Introduction to Federated Learning

Suyeong Park

May 23, 2022

Machine Learning and Vision Lab, UNIST

Table of Contents

1. Introduction

Motivation

- Decentralized data
- Data privacy preserving
- Local device HW resources

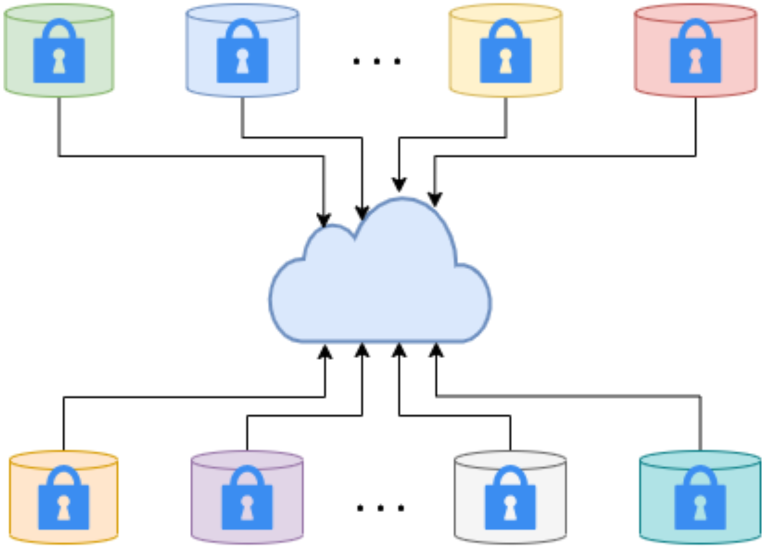


Figure 1: decentralized setting with data privacy

Examples

- Gboard on Android
- Media playback preferences in Safari
- Voice assistant in Siri
- Popular health data types

Examples

Figure 2: Gboard on Android [1]

[1] source: https://ai.googleblog.com/2017/04/federated-learning-collaborative.html

Federated Learning

Examples

MIT Technology Review

Sign in

Subscribe

Artificial intelligence / Machine learning

How Apple personalizes Siri without hoovering up your data

The tech giant is using privacy-preserving machine learning to improve its voice assistant while keeping your data on your phone.

by Karen Hao

December 11, 2019

Figure 3: Apple [2]

[2] source: `https://www.technologyreview.com/2019/12/11/131629/apple-ai-personalizes-siri-federated-learning/`

Examples

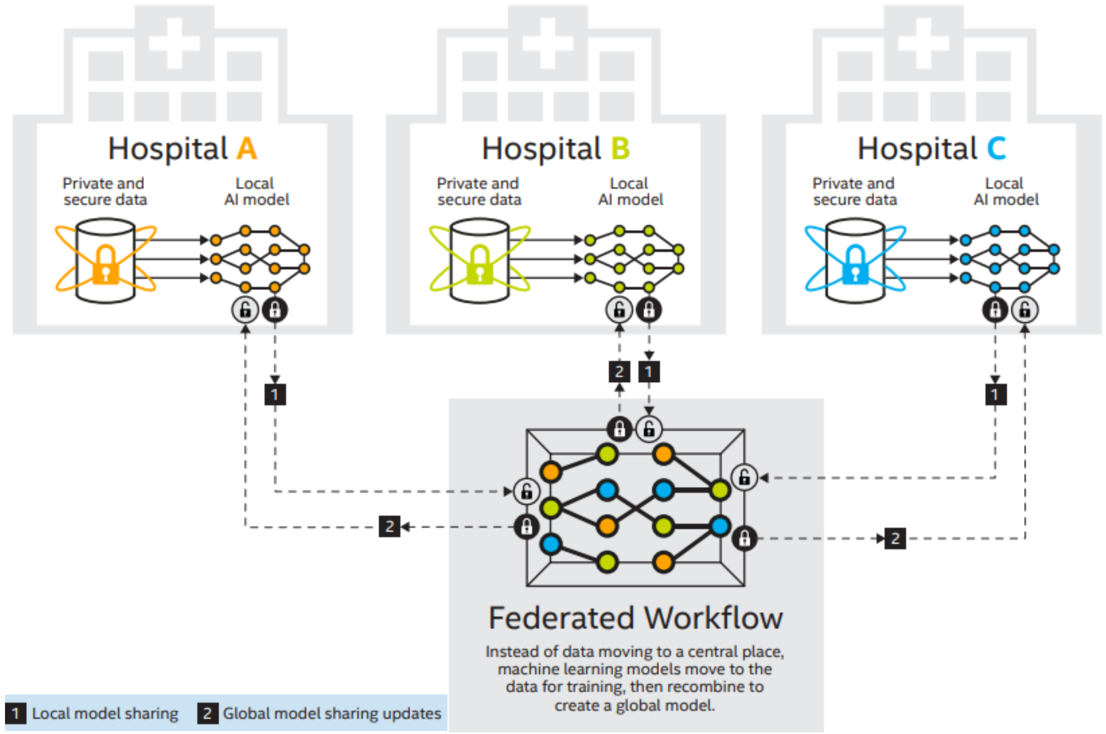


Figure 4: Intel & Hospitals [3]

[3] source: https://newsroom.intel.com/news/intel-works-university-pennsylvania-using-privacy-preserving-ai-identify-brain-tumors/

Federated Learning

Definition [1]

Federated learning(FL) is a machine learning setting where multiple clients collaborate in solving a ML problem, under the coordination of a central server. **Each client's raw data is stored locally and not exchanged or transferred**; instead, updates intended for immediate aggregation are used to achieve the learning objective.

Federated Learning

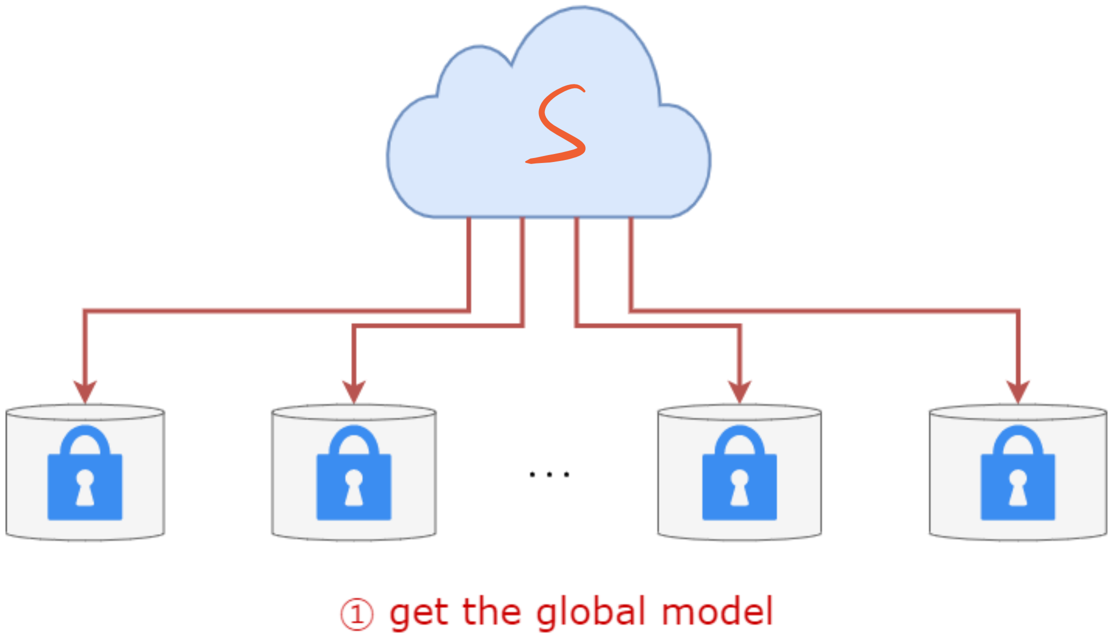


Figure 5: Federated Learning workflow - 1 (client-side)

