ECE 6960-021 Course Project

Semi-Cyclic Gradient Descent Outperforms Federated Averaging with Sufficiently Small Learning Rate

Shaocong Ma

April 29, 2020

Abstract

This report proposes the convergence analysis for the semi-cyclic gradient descent (SCGD) algorithm under the setting of distributed learning with heterogeneous data. For the strongly quasi-convex and smooth objective function, we show that given n devices, in order to achieve the ϵ error, the semi-cyclic gradient descent algorithm requires $\mathcal{O}(\frac{n}{\sqrt{\epsilon}})$ rounds of communications. It outperforms the performance of the traditional Federated Averaging (FedAvg) algorithm which requires $\mathcal{O}(\frac{1}{\epsilon})$ rounds for the strongly convex and smooth objective function. Moreover, our theoretical results are consistent with our numerical experiments.

1 Introduction

We focus on the distributed optimization algorithms for the heterogeneous data, also known as the federated learning (see [Li et al., 2019a], [Yang et al., 2019], [Kairouz et al., 2019], etc.); it generally considers the scenario where users' data is of high privacy and cannot be visited by other users, but it still demands to train a machine learning model based on these local datasets. For example, in [Hard et al., 2018], the language model used to predict users' next input was trained on users' behavior dataset; users' input behavior is highly private so it should be avoided to upload related data to the server; besides the data privacy, it is also important to deal with the heterogeneity of dataset.

The heterogeneity of dataset comes from the diversity among users; it is reasonable to assume the data belongs to different user would have different distribution. There are many well-studied algorithms specially designed to solve this problem, such as Federated SVRG ([Konečný et al., 2016]), Federated Averaging ([McMahan et al., 2016]), FedProx ([Li et al., 2018]), etc. Moreover, the influence of data heterogeneity has been studied in [Li et al., 2019b] and [Khaled et al., 2019]; this kind of heterogeneity leads to an asymptotic error term with dependence on the degree of heterogeneity which is non-vanishing and usually large for the heterogeneous data.

In more realistic setting, the distributed algorithm should not only consider the heterogeneity of data distribution, but also the heterogeneity of responding time. Specifically, users may live in the different time zone so the activity of their devices are forced to follow a specific time pattern. For example, usually people's devices are charging and connected to Wi-Fi at the night; to avoid influence users' experience, the model training cannot be proceeded during the daytime. Related setting has been considered in [Bonawitz et al., 2019] and [Eichner et al., 2019], and usually applies semi-cyclic gradient-based algorithm. Also, [Eichner et al., 2019] point it out that the fixed sampling pattern may lead the gradient-based algorithm to suffer much worse convergence performance than the uniform sampling scheme, so they suggest to modify this algorithm to the multi-task learning.

However, the examples given in [Eichner et al., 2019] and [Woodworth et al., 2018] rely on selecting a relatively large learning rate; so their lower bound of semi-cyclic SGD can not be consistent with the existing result of SGD with cyclic sampling ([HaoChen and Sra, 2018], [Safran and Shamir, 2019] and [Ying et al., 2018] for the worst-case cyclic sampling; [Jain et al., 2019], [Nguyen et al., 2020], [Safran and Shamir, 2019] and [Rajput et al., 2020] for cyclic sampling with random reshuffle). This motivates us to build a non-asymptotic upper bound for the semi-cyclic gradient-based algorithm.

Structure of Report The report mainly includes three parts: In Section 2, we introduce the basic notations and two algorithms (FedAvg and SCGD) which we will compare in this report. In Section

3, we give the non-asymptotic convergence analysis for the SCGD algorithm and compare it with the non-asymptotic upper bound of FedAvg algorithm; we conclude SCGD theoretically outperforms FedAvg with sufficiently small learning rate. Then we give the convergence rate of SCGD under diminishing stepsize; and it is also faster than the FedAvg algorithm. In Section 4, we numerically verify our statement by applying FedAvg and SCGD to the classical linear regression problem; we conclude SCGD in practice can also outperform FedAvg with sufficiently small learning rate or with diminishing learning rate.

2 Setup and Related Works

To formulate the distributed learning setting, we assume a dataset is distributed to n devices. And we aim to minimize the following objective function

$$F(\theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\theta) := \frac{1}{n}\sum_{i=1}^{n} f(\theta; x_i),$$

where x_i is the local dataset in the i-th device, and $f_i(\theta) := f(\theta; x_i)$ represents the loss evaluated at x_i when the model parameter is set to be θ. Here we introduce two algorithms we will compare in this report.

Federated Averaging (FedAvg) Federated averaging algorithm, also known as FedAvg, is broadly used in practice ([McMahan et al., 2016], [Hard et al., 2018], etc.). It includes two main steps

- **Broadcast**: it distributes the model parameter to the sampled active devices; usually, the number of sampled devices is much smaller than the total number of devices and should be determined before training. Then the local device updates the parameter using the local data for multiple steps.
- **Aggregation**: After each selected device completes the local training procedure, the updated parameter will be sent back to the central server. The central server computes the average of all returned parameters; it will be used in the next step of broadcast.

Based on the current convergence upper bound for the strongly convex and smooth objective function ([Li et al., 2019b]), the best possible result for the FedAvg algorithm can achieve a sublinear convergence rate $\mathcal{O}(\frac{1}{T})$ with diminishing stepsize, and can achieve exponential convergence but with a non-vanishing $\mathcal{O}(\eta)$-level asymptotic error, where T is the number of communications and η is the fixed learning rate.

