Convergence Analysis of Federated Learning in the Overparameterized Regime

Seminar for 'Deep learning theory'

a.y. 2023/24

Sebastiano Scardera

June 7, 2024

Contents

Preface

This is the report for the seminar of the course 'Deep learning theory'. I presented the key results and experiments from the paper [1]. The presentation lasted for one hour, focusing on the main points. The detailed proofs can be found in the original paper.

1 Overview of Federated learning

Federated learning is a decentralized approach to machine learning where multiple devices or clients collaborate to train a shared model without sharing their raw data. This leads to a double benefit: preservation of privacy and avoiding to transfer large volumes of data.

Federated learning has gained significant attention in recent years due to its potential applications in various domains:

- Smartphone: next-word prediction (used by Google's G-board), voice and face recognition (used by Apple's voice assistant Siri), etc.
- Healthcare: training a model on patient data without sharing it for drug discovery, tumor detection, etc.
- IoT: smart home, autonomous driving, etc.
- Finance: fraud detection, etc.

The baseline algorithm is the Federated Averaging algorithm (FedAvg) proposed by Google in 2017. In FedAvg, each device trains a model on its local data and then shares only the local parameters updates with a central server. The server then aggregates the shared model parameters to update the global model, and broadcasts the updated global model to all devices.

1.1 FedAvg algorithm scheme

Let $u(t) \in \mathbb{R}^{d\times m}$ be the global model at round t, $w_{k,c}(t) \in \mathbb{R}^{d\times m}$ denote the c-th client's model at round t after k local steps. The FedAvg algorithm can be summarized as follows:

- In t-th communication round, server broadcasts the global model $u(t) \in \mathbb{R}^{d\times m}$ to every clients;
- Each client c starts with $w_{0,c}(t) = u(t)$ and takes K gradient descent steps to get $w_{K,c}(t)$;
- Each client sends the parameters update $\Delta u_c(t) = w_{K,c}(t) - w_{0,c}(t)$ to the server;
- Server aggregates the client models to obtain the updated global model $u(t+1)$ as follows:

$$u(t+1) = u(t) + \frac{\eta_{\text{global}}}{N} \sum_{c=1}^{N} \Delta u_c(t),$$

- Repeat the above steps for T rounds.

Tipically, the local data points are sampled from a non independent and identically distributed distribution (non-IID), which can lead to poor performance of the global model and the convergence may not be guaranteed. The main reason is the 'Client-drift' phenomenon: the local dynamics tends to the local optimum, which can be very different from the global optimum.

2 Introduction

The paper [1] presents a theoretical analysis of the convergence of the FedAvg algorithm in the overparameterized regime. The main result is that the convergence of the FedAvg algorithm can be guaranteed if the parameters are properly tuned. Up to the authors, it provides the first proof of convergence of federated learning concerning neural networks with multi-step local updates. The training analysis is non-trivial because the dynamics of the neural network does not follow the gradient direction and the convergence is not guaranteed. Moreover, it holds without assumptions on the convexity of the loss function or distribution of data.

The main idea is to analyze the dynamics of the ReLU neural networks in the overparameterized regime by using the Neural Tangent Kernel (NTK) theory. The classical NTK theory cannot be directly applied to the federated learning and the core of the paper is to extend the NTK theory to the federated learning setting.

Furthermore, with additional distributional assumption, the authors provide a good generlization bound for the global model.

3 Problem formulation

3.1 Notations

Let N be the number of clients (c its index), T the number of communications rounds (t its index), K the number of local steps (k its index). $u(t) \in \mathbb{R}^{d\times m}$ is the global model at round t, $w_{k,c}(t)$ denote the c-th client's model at round t after k local steps.

Let $S_1 \cup S_2 \cup \cdots \cup S_N = [n]$ be the partition of the data points and $S_i \cap S_j = \emptyset$, where S_c is the set of data points on client c. The whole dataset is $\{(x_1, y_1), ..., (x_n, y_n)\} \subseteq \mathbb{R}^d \times \mathbb{R}$, for each client c: $y_c \in \mathbb{R}^{|S_c|}$ denotes the ground truth with regard of its data, $y_c^{(k)}(t) \in \mathbb{R}^{|S_c|}$ denotes the local model's prediction at round t after k local steps, $y^{(k)}(t) \in \mathbb{R}^n$ is the aggregated global output at round t after k local steps.