Federated Learning

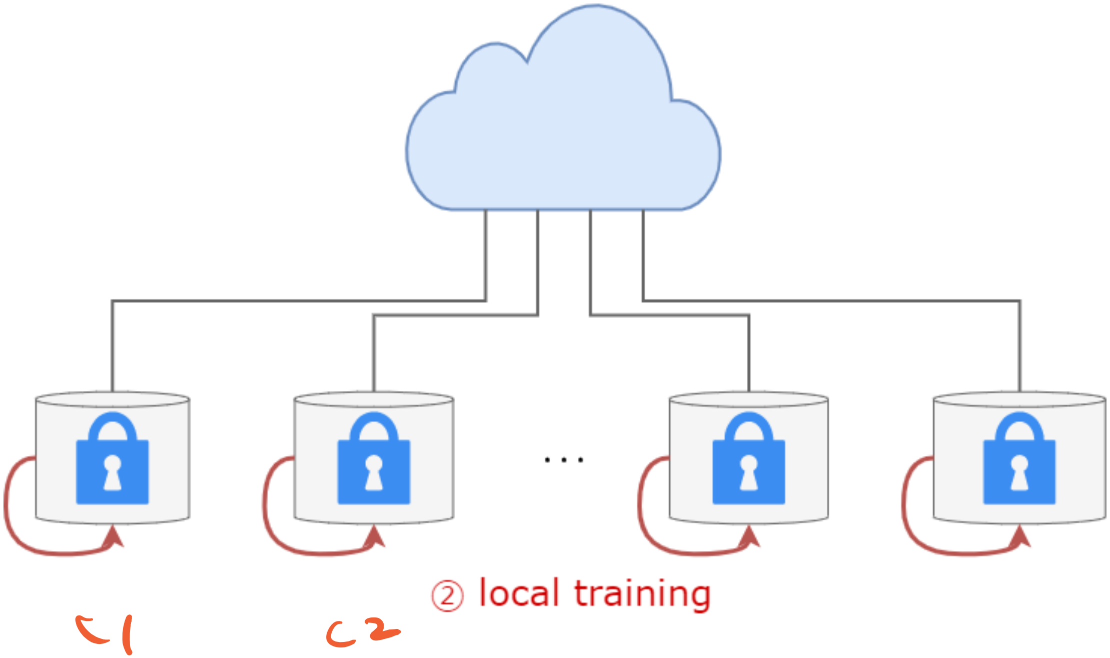


Figure 6: Federated Learning workflow - 2 (client-side)

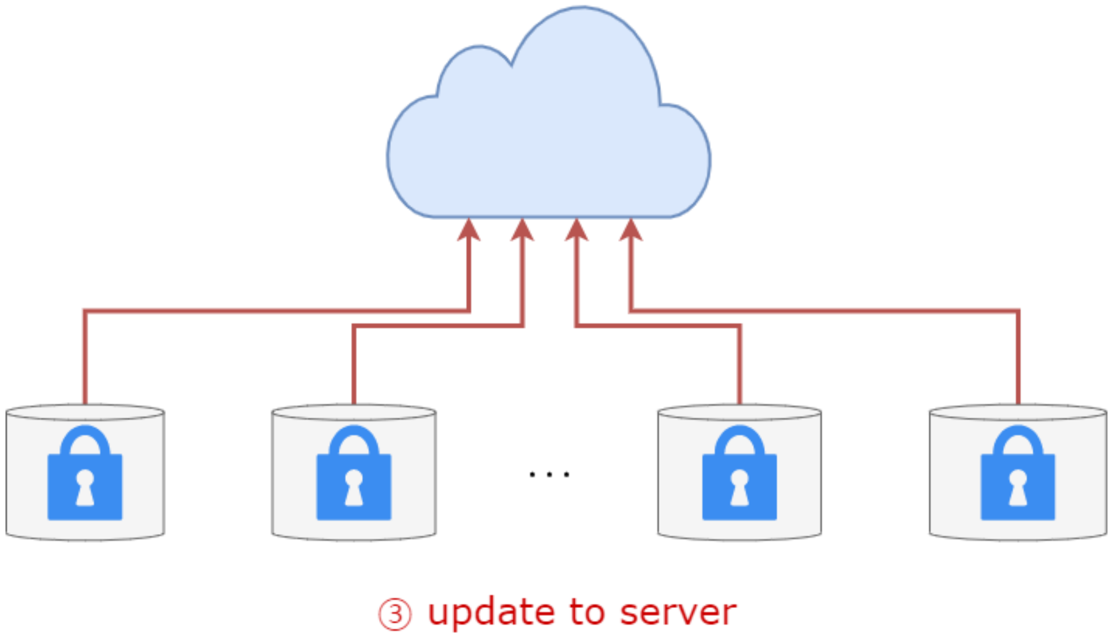


Figure 7: Federated Learning workflow - 3 (client-side)

Federated Learning

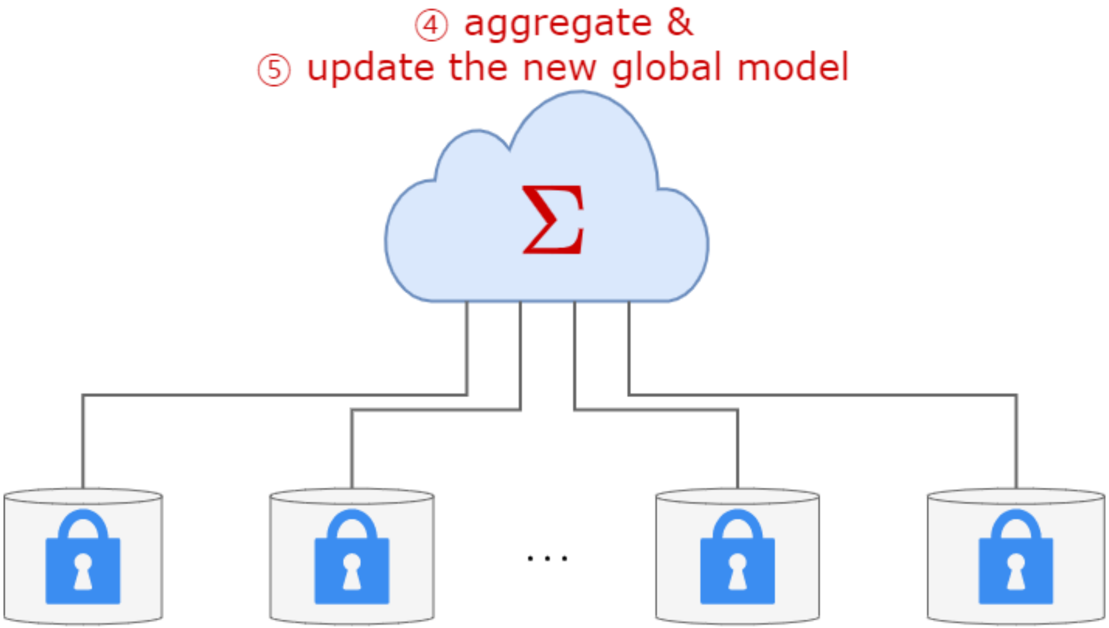


Figure 8: Federated Learning workflow - 4 (server-side)

