Federated Learning

Mingyi Hong

University Of Minnesota

Outline

- Introduction to Federated Learning
- Assumptions, and popular algorithms
- Convergence analysis for FedAvg
- Other research issues

Introduction

What is Federated Learning (FL)

Federated Learning (FL) is a distributed machine learning approach which enables model training on decentralized data residing on different devices.

What is Federated Learning (FL)

Federated Learning (FL) is a distributed machine learning approach which enables model training on decentralized data residing on different devices.

- Property:
  - Distributed private data
  - Local model training
  - Aggregated at center node(s)

What is Federated Learning (FL)

Federated Learning (FL) is a distributed machine learning approach which enables model training on decentralized data residing on different devices.

- Property:
    - Distributed private data
    - Local model training
    - Aggregated at center node(s)
- Core Issues:
    - Unbalanced data
    - Asynchronous Communication
    - Privacy & Security

Federated Learning (FL)

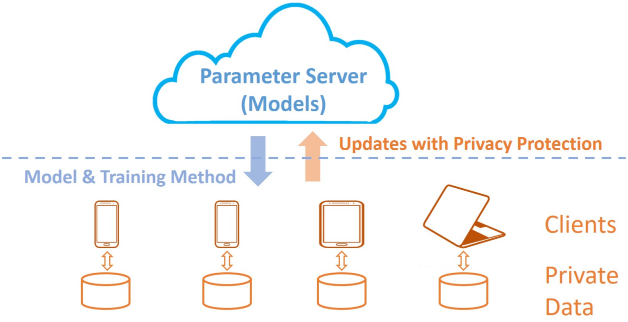


Figure 1.1: System structure of federated learning

- Parameter server network
- Massively distributed data
- Communication compression

FL System Structure

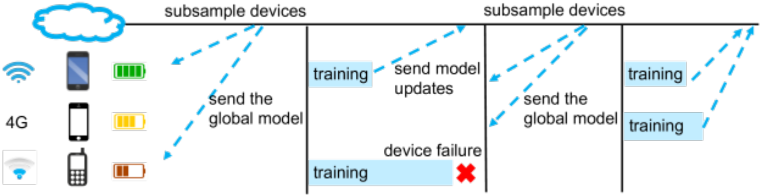


Figure 1.2: Workflow of federated learning

Figure [Li+19] illustrates two rounds of global update with possible local failure.

Server Aspect

- Coordinators
  - coordinate global synchronization
  - instruct selectors to select agents
  - create aggregators
- Aggregators
  - manage training procedures
  - aggregate the local updates
- Selectors
  - accept and forward agents to aggregators
  - receive instructions to select agents

Agent Aspect

- Configure
  - setup FL application
  - connects to the server
- Task Execution
  - receives model and metrics and train the model
- Report
  - reports the model and logs to the server.

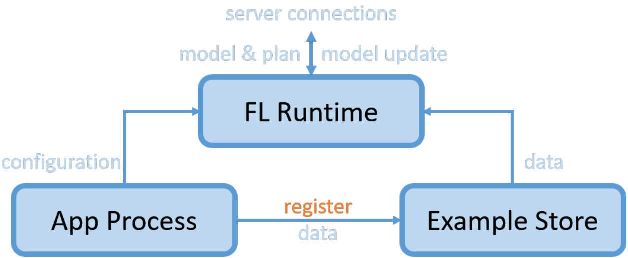
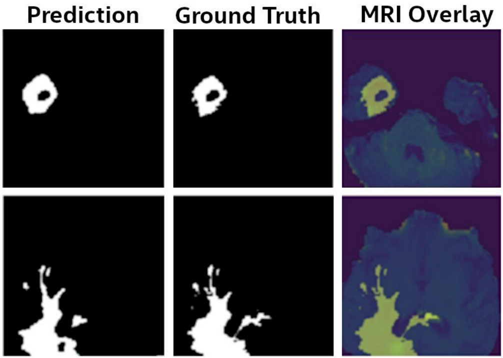