Semi-Cyclic Gradient Descent (SCGD) The SCGD algorithm is the natural generalization of traditional gradient decent algorithm to the distributed optimization problem. When it is unable to simultaneously visit all of observations, we can visit each local dataset in a fixed order. It is usually used to deal with the pattern of devices responding time ([Eichner et al., 2019]). We describe the details of SCGD in Algorithm 1. Currently, as far as we know, there is no convergence analysis for the SCGD.

2.1 Assumptions

First, we make the *μ-strongly quasi-convex* assumptions for the objective function as used in [Gower et al., 2019]. This assumption has been discussed in [Necoara et al., 2019] and [Karimi et al., 2016]; it generalizes the traditional strongly convex assumption. Note that it doesn't require the convexity and the uniqueness of minimizer, and it also works for a large class of non-convex functions. We give an example of μ-strongly quasi-convex function without convexity in Appendix B.

Assumption 2.1 (μ-strong quasi-convexity). *The objective function $F : \mathbb{R}^d \to \mathbb{R}$ is μ-strongly quasi-convex; that is, there exists $\mu > 0$ such that for all $\theta \in \mathbb{R}^d$ and $\theta^* = \mathrm{proj}_{\Theta^*}(\theta)$,*

$$F(\theta^*) - F(\theta) \geq \langle \nabla F(\theta), \theta^* - \theta \rangle + \frac{\mu}{2}\|\theta - \theta^*\|^2$$

where it is also assumed $\Theta^* := \arg\min_{\theta \in \mathbb{R}^d} F(\theta)$ exists and the projection operator is defined as

$$\mathrm{proj}_{\Theta^*} : \theta \mapsto \arg\min_{\theta^* \in \Theta^*}(\theta, \theta^*).$$

By quasi-convexity, the set of global minimizers is always convex, so the projection operator is well-defined. Next, we have the following standard assumption on each component:

Assumption 2.2 (L-Lipschitz continuous gradient). *Each component $f_i(\theta)$ has L-Lipschitz continuous gradient; that is, for all $\theta, \theta' \in \mathbb{R}^d$, there exists $L > 0$ such that*

$$\|\nabla f_i(\theta) - \nabla f_i(\theta')\| \leq L\|\theta - \theta'\|.$$

Note that we do not assume the convexity on each component; they can be non-convex or concave. However, we note that for the FedAvg algorithm, the convexity of each component is necessary ([Li et al., 2019b] assumes each component is strongly convex; and [Khaled et al., 2019] assumes

Algorithm 1 Semi-Cyclic Gradient Descent

```
Set $\theta_{1,0}^{0}$ as the initial parameter, $\eta$ as the learning rate, and $K_{\max}$ as the maximum iterations.
Generate the device list $\{\xi_1, \dots, \xi_n\}$.                    ▷ $\xi_i \neq \xi_j$ for $i \neq j$
for $K$ from 0 to $K_{\max}$ do
    for $i$ from 1 to $n$ do
        Broadcast the parameter $\theta_{i,0}^{K}$ to the device $\xi_i$.
        for $j$ from 0 to $E-1$ do
            Local update: $\theta_{i,j+1}^{K} = \theta_{i,j}^{K} - \eta \nabla f_{\xi_i}(\theta_{i,j}^{K})$
        end for
    end for
    Set $\theta_{1,0}^{K+1} = \theta_{n,E}^{K}$.
end for
```

each component is convex); and we also give an example that FedAvg will diverge when there are some components are concave (see Appendix C) while the convergence analysis for SCGD still holds for this case. Lastly, to characterize the heterogeneity among the data set, we define the *local gradient noise* at i-th device as

$$\sigma_i^2 := \sup_{\theta^* \in \Theta^*} \|\nabla f_i(\theta^*)\|^2.$$

And we further require the following mild assumption:

Assumption 2.3 (Finite gradient noise). *For all $i \in \{1, 2, \dots, n\}$, the local gradient noise is finite; that is*

$$\sigma_i^2 < \infty.$$

We note that if the set of minimizers is compact, the smoothness automatically implies this assumption.

3 Main Results

3.1 Main Lemmas

The following lemmas would be used in proving our main results:

Lemma 3.1. *Under Assumption 2.1, 2.2, and 2.3, the semi-cyclic gradient descent algorithm starts from the initial point $\theta_{1,0}^{K}$ and runs for n rounds of communications. Moreover, assume the learning rate η satisfies*

$$\eta \leq \frac{C}{nE} \tag{1}$$

where C is given by

$$C := \min\{\tfrac{1}{2L}, \sqrt{\tfrac{\mu^2}{24L^4} + \left(\tfrac{4L^2-\mu^2}{16L^4}\mu\right)^2} - \tfrac{4L^2-\mu^2}{16L^4}\mu\}. \tag{2}$$

Then

$$\begin{aligned}\mathrm{dist}_{\Theta^*}^2(\theta_{1,0}^{K+1}) \leq (1 - \frac{1}{3}\mu n E \eta) \cdot \mathrm{dist}_{\Theta^*}^2(\theta_{1,0}^{K}) \\ + \frac{4L^2 n^3 E^3 \eta^3}{\mu} \cdot \frac{1}{n}\sum_{i=1}^{n} \sigma_i^2.\end{aligned}$$

Proof. See Appendix A. □

Lemma 3.2 (Chung's lemma). *Let $u_k \geq 0$ be a sequence of real numbers. Given two positive real number $a > 2$ and $b > 0$; assume there exists k_0 such that*

$$u_{k+1} \leq (1 - \frac{a}{k})u_k + \frac{b}{k^3}$$

holds for all $k_0 \geq k$. Then

$$\limsup_{k \to \infty} k^2 u_k \leq \frac{b}{a-2} < \infty.$$

Remark. This well-known lemma is borrowed from Lemma 2.1 in [Gürbüzbalaban et al., 2015]. See Lemma 4 in [Chung, 1954] for a proof.