3.2 Setup

Let $\phi(z) = \max(0, z)$ be the ReLU activation function, $f : \mathbb{R}^d \to \mathbb{R}$ be the one-hidden layer neural network:

$$f(u, x) := \frac{1}{\sqrt{m}} \sum_{r=1}^{m} a_r \phi(u_r^T x),$$

where every column u_r of $u \in \mathbb{R}^{d\times m}$ is sampled from $\mathcal{N}(0, \sigma^2 I_d)$ and $a \in \mathbb{R}^m$ is sampled from $\{-1, 1\}^m$ uniformly. The loss functions are the following:

$$L_c(u, x) := \frac{1}{2} \sum_{i\in S_c} (f(u, x_i) - y_i)^2, \quad L(u) := \frac{1}{N} \sum_{c=1}^{N} L_c(u).$$

For the computation of the gradients of f we use the distributional derivative of the ReLU function: $\phi'(z) = \mathbb{I}_{z>0}$. We want to minimize the global loss $L(u)$ by using the FedAvg algorithm. The local update $\Delta u_c(t)$ is computed by taking K gradient descent steps on the local loss $L_c(u)$:

$$\Delta u_{c,r}(t) = \sum_{k=1}^{K} \eta_{\text{local}} \nabla L_c(w_{k,c}(t)) = \sum_{k=1}^{K} \eta_{\text{local}} \frac{a_r}{\sqrt{m}} \sum_{j\in S_c} -(y_c^{(k)}(t) - y_j) x_j \mathbb{I}_{w_{k,c,r}(t)^T x_j \geq 0}.$$

3.3 NTK analysis

In the centralized overparameterized setting, by [2], the dynamics of the neural network can be described by the Neural Tangent Kernel:

$$H_{i,j}^{\infty} = \mathbb{E}_{u\sim\mathcal{N}(0,\sigma^2 I_d)}\left[\phi'(u_i^T x)\phi'(u_j^T x)\right].$$

In the federated learning setting, we are not in the same setting but we can use the local NTK to analyze the training dynamics.

Let $y \in \mathbb{R}^n$ be the ground truth and $y(t) = (y_1(t), ..., y_n(t))$ the aggregated global output at round t, where $y_i(t) = f(u(t), x_i)$. We can compute:

$$\begin{aligned}
&\|y - y(t+1)\|_2^2 \\
=&\|y - y(t) - (y(t+1) - y(t))\|_2^2 \\
=&\|y - y(t)\|_2^2 - 2(y - y(t))^T(y(t+1) - y(t)) + \|y(t+1) - y(t)\|_2^2.
\end{aligned}$$

Now we focus on $y(t+1) - y(t)$, for each $i \in [n]$:

$$\begin{aligned}
&y_i(t+1) - y_i(t) \\
=&\frac{1}{\sqrt{m}}\sum_{r=1}^{m} a_r(\phi(u_r(t+1)^T x_i) - \phi(u_r(t)^T x_i)) \\
=&\frac{1}{\sqrt{m}}\sum_{r=1}^{m} a_r\Big(\phi((u_r(t) + \eta_{\text{global}}\Delta u_r(t))^T x_i) - \phi(u_r(t)^T x_i)\Big)
\end{aligned}$$

Now, we give a definition that will be crucial for the analysis. For each data point x_i, we distinguish the set of the neurons whose activation pattern changes over time from the set of the neurons whose activation pattern remains the same.

Definition 1. *Fixed $R > 0$, for each $i \in [n]$, we define the set Q_i as follows:*

$$Q_i := \{r \in [m] : \forall w \in \mathbb{R}^d \text{ s.t. } \|w - w_r(0)\| \leq R, \quad \mathbb{I}_{w_r(0)^T x_i \geq 0} = \mathbb{I}_{w^T x_i \geq 0}\},$$

and let $\bar{Q}_i$ its complement.

In this way, we can rewrite the difference $y_i(t+1) - y_i(t) = v_{1,i} + v_{2,i}$, where:

$$\begin{aligned}
v_{1,i} &= \frac{1}{\sqrt{m}}\sum_{r\in Q_i} a_r\Big(\phi((u_r(t) + \eta_{\text{global}}\Delta u_r(t))^T x_i) - \phi(u_r(t)^T x_i)\Big), \\
v_{2,i} &= \frac{1}{\sqrt{m}}\sum_{r\in \bar{Q}_i} a_r\Big(\phi((u_r(t) + \eta_{\text{global}}\Delta u_r(t))^T x_i) - \phi(u_r(t)^T x_i)\Big).
\end{aligned}$$

