})^6$

that FedAvg with certain schemes (see Table 1) can achieve $\mathcal{O}(\frac{1}{T})$ convergence rate in non-iid federated setting. We show that heterogeneity of training data and partial device participation slow down the convergence. We empirically verify our results through numerical experiments.

Our theoretical analysis requires the decay of learning rate (which is known to hinder the convergence rate.) Unfortunately, we show in Theorem 4 that the decay of learning rate is necessary for FedAvg with $E > 1$, even if full gradient descent is used.[4] If the learning rate is fixed to η throughout, FedAvg would converge to a solution at least $\Omega(\eta(E-1))$ away from the optimal. To establish Theorem 4, we construct a specific ℓ_2-norm regularized linear regression model which satisfies our strong convexity and smoothness assumptions.

Paper organization. In Section 2, we elaborate on FedAvg. In Section 3, we present our main convergence bounds for FedAvg. In Section 4, we construct a special example to show the necessity of learning rate decay. In Section 5, we discuss and compare with prior work. In Section 6, we conduct empirical study to verify our theories. All the proofs are left to the appendix.

2 Federated Averaging (FedAvg)

Problem formulation. In this work, we consider the following distributed optimization model:

$$\min_{\mathbf{w}} \left\{ F(\mathbf{w}) \triangleq \sum_{k=1}^{N} p_k F_k(\mathbf{w}) \right\}, \tag{2}$$

where N is the number of devices, and p_k is the weight of the k-th device such that $p_k \geq 0$ and $\sum_{k=1}^{N} p_k = 1$. Suppose the k-th device holds the n_k training data: $x_{k,1}, x_{k,2}, \cdots, x_{k,n_k}$. The local objective $F_k(\cdot)$ is defined by

$$F_k(\mathbf{w}) \triangleq \frac{1}{n_k} \sum_{j=1}^{n_k} \ell(\mathbf{w}; x_{k,j}), \tag{3}$$

where $\ell(\cdot;\cdot)$ is a user-specified loss function.

Algorithm description. Here, we describe one around (say the t-th) of the *standard* FedAvg algorithm. First, the central server **broadcasts** the latest model, $\mathbf{w}_t$, to all the devices. Second, every device (say the k-th) lets $\mathbf{w}_t^k = \mathbf{w}_t$ and then performs E (≥ 1) **local updates**:

$$\mathbf{w}_{t+i+1}^k \longleftarrow \mathbf{w}_{t+i}^k - \eta_{t+i} \nabla F_k(\mathbf{w}_{t+i}^k, \xi_{t+i}^k), i = 0, 1, \cdots, E-1$$

where η_{t+i} is the learning rate (a.k.a. step size) and ξ_{t+i}^k is a sample uniformly chosen from the local data. Last, the server **aggregates** the local models, $\mathbf{w}_{t+E}^1, \cdots, \mathbf{w}_{t+E}^N$, to produce the new global model, $\mathbf{w}_{t+E}$. Because of the non-iid and partial device participation issues, the aggregation step can vary.

[4] It is well know that the full gradient descent (which is equivalent to FedAvg with $E = 1$ and full batch) do not require the decay of learning rate.

[5] The sampling scheme is proposed by Sahu et al. (2018) for FedAvg as a baseline, but this convergence rate is our contribution.

[6] The convergence relies on the assumption that data are balanced, i.e., $n_1 = n_2 = \cdots = n_N$. However, we can use a rescaling trick to get rid of this assumption. We will discuss this point later in Section 3.

IID versus non-iid. Suppose the data in the k-th device are i.i.d. sampled from the distribution $\mathcal{D}_k$. Then the overall distribution is a mixture of all local data distributions: $\mathcal{D} = \sum_{k=1}^{N} p_k \mathcal{D}_k$. The prior work Zhang et al. (2015a); Zhou and Cong (2017); Stich (2018); Wang and Joshi (2018); Woodworth et al. (2018) assumes the data are iid generated by or partitioned among the N devices, that is, $\mathcal{D}_k = \mathcal{D}$ for all $k \in [N]$. However, real-world applications do not typically satisfy the iid assumption. One of our theoretical contributions is avoiding making the iid assumption.