Federated Learning

Key issues

- **Privacy**
- **Communication costs**

  **communication: transmission between server or clients*
- **Data heterogeneity**: violation of I.I.D. assumption (Non-IID)
- **System heterogeneity**: network bandwidth, asynchronous Internet connections, etc

Two main settings

	Cross-device FL	Cross-silo FL
Example	mobile or IoT devices	medical or financial institutes
Data availability	available only a fraction of clients	available all clients
Distribution scale	massively parallel	2-100 clients
Addressability	not accessible	accessible to client ids
Client statefulness	stateless	stateful
Client reliability	highly unreliable	relatively few failures
Primary bottleneck	connection and communication	computation or communication
Data partition axis	fixed (HFL)	fixed (HFL&VFL)

Table 1: Federated learning settings

2. Algorithms

FedSGD & FedAVG

AISTATS, 2017

FedAVG [2]

Summary

- The first approach to federated learning (FL).
- It simply extended **SGD** to FL setting by **averaging**.
- It proposed two simple algorithms: FedSGD and FedAVG.
- Empirical results show that the FL performance depends on various **hyperparameters**: number of participation clients, number of local epochs and batch size.

FedSGD & FedAVG

Notation	Description
$\mathbf{w}_t$	**model** at t-th round
$\mathbf{w}_t^k$	**model** of k-th client at t-th round
∇f_k	**gradient** of objective on the model of k-th client
n_k	number of local data points of k-th client
K	**number of all clients**
$n = \sum_{k=1}^{K} n_k$	total number of local data points of each client
η	learning rate
C	participation ratio of clients at each round
E	number of local epochs
B	local (mini)batch size
$u_k = E \cdot \frac{n_k}{B}$	number of local updates of k-th client on each round

Table 2: Notations

FedSGD: Federated Stochastic Gradient Descent

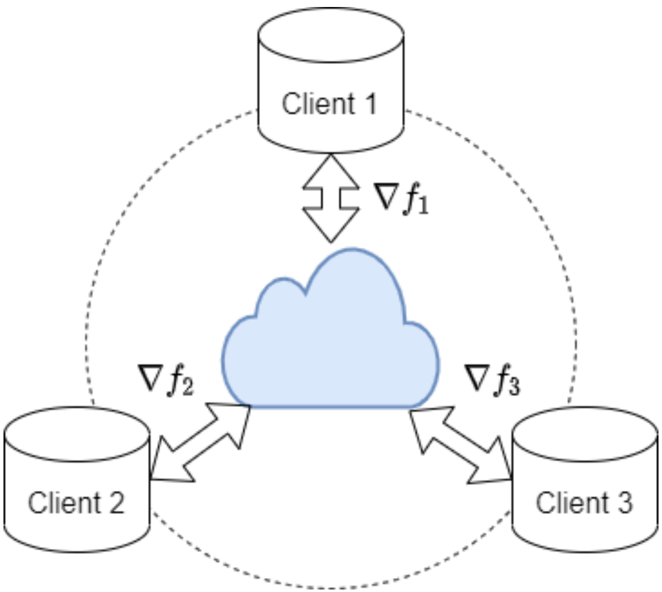


Figure 9: FedSGD

$$\nabla f = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i$$

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \eta \cdot \sum_{k=1}^{K} \frac{n_k}{n} \nabla f_k \tag{1}$$

$\Longleftrightarrow$ GD

identical

FedAVG: Federated Averaging

client

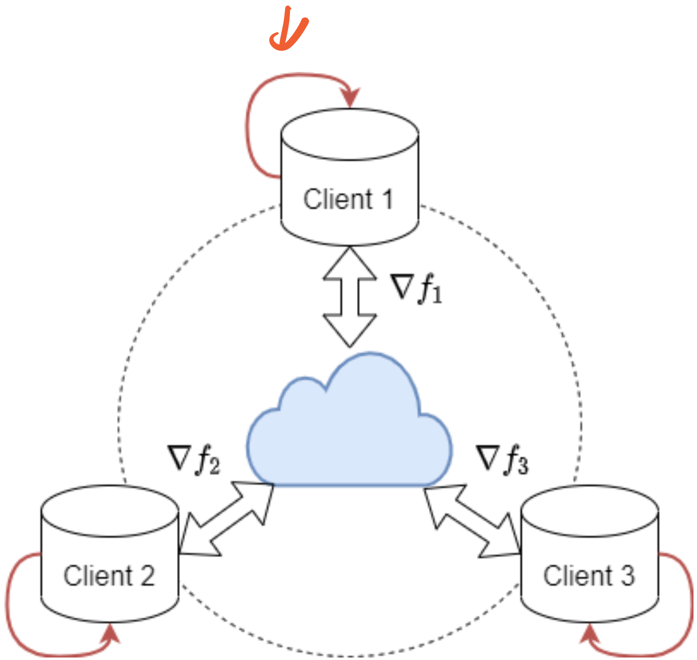


Figure 10: FedAVG

local SGD

$$\mathbf{w}_t^k \leftarrow \mathbf{w}_t^k - \eta \cdot \nabla f_k \qquad (2)$$

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \eta \cdot \sum_{k=1}^{K} \frac{n_k}{n} \nabla f_k \qquad (3)$$

aggregation step at server

FedSGD & FedAVG

FedAVG

Objective of FedAVG

$$\min_{\mathbf{w}} [F(\mathbf{w}) = \sum_{k=1}^{K} p_k F_k(\mathbf{w})] \tag{4}$$

$$\text{where } F_k(\mathbf{w}) = \sum_{\xi \in \mathcal{D}_k} f(\mathbf{w}, \xi)/n_k \tag{5}$$

$$p_k = n_k / \sum_{k=1}^{K} n_k \tag{6}$$

n

FedAVG

Algorithm 1 FedAVG

SERVERUPDATE(K, B, E, η) ▷ server-side

initialize $\mathbf{w}_0$

comm'n

for $t \leftarrow 1, 2, \cdots$ **do**

$m \leftarrow \max(C \cdot K, 1)$

$S_t \leftarrow$ random set of m clients

for $k \leftarrow 1, \cdots, |S_t|$ **in parallel do**

$\mathbf{w}_{t+1}^{k} \leftarrow \text{ClientUpdate}(k, \mathbf{w}_{k}^{t})$ local SGD

Aggregate $\mathbf{w}^{t+1} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} \mathbf{w}_k^{t+1}$