Assuming that the dynamics remains bounded, the key observation is that v_1 can be written as:

$$\begin{aligned}
v_{1,i} &= \frac{1}{\sqrt{m}}\sum_{r\in Q_i} a_r \eta_{\text{global}} x_i^T \Delta u_r(t) \mathbb{I}_{u_r(t)^T x_i \geq 0} \\
&= \frac{\eta_{\text{global}}\eta_{\text{local}}}{Nm}\sum_{r\in Q_i}\sum_{k\in[K]}\sum_{c\in[N]}\sum_{j\in S_c} -(y_c^{(k)}(t)_j - y_j) x_i^T x_j \mathbb{I}_{u_r(t)^T x_i \geq 0,\ w_{k,c,r}(t)^T x_j \geq 0}.
\end{aligned}$$

Now we can define the following Gram matrix:

Definition 2. *For any $t \in [T]$, $k \in [K]$, $c \in [N]$, we define $H(t,k,c)$ as follows:*

$$H(t,k,c)_{i,j} := \frac{1}{m}\sum_{r=1}^{m} x_i^T x_j \mathbb{I}_{u_r(t)^T x_i \geq 0,\ w_{k,c,r}(t)^T x_j \geq 0},$$
$$H(t,k,c)^{\perp}_{i,j} := \frac{1}{m}\sum_{r \in \bar{Q}_i} x_i^T x_j \mathbb{I}_{u_r(t)^T x_i \geq 0,\ w_{k,c,r}(t)^T x_j \geq 0}.$$

With this definition, we can rewrite $v_{1,i}$ as:

$$v_{1,i} = \frac{\eta_{\text{global}}\eta_{\text{local}}}{N}\sum_{k}\sum_{c}\sum_{j} -(y_c^{(k)}(t)_j - y_j)\big(H(t,k,c)_{i,j} - H(t,k,c)^{\perp}_{i,j}\big). \tag{1}$$

4 Properties and main results

4.1 Properties

Now we can show some properties that are classical in the NTK framework.

Proposition 1. *For all $t \in [T]$,*

i) ***Weights change lazily****, for all $r = 1, ..., m$:*

$$\|\Delta u_r(t)\|_2 \leq O\Big(\frac{\|y - y(t)\|_2}{N\sqrt{m}}\mathcal{C}(\eta_{local} \cdot K)\Big)$$

and

$$\|u_r(t) - u_r(0)\|_2 \leq O\Big(\frac{\|y - y(0)\|_2}{\lambda\sqrt{m}}\Big);$$

ii) ***Activation patterns remain roughly the same****, $k \in [K]$, $c \in [N]$, with probability higher than $1 - n\exp(-mR)$:*

$$\|H(t,k,c)^{\perp}\|_F \leq 4nR = O\big(1\big)$$

and

$$\|v_2\|_2 \leq O\Big(\frac{\|y - y(t)\|_2}{N}\mathcal{C}(\eta_{local} \cdot K, \eta_{global}\eta_{local} \cdot K)\Big).$$

iii) ***Global error controls model updates****:*

$$\|y(t+1) - y(t)\|_2 \leq O\Big(\frac{\|y - y(t)\|_2}{N^2}\mathcal{C}(\eta_{local} \cdot K, \eta_{global}\eta_{local} \cdot K)\Big).$$

Definition 3. *Let's define the matrix $H(t,k)$ coming from the combination of the S_c columns of $H(t,k,c)$ for all $c \in [N]$:*

$$H(t,k)_{i,j} := H(t,k,c)_{i,j}, \quad \forall j \in S_c.$$

When $k = 0$, we can change the notation:

$$H(t)_{i,j} := H(t,0)_{i,j} = \frac{1}{m}\sum_{r=1}^{m} x_i^T x_j \mathbb{I}_{u_r(t)^T x_i \geq 0,\ u_r(t)^T x_j \geq 0},$$

Observe that the definition is well posed because: $S_1 \cup S_2 \cup \cdots \cup S_N = [n]$ and $S_i \cap S_j = \emptyset$. On the basis of these properties, we can see that the dynamics of the error behaves in the following way:

$$\|y - y(t+1)\|_2^2 \simeq \|y - y(t)\|_2^2 - 2 \sum_{k \in [K]} (y - y(t))^T H(t,k)(y - y^{(k)}(t)).$$

$H(t,k)$ is not symmetric (unlike the classical NTK) and the intermediate model states influence the globale update. The following facts will be useful to address these issues.