Full device participation. The prior work Coppola (2015); Zhou and Cong (2017); Stich (2018); Yu et al. (2019); Wang and Joshi (2018); Wang et al. (2019) requires the full device participation in the aggregation step of FedAvg. In this case, the aggregation step performs

$$\mathbf{w}_{t+E} \longleftarrow \sum_{k=1}^{N} p_k \mathbf{w}_{t+E}^k.$$

Unfortunately, the full device participation requirement suffers from serious "straggler's effect" (which means everyone waits for the slowest) in real-world applications. For example, if there are thousands of users' devices in the FL system, there are always a small portion of devices offline. Full device participation means the central server must wait for these "stragglers", which is obviously unrealistic.

Partial device participation. This strategy is much more realistic because it does not require all the devices' output. We can set a threshold K ($1 \leq K < N$) and let the central server collect the outputs of the first K responded devices. After collecting K outputs, the server stops waiting for the rest; the $K+1$-th to N-th devices are regarded stragglers in this iteration. Let $\mathcal{S}_t$ ($|\mathcal{S}_t| = K$) be the set of the indices of the first K responded devices in the t-th iteration. The aggregation step performs

$$\mathbf{w}_{t+E} \longleftarrow \frac{N}{K} \sum_{k \in \mathcal{S}_t} p_k \mathbf{w}_{t+E}^k.$$

It can be proved that $\mathbb{E}\left[\frac{N}{K} \sum_{k \in \mathcal{S}_t} p_k\right]$ equals one in expectation.

Communication cost. The FedAvg requires two rounds communications— one broadcast and one aggregation— per E iterations. If T iterations are performed totally, then the number of communications is $\lfloor \frac{2T}{E} \rfloor$. During the broadcast, the central server sends $\mathbf{w}_t$ to all the devices. During the aggregation, all or part of the N devices sends its output, say $\mathbf{w}_{t+E}^k$, to the server.

3 Convergence Analysis of FedAvg in Non-iid Setting

In this section, we show that FedAvg converges to the global optimum at a rate of $\mathcal{O}(1/T)$ for strongly convex and smooth functions and non-iid data. The main observation is that when the learning rate is sufficiently small, the effect of E steps of local updates is similar to one step update with a larger learning rate. This coupled with appropriate sampling and averaging schemes would make each global update behave like an SGD update. Partial device participation ($K < N$) only makes the averaged sequence $\{\mathbf{w}_t\}$ have a larger variance, which, however, can be controlled by learning rates. These imply the convergence property of FedAvg should not differ too much from SGD. Next, we will first give the convergence result with full device participation (i.e., $K = N$) and then extend this result to partial device participation (i.e., $K < N$).

3.1 Notation and Assumptions

We make the following assumptions on the functions $F_1, \cdots, F_N$. Assumption 1 and 2 are standard; typical examples are the ℓ_2-norm regularized linear regression, logistic regression, and softmax classifier.

Assumption 1. *$F_1, \cdots, F_N$ are all L-smooth: for all $\mathbf{v}$ and $\mathbf{w}$, $F_k(\mathbf{v}) \leq F_k(\mathbf{w}) + (\mathbf{v} - \mathbf{w})^T \nabla F_k(\mathbf{w}) + \frac{L}{2} \|\mathbf{v} - \mathbf{w}\|_2^2$.*

Assumption 2. *$F_1, \cdots, F_N$ are all μ-strongly convex: for all $\mathbf{v}$ and $\mathbf{w}$, $F_k(\mathbf{v}) \geq F_k(\mathbf{w}) + (\mathbf{v} - \mathbf{w})^T \nabla F_k(\mathbf{w}) + \frac{\mu}{2} \|\mathbf{v} - \mathbf{w}\|_2^2$.*

Assumptions 3 and 4 have been made by the works Zhang et al. (2013); Stich (2018); Stich et al. (2018); Yu et al. (2019).