CLIENTUPDATE$(k, \mathbf{w})$ ▷ client-side

$\mathcal{B} \leftarrow$ split $\mathcal{P}_k$ into batches of size B

for $i \leftarrow 1, \cdots, E$ **do**

for $b \in \mathcal{B}$ **do**

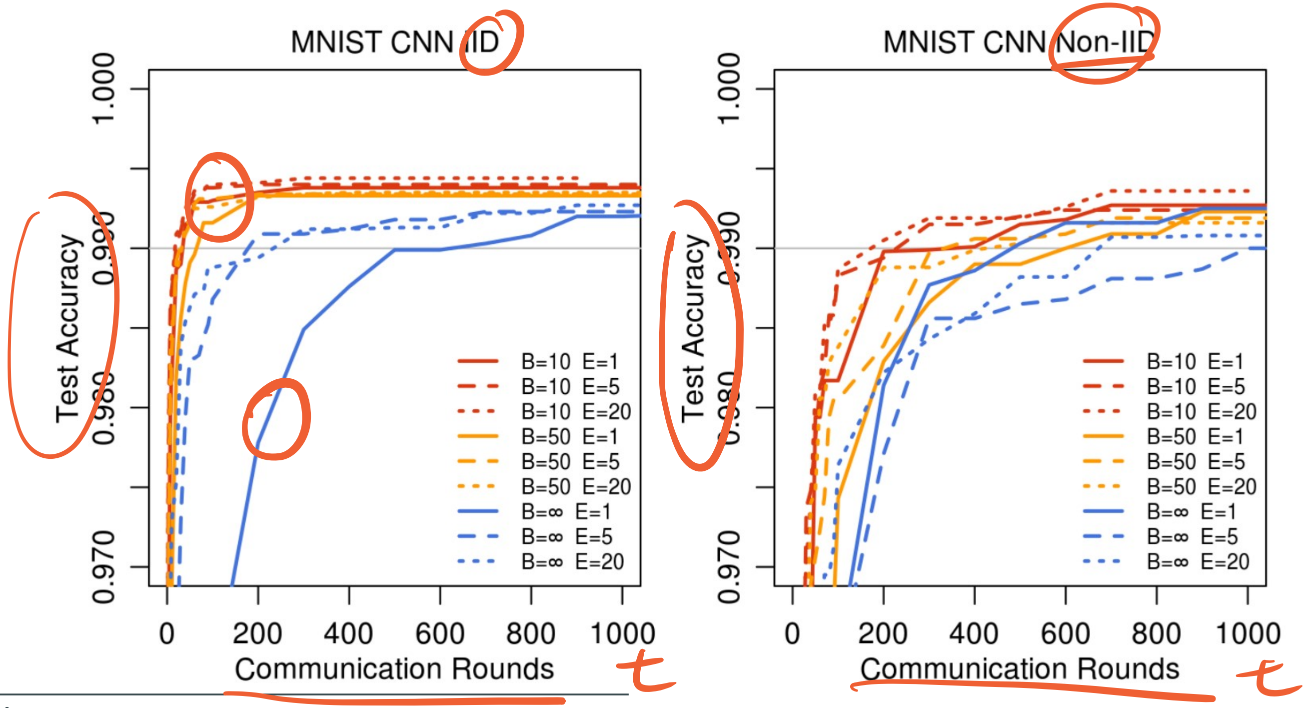
$\mathbf{w} \leftarrow \mathbf{w} - \eta \nabla l(\mathbf{w}; b)$ SGD

return $\mathbf{w}$ to server

FedAVG

Experiments [4]

Test set accuracy over communication rounds ($C = 0.1$) of FedSGD ($B = \infty$) and FedAVG ($B < \infty$).



[4] See Appendix A (75) for experimental settings.

Convergence Analysis of FedAvg [3]

Convergence Analysis Summary

Under assumptions (27) and decaying the learning rate,

$$\mathbb{E}[F(\mathbf{w}_T) - F^*] \leq \mathcal{O}\left(\frac{B+C}{T}\right) \tag{7}$$

$$\text{where } B = \Gamma + (E-1)^2$$

Same as SGD

Remarks

The convergence rate depends on ...

- data heterogeneity $\Gamma := F^* - \sum_{k=1}^{N} p_k F_k^*$
- number of local updates E
- total number of communication updates T

Convergence Analysis of FedAvg

Summary

- Data heterogeneity Γ can lead to slow convergence.
- Too many ~~or less~~ local updates E can lead to slow convergence.
- Too ~~many~~ small participation clients K can lead to slow convergence.
- Sampling with replacement can lead to faster convergence.
- Fixed learning rate ($\eta_t = \eta$) can lead to sub-optimal point when $E > 1$.

Convergence Analysis of FedAvg

cross - device FL

Notation	Description
N	total number of clients
K	number of clients that participate in every round
T	total number of every **local SGD**
E	number of **local iterations** btw 2 communications
$\frac{T}{E}$	number of communications
p_k	weight of k-th client; $p_k \leq 0, \sum_k^N p_k = 1$
$\{x_{k,l}\}_{l=1}^{n_k}$	n_k training data of k-th client
$F_k(\cdot), l(\cdot)$	local objective, local loss function
η_{t+i}	learning rate of i-th update at t-th round
ξ_{t+i}^k	sample uniformly chosen from the local data
$\mathbf{w}_{t+i}^k$	local models of k-th client of i-th update at t-th round

Table 3: Notations

Notations: Data heterogeneity

$$\Gamma = F^* - \sum_{k=1}^{N} p_k F_k^* \tag{8}$$

- F^*, F_k^*: minimum value of F, F_k
- $\mathcal{I}_E$: set of global synchronization steps; $\mathcal{I}_E = \{nE | n = 1, 2, \dots\}$
- $t + 1 \in \mathcal{I}_E$: time step to communication

Convergence Analysis of FedAvg

Notations: Problem formulation

Local objective

$$F_k(\mathbf{w}) = \frac{1}{n_k}\sum_{j}^{n_k} l(\mathbf{w}; x_{k,j}) \tag{9}$$

Local update – client

$$\mathbf{w}_{t+i+1}^k \leftarrow \mathbf{w}_{t+i}^k - \eta_{t+i}\nabla F_k(\mathbf{w}_{t+i}^k, \xi_{t+i}^k), \quad i = 0, 1, \ldots, E-1 \tag{10}$$

Aggregation step – server

$$\mathbf{w}_{t+E} \leftarrow \frac{N}{K}\sum_{k\in\mathcal{S}_t} p_k \mathbf{w}_{t+E}^k \tag{11}$$

Convergence Analysis of FedAvg

Assumptions

Assumption 1 (L-smooth)

$$F_k(\mathbf{v}) \leq F_k(\mathbf{w}) + (\mathbf{v} - \mathbf{w})^T \nabla F_k(\mathbf{w}) + \frac{L}{2} \|\mathbf{v} - \mathbf{w}\|^2, \forall \mathbf{v}, \mathbf{w} \tag{12}$$

Assumption 2 (μ-strongly convex)

$$F_k(\mathbf{v}) \geq F_k(\mathbf{w}) + (\mathbf{v} - \mathbf{w})^T \nabla F_k(\mathbf{w}) + \frac{\mu}{2} \|\mathbf{v} - \mathbf{w}\|^2, \forall \mathbf{v}, \mathbf{w} \tag{13}$$

Assumption 3 (bounded variance of stochastic gradients)

$$\mathbb{E}\|\nabla F_k(\mathbf{w}_t^k, \xi^k) - \nabla F_k(\mathbf{w}_t^k)\|^2 \leq \sigma_k^2, \quad k = 1, \ldots, N \tag{14}$$

Assumption 4 (uniformly bounded expected L2 norm of stochastic gradients)

$$\mathbb{E}\|\nabla F_k(\mathbf{w}_t^k, \xi^k)\|^2 \leq G^2, \quad k = 1, \ldots, N, t = 1, \ldots, T-1 \tag{15}$$

cross-silo

Theorem 1: Convergence when full participation

Theorem 2: Convergence when partial participation (Scheme 1)

Theorem 3: Convergence when partial participation (Scheme 2)

Convergence Analysis of FedAvg

Theorem 1: Convergence when full participation

Under Assumptions 1 to 4, choose $\kappa = \frac{L}{\mu}, \gamma = \max(8\kappa, E)$, $\eta_t = \frac{2}{\mu(\gamma+t)}$ and then FedAvg satisfies

$$\mathbb{E}[F(\mathbf{w}_T)] - F^* \leq \frac{\kappa}{\gamma + T - 1}\left(\frac{2B}{\mu} + \frac{\mu\gamma}{2}\mathbb{E}\|\mathbf{w}_1 - \mathbf{w}^*\|^2\right), \quad (16)$$

$$\text{where } B = \sum_{k=1}^{N} p_k^2 \sigma_k^2 + 6L\Gamma + 8(E-1)^2 G^2. \quad (17)$$

Proof sketch of Theorem 1

1. Derive inequality from Lemma 1-3 using $\Delta_t = \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}^*\|^2$.
2. Prove inequality about Δ_t using **induction**.
3. Drive final result (Theorem 1 (29)).