Figure 1.3: Agent side system structure

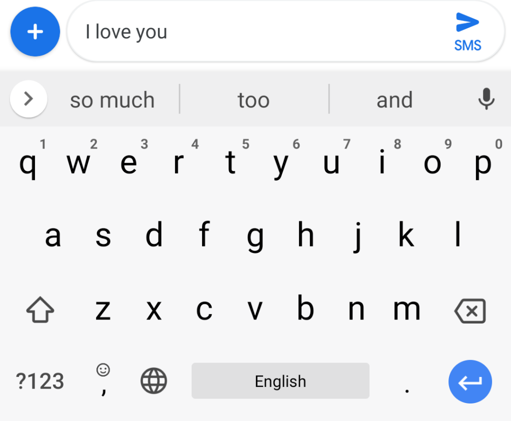
Applications of FL

Figure 1.4: Medical Imaging [She+19]



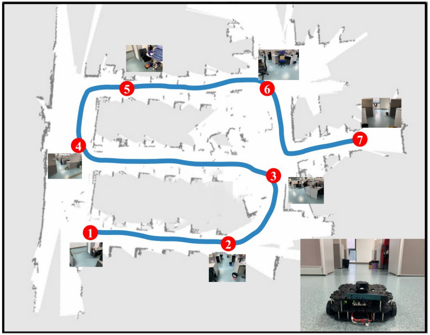
Applications of FL

Figure 1.5: Keyboard Prediction [Yan+18]



Applications of FL

Figure 1.6: Robot Control [Liu+19]



Connection of FL and Decentralized Learning

- From network topology, FL can be viewed as Decentralized Learning + star network
- So algorithms for the latter case can be modified to apply to FL
- FL itself have some distinctive features, so require new algorithm design and analysis
  - Users typically are asynchronous, and prefer to perform multiple local updates before communicating to the server
  - Typically the models (that is, algorithm parameters $\mathbf{x}$'s) are transmitted, but not the local gradients (which could leak useful information about local data)
  - Explicitly need to deal with privacy / security issues

Algorithms

Related Work

- Framework
  - FL Framework (Jakub Konecny et al. '16)
  - FL at Scale (Keith Bonawitz et al. '19)
- Overview
  - Overview on FL (Smith, Virginia et al. '19)
  - FL in Mobile Edge Networks (Qiang Yang et al. '19)
  - FL for Wireless Communication (Jeffery H. Reed et al. '19)
- Algorithm & Applications
  - SecureBoost (Vertical FL) (Qiang Yang et al. '19)
  - Brain Tumor Segmentation (Micah J. Sheller et al. '19)
  - In-Edge AI (Xiaofei Wang et al. '18)
  - Google Keyboard (Timothy Yang et al. '19)

FL Algorithm Design

- FedAvg [Sti19; Li+19]:
  - skips communication of centralized algorithm,
  - requires bounded local update number;
- Distributed-SVRG [Cen+19]:
  - naturally distributed algorithm,
  - requires more server operation;
- FedProx [Sah+18]:
  - local functions different from global function,
  - locally solves to certain accuracy,

Finite-sum Problem

Assume we have N clients with private data sets $\mathcal{D}_i$, each with $n_i = |\mathcal{D}_i|$ data points on client i, then we can write the problem as

$$\min_x f(x) \triangleq \frac{1}{N}\sum_{i=1}^{N} f_i(x) \quad \text{where} \quad f_i(x) \triangleq \frac{1}{n_i}\sum_{\xi_i \in \mathcal{D}_i} F(x;\xi_i) \tag{2.1}$$

Related Algorithms: Local SGD, Parallel Restarted SGD, FedAvg, FedProx, Communication Efficient SGD, Q-Sparse SGD, Cooperative SGD, etc.