Assumption 3. *Let ξ_t^k be sampled from the k-th device's local data uniformly at random. The variance of stochastic gradients in each device is bounded: $\mathbb{E}\left\|\nabla F_k(\mathbf{w}_t^k, \xi_t^k) - \nabla F_k(\mathbf{w}_t^k)\right\|^2 \leq \sigma_k^2$ for $k = 1, \cdots, N$.*

Assumption 4. *The expected squared norm of stochastic gradients is uniformly bounded, i.e., $\mathbb{E}\left\|\nabla F_k(\mathbf{w}_t^k, \xi_t^k)\right\|^2 \leq G^2$ for all $k = 1, \cdots, N$ and $t = 1, \cdots, T-1$.*

Quantifying the degree of non-iid (heterogeneity). Let F^* and F_k^* be the minimum values of F and F_k, respectively. We use the term $\Gamma = F^* - \sum_{k=1}^N p_k F_k^*$ for quantifying the degree of non-iid. If the data are iid, then Γ obviously goes to zero as the number of samples grows. If the data are non-iid, then Γ is nonzero, and its magnitude reflects the heterogeneity of the data distribution.

3.2 Convergence Result: Full Device Participation

Here we analyze the case that all the devices participate in the aggregation step; see Section 2 for the algorithm description. Let the `FedAvg` algorithm terminate after T iterations and return $\mathbf{w}_T$ as the solution. We always require T is evenly divisible by E so that `FedAvg` can output $\mathbf{w}_T$ as expected.

Theorem 1. *Let Assumptions 1 to 4 hold and L, μ, σ_k, G be defined therein. Choose $\kappa = \frac{L}{\mu}$, $\gamma = \max\{8\kappa, E\}$ and the learning rate $\eta_t = \frac{2}{\mu(\gamma + t)}$. Then `FedAvg` with full device participation satisfies*

$$\mathbb{E}\left[F(\mathbf{w}_T)\right] - F^* \leq \frac{\kappa}{\gamma + T - 1}\left(\frac{2B}{\mu} + \frac{\mu\gamma}{2}\mathbb{E}\|\mathbf{w}_1 - \mathbf{w}^*\|^2\right), \tag{4}$$

where

$$B = \sum_{k=1}^N p_k^2 \sigma_k^2 + 6L\Gamma + 8(E-1)^2 G^2. \tag{5}$$

3.3 Convergence Result: Partial Device Participation

As discussed in Section 2, partial device participation has more practical interest than full device participation. Let the set $\mathcal{S}_t$ ($\subset [N]$) index the active devices in the t-th iteration. To establish the convergence bound, we need to make assumptions on $\mathcal{S}_t$.

Assumption 5 assumes the K indices are selected from the distribution p_k independently and with replacement. The aggregation step is simply averaging. This is first proposed in (Sahu et al., 2018), but they did not provide theoretical analysis.

Assumption 5 (Scheme I). *Assume $\mathcal{S}_t$ contains a subset of K indices randomly selected with replacement according to the sampling probabilities $p_1, \cdots, p_N$. The aggregation step of `FedAvg` performs $\mathbf{w}_t \leftarrow \frac{1}{K}\sum_{k \in \mathcal{S}_t} \mathbf{w}_t^k$.*

Theorem 2. *Let Assumptions 1 to 4 hold and L, μ, σ_k, G be defined therein. Let κ, γ, η_t, and B be defined in Theorem 1. Let Assumption 5 hold and define $C = \frac{4}{K}E^2G^2$. Then*

$$\mathbb{E}\left[F(\mathbf{w}_T)\right] - F^* \leq \frac{\kappa}{\gamma + T - 1}\left(\frac{2(B+C)}{\mu} + \frac{\mu\gamma}{2}\mathbb{E}\|\mathbf{w}_1 - \mathbf{w}^*\|^2\right)$$

Alternatively, we can select K indices from $[N]$ uniformly at random without replacement. As a consequence, we need a different aggregation strategy. Assumption 6 assumes the K indices are selected uniformly without replacement and the aggregation step is the same as in Section 2. However, to guarantee convergence, we require an additional assumption of balanced data.