3.2 Non-Asymptotic Analysis with Constant Learning Rate

In this section, we give the non-asymptotic upper bound for a fixed constant learning rate as follows:

Theorem 3.3. *Under Assumption 2.1, 2.2, and 2.3, the semi-cyclic gradient descent algorithm starts from the initial point $\theta_{1,0}^{0}$ and runs for T rounds of communication such that each device has been visited for K times (that is, $T = nK$). If the learning rate satisfies*

$$\eta \leq \frac{C}{nE} \tag{3}$$

where C is given by (2), then

$$\begin{aligned}\mathbb{E}\mathrm{dist}_{\Theta^*}^2(\theta_{1,0}^{K}) \leq (1 - \frac{1}{3}\mu n E \eta)^K \cdot \mathbb{E}\mathrm{dist}_{\Theta^*}^2(\theta_{1,0}^{0}) \\ + \frac{12L^2 n^2 E^2 \eta^2}{\mu^2} \cdot \frac{1}{n}\sum_{i=1}^{n} \sigma_i^2.\end{aligned}$$

Proof. By Lemma 3.1,

$$\mathbb{E}\mathrm{dist}^2_{\Theta^*}(\theta^{K+1}_{1,0}) \leq (1-\frac{1}{3}\mu nE\eta)\cdot\mathbb{E}\mathrm{dist}^2_{\Theta^*}(\theta^{K}_{1,0}) + \frac{4L^2n^3E^3\eta^3}{\mu}\cdot\frac{1}{n}\sum_{i=1}^{n}\sigma_i^2.$$

Unrolling this inequality, we obtain

$$\mathbb{E}\mathrm{dist}^2_{\Theta^*}(\theta^{K}_{1,0}) \leq (1-\frac{1}{3}\mu nE\eta)^K\cdot\mathbb{E}\mathrm{dist}^2_{\Theta^*}(\theta^{0}_{1,0}) + \frac{12L^2n^2E^2\eta^2}{\mu^2}\cdot\frac{1}{n}\sum_{i=1}^{n}\sigma_i^2.$$

□

Remark. The first term represents the dependence on the initialization; with the number of communications T increasing, this term exponentially tends to zero. The second term is the asymptotic error, which is always non-vanishing. Unlike the FedAvg algorithm which asymptotic error is of $\mathcal{O}(\eta)$-level, the SCGD algorithm has asymptotic error of $\mathcal{O}(\eta^2)$-level. This property makes it possible to achieve a higher precision when the learning rate is sufficiently small.

Heterogeneity of Data Distribution As given in the Theorem 3.3, the asymptotic error term depends on the local gradient noise $\frac{1}{n}\sum_{i=1}^{n}\sigma_i^2$; this result is consistent with the FedAvg algorithm ([Khaled et al., 2019] and [Li et al., 2019b]), and also reveals the phenomenon where higher heterogeneity would make it more difficult to achieve the desired precision. And the dependence of heterogeneity can be removed by applying the diminishing step-size; see more discussion in Section 3.3.

Choice of E For the fixed maximum communication rounds T, we set the learning rate

$$\eta = \ell\cdot\frac{\log T}{T}.$$

for some constant ℓ such that (3) holds. Then we have

$$\mathbb{E}\|\theta^{K+1}_{1,0}-\theta^*\|^2 \leq T^{-\frac{\mu\ell}{3}E}\cdot\mathbb{E}\|\theta^0_{1,0}\|^2 + \frac{12L^2\ell^2}{\mu^2}\cdot\frac{1}{n}\sum_{i=1}^{n}\sigma_i^2\cdot\frac{n^2E^2(\log T)^2}{T^2}.$$

It is easy to notice that if we set E to be large, the first term will be smaller, but the error term will be worse. To balance both terms, the optimal choice of E is to set

$$E = \left\lceil\frac{4}{\mu\ell}\right\rceil;$$

that means, E is set to be the smallest integer such that $E \geq \frac{4}{\mu\ell}$.

ϵ-Time Complexity Under the optimal setting, we have

$$\mathbb{E}\|\theta^{K+1}_{1,0}-\theta^*\|^2 = \mathcal{O}(\frac{log(T)^2n^2}{T^2}).$$

To achieve an ϵ error, this setting requires $\mathcal{O}(\frac{n}{\sqrt{\epsilon}})$ rounds of communications. For the traditional Federated Averaging algorithm, it requires $\mathcal{O}(\frac{1}{\epsilon})$ rounds of communications for the strongly convex objective function ([Li et al., 2019b]). It means the semi-cyclic gradient descent algorithm can outperform the traditional algorithms under some scenarios, especially when a high-precision is required.

3.3 Asymptotic Analysis with Diminishing Learning Rate

We just show that with sufficiently small learning rate, the SCGD algorithm can have smaller asymptotic error. Usually, it is preferred to use diminishing step-size in FedAvg algorithm (see [Li et al., 2019b]) to avoid the influence of large local gradient noise caused by heterogeneity; therefore, in this subsection, we theoretically show that in this case SCGD can also outperform FedAvg with respect to the convergence speed.