Algorithm Design

Input: Max iteration # T, initial point $\mathbf{x}^0$, local iteration # Q.
Initialize: $\mathbf{x}_i^0 \triangleq \mathbf{x}^0, i = 1, \ldots, N$
for $r = 0, \ldots, T-1$ **do**
    for $i = 1, \ldots, N$ *in parallel* **do**
        Randomly samples ξ_i^r form $\mathcal{D}_i$
        $\mathbf{x}_i^{r+1} \triangleq \mathbf{x}_i^r - \gamma^r \nabla F(\mathbf{x}_i^r; \xi_i^r)$
    end
    if $r \mod Q = 0$ **then**
        $\mathbf{x}^{r+1} \triangleq \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i^r$
        $\mathbf{x}_i^{r+1} \triangleq \mathbf{x}^{r+1}, i = 1, \ldots, N$
    end
end
Output: Randomly samples $\mathbf{x}^r \in \{\mathbf{x}^0, \ldots, \mathbf{x}^T\}$.

Algorithm 1: Local SGD (PR-SGD/FedAvg)

Algorithm Design

Input: Max iteration # T, initial point $\mathbf{x}^0$, local iteration # Q.
Initialize: $\mathbf{x}_i^0 \triangleq \mathbf{x}^0, i = 1, \ldots, N$
for $r = 0, \ldots, T-1$ **do**
  for $i = 1, \ldots, N$ *in parallel* **do**
    $\mathbf{x}_i^{r+1} \triangleq \mathbf{x}_i^r - \gamma^r \nabla f_i(\mathbf{x}_i^r)$
  end
  if $r \mod Q = 0$ **then**
    $\mathbf{x}^{r+1} \triangleq \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i^r$
    $\mathbf{x}_i^{r+1} \triangleq \mathbf{x}^{r+1}, i = 1, \ldots, N$
  end
end
Output: $\mathbf{x} = \sum_{r=0}^{T-1} \bar{\mathbf{x}}^r$.

Algorithm 2: Local GD

Assumptions

A1 (Smoothness)

$f(\cdot)$ is L-smooth, $f_i(\cdot)$ are L-smooth

A2 (Unbiased Gradient Estimation)

$\mathbb{E}_{\xi_i \in \mathcal{D}_i} \nabla F(\mathbf{x}; \xi_i) = \nabla f_i(\mathbf{x}),\ \forall\ i, \mathbf{x}$

A3 (Bounded gradient variance)

$\mathbb{E}_{\xi_i \in \mathcal{D}_i} \|\nabla F(\mathbf{x}; \xi_i) - \nabla f_i(\mathbf{x})\|^2 \leq \sigma^2,\ \forall\ i, \mathbf{x}$

A4 (Bounded gradient)

$\|\nabla f_i(\mathbf{x})\|^2 \leq G^2,\ \forall\ f_i, \mathbf{x}$

The FedAvg-type algorithm

- **Question:** FedAvg seems very simple and intuitive, but is it a good algorithm (from algorithmic perspective)?
- Compared with what we studied before, what's the difference / similarities?

Divergence of FedAvg

Lemma 2.1

Suppose that Assumption 1-2 holds true, but without BG, or without both BG and i.i.d. Then FedAvg with local-GD and local SGD can diverge to infinity for any $Q > 1$.

- Both BG and i.i.d. are essential for FedAvg
- Otherwise meaningless solution could be generated
- Why this happens? Centralized algorithm will not have this; Because bad directions? or we should not perform averaging?

Data Heterogeneity

- $\frac{1}{N}\sum_{i=1}^{N}\|\nabla f_i(\mathbf{x}^\star)\|^2 \leq \sigma_f^2,\ i=1,\ldots,N,$ where σ_f is a constant [KMR19],
- $\frac{1}{N}\sum_{i=1}^{N}\|\nabla f_i(\mathbf{x}) - \nabla f(\mathbf{x})\|^2 \leq \kappa,\ \forall \mathbf{x} \in \Re^d$, where κ is a constant [YJY19],
- $|\langle \nabla f_i(\mathbf{x}_i), \nabla f_j(\mathbf{x}_j)\rangle| \leq \beta,\ \forall i \neq j, \mathbf{x}_i \in \{\mathbf{x}_i^{r,q}\}$, where β is a constant [Had+19],
- $\frac{1}{N}\sum_{i=1}^{N}\|\nabla f_i(\mathbf{x})\|^2 \leq \|\nabla f(\mathbf{x})\|^2 B^2,\ \forall \|\nabla f(\mathbf{x})\|^2 \geq \epsilon$, where B is a constant [Sah+18].