Assumption 6 (Scheme II). *Assume $\mathcal{S}_t$ contains a subset of K indices uniformly sampled from $[N]$ without replacement. Assume the data is balanced in the sense that $p_1 = \cdots = p_N = \frac{1}{N}$. The aggregation step of `FedAvg` performs $\mathbf{w}_t \leftarrow \frac{N}{K}\sum_{k \in \mathcal{S}_t} p_k \mathbf{w}_t^k$.*

Theorem 3. *Replace Assumption 5 by Assumption 6 and C by $C = \frac{N-K}{N-1}\frac{4}{K}E^2G^2$. Then the same bound in Theorem 2 holds.*

Scheme II requires $p_1 = \cdots = p_N = \frac{1}{N}$ which obviously violates the unbalance nature of FL. Fortunately, this can be addressed by the following transformation. Let $\widetilde{F}_k(\mathbf{w}) = p_k N F_k(\mathbf{w})$ be a scaled local objective F_k. Then the global objective becomes a simple average of all scaled local objectives:

$$F(\mathbf{w}) = \sum_{k=1}^{N} p_k F_k(\mathbf{w}) = \frac{1}{N} \sum_{k=1}^{N} \widetilde{F}_k(\mathbf{w}).$$

Theorem 3 still holds if L, μ, σ_k, G are replaced by $\widetilde{L} \triangleq \nu L$, $\widetilde{\mu} \triangleq \varsigma \mu$, $\widetilde{\sigma}_k = \sqrt{\nu}\sigma$, and $\widetilde{G} = \sqrt{\nu}G$, respectively. Here, $\nu = N \cdot \max_k p_k$ and $\varsigma = N \cdot \min_k p_k$.

3.4 Discussions

Choice of E. Since $\|\mathbf{w}_0 - \mathbf{w}^*\|^2 \leq \frac{4}{\mu^2} G^2$ for μ-strongly convex F, the dominating term in eqn. (6) is

$$\mathcal{O}\left(\frac{\sum_{k=1}^{N} p_k^2 \sigma_k^2 + L\Gamma + \left(1 + \frac{1}{K}\right) E^2 G^2 + \gamma G^2}{\mu T}\right). \tag{7}$$

Let T_ϵ denote the number of required steps for FedAvg to achieve an ϵ accuracy. It follows from eqn. (7) that the number of required communication rounds is roughly

$$\frac{T_\epsilon}{E} \propto \left(1 + \frac{1}{K}\right) E G^2 + \frac{\sum_{k=1}^{N} p_k^2 \sigma_k^2 + L\Gamma + \kappa G^2}{E} + G^2. \tag{8}$$

Thus, $\frac{T_\epsilon}{E}$ is a function of E that first decreases and then increases, which implies that over-small or over-large E may lead to high communication cost and that the optimal E exists.

Stich (2018) showed that if the data are iid, then E can be set to $\mathcal{O}(\sqrt{T})$. However, this setting does not work if the data are non-iid. Theorem 1 implies that E must not exceed $\Omega(\sqrt{T})$; otherwise, convergence is not guaranteed. Here we give an intuitive explanation. If E is set big, then $\mathbf{w}_t^k$ can converge to the minimizer of F_k, and thus FedAvg becomes the one-shot average Zhang et al. (2013) of the local solutions. If the data are non-iid, the one-shot averaging does not work because weighted average of the minimizers of $F_1, \cdots, F_N$ can be very different from the minimizer of F.

Choice of K. Stich (2018) showed that if the data are iid, the convergence rate improves substantially as K increases. However, under the non-iid setting, the convergence rate has a weak dependence on K, as we show in Theorems 2 and 3. This implies FedAvg is unable to achieve linear speedup. We have empirically observed this phenomenon (see Section 6). Thus, in practice, the participation ratio $\frac{K}{N}$ can be set small to alleviate the straggler's effect without affecting the convergence rate.

Choice of sampling schemes. We considered two sampling and averaging schemes in Theorems 2 and 3. Scheme I selects K devices according to the probabilities $p_1, \cdots, p_N$ with replacement. The non-uniform sampling results in faster convergence than uniform sampling, especially when $p_1, \cdots, p_N$ are highly non-uniform. If the system can choose to activate any of the N devices at any time, then Scheme I should be used.