Proposition 2.

iv) If $R \in (0,1)$ and $u_1(0), ..., u_m(0) \overset{iid}{\sim} \mathcal{N}(0, I)$, then with probability at least $1 - n^2 \exp(-mR/10)$, for all $t \in [T]$ and $k \in [K]$:

$$\|H(t,k) - H(0)\|_F < 2nR,$$

if weights change lazily, $\forall k \in [K]$, $c \in [N]$, $r \in [m]$, $t \in [T]$, $\|w_{k,c,r}(t) - u_r(0)\|_2 \leq R$.

*v) **Global error controls local error**, for all $t \in [T]$ and $k \in [K]$:*

$$\|y - y^{(k)}(t)\|_2 \leq O\Big(\|y - y(t)\|_2 \cdot \mathcal{C}(\eta_{local} \cdot K)\Big).$$

4.2 Main result

Now we can state the main result of the paper.

Theorem 1 (Convergence)**.** *Let $\delta > 0$, $\lambda = \lambda_{\min}(H(0)) > 0$. Let $m = \Omega(\lambda^{-4} n^4 \log(n/\delta))$, we iid initialize $u_r(0) \sim \mathcal{N}(0, I)$, a_r sampled from $\{-1, 1\}$ uniformly, for all $r \in [m]$.*
Set $\eta_{local} = O\big(\lambda/(\kappa K n^2)\big)$, $\eta_{global} = O\big(1\big)$, then with probability at least $1 - \delta$ we have for $t \in [T]$:

$$\|y - y(t)\|_2^2 \leq \Big(1 - \frac{\eta_{local}\eta_{global}\lambda K}{2N}\Big)^t \|y - y(0)\|_2^2.$$

Proof sketch. *In order to prove the linear convergence of Theorem 1, we can show that the global error decreases at each round: for all $t = 0, 1, ...$:*

$$\|y - y(t+1)\|_2^2 \leq \Big(1 - \frac{\eta_{local}\eta_{global}\lambda K}{2N}\Big) \|y - y(t)\|_2^2.$$

As we have seen above:

$$\|y - y(t+1)\|_2^2 = \|y - y(t)\|_2^2 - 2(y - y(t))^T (y(t+1) - y(t)) + \|y(t+1) - y(t)\|_2^2. \tag{2}$$

Focusing on the second term of the sum and using (1), we can write:

$$\begin{aligned}
&- 2(y - y(t))^T (y(t+1) - y(t)) \\
=& - 2(y - y(t))^T (v_1 + v_2) \\
=& - \frac{2\eta_{global}\eta_{local}}{N} \sum_{i \in [n]} \sum_{k \in [K]} \sum_{c \in [N]} \sum_{j \in S_c} (y_i - y_i(t))(y_j - y_c^{(k)}(t)_j)\big(H(t,k,c)_{i,j} - H(t,k,c)^{\perp}_{i,j}\big) \\
&- 2(y - y(t))^T v_2.
\end{aligned}$$

We can rewrite (2) as:

$$\|y - y(t+1)\|_2^2 = \|y - y(t)\|_2^2 + C_1 + C_2 + C_3 + C_4,$$

where

$$C_1 = -\frac{2\eta_{global}\eta_{local}}{N}\sum_{i\in[n]}\sum_{k\in[K]}\sum_{c\in[N]}\sum_{j\in S_c}(y_i - y_i(t))(y_j - y_c^{(k)}(t)_j)H(t,k,c)_{i,j},$$

$$C_2 = \frac{2\eta_{global}\eta_{local}}{N}\sum_{i\in[n]}\sum_{k\in[K]}\sum_{c\in[N]}\sum_{j\in S_c}(y_i - y_i(t))(y_j - y_c^{(k)}(t)_j)H(t,k,c)^{\perp}_{i,j},$$

$$C_3 = -2(y - y(t))^T v_2,$$

$$C_4 = \|y(t+1) - y(t)\|_2^2.$$

Now, exploiting the properties of the section 4.1, we can bound these quantities. In particular:

- *using properties iv) and v), we can prove that with probability at least* $1 - n^2\exp(-mR/10)$*:*

$$C_1 \leq \frac{2\eta_{global}\eta_{local}}{N}\|y - y(t)\|_2^2\big(-K\lambda + 4nRK(1 + 2\eta_{local}Kn) + 2\eta_{local}\kappa\lambda K^2 n\big);$$