Convergence Analysis of FedAvg

Proof sketch of Theorem 1

Lemma 1 Result of one step SGD

Under Assumption 1-2, $\eta_t \leq \frac{1}{4L}, \Gamma \geq 0$,

$$\begin{aligned}\mathbb{E}\|\bar{\mathbf{v}}_{t+1} - \mathbf{w}^*\|^2 \leq & (1 - \eta_t \mu)\mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}^*\|^2 + \eta_t^2 \mathbb{E}\|\mathbf{g}_t - \bar{\mathbf{g}}_t\|^2 \\ & + 6L\eta_t^2\Gamma + 2\mathbb{E}\sum_{k=1}^{N} p_k \|\bar{\mathbf{w}}_t - \mathbf{w}_k^t\|^2 \end{aligned} \tag{18}$$

Lemma 2 Bounding the variance

Under Assumption 3,

$$\mathbb{E}\|\mathbf{g}_t - \bar{\mathbf{g}}_t\|^2 \leq \sum_{k=1}^{N} p_k^2 \sigma_k^2 \tag{19}$$

Lemma 3 Bounding the divergence of $\{\mathbf{w}_t^k\}$

Under Assumption 4, η_t is non-increasing, $\eta_t \leq 2\eta_{t+E}, \forall t \geq 0$

$$\mathbb{E}\left[\sum_{k=1}^{N} p_k \|\bar{\mathbf{w}}_t - \mathbf{w}_k^t\|^2\right] \leq 4\eta_t^2 (E-1)^2 G^2 \tag{20}$$

Theorem 2: Convergence when partial participation (Scheme1)

Under Assumptions 1 to 5 and Scheme 1 (random sampling with replacement), define $\kappa, \gamma, \eta_t, B$ from Theorem 1, $C = \frac{4}{K}E^2G^2$ and then FedAvg satisfies

$$\mathbb{E}[F(\mathbf{w}_T)] - F^* \leq \frac{\kappa}{\gamma + T - 1}\left(\frac{2(B+C)}{\mu} + \frac{\mu\gamma}{2}\mathbb{E}\|\mathbf{w}_1 - \mathbf{w}^*\|^2\right). \tag{21}$$

Proof sketch of Theorem 2

1. Derive inequality from Lemma 1-5 using $\Delta_t = \mathbb{E}\|\bar{\mathbf{w}}_t - \mathbf{w}^*\|^2$.
2. Prove inequality about Δ_t using **induction**.
3. Drive final result (32).

Proof sketch of Theorem 2

Lemma 4 Unbiased sampling scheme

$$\mathbb{E}_{\mathcal{S}_t}[\bar{\mathbf{w}}_t] = \bar{\mathbf{v}}_{t+1}, \quad t+1 \in \mathcal{I}_E \tag{22}$$

Lemma 5 Bounding the variance of $\bar{\mathbf{w}}_t$

For $t+1 \in \mathcal{I}$, assume η_t is non-increasing and $\eta_t \leq 2\eta_{t+E}$ for all $t \geq 0$, the expected difference between $\bar{\mathbf{v}}_{t+1}$ and $\bar{\mathbf{w}}_{t+1}$ is bounded.

(i) Scheme 1,

$$\mathbb{E}_{\mathcal{S}_t}\left\|\bar{\mathbf{v}}_{t+1} - \bar{\mathbf{w}}_{t+1}\right\|^2 \leq \frac{4}{K}\eta_t^2 E^2 G^2 \tag{23}$$

(ii) Scheme 2, assume $p_1 = \cdots = p_N = \frac{1}{N}$

$$\mathbb{E}_{\mathcal{S}_t}\left\|\bar{\mathbf{v}}_{t+1} - \bar{\mathbf{w}}_{t+1}\right\|^2 \leq \frac{N-K}{N-1}\frac{4}{K}\eta_t^2 E^2 G^2 \tag{24}$$

Convergence rate

Under Assumption 2, the dominating term of (29):

$$\mathcal{O}\left(\frac{\sum_{k=1}^{N} p_k^2 \sigma_k^2 + L\Gamma + \left(1 + \frac{1}{K}\right) E^2 G^2 + \gamma G^2}{\mu T}\right) \tag{25}$$

Let T_ϵ as the number of required steps to achieve an ϵ accuracy.

$$\frac{T_\epsilon}{E} \propto \left(1 + \frac{1}{K}\right) E^2 G^2 + \frac{\sum_{k=1}^{N} p_k^2 \sigma_k^2 + L\Gamma + \kappa G^2}{E} + G^2 \tag{26}$$

Theorem 3: Convergence when partial participation (Scheme2)

Under Assumptions 1 to 4 & 6 and Scheme 2 (random sampling without replacement), define $\kappa, \gamma, \eta_t, B$ from Theorem 1, $C = \frac{N-K}{N-1}\frac{4}{K}E^2G^2$ and then FedAvg satisfies

$$\mathbb{E}[F(\mathbf{w}_T)] - F^* \leq \frac{\kappa}{\gamma + T - 1}\left(\frac{2(B+C)}{\mu} + \frac{\mu\gamma}{2}\mathbb{E}\|\mathbf{w}_1 - \mathbf{w}^*\|^2\right). \tag{27}$$

Convergence Analysis of FedAvg

Theorem 4

With full batch size, $E > 1$, any fixed (small) learning rate,

$$\|\bar{\mathbf{w}}^* - \mathbf{w}^*\|_2 = \Omega((E-1)\eta) \cdot \|\mathbf{w}^*\|_2. \tag{28}$$

Remarks

Fixed learning rate ($\eta_t = \eta$) can lead to sub-optimal point when $E > 1$.