However, oftentimes the system has no control over the sampling; instead, the server simply uses the first K returned results for the update. In this case, we can assume the K devices are uniformly sampled from all the N devices and use Theorem 3 to guarantee the convergence. If $p_1, \cdots, p_N$ are highly non-uniform, then $\nu = N \cdot \max_k p_k$ is big and $\varsigma = N \cdot \min_k p_k$ is small, which makes the convergence of FedAvg slow. This point of view is empirically verified in our experiments.

4 Necessity of Learning Rate Decay

In this section, we point out that diminishing learning rates are crucial for the convergence of FedAvg in the non-iid setting. Specifically, we establish the following theorem by constructing a ridge regression model (which is strongly convex and smooth).

[7] Here we use $\gamma = \mathcal{O}(\kappa + E)$.

Theorem 4. *We artificially construct a strongly convex and smooth distributed optimization problem. With full batch size, $E > 1$, and any fixed step size, FedAvg will converge to sub-optimal points. Specifically, let $\bar{\mathbf{w}}^*$ be the solution produced by FedAvg with a small enough and constant η, and $\mathbf{w}^*$ the optimal solution. Then we have*

$$\|\bar{\mathbf{w}}^* - \mathbf{w}^*\|_2 = \Omega((E-1)\eta) \cdot \|\mathbf{w}^*\|_2,$$

where we hide some problem dependent constants.

Theorem 4 and its proof provide several implications. First, the decay of learning rate is necessary of FedAvg. On the one hand, Theorem 1 shows with $E > 1$ and a decaying learning rate, FedAvg converges to the optimum. On the other hand, Theorem 4 shows that with $E > 1$ and any fixed learning rate, FedAvg does not converges to the optimum.

Second, FedAvg behaves very differently from gradient descent. Note that FedAvg with $E = 1$ and full batch size is exactly the Full Gradient Descent; with a proper and fixed learning rate, its global convergence to the optimum is guaranteed Nesterov (2013). However, Theorem 4 shows that FedAvg with $E > 1$ and full batch size cannot possibly converge to the optimum. This conclusion doesn't contradict with Theorem 1 in Khaled et al. (2019), which, when translated into our case, asserts that $\bar{\mathbf{w}}^*$ will locate in the neighborhood of $\mathbf{w}^*$ with a constant learning rate.

Third, Theorem 4 shows the requirement of learning rate decay is not an artifact of our analysis; instead, it is inherently required by FedAvg. An explanation is that constant learning rates, combined with E steps of possibly-biased local updates, form a sub-optimal update scheme, but a diminishing learning rate can gradually eliminate such bias.

The efficiency of FedAvg principally results from the fact that it performs several update steps on a local model before communicating with other workers, which saves communication. Diminishing step sizes often hinders fast convergence, which may counteract the benefit of performing multiple local updates. Theorem 4 motivates more efficient alternatives to FedAvg.

5 Related Work

Federated learning (FL) was first proposed by McMahan et al. (2017) for collaboratively learning a model without collecting users' data. The research work on FL is focused on the communication-efficiency Konevcnỳ et al. (2016); McMahan et al. (2017); Sahu et al. (2018); Smith et al. (2017) and data privacy Bagdasaryan et al. (2018); Bonawitz et al. (2017); Geyer et al. (2017); Hitaj et al. (2017); Melis et al. (2019). This work is focused on the communication-efficiency issue.

FedAvg, a synchronous distributed optimization algorithm, was proposed by McMahan et al. (2017) as an effective heuristic. Sattler et al. (2019); Zhao et al. (2018) studied the non-iid setting, however, they do not have convergence rate. A contemporaneous and independent work Xie et al. (2019) analyzed asynchronous FedAvg; while they did not require iid data, their bound do not guarantee convergence to saddle point or local minimum. Sahu et al. (2018) proposed a federated optimization framework called FedProx to deal with statistical heterogeneity and provided the convergence guarantees in non-iid setting. FedProx adds a proximal term to each local objective. When these proximal terms vanish, FedProx is reduced to FedAvg. However, their convergence theory requires the proximal terms always exist and hence fails to cover FedAvg.