- *using properties ii) and v), we can prove that with probability at least* $1 - n\exp(-mR)$*:*

$$C_2 \leq \frac{16\eta_{global}\eta_{local}}{N}\|y - y(t)\|_2^2 KnR\big(1 + 2\eta_{local}nK\big);$$

- *thanks to lazy changes in parameters and activation patterns (properties i) and ii)), we can show that with probability at least* $1 - n\exp(-mR)$*:*

$$C_3 = -2(y - y(t))^T v_2 \leq \frac{16\eta_{global}\eta_{local}K}{N}\big(1 + 2\eta_{local}nK\big)nR\|y - y(t)\|_2^2;$$

- *by property iii), we have that:*

$$C_4 = \|y(t+1) - y(t)\|_2^2 \leq \frac{4\eta_{global}^2\eta_{local}^2 n^2K^2\big(1 + 2\eta_{local}nK\big)^2}{N^2}\|y - y(t)\|_2^2.$$

Let $R = \max_{r\in[m]}\|u_r(t) - u_r(0)\|_2$ be the maximal movement of the weights. Thanks to the property i), R is infinitesimal in m.
Choosing $R \leq \frac{\lambda}{(1000n)}$, $\eta_{\text{local}} \leq \frac{\lambda}{(1000n^2K)}$ and $\eta_{\text{local}}\eta_{\text{global}} \leq \frac{\lambda}{(1000n^2K)}$ and exploiting the bounds of the C_is, we can show:

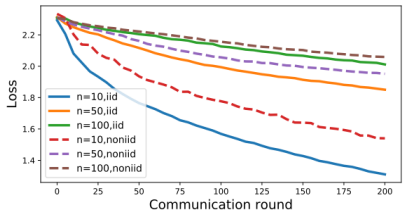
$$\|y - y(t+1)\|_2^2 \leq \|y - y(t)\|_2^2 - \frac{1}{2}\frac{\eta_{\text{local}}\eta_{\text{global}}\lambda K}{N}\|y - y(t)\|_2^2.$$

□

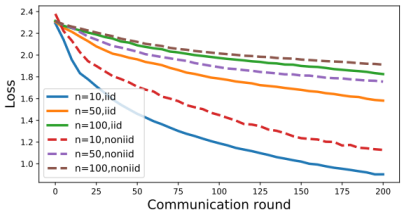
5 Experiments

The experiments consist in a 10 class classification tasks using ResNet56. For fair convergence comparison,the total number of samples n is fixed. Based on our main result Theorem 4.1, the figures show the convergence with respect to the number of client N. There are the two settings: non-iid and iid clients.

- **iid Data distribution**: is homogeneous in all the clients. Specifically, the label distribution over 10 classes is a uniform distribution.
- **non-iid Data distribution**: is heterogeneous in all the clients. For non-IID splits,on every client, training examples are drawn independently with class labels following a categorical distribution over 10 classes parameterized by a vector $\mathbf{q}$ ($q_i \geq 0, i \in [N]$ and $\|\mathbf{q}\|_1 = 1$). To synthesize a population of non-identical clients, $\mathbf{q} \sim Dir(\alpha\mathbf{p})$ is dawn from a Dirichlet distribution, where $\mathbf{p}$ characterizes a prior class distribution over 10 classes, and α, set to 0.5, is a concentration parameter controlling the identicalness among clients.



(a) Local update epoch K =2



(b) Local update epoch K =5

Figure 1: Training loss vs. communication rounds when number of clients N = 10, 50, 100 with iid and non-iid setting using mini-batch SGD optimizer.

6 Comments

i) This paper provides the first general framework able to analyze the convergence of FedAvg with multiple local steps in the overparameterized regime. In particular, the dynamics, which does not follow the gradient direction, is described using an asymmetric matrix;

ii) A notable result is that the convergence is achieved without any assumptions on the convexity of the loss function or the distribution of the data;

iii) An important hypotesis is the choice of η_{local} inversely proportional to K to limit the 'Client-drift' phenomenon;

iv) The authors provide a generalization bound for the global model, with additional assumptions on the distribution of the data;

v) It would be interesting to extend the analysis to different architectures and activation functions. One problem might be the highly explicit approach used in the paper.

References

[1] Baihe Huang et al. "Fl-ntk: A neural tangent kernel-based framework for federated learning analysis". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 4423–4434.

[2] Arthur Jacot, Franck Gabriel, and Clément Hongler. "Neural tangent kernel: Convergence and generalization in neural networks". In: *Advances in neural information processing systems* 31 (2018).