Convergence Results

Table 1: The convergence of federated learning algorithms, the Local GD algorithm is a deterministic algorithm and D-SVRG use global full gradient.

Algorithm	CVX	i.i.d	BG	Convergence Rate
FedAvg [Sti19]	$+$	Yes	No	$\mathcal{O}(1/QT) + \mathcal{O}(1/T^2)$
FedAvg [Li+19]	$+$	No	Yes	$\mathcal{O}(1/QT) + \mathcal{O}(Q/T)$
Coop-SGD [WJ18]	-	Yes	No	$\mathcal{O}(1/\sqrt{QT}) + \mathcal{O}(1/T)$
Moment-PRSGD [YJY19]	-	No	Yes	$\mathcal{O}(1/\sqrt{QT}) + \mathcal{O}(Q/T)$
FedProx [Sah+18]	-	No	Yes	$\mathcal{O}(1/T)$
Local-GD [KMR19]	0	No	No	$\mathcal{O}(1/\sqrt{QT}) + \mathcal{O}(Q/T)$
D-SVRG [Cen+19]	-	No	No	$\mathcal{O}(1/T)$

I.I.D: best rate $\mathcal{O}(1/T^2)$ without bounded gradient;

Non-I.I.D: $\mathcal{O}(1/T)$ or slower with bounded gradient or full gradient.

"$+$" strongly convex, "0" convex, "-" non-convex

Convergence Results (cont.)

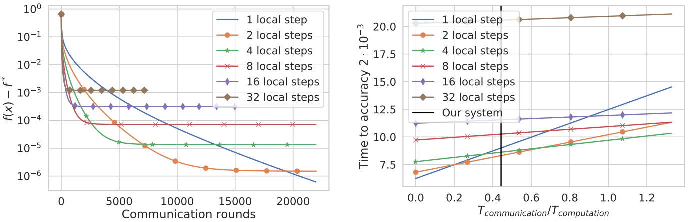


Figure 2.1: Convergence of local GD methods with different number of local steps. 1 local step corresponds to fully synchronized gradient descent. The left plot shows convergence in terms of communication rounds, showing a clear advantage of local GD when only limited accuracy is required. The right plot shows what changes with different communication cost.

Main Result: Local GD

Notations:

$$\bar{\mathbf{x}}^r \triangleq \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i^r, \quad V_r \triangleq \frac{1}{N}\sum_{i=1}^{N}\left\|\mathbf{x}_i^r - \bar{\mathbf{x}}^r\right\|^2, \quad g_r \triangleq \frac{1}{N}\sum_{i=1}^{N}\nabla f_i(\mathbf{x}_i^r)$$

$$e_r \triangleq \bar{\mathbf{x}}^r - \mathbf{x}^\star, D_f(x,y) \triangleq f(x) - f(y) - \langle \nabla f(y), x - y\rangle .$$

Theorem 2.2

For local GD run with a constant stepsize $0 < \gamma \leq \frac{1}{4LQ}$ *and under Assumption 1, if each* $f_i(\cdot)$ *is convex, we have*

$$\frac{1}{T}\sum_{r=0}^{T-1} f(\bar{\mathbf{x}}^r) - f(\mathbf{x}^\star) \leq \frac{2\left\|\mathbf{x}^0 - \mathbf{x}^\star\right\|^2}{\gamma T} + 24\gamma^2\sigma_f^2 Q^2 L. \tag{2.2}$$

Proof steps summarize from [KMR19].

Main Result: Local GD (cont.)