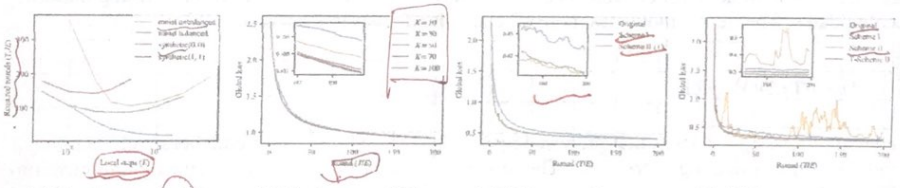
When data are iid distributed and all devices are active, FedAvg is referred to as LocalSGD. Due to the two assumptions, theoretical analysis of LocalSGD is easier than FedAvg. Stich (2018) demonstrated LocalSGD provably achieves the same linear speedup with strictly less communication for strongly-convex stochastic optimization. Coppola (2015); Zhou and Cong (2017); Wang and Joshi (2018) studied LocalSGD in the non-convex setting and established convergence results. Yu et al. (2019); Wang et al. (2019) recently analyzed LocalSGD for non-convex functions in heterogeneous settings. In particular, Yu et al. (2019) demonstrated LocalSGD also achieves $\mathcal{O}(1/\sqrt{NT})$ convergence (i.e., linear speedup) for non-convex optimization. Lin et al. (2018) empirically shows variants of LocalSGD increase training efficiency and improve the generalization performance of large batch sizes while reducing communication. For LocalGD on non-iid data (as opposed to LocalSGD), the best result is by the contemporaneous work (but slightly later than our first version) (Khaled et al., 2019). Khaled et al. (2019) used fixed learning rate η and showed $\mathcal{O}(\frac{1}{T})$

convergence to a point $\mathcal{O}(\eta^2 E^2)$ away from the optimal. In fact, the suboptimality is due to their fixed learning rate. As we show in Theorem 4, using a fixed learning rate η throughout, the solution by `LocalGD` is at least $\Omega((E-1)\eta)$ away from the optimal.

If the data are iid, distributed optimization can be efficiently solved by the second-order algorithms Mahajan et al. (2018); Reddi et al. (2016); Shamir et al. (2014); Shusen Wang et al. (2018); Zhang and Lin (2015) and the one-shot methods Lee et al. (2017); Lin et al. (2017); Wang (2019); Zhang et al. (2013; 2015b). The primal-dual algorithms Hong et al. (2018); Smith et al. (2016; 2017) are more generally applicable and more relevant to FL.

6 Numerical Experiments

Models and datasets We examine our theoretical results on a logistic regression with weight decay $\lambda = 1e-4$. This is a stochastic convex optimization problem. We distribute MNIST dataset (LeCun et al., 1998) among $N = 100$ workers in a non-iid fashion such that each device contains samples of only two digits. We further obtain two datasets: `mnist balanced` and `mnist unbalanced`. The former is balanced such that the number of samples in each device is the same, while the latter is highly unbalanced with the number of samples among devices following a power law. To manipulate heterogeneity more precisely, we synthesize unbalanced datasets following the setup in Sahu et al. (2018) and denote it as `synthetic(`α`,` β`)` where α controls how much local models differ from each other and β controls how much the local data at each device differs from that of other devices. We obtain two datasets: `synthetic(0,0)` and `synthetic(1,1)`. Details can be found in Appendix D.



(a) The impact of E (b) The impact of K (c) Different schemes (d) Different schemes

Figure 1: (a) To obtain an accuracy, the required rounds first decrease and then increase when we increase the local steps E. (b) In `Synthetic(0,0)` dataset, decreasing the numbers of active devices each round has little effect on the convergence process. (c) In `mnist balanced` dataset, Scheme I slightly outperforms Scheme II. They both performs better than the original scheme. Here transformed Scheme II coincides with Scheme II due to the balanced data. (d) In `mnist unbalanced` dataset, Scheme I performs better than Scheme II and the original scheme. Scheme II suffers from instability while transformed Scheme II has a lower convergence rate.