Experiments [5]

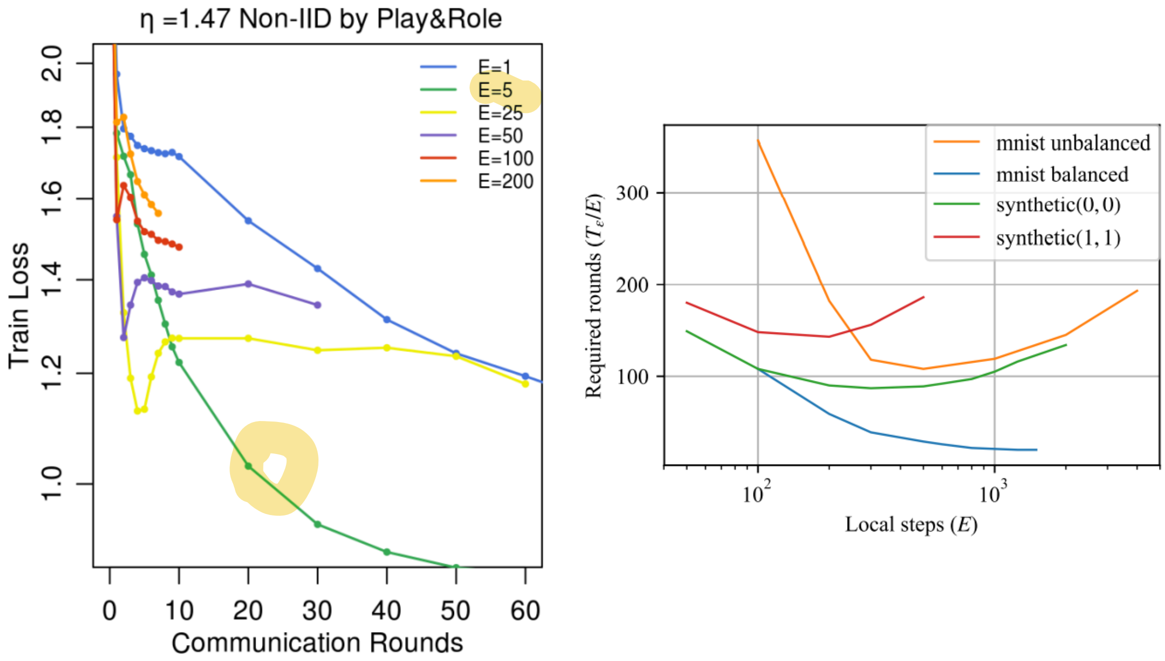


Figure 11: Required rounds to obtain an ϵ accuracy [2, 3]

Too large or small number of local updates E leads to slow convergence.

[5]See Appendix A (75)

Experiments

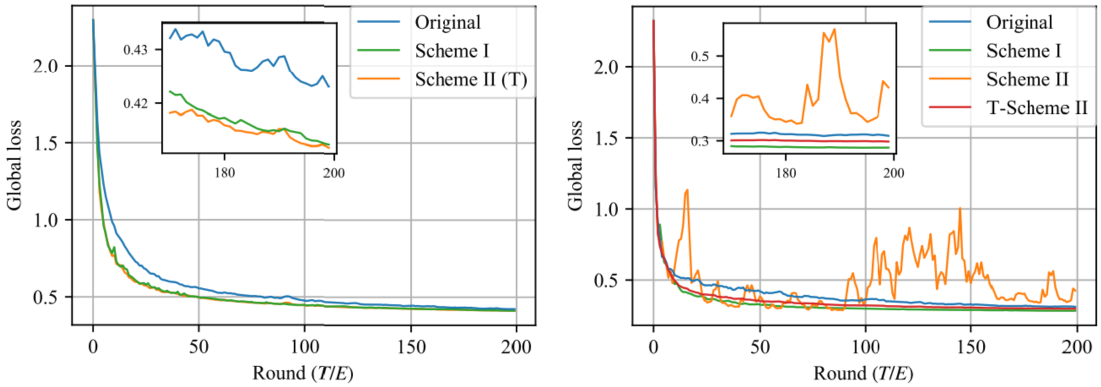


Figure 12: Comparisons of global loss over communication rounds

Compared to two different random sampline schemes (1: w/ replacement, 2: w/o replacement), sampling with replacement performs faster convergence.

Convergence Analysis of FedAvg

Take-aways

The performance of federated learning (FedAVG) depends on various hyperparameters such as:

- data heterogeneity Γ,
- number of local updates E,
- number of participation clients K,
- sampling scheme,
- dynamic learning rate.

2.2 FedOpt

ICLR, 2021

FedOpt [4]

Summary

- **Motivation**: FedAVG is unsuitable for settings with heavy-tail stochastic gradient noise distribution.
- **Challenges**: Client performing multiple local updates, data heterogeneity, communication costs.
- **Approach**: It applied adaptive server optimizer to FedAVG without enlarging convergence rate.
- **Contribution**: It showed that there's a relation between number of clients' updates and client heterogeneity.

FedOpt

Notation	Description
m	total number of clients
$F_i(x)$	loss function of i-th client
$\mathcal{D}_i$	data distribution of i-th client
σ_l^2	local variance
σ_g^2	global variance (client heterogeneity)
$\mathcal{S}$	set of selected clients
x_i^t	local model of i-th client at round t, $i \in \mathcal{S}$
η	learning rate
τ	degree of adaptivity
K	number of client updates taken per round

Table 4: FedOpt Notations

FedOpt

FedOpt Global model x

$$x_{t+1} = \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} x_i^t = x_t - \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}}(x_t - x_i^t)$$
$$= x_t + 1 \cdot \Delta_t$$

where $\Delta_i^t := x_i^t - x_t$ and $\Delta_t := \frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} \Delta_i^t$.

Client optimizer minimizes $F_i(x)$ based on each client's local data.
Server optimizer minimizes $f(x) = \frac{1}{m}\sum_{i=1}^{m} F_i(x)$.

Δ_t can be a **pseudo-gradient**.

(negative)

FedOpt

Algorithm 1 FEDOPT

Input: x_0, CLIENTOPT, SERVEROPT

for $t = 0, \cdots, T-1$ **do**

  Sample a subset $\mathcal{S}$ of clients

  $x_{i,0}^t = x_t$

  for each client $i \in \mathcal{S}$ **in parallel do**

    for $k = 0, \cdots, K-1$ **do**

      Compute an unbiased estimate $g_{i,k}^t$ of $\nabla F_i(x_{i,k}^t)$

      $x_{i,k+1}^t = \text{CLIENTOPT}(x_{i,k}^t, g_{i,k}^t, \eta_l, t)$

    $\Delta_i^t = x_{i,K}^t - x_t$

  $\Delta_t = \frac{1}{|\mathcal{S}|} \sum_{i \in \mathcal{S}} \Delta_i^t$

  $x_{t+1} = \text{SERVEROPT}(x_t, -\Delta_t, \eta, t)$

Figure 13: FedOpt (1)

adaptive optimization

FedOpt

Algorithm 2 **FedAdagrad**, **FedYogi**, and **FedAdam**

Initialization: $x_0, v_{-1} \geq \tau^2$, decay parameters $\beta_1, \beta_2 \in [0, 1)$
for $t = 0, \cdots, T-1$ **do**
    Sample subset $\mathcal{S}$ of clients
    $x_{i,0}^t = x_t$
    for each client $i \in \mathcal{S}$ **in parallel do**
        for $k = 0, \cdots, K-1$ **do**
            Compute an unbiased estimate $g_{i,k}^t$ of $\nabla F_i(x_{i,k}^t)$
            $x_{i,k+1}^t = x_{i,k}^t - \eta_l g_{i,k}^t$
        $\Delta_i^t = x_{i,K}^t - x_t$
    $\Delta_t = \frac{1}{|\mathcal{S}|} \sum_{i \in \mathcal{S}} \Delta_i^t$
    $m_t = \beta_1 m_{t-1} + (1 - \beta_1) \Delta_t$
    $v_t = v_{t-1} + \Delta_t^2$ (**FedAdagrad**)
    $v_t = v_{t-1} - (1 - \beta_2) \Delta_t^2 \operatorname{sign}(v_{t-1} - \Delta_t^2)$ (**FedYogi**)
    $v_t = \beta_2 v_{t-1} + (1 - \beta_2) \Delta_t^2$ (**FedAdam**)
    $x_{t+1} = x_t + \eta \frac{m_t}{\sqrt{v_t} + \tau}$

second moment

Figure 14: FedOpt (2)