- We can also quantify the communication efficiency

- If desired accuracy is

$$\epsilon \triangleq \frac{1}{T}\sum_{r=0}^{T-1} f(\bar{\mathbf{x}}^r) - f(\mathbf{x}^\star) \geq 3\sigma_f^2/L,$$

  Then we should choose $T/Q = \mathcal{O}(1/\epsilon)$

- Else, if $\epsilon < 3\sigma_f^2/L$, then $T/Q = \mathcal{O}(1/\epsilon^{3/2})$, [e.g., $T = \mathcal{O}(\epsilon^{-2}), Q = \mathcal{O}(\epsilon^{1/2})$]

- To get a convergence rate of $1/\sqrt{NT}$ we choose $\gamma = \frac{\sqrt{N}}{4L\sqrt{T}}$, $Q = \mathcal{O}(T^{1/4}N^{-3/4})$, $T/Q = \Omega(T^{3/4}N^{3/4})$. If a rate of $1/\sqrt{T}$ is desired instead, we can choose $Q = \mathcal{O}(T^{1/4})$.

Proof Outline: Step 1

Lemma 2.3

For any $\gamma \geq 0$ *we have*

$$\|e_{r+1}\|^2 \leq \|e_r\|^2 + \gamma L(1+2\gamma L)V_r - 2\gamma(1-2\gamma L)D_f(\bar{\mathbf{x}}^r, \mathbf{x}^\star). \tag{2.3}$$

In particular, if $\gamma \leq \frac{1}{4L}$, *then* $\|e_{r+1}\|^2 \leq \|e_r\|^2 + \frac{3}{2}\gamma L V_r - \gamma D_f(\bar{\mathbf{x}}_r, \mathbf{x}^\star)$.

Proof Outline: Step 2

Lemma 2.4

Suppose that A1 holds and each $f_i(\cdot)$ convex, let $r_0 \mod Q = 0$ denotes the communication iterations, define $v \triangleq r_0 + Q$.
Suppose Algorithm 2 is run with a constant stepsize $\gamma > 0$ such that $\gamma \leq \frac{1}{4LQ}$. Then the following inequalities hold:

$$\sum_{r=r_0+1}^{v} V_r \leq 5L\gamma^2 Q^2 \sum_{r=r_0+1}^{v} D_f(\bar{\mathbf{x}}^r, \mathbf{x}^\star) + 8\gamma^2 Q^3 \sigma_f^2,$$

$$\sum_{r=r_0+1}^{v} \left(\frac{3}{2} L V_r - D_f(\bar{\mathbf{x}}^r, \mathbf{x}^\star) \right) \leq -\frac{1}{2} \sum_{r=r_0+1}^{v} D_f(\bar{\mathbf{x}}^r, \mathbf{x}^\star) + 12 L\gamma^2 Q^3 \sigma_f^2.$$

Note: Recall that since $\nabla f(\mathbf{x}^\star) = 0$, we have

$$D_f(\bar{\mathbf{x}}^r, \mathbf{x}^\star) = f(\bar{\mathbf{x}}^r) - f(\mathbf{x}^\star) \tag{2.4}$$

Preliminary

Lemma 2.5

Suppose that A1 holds and each $f_i(\cdot)$ *convex, then*

$$\|g_r\|^2 \le 2L^2 V_r + 4LD_f(\bar{\mathbf{x}}^r, x^\star). \tag{2.5}$$

Lemma 2.6

Suppose that A1 holds and each $f_i(\cdot)$ *convex. Then,*

$$-\frac{2}{N}\sum_{i=1}^{N}\langle \bar{\mathbf{x}}^r - x^\star, \nabla f_i(x_i^r)\rangle \le -2D_f(\bar{x}^r, x^\star) + LV_r. \tag{2.6}$$

Proof of Lemma 2.5

Starting with the left-hand side,

$$
\begin{aligned}
\|g_r\|^2 &\le 2\left\|g_r - \nabla f(\bar{x}^r)\right\|^2 + 2\left\|\nabla f(\bar{x}^r)\right\|^2 \\
&= 2\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla f_i(x_i^r) - \frac{1}{N}\sum_{i=1}^{N}\nabla f_i(\bar{x}^r)\right\|^2 + 2\left\|\nabla f(\bar{x}^r)\right\|^2 \\
&\le \frac{2}{N}\sum_{i=1}^{N}\left\|\nabla f_i(x_i^r) - \nabla f_i(\bar{x}^r)\right\|^2 + 2\left\|\nabla f(\bar{x}^r)\right\|^2 \\
&\le \frac{2L^2}{N}\sum_{i=1}^{N}\left\|x_i^r - \bar{x}^r\right\|^2 + 2\left\|\nabla f(\bar{x}^r)\right\|^2.
\end{aligned}
$$