Experiment settings For all experiments, we initialize all runnings with $\mathbf{w}_0 = 0$. In each round, all selected devices run E steps of SGD in parallel. We decay the learning rate at the end of each round by the following scheme $\eta_t = \frac{\eta_0}{1+t}$, where η_0 is chosen from the set $\{1, 0.1, 0.01\}$. We evaluate the averaged model after each global synchronization on the corresponding global objective. For fair comparison, we control all randomness in experiments so that the set of activated devices is the same across all different algorithms on one configuration.

Impact of E We expect that T_ϵ/E, the required communication round to achieve curtain accuracy, is a hyperbolic function of E as equ (8) indicates. Intuitively, a small E means a heavy communication burden, while a large E means a low convergence rate. One needs to trade off between communication efficiency and fast convergence. We empirically observe this phenomenon on unbalanced datasets in Figure 1a. The reason why the phenomenon does not appear in `mnist balanced` dataset requires future investigations.

Impact of K Our theory suggests that a larger K may slightly accelerate convergence since T_ϵ/E contains a term $\mathcal{O}\left(\frac{EG^2}{K}\right)$. Figure 1b shows that K has limited influence on the convergence of FedAvg in synthetic(0, 0) dataset. It reveals that the curve of a large enough K is slightly better. We observe similar phenomenon among the other three datasets and attach additional results in Appendix D. This justifies that when the variance resulting sampling is not too large (i.e., $B \gg C$), one can use a small number of devices without severely harming the training process, which also removes the need to sample as many devices as possible in convex federated optimization.

Effect of sampling and averaging schemes. We compare four schemes among four federated datasets. Since the original scheme involves a history term and may be conservative, we carefully set the initial learning rate for it. Figure 1c indicates that when data are balanced, Schemes I and II achieve nearly the same performance, both better than the original scheme. Figure 1d shows that when the data are unbalanced, i.e., p_k's are uneven, Scheme I performs the best. Scheme II suffers from some instability in this case. This is not contradictory with our theory since we don't guarantee the convergence of Scheme II when data is unbalanced. As expected, transformed Scheme II performs stably at the price of a lower convergence rate. Compared to Scheme I, the original scheme converges at a slower speed even if its learning rate is fine tuned. All the results show the crucial position of appropriate sampling and averaging schemes for FedAvg.

7 Conclusion

Federated learning becomes increasingly popular in machine learning and optimization communities. In this paper we have studied the convergence of FedAvg, a heuristic algorithm suitable for federated setting. We have investigated the influence of sampling and averaging schemes. We have provided theoretical guarantees for two schemes and empirically demonstrated their performances. Our work sheds light on theoretical understanding of FedAvg and provides insights for algorithm design in realistic applications. Though our analyses are constrained in convex problems, we hope our insights and proof techniques can inspire future work.

Acknowledgements

Li, Yang and Zhang have been supported by the National Natural Science Foundation of China (No. 11771002 and 61572017), Beijing Natural Science Foundation (Z190001), the Key Project of MOST of China (No. 2018AAA0101000), and Beijing Academy of Artificial Intelligence (BAAI).

References

Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. How to backdoor federated learning. *arXiv preprint arXiv:1807.00459*, 2018. 7

Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017.

Gregory Francis Coppola. Iterative parameter mixing for distributed large-margin training of structured predictors for natural language processing. *PhD thesis*, 2015. 4, 7

Robin C Geyer, Tassilo Klein, Moin Nabi, and SAP SE. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*, 2017. 7

Briland Hitaj, Giuseppe Ateniese, and Fernando Pérez-Cruz. Deep models under the GAN: information leakage from collaborative deep learning. In *ACM SIGSAC Conference on Computer and Communications Security*, 2017. 7

Mingyi Hong, Meisam Razaviyayn, and Jason Lee. Gradient primal-dual algorithm converges to second-order stationary solution for nonconvex distributed optimization over networks. In *International Conference on Machine Learning (ICML)*, 2018. 8