Convergence Analysis of FedOpt

Under assumptions (48) and sufficiently large $T = G/L$,
$\sigma^2 = \sigma_l^2 + 6K\sigma_g^2$, $\eta_l \leq \min\left\{\frac{1}{16L}, \frac{1}{T^{1/6}}\left[\frac{\tau}{120L^2G}\right]^{1/3}\right\}$,
$\eta_l = \Theta(1/KL\sqrt{T})$, $\eta = \Theta(\sqrt{Km})$,

$$\min_{0\leq t\leq T-1} \mathbb{E}\|\nabla f(x_t)\|^2$$
$$= \mathcal{O}\left(\frac{f(x_0) - f(x^*)}{\sqrt{mKT}} + \frac{2\sigma_l^2 L}{G^2\sqrt{mKT}} + \frac{\sigma^2}{GKT} + \frac{\sigma^2 L\sqrt{m}}{G^2\sqrt{K}T^{3/2}}\right) \tag{29}$$

Convergence Analysis: Assumptions

- Lipschitz gradient of F_i (27)
- Bounded variance σ_l^2, σ_g^2 of F_i (27)
- Bounded gradients of f_i (27)

Remarks

- Convergence rate is almost same with FedAVG when $T \gg K$.
- Local learning rate η_l and its decay are $\frac{1}{\sqrt{T}}, \frac{1}{\sqrt{t}}$.
- Communication costs depend on T, which also depends on K.
- For selected sample clients and η properly, the effect of client heterogeneity σ_g can be reduced.

Experiments [6]

FedOpt (FedAdagrad, FedAdam, FedYogi) outperforms other FL algorithms.

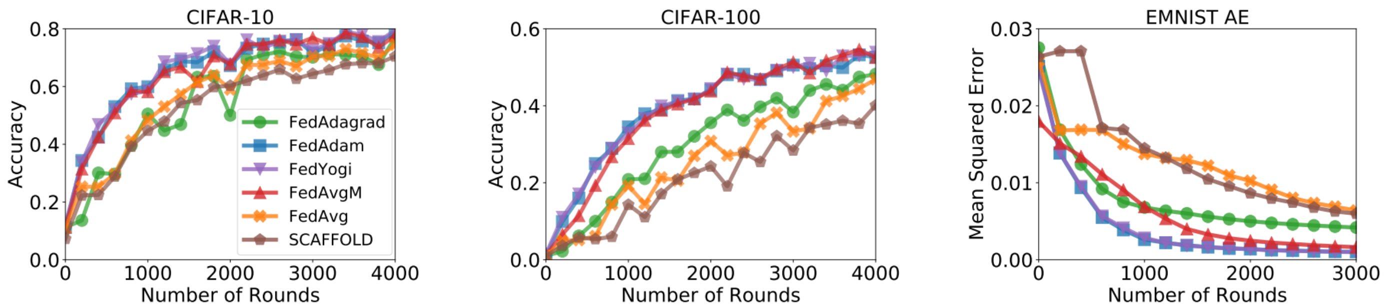


Figure 15: FedOpt results

[6] see Appendix B (76) for experimental settings.

3. Challenges

Existing Problems

- In practice, all clients cannot be participated in one communication stage because of the network bandwidth or system limitation [2].
- Although it's possible, training with too many clients in FL can negatively impact generalization and data-efficiency [5].
- For communication-efficient federated learning to achieve faster convergence, one possible way is focusing informative clients [6, 7, 8, 9].

3.1 On Large-Cohort Training

(Impact of number of participating clients)

NeurIPS, 2021

On Large-Cohort Training for FL [5]

Summary

- **Challenge**: Cohort size (number of participating clients at every communication) affects convergence improvements and generalization.
- **Contribution**: It showed the empirical findings about the cohort size.

Key findings

Increasing the cohort size may not lead to significant convergence improvements in practice.

Problem formulation

Objectives

Minimize a weighted average of client loss functions:

$$\min_{x} f(x) := \sum_{k=1}^{K} p_k f_k(x). \tag{30}$$

Notation	Description
K	total number of clients
p_k	weights of client k (number of local data)
f_k	loss function of client k

Table 5: Notations

Notation	Description
C	cohort of clients
M	cohort size (number of participating clients per round)
E	number of local epochs
x	server model
x_k	local model
Δ_k	client update ($\Delta_k := x_k - x$)
η_c, η_s	learning rate of client and server
g	gradient estimate
Δ	pseudo-gradient

Table 6: Notations

Experimental results[7]

Challenges 1) Catastrophic training failures (for large M)

Training accuracy decreased by a factor of at least 1/2 in a single round due to data heterogeneity.

80%

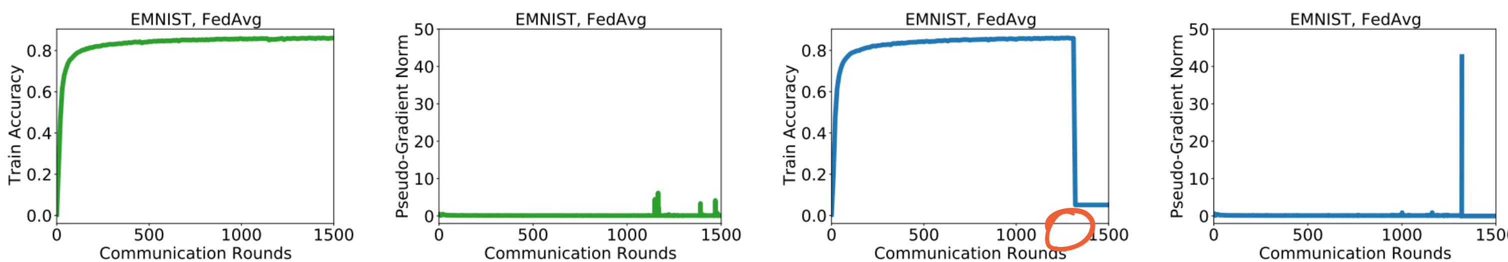


Figure 16: Catastrophic training failures (M=10)

[7] see Appendix B (76) for experiment settings

Experimental results

Challenges 2) Generalization failures

Large cohorts (large participation rate) lead to worse generalization in some datasets.

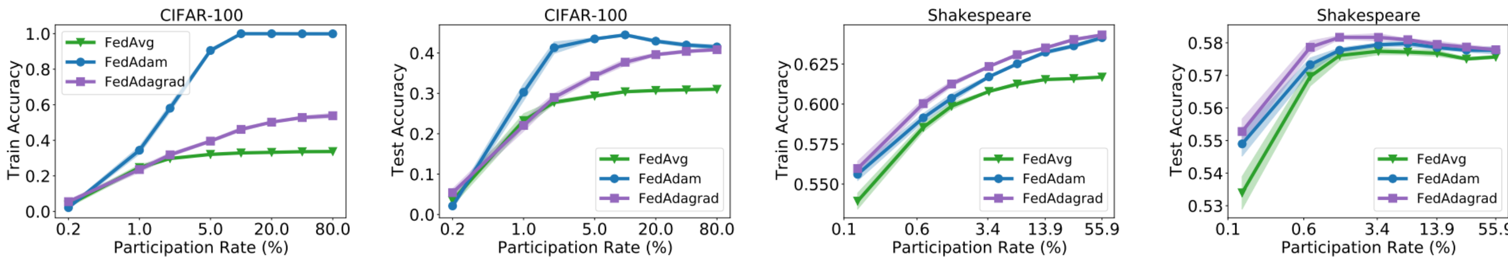


Figure 17: Generalization failures

Experimental results
Challenges diagnosis

Pseudo-gradient Δ is an average of nearly orthogonal vectors.