The claim of the lemma follows by noting that

$$\left\|\nabla f(\bar{x}^r)\right\|^2 = \left\|\nabla f(\bar{x}^r) - \nabla f(x^\star)\right\|^2 \le 2LD_f(\bar{x}^r, x^\star).$$

Proof of Lemma 2.6

Starting with the left-hand side,

$$
\begin{aligned}
-2\left\langle \bar{x}^r - x^\star, \nabla f_i(x_i^r)\right\rangle &= -2\left\langle \bar{x}^r - x_i^r + x_i^r - x^\star, \nabla f_i(x_i^r)\right\rangle \\
&\overset{(a)}{\le} 2(f_i(x^\star) - f_i(x_i^r)) - 2\left\langle \bar{x}^r - x_i^r, \nabla f_i(x_i^r)\right\rangle \\
&\overset{(b)}{\le} 2(f_i(x^\star) - f_i(x_i^r)) - 2(f_i(x_i^r) - f_i(\bar{x}^r) + \frac{L}{2}\left\|x_i^r - \bar{x}^r\right\|^2) \\
&= 2(f_i(x^\star) - f_i(\bar{x}^r) + \frac{L}{2}\left\|x_i^r - \bar{x}^r\right\|^2).
\end{aligned}
\tag{2.7}
$$

where (a) comes from convexity, and (b) we use L-smoothness. Averaging over i,

$$
\begin{aligned}
-\frac{2}{N}\sum_{i=1}^{N}\left(\bar{x}^r - x^\star, \nabla f_i(x_i^r)\right) &\le -2(f(\bar{x}^r) - f(x^\star)) + \frac{L}{N}\sum_{i=1}^{N}\left\|x_i^r - \bar{x}^r\right\|^2 \\
&= -2D_f(\bar{x}^r, x^\star) + LV_r,
\end{aligned}
$$

which is the claim of this lemma.

Proof of Lemma 2.3

- Then we go back to our main steps of showing descent
- We will first show Lemma 2.3

Proof of Lemma 2.3

Note that $\bar{x}_{t+1} = \bar{x}^r - \gamma g_r$ always holds (average update).
Then we have,

$$
\begin{aligned}
\|e_{r+1}\|^2 &= \|\bar{x}^r - \gamma g_r - x^\star\|^2 \\
&= \|e_r\|^2 + \gamma^2 \|g_r\|^2 - 2\gamma \langle \bar{x}^r - x^\star, g_r \rangle \\
&= \|e_r\|^2 + \gamma^2 \|g_r\|^2 - \frac{2\gamma}{N} \sum_{i=1}^{N} \langle \bar{x}^r - x^\star, \nabla f_i(x_i^r) \rangle \\
&\overset{(2.5)}{\leq} \|e_r\|^2 + \gamma^2 (2L^2 V_r + 4L D_f(\bar{x}^r, x^\star)) \\
&\qquad - \frac{2\gamma}{N} \sum_{i=1}^{N} \langle \bar{x}^r - x^\star, \nabla f_i(x_i^r) \rangle \\
&\overset{(2.6)}{\leq} \|e^r\|^2 + \gamma L(1 + 2\gamma L) V_r - 2\gamma(1 - 2\gamma L) D_f(\bar{x}^r, x^\star).
\end{aligned}
$$

Proof of Lemma 2.3

In short:

$$\begin{aligned}\|e_{r+1}\|^2 &= \|\bar{x}^r - \gamma g_r - x^\star\|^2 \\ &\le \|e^r\|^2 + \gamma L(1+2\gamma L)V_r - 2\gamma(1-2\gamma L)D_f(\bar{x}^r, x^\star).\end{aligned}$$