Theorem 3.4. *Under Assumption 2.1, 2.2, and 2.3, the semi-cyclic gradient descent algorithm starts from the initial point $\theta^0_{1,0}$ and runs for T rounds of communication such that each device has been visited for K times (that is, $T = nK$). Moreover, assume the learning rate $\eta_K := \frac{\eta_0}{K}$ satisfies*

$$\frac{1}{3}\mu nE\eta_0 > 2.$$

Then

$$\limsup_{K\to\infty} K^2\mathbb{E}\mathrm{dist}^2_{\Theta^*}(\theta^K_{1,0}) < \infty.$$

Proof. Since $\eta_K := \frac{\eta_0}{K}$, there always exists K_0 such that for all $K \geq K_0$, the condition (1) is satisfied. Then by Lemma 3.1,

$$\mathbb{E}\mathrm{dist}^2_{\Theta^*}(\theta^{K+1}_{1,0}) \leq (1-\frac{1}{3}\mu nE\eta_0\cdot\frac{1}{K})\cdot\mathbb{E}\mathrm{dist}^2_{\Theta^*}(\theta^{K}_{1,0}) + \frac{4L^2n^3E^3\eta_0^3}{\mu}\cdot\frac{1}{n}\sum_{i=1}^{n}\sigma_i^2\cdot\frac{1}{K^3}.$$

By Lemma 3.2, setting

$$a = \frac{1}{3}\mu nE\eta_0$$

and

$$b = \frac{4L^2n^3E^3\eta_0^3}{\mu},$$

we obtain

$$\limsup_{K\to\infty} k^2 \mathbb{E}\mathrm{dist}^2_{\Theta^*}(\theta^K_{1,0}) < \infty.$$

□

Heterogeneity of Data Distribution Note that with diminishing step-size, the optimizer iteration can exactly reach the minimizer set Θ^*, while SCGD with a constant step-size always oscillates around Θ^*.

Convergence Rate Theorem 3.4 shows that with the diminishing learning rate, the distance between the iteration $\theta^K_{1,0}$ and the minimizer set is decreasing with rate

$$\mathcal{O}(\frac{1}{K^2}) = \mathcal{O}(\frac{n^2}{T^2})$$

for the smooth and strongly quasi-convex objective function. As given in [Li et al., 2019b], FedAvg can only achieve $\mathcal{O}(\frac{1}{T})$ convergence rate with the same diminishing setting for the smooth and strongly convex objective function.

4 Numerical Experiments

In this section, we set up a numerical experiment to verify our statements. The basic setting is to solve the linear regression problem: given a data set (X, y) where $X \in \mathbb{R}^{n\times d}$ is the feature matrix and $y \in \mathbb{R}^n$ is the response, we assume there is a linear relation between X and y and aim to find the relation by solving the optimization problem

$$\min_{\beta\in\mathbb{R}^d} \quad \|y - X\beta\|^2.$$

The distributed learning environment is constructed as follows: First, assume there are 1000 observations in total; the whole data set is divided into 20 devices randomly (each device may have different number of observations). Next, we generate the local data set using the normal random variables with the dimension 8. Lastly, we evaluate the performance of both algorithms using the mean square error (MSE).

Sufficiently small constant learning rate First, we compare the convergence error of both algorithms under the sufficiently small learning rate. For the optimizer, we set the learning rate to be $\eta = 10^{-5}$ and set the number of local updates to be $E = 4$. For the FedAvg algorithm, we assume there are 15% devices active for each communication (that is, for each communication round, the central server can visit 3 devices in our setting). For the SCGD algorithm, we only require 5% devices active (that is, for each communication round, the central server can only visit 1 device). Note that for a fixed constant learning rate, both algorithms cannot achieve the exactly minimizer of the objective function under the distributed setting.

Both algorithms run for 10000 rounds of communications and are evaluated the MSE after each round. The difference error curve is shown in Figure 1, where the y-axis is the difference of MSE for two algorithms; more explicitly, is computed as

$$\mathrm{MSE}_{\mathrm{FedAvg}} - \mathrm{MSE}_{\mathrm{SCGD}},$$

where $\mathrm{MSE}_{\mathrm{FedAvg}}$ is the MSE of FedAvg, and $\mathrm{MSE}_{\mathrm{SCGD}}$ is the MSE of SCGD. When the difference at the round t is larger than 0, it represents that the loss of SCGD is smaller than the loss of FedAvg at the round t. And we repeat the experiment for same initialization for 100 times; the upper curve and the lower curve are 95% percentile and 5% percentile, respectively.

As shown in Figure 1, by setting a sufficiently small learning rate $\eta = 10^{-5}$, the SCGD always has smaller errors after 6000 rounds of communications, so it verifies our theoretical result.

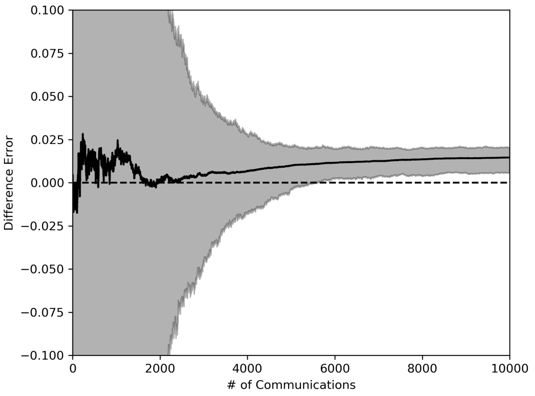

Figure 1: Sufficiently Small Constant Learning Rate

Diminishing learning rate Second, we compare the convergence of both algorithms under diminishing learning rate. We follow the same distributed linear regression problem, instead of setting a small constant learning rate, we set a relatively large one $\eta_0 = 10^{-3}$ and decrease it whenever all devices have been visited once by setting

$$\eta_K = \frac{\eta_0}{K}$$

where $K := \frac{T}{n}$. Figure 2 shows that SCGD converges faster than FedAvg under diminishing learning rate; it is consistent with our Theorem 3.4.