If $\gamma \le \frac{1}{4L}$, then $1 - 2\gamma L \ge \frac{1}{2}$ and $1 + 2\gamma L \le \frac{3}{2}$, and hence

$$\|e_{t+1}\|^2 \le \|e_r\|^2 + \frac{3}{2}\gamma L V_r - \gamma D_f(\bar{x}^r, x^\star).$$

The proof is completed

Proof outline of Lemma 2.4

First we prove [easy, omitted]

$$V_r \leq \frac{\gamma^2 Q}{N} \sum_{i=1}^{N} \sum_{\tau=r_0+1}^{r} \left\| \nabla f_i(x_i^\tau) \right\|^2 ,$$

$$\left\| \nabla f_i(x_i^r) \right\|^2 \leq 3L^2 \left\| x_i^r - \bar{x}^r \right\|^2 + 4L D_{f_i}(\bar{x}^r, x^\star) + 6 \left\| \nabla f_i(x^\star) \right\|^2 .$$

If the above are true, then sum from $r_0 + 1$ to $v = r_0 + Q$

$$\sum_{r=r_0+1}^{v} V_r \leq 3L^2 \gamma^2 Q^2 \sum_{r=r_0+1}^{v} V_r + 4L\gamma^2 Q^2 \sum_{r=r_0+1}^{v} D_f(\bar{x}^r, x^\star)$$

$$+ \sum_{r=r_0+1}^{v} 6\gamma^2 Q^2 \sigma_f^2 .$$

Proof outline of Lemma 2.4 (cont.)

Move the terms of V_r to the left we have

$$(1 - 3L^2\gamma^2Q^2) \sum_{r=r_0+1}^{v} V_r \le 4L\gamma^2Q^2 \sum_{r=r_0+1}^{v} D_f(\bar{x}^r, x^\star) + 6\gamma^2Q^3\sigma_f^2. \quad (2.8)$$

Multiply both side by $3L/2$ and subtract $\sum_{i=r_0+1}^{v} D_f(\bar{x}^r, x^\star)$, we also have

$$\sum_{r=r_0+1}^{v} \frac{3}{2}LV_r - \sum_{r=r_0+1}^{v} D_f(\bar{x}^r, x^\star) \le (\frac{15}{2}L^2\gamma^2Q^2 - 1) \sum_{r=r_0+1}^{v} D_f(\bar{x}^r, x^\star) + \frac{45}{4}L\gamma^2Q^3\sigma_f^2.$$

Note that because $\gamma \le \frac{1}{4LQ} \le \frac{1}{\sqrt{15}LQ}$, then our choice of γ implies that $1 - 3L^2\gamma^2Q^2 \ge \frac{4}{5}$ and $\frac{15}{2}L^2\gamma^2H^2 - 1 \le -\frac{1}{2}$.

Other Issues

Heterogeneous Data Issues

- Most of the decentralized algorithms do not have heterogeneous data issues
- By FedAvg-type algorithms have
- The reason is that if a node is too focused on the local update, it can go too far away to the wrong directions
- Need generic algorithm design to deal with heterogeneity, while being able to harness homogeneity

Communication Issues

- Communication efficiency
  - I.I.D: $Q = \mathcal{O}(T)$
  - Non-I.I.D: $Q \leq \mathcal{O}(T^{1/3})$
- Asynchronous update
  - Hodwild! [Ngu+18]
    bounded delay between communication
  - Event-triggered [Li+19] algorithm
    bounded distance from global (bounded local update)
    diminishing distance (increasing communication frequency)

Privacy Issue

- No privacy issue
- Server level privacy: against third-party
- Client level privacy: against server

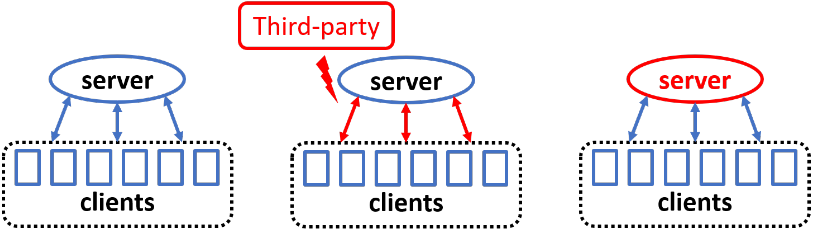


Figure 4.1: Three privacy issues in federated learning [Li+19], the system on the left has no privacy issue, the system in the middle needs to defend against the third-party and the system on the right has a malicious server.