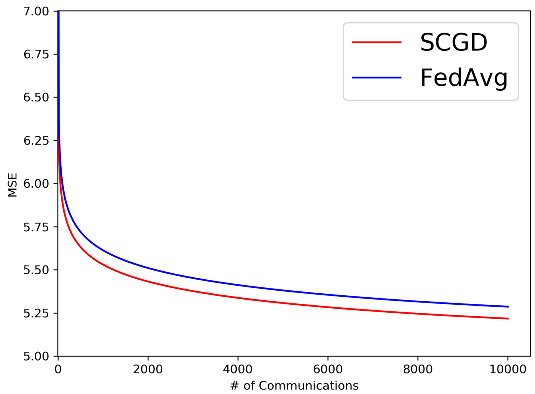

Figure 2: Diminishing Learning Rate

5 Conclusion

Lastly, we summarize this work below.

- First, we build the non-asymptotic upper bound of the SCGD algorithm. It concludes that with a sufficient small learning rate, the semi-cyclic gradient descent can achieve $\mathcal{O}(\frac{n}{\sqrt{\epsilon}})$ complexity while the FedAvg algorithm can only achieve $\mathcal{O}(\frac{1}{\epsilon})$ complexity.

- Second, we give the asymptotic convergence analysis of the SCGD algorithm. It shows the convergence rate of SCGD is $\mathcal{O}(\frac{n^2}{T^2})$ while that of the FedAvg algorithm is $\mathcal{O}(\frac{1}{T})$.

- Third, we show that the convergence analysis of SCGD is built on a more general setting (strongly quasi-convex objective function), and we do not require the convexity on components; however, a counterexample is given to show that the convexity of each component is necessary for the FedAvg algorithm.

- Lastly, our empirical experiments show that the asymptotic error of SCGD algorithm is significantly smaller than the FedAvg algorithm when the learning rate η is sufficiently small. And with a diminishing learning rate, the SCGD algorithm also converges faster than the FedAvg algorithm. These results are consistent with our two main theorems.

References

[Bonawitz et al., 2019] Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., Kiddon, C., Konecny, J., Mazzocchi, S., McMahan, H. B., et al. (2019). Towards federated learning at scale: System design. arXiv preprint arXiv:1902.01046.

[Chung, 1954] Chung, K. L. (1954). On a stochastic approximation method. The Annals of Mathematical Statistics, pages 463–483.

[Eichner et al., 2019] Eichner, H., Koren, T., McMahan, H. B., Srebro, N., and Talwar, K. (2019). Semi-cyclic stochastic gradient descent. arXiv preprint arXiv:1904.10120.

[Gower et al., 2019] Gower, R. M., Loizou, N., Qian, X., Sailanbayev, A., Shulgin, E., and Richtárik, P. (2019). Sgd: General analysis and improved rates. arXiv preprint arXiv:1901.09401.

[Gürbüzbalaban et al., 2015] Gürbüzbalaban, M., Ozdaglar, A., and Parrilo, P. (2015). Convergence rate of incremental gradient and newton methods. arXiv preprint arXiv:1510.08562.

[HaoChen and Sra, 2018] HaoChen, J. Z. and Sra, S. (2018). Random shuffling beats sgd after finite epochs. arXiv preprint arXiv:1806.10077.

[Hard et al., 2018] Hard, A., Rao, K., Mathews, R., Ramaswamy, S., Beaufays, F., Augenstein, S., Eichner, H., Kiddon, C., and Ramage, D. (2018). Federated learning for mobile keyboard prediction. arXiv preprint arXiv:1811.03604.

[Jain et al., 2019] Jain, P., Nagaraj, D., and Netrapalli, P. (2019). Sgd without replacement: Sharper rates for general smooth convex functions. arXiv preprint arXiv:1903.01463.

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

[Karimi et al., 2016] Karimi, H., Nutini, J., and Schmidt, M. (2016). Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pages 795–811. Springer.

[Khaled et al., 2019] Khaled, A., Mishchenko, K., and Richtárik, P. (2019). First analysis of local gd on heterogeneous data. arXiv preprint arXiv:1909.04715.

[Konečnỳ et al., 2016] Konečnỳ, J., McMahan, H. B., Ramage, D., and Richtárik, P. (2016). Federated optimization: Distributed machine learning for on-device intelligence. arXiv preprint arXiv:1610.02527.

[Li et al., 2019a] Li, T., Sahu, A. K., Talwalkar, A., and Smith, V. (2019a). Federated learning: Challenges, methods, and future directions. arXiv preprint arXiv:1908.07873.

[Li et al., 2018] Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., and Smith, V. (2018). Federated optimization in heterogeneous networks. arXiv preprint arXiv:1812.06127.

[Li et al., 2019b] Li, X., Huang, K., Yang, W., Wang, S., and Zhang, Z. (2019b). On the convergence of fedavg on non-iid data. arXiv preprint arXiv:1907.02189.