Privacy Preserving

- Add noise to the aggregation step on the server [GKN17], defend against the third-party, the server need to be trusted;
- Add noise to the updated model [HMV15],
- Secure aggregation [Bon+17], defend against the malicious server; cannot defend against the third-party.

System Security

- Attacks
  - degrade the performance
  - meet targeted behavior
- Solutions
  - using medium instead of mean
  - active sampling the agents
  - adptive weighting the model

References

Keith Bonawitz et al. "Practical Secure Aggregation for Privacy-Preserving Machine Learning". In: *Conference on Computer and Communications Security*. CCS '17. Dallas, Texas, USA: ACM, 2017, pp. 1175–1191. ISBN: 978-1-4503-4946-8.

Shicong Cen et al. "Convergence of Distributed Stochastic Variance Reduced Methods without Sampling Extra Data". In: *arXiv preprint arXiv:1905.12648* (2019).

Robin C Geyer, Tassilo Klein, and Moin Nabi. "Differentially private federated learning: A client level perspective". In: *arXiv preprint arXiv:1712.07557* (2017).

Farzin Haddadpour et al. "Trading Redundancy for Communication: Speeding up Distributed SGD for Non-convex Optimization". In: *International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. Long Beach, California, USA: PMLR, 2019, pp. 2545–2554.

References

Zhenqi Huang, Sayan Mitra, and Nitin Vaidya. "Differentially private distributed optimization". In: *International Conference on Distributed Computing and Networking*. ACM. 2015, p. 4.

Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. "First analysis of local gd on heterogeneous data". In: *arXiv preprint arXiv:1909.04715* (2019).

Tian Li et al. "Federated learning: Challenges, methods, and future directions". In: *arXiv preprint arXiv:1908.07873* (2019).

W. Li et al. "COLA: Communication-censored Linearized ADMM for Decentralized Consensus Optimization". In: *International Conference on Acoustics, Speech and Signal Processing*. 2019, pp. 5237–5241.

Xiang Li et al. "On the convergence of fedavg on non-iid data". In: *arXiv preprint arXiv:1907.02189* (2019).

References

Boyi Liu et al. "Lifelong federated reinforcement learning: a learning architecture for navigation in cloud robotic systems". In: *arXiv preprint arXiv:1901.06455* (2019).

Lam M Nguyen et al. "SGD and Hogwild! convergence without the bounded gradients assumption". In: *arXiv preprint arXiv:1802.03801* (2018).

Anit Kumar Sahu et al. "On the convergence of federated optimization in heterogeneous networks". In: *arXiv preprint arXiv:1812.06127* (2018).

Micah J. Sheller et al. "Multi-institutional Deep Learning Modeling Without Sharing Patient Data: A Feasibility Study on Brain Tumor Segmentation". In: *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*. Ed. by Alessandro Crimi et al. Springer International Publishing, 2019, pp. 92–104. ISBN: 978-3-030-11723-8.

References

Sebastian Urban Stich. "Local SGD Converges Fast and Communicates Little". In: *International Conference on Learning Representations* (2019), p. 17.

Jianyu Wang and Gauri Joshi. "Cooperative SGD: A unified framework for the design and analysis of communication-efficient SGD algorithms". In: *arXiv preprint arXiv:1808.07576* (2018).

Timothy Yang et al. "Applied federated learning: Improving google keyboard query suggestions". In: *arXiv preprint arXiv:1812.02903* (2018).

Hao Yu, Rong Jin, and Sen Yang. "On the Linear Speedup Analysis of Communication Efficient Momentum SGD for Distributed Non-Convex Optimization". In: *International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. Long Beach, California, USA: PMLR, 2019, pp. 7184–7193.