[McMahan et al., 2016] McMahan, H. B., Moore, E., Ramage, D., Hampson, S., et al. (2016). Communication-efficient learning of deep networks from decentralized data. arXiv preprint arXiv:1602.05629.

[Necoara et al., 2019] Necoara, I., Nesterov, Y., and Glineur, F. (2019). Linear convergence of first order methods for non-strongly convex optimization. Mathematical Programming, 175(1-2):69–107.

[Nguyen et al., 2020] Nguyen, L. M., Tran-Dinh, Q., Phan, D. T., Nguyen, P. H., and van Dijk, M. (2020). A unified convergence analysis for shuffling-type gradient methods. arXiv preprint arXiv:2002.08246.

[Rajput et al., 2020] Rajput, S., Gupta, A., and Papailiopoulos, D. (2020). Closing the convergence gap of sgd without replacement. arXiv preprint arXiv:2002.10400.

[Safran and Shamir, 2019] Safran, I. and Shamir, O. (2019). How good is sgd with random shuffling? arXiv preprint arXiv:1908.00045.

[Woodworth et al., 2018] Woodworth, B. E., Wang, J., Smith, A., McMahan, B., and Srebro, N. (2018). Graph oracle models, lower bounds, and gaps for parallel stochastic optimization. In Advances in neural information processing systems, pages 8496–8506.

[Yang et al., 2019] Yang, Q., Liu, Y., Chen, T., and Tong, Y. (2019). Federated machine learning: Concept and applications. ACM Transactions on Intelligent Systems and Technology (TIST), 10(2):1–19.

[Ying et al., 2018] Ying, B., Yuan, K., Vlaski, S., and Sayed, A. H. (2018). Stochastic learning under random reshuffling with constant step-sizes. IEEE Transactions on Signal Processing, 67(2):474–489.

A Proof of Lemma 3.1

In this lemma, we quantify how much the model parameter θ becomes closer to the set of minimizers after every device has been visited for once. Based on the Algorithm 1, we have

$$\begin{aligned}
\theta_{1,0}^{K+1} &= \theta_{n,E}^{K} \\
&= \theta_{n,0}^{K} - \eta \sum_{j=0}^{E-1} \nabla f_{\xi_n}(\theta_{n,j}^{K}) \\
&= \theta_{1,0}^{K} - \eta \sum_{i=1}^{n} \sum_{j=0}^{E-1} \nabla f_{\xi_i}(\theta_{i,j}^{K}) \\
&= \theta_{1,0}^{K} - \eta \sum_{i=1}^{n} \sum_{j=0}^{E-1} \left[\nabla f_{\xi_i}(\theta_{i,j}^{K}) - \nabla f_{\xi_i}(\theta_{1,0}^{K}) + \nabla f_{\xi_i}(\theta_{1,0}^{K})\right] \\
&= \theta_{1,0}^{K} - \eta \sum_{i=1}^{n} \sum_{j=0}^{E-1} \left[\nabla f_{\xi_i}(\theta_{i,j}^{K}) - \nabla f_{\xi_i}(\theta_{1,0}^{K})\right] - nE\eta \nabla F(\theta_{1,0}^{K})
\end{aligned}$$

Set $\theta^* = \mathrm{proj}_{\Theta^*}(\theta_{n,E}^{K})$. Subtract θ^* and take square on both sides.

$$\begin{aligned}
\|\theta_{1,0}^{K+1} - \theta^*\|^2 &= \|\theta_{1,0}^{K} - \theta^* - nE\eta\nabla F(\theta_{1,0}^{K}) - \eta \sum_{i=1}^{n} \sum_{j=0}^{E-1} \left[\nabla f_{\xi_i}(\theta_{i,j}^{K}) - \nabla f_{\xi_i}(\theta_{1,0}^{K})\right] \|^2 \\
&\leq \frac{1}{1 - \frac{1}{2}\mu nE\eta} \|\theta_{1,0}^{K} - \theta^* - nE\eta\nabla F(\theta_{1,0}^{K})\|^2 + \frac{2\eta^2}{\mu nE\eta} \| \sum_{i=1}^{n} \sum_{j=0}^{E-1} \left[\nabla f_{\xi_i}(\theta_{i,j}^{K}) - \nabla f_{\xi_i}(\theta_{1,0}^{K})\right] \|^2 \\
&\leq \frac{1}{1 - \frac{1}{2}\mu nE\eta} \|\theta_{1,0}^{K} - \theta^* - nE\eta\nabla F(\theta_{1,0}^{K})\|^2 + \frac{2\eta}{\mu} \cdot \sum_{i=1}^{n} \sum_{j=0}^{E-1} \|\nabla f_{\xi_i}(\theta_{i,j}^{K}) - \nabla f_{\xi_i}(\theta_{1,0}^{K})\|^2
\end{aligned}$$

where we use Jensen's inequality in the second step; more explicitly, by the convexity of $\|\cdot\|^2$,

$$\|a+b\|^2 = \|(1-t) \cdot \frac{1}{1-t} a + t \cdot \frac{1}{t} b\|^2 \leq (1-t)\|\frac{1}{1-t} a\|^2 + t\|\frac{1}{t} b\|^2 = \frac{1}{1-t}\|a\|^2 + \frac{1}{t}\|b\|^2.$$

Moreover, we require the learning rate sufficiently small such that

$$\eta < \frac{1}{nE}\frac{2}{\mu}. \tag{4}$$

Now let $T_1 := \|\theta_{1,0}^{K} - \theta^* - nE\eta\nabla F(\theta_{1,0}^{K})\|^2$ and $T_2 := \sum_{i=1}^{n} \sum_{j=0}^{E-1} \|\nabla f_{\xi_i}(\theta_{i,j}^{K}) - \nabla f_{\xi_i}(\theta_{1,0}^{K})\|^2$. We bound T_1 and T_2 respectively.

- We bound T_1 as follows:

$$\begin{aligned}
T_1 &:= \|\theta_{1,0}^{K} - \theta^* - nE\eta\nabla F(\theta_{1,0}^{K})\|^2 \\
&= \|\theta_{1,0}^{K} - \theta^*\|^2 + n^2E^2\eta^2\|\nabla F(\theta_{1,0}^{K})\|^2 - 2nE\eta \cdot \langle \theta_{1,0}^{K} - \theta^*, \nabla F(\theta_{1,0}^{K})\rangle \\
&= \|\theta_{1,0}^{K} - \theta^*\|^2 + n^2E^2\eta^2\|\nabla F(\theta_{1,0}^{K})\|^2 - 2nE\eta \cdot \langle \theta_{1,0}^{K} - \theta^*, \nabla F(\theta_{1,0}^{K}) - \nabla F(\theta^*)\rangle \\
&\leq \|\theta_{1,0}^{K} - \theta^*\|^2 + n^2E^2\eta^2 \cdot L^2\|\theta_{1,0}^{K} - \theta^*\|^2 - nE\eta \cdot \mu\|\theta_{1,0}^{K} - \theta^*\|^2 \\
&= \left(1 - \mu nE\eta + L^2n^2E^2\eta^2\right)\|\theta_{1,0}^{K} - \theta^*\|^2
\end{aligned}$$

where we apply the Lipschitz gradient assumption

$$\|\nabla F(\theta_{1,0}^{K})\|^2 \leq L^2\|\theta_{1,0}^{K} - \theta^*\|^2$$

and μ-strongly quasi-convexity of F

$$\langle \nabla F(\theta^K_{1,0}), \theta^K_{1,0} - \theta^* \rangle \geq F(\theta^K_{1,0}) - F(\theta^*) + \frac{\mu}{2}\|\theta^K_{1,0} - \theta^*\|^2.$$

in the inequality step. Note that since $F(\theta^K_{1,0}) - F(\theta^*)$ is always non-negative so we omit it.

- Now we bound T_2.

$$\begin{aligned} T_2 &:= \sum_{i=1}^{n}\sum_{j=0}^{E-1} \|\nabla f_{\xi_i}(\theta^K_{i,j}) - \nabla f_{\xi_i}(\theta^K_{1,0})\|^2 \\ &\leq L^2 \sum_{i=1}^{n}\sum_{j=0}^{E-1} \|\theta^K_{i,j} - \theta^K_{1,0}\|^2 \\ &\leq \frac{L^2 n^2 E^2}{2} \sum_{i=1}^{n}\sum_{j=0}^{E-1} \|\theta^K_{i,j+1} - \theta^K_{i,j}\|^2 \end{aligned}$$

Notice that

$$\begin{aligned} \|\theta^K_{i,j+1} - \theta^K_{i,j}\|^2 &= \eta^2 \|\nabla f_{\xi_i}(\theta^K_{i,j})\|^2 \\ &\leq 2\eta^2\|\nabla f_{\xi_i}(\theta^K_{i,j}) - \nabla f_{\xi_i}(\theta^*)\| + 2\eta^2\|\nabla f_{\xi_i}(\theta^*)\|^2 \\ &\leq 2\eta^2 L^2 \|\theta^K_{i,j} - \theta^*\|^2 + 2\eta^2\|\nabla f_{\xi_i}(\theta^*)\|^2 \\ &\leq 4\eta^2 L^2\|\theta^K_{1,0} - \theta^*\|^2 + 4\eta^2 L^2 \|\theta^K_{i,j} - \theta^K_{1,0}\|^2 + 2\eta^2\|\nabla f_{\xi_i}(\theta^*)\|^2 \end{aligned}$$

Sum it over i, j:

$$\begin{aligned} \sum_{i=1}^{n}\sum_{j=0}^{E-1} \|\theta^K_{i,j+1} - \theta^K_{i,j}\|^2 &\leq 4\eta^2 L^2 nE \|\theta^K_{1,0} - \theta^*\|^2 + 4\eta^2 L^2 \sum_{i=1}^{n}\sum_{j=0}^{E-1}\|\theta^K_{i,j} - \theta^K_{1,0}\|^2 + 2\eta^2 E \sum_{i=1}^{n}\|\nabla f_{\xi_i}(\theta^*)\|^2 \\ &\leq 4\eta^2 L^2 nE\|\theta^K_{1,0} - \theta^*\|^2 + 2\eta^2 L^2 n^2 E^2 \sum_{i=1}^{n}\sum_{j=0}^{E-1}\|\theta^K_{i,j+1} - \theta^K_{i,j}\|^2 + 2\eta^2 nE \cdot \frac{1}{n}\sum_{i=1}^{n}\sigma_i^2 \end{aligned}$$

Finally, we get

$$T_2 \leq \frac{L^2 n^3 E^3}{1 - 2L^2 n^2 E^2 \cdot \eta^2} \cdot \eta^2 \left(2L^2 \|\theta^K_{1,0} - \theta^*\|^2 + \frac{1}{n}\sum_{i=1}^{n}\sigma_i^2 \right)$$

Combine T_1 and T_2.

$$\begin{aligned} \|\theta^{K+1}_{1,0} - \theta^*\|^2 \leq & \left[\frac{1 - \mu nE\eta + L^2 n^2 E^2 \eta^2}{1 - \frac{1}{2}\mu nE\eta} + \frac{2\eta^3}{\mu} \cdot 2L^2 \cdot \frac{L^2 n^3 E^3}{1 - 2L^2 n^2 E^2 \cdot \eta^2} \cdot \right] \|\theta^K_{1,0} - \theta^*\|^2 \\ & + \frac{2\eta^3}{\mu} \cdot \frac{L^2 n^3 E^3}{1 - 2L^2 n^2 E^2 \cdot \eta^2} \cdot \frac{1}{n}\sum_{i=1}^{n}\sigma_i^2 \end{aligned}$$

Then we require

a) $1 - \frac{1}{2}\mu nE\eta \geq \frac{1}{2}$

b) $1 - 2L^2 n^2 E^2 \eta^2 \geq \frac{1}{2}$

c) $1 - \frac{1}{2}\mu nE\eta + 2(L^2 - \frac{\mu^2}{4})n^2 E^2 \eta^2 + \frac{4L^4}{\mu}\eta^3 n^3 E^3 \leq 1 - \frac{1}{3}\mu nE\eta$

And the requirements above with (4) are equivalent to

$$\eta \leq \frac{C}{nE} \tag{5}$$

where C is given by

$$C := \min\{\frac{1}{2L}, \sqrt{\frac{\mu^2}{24L^4} + \left(\frac{4L^2-\mu^2}{16L^4}\mu\right)^2} - \frac{4L^2-\mu^2}{16L^4}\mu\}.$$

Finally, due to $\theta^* = \mathrm{proj}_{\Theta^*}(\theta_{1,0}^K)$, we obtain

$$\begin{aligned}\mathrm{dist}^2_{\Theta^*}(\theta_{1,0}^{K+1}) &\leq \|\theta_{1,0}^{K+1} - \theta^*\|^2 \\ &\leq (1 - \frac{1}{3}\mu nE\eta)\|\theta_{1,0}^K - \theta^*\|^2 + \frac{4L^2n^3E^3\eta^3}{\mu} \cdot \frac{1}{n}\sum_{i=1}^n \sigma_i^2 \\ &= (1 - \frac{1}{3}\mu nE\eta) \cdot \mathrm{dist}^2_{\Theta^*}(\theta_{1,0}^K) + \frac{4L^2n^3E^3\eta^3}{\mu} \cdot \frac{1}{n}\sum_{i=1}^n \sigma_i^2.\end{aligned}$$

B Example: μ-strongly quasi-convex but non-convex

Consider the following one-dimensional function

$$F(x) = \begin{cases} -\frac{1}{4} & x \leq \frac{1}{2} \\ x^2 - x & \frac{1}{2} < x \leq 1 \\ \log x & 1 < x \leq \frac{3}{2} \\ \frac{2}{9}x^2 + \log(\frac{3}{2}) - \frac{1}{2} & x > \frac{3}{2} \end{cases} \tag{6}$$

Obviously, it is a non-convex function since its Hessian on $(1, \frac{3}{2}]$ is strictly negative definite. And the minimizer set is $\Theta = (-\infty, \frac{1}{2}]$. It suffices to verify the μ-strong quasi-convexity on $[\frac{1}{2}, +\infty)$. (And in this case we always have $\theta^* = \mathrm{proj}_{\Theta^*}(\theta) = \frac{1}{2}$). For $x \in [\frac{1}{2}, 1]$,

$$-\frac{1}{4} - (x^2 - x) \geq (2x-1)(\frac{1}{2} - x) + \frac{\mu}{2}(x - \frac{1}{2})^2,$$

holds for all $\mu \in (0, 2]$. And for $x \in [1, \frac{3}{2}]$,

$$-\frac{1}{4} - \log x \geq \frac{\frac{1}{2} - x}{x} + \frac{\mu}{2}(x - \frac{1}{2})^2,$$

holds for all $\mu \in (0, \frac{5}{6} - \log(\frac{9}{4})]$. For $x \in [\frac{3}{2}, \infty)$,

$$-\frac{1}{4} - \frac{2}{9}x^2 - \log(\frac{3}{2}) + \frac{1}{2} \geq \frac{4}{9}x(\frac{1}{2} - x) + \frac{\mu}{2}(x - \frac{1}{2})^2,$$

also holds for all $\mu \in (0, \frac{5}{6} - \log(\frac{9}{4})]$.
Therefore, we show that $F(x)$ is μ-strongly quasi-convex with $\mu = \frac{5}{6} - \log(\frac{9}{4}) \approx 0.02240$.

C Example: Divergence of FedAvg with Concave Components

Let $F(x) = \frac{1}{3}\sum_{i=1}^3 f_i(x)$ where $f_1(x) = f_2(x) = -x^2$ and $f_3(x) = 3x^3$. Set the initialization $x_0 = 1$, learning rate $\eta = 0.1$, and the number of local updates $E = 2$. For the FedAvg algorithm, we assume all devices can be visited in one communication round. Then after one round, the local parameter on the first two devices becomes $x^{(1)} = x^{(2)} = 1.44$, the local parameter on the third device become $x^{(3)} = 0.16$. So their aggregation is $x_1 = \frac{1.44+1.44+0.16}{3} > 1$. In the next round, the averaged parameter will keep leaving from the global minima $x^* = 0$. And for the SCGD algorithm, after three rounds of communications, the tracking parameter becomes 0.331776, which verifies our Theorem 3.3.