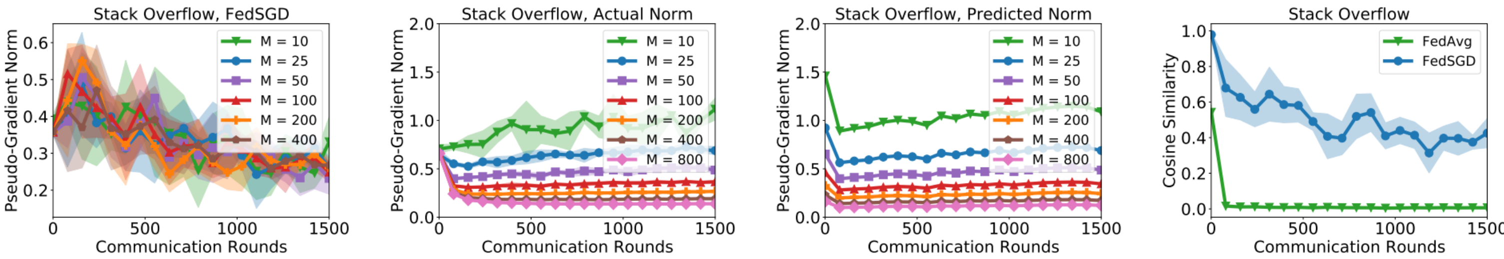


Figure 18: Diagnosing large-cohort challenges

On Large-Cohort Training for FL

Take-aways

- Large-cohort training for federated learning can negatively impact generalization and data-efficiency.
- Clarifying and breaking through impacts of cohort sizes is still open problem.

3.2 Client Selection

Client Selection problem

- In practice, all clients cannot be participated in one communication stage because of the network bandwidth or system limitation.
- However, only important clients might be helpful for training because of stragglers or outliers.

Figure 19: Full participation

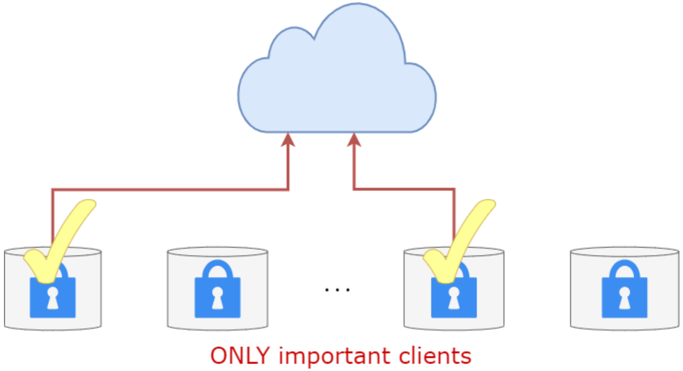


Figure 20: Partial participation

Client Selection problem

Problem Statement

Client selection problem aims to select some **informative clients** from all to show faster convergence at the earlier communication round to reduce communication cost of Federated Learning.

Client Selection problem

Related works

1. **Loss-based sampling** [6, 8]
   - ∘ selecting clients with high loss value.
   - \+ simple computation
   - − hyperparameter-sensitive, unexpected impacts of outliers
2. **Sample size-based sampling** [7]
   - ∘ selecting clients with large number of local samples.
   - \+ simple computation
   - − not robust on non-IID setting
3. **Similarity-based sampling** [9, 7]
   - ∘ selecting similar or diverse clients based on its gradient.
   - \+ less information redundancy of clients
   - − inefficient communication, heavy computation

DivFL [9]

Summary

- **Motivation**: For client selection, redundant client information is inefficient while selecting clients.
- **Approach**: For diversity, it finds the best subset to minimize the gap between gradient information of selected clients and the whole clients.

DivFL

Notation	Description
$F_k(\cdot)$	loss function of k-th client
∇F_k	gradient on local data of k-th client
N	total number of clients
K	(maximum) number of selected clients
V	the set of total clients ($\lvert V \rvert = N$)
S	the set of selected clients ($\lvert S \rvert \leq K$)
$\sigma(\cdot)$	selecting function ($V \to S$)
v^k	local model of k-th client ($v^k \in V$)
T	total number of communications
E	the number of local SGD updates
η	local learning rate
w_t, w^k	global model of t-th round, local model of k-th client

Table 7: Notations

DivFL

Objective

Difference between gradient information of selected clients and all clients:

$$\sum_{k\in[N]} \nabla F_k(v^k) = \sum_{k\in[N]} \left[\nabla F_k(v^k) - \nabla F_{\sigma(k)}(v^{\sigma(k)})\right] + \sum_{k\in S} \gamma_k \nabla F_k(v^k) \tag{31}$$

$$\therefore \sum_{k\in[N]} \nabla F_k(v^k) - \sum_{k\in S} \gamma_k \nabla F_k(v^k) = \sum_{k\in[N]} \left[\nabla F_k(v^k) - \nabla F_{\sigma(k)}(v^{\sigma(k)})\right] \tag{32}$$

Objective

To minimize the difference between gradient information of selected clients and the whole clients:

$$\begin{aligned}\|\sum_{k\in[N]} \nabla F_k(v^k) - \sum_{k\in S} \gamma_k \nabla F_k(v^k)\| \\ \leq \sum_{k\in[N]} \|\nabla F_k(v^k) - \nabla F_{\sigma(k)}(v^{\sigma(k)})\|\end{aligned} \tag{33}$$

$$\begin{aligned}\|\sum_{k\in[N]} \nabla F_k(v^k) - \sum_{k\in S} \gamma_k \nabla F_k(v^k)\| \\ \leq \sum_{k\in[N]} \min_{i\in S} \|\nabla F_k(v^k) - \nabla F_i(v^i)\| = G(S)\end{aligned} \tag{34}$$

Objective

To minimize the gap, they minimize the upper bound $G(S)$ of the approximation error ($=$ to maximize a constant its negation: $\bar{G}(S)$).

Diverse Client Selection

To find the best subset S,

$$\max_{S} \; [\bar{G}(S) = C - \sum_{k \in [N]} \min_{i \in S} \|\nabla F_k(v^k) - \nabla F_i(v^i)\|] \tag{35}$$

$$\text{where } \bar{G}(S) = C - G(S). \tag{36}$$

We call this $\bar{G}(\cdot)$ as **submodular function** [8].

[8] See Appendix D.1 (78) for details.

Greedy selection for Objective

$$S \leftarrow S \cup k^*, k^* \in \arg\max_{k \in V \setminus S} [\bar{G}(S) - \bar{G}(\{k\} \cup S)] \qquad (37)$$

where a accelerated greedy algorithm (stochastic-greedy [9]) was used.

[9] STOCHASTIC-GREEDY algorithm [10] is a linear-time algorithm for maximizing a non-negative monotone submodular function subject to a cardinality constraint k. See Appendix D.2 (79) for details.

Algorithm 1 `DivFL`

Input: T, E, η, w_0

for $t = 0, \cdots, T-1$ **do**

  Server selects a subset of K active clients S_t using the stochastic greedy algorithm in Eq. (6), and sends w_t to them.

  for *device* $k \in S_t$ *in parallel* **do**

    $w^k \leftarrow w_t$

    Solve the local sub-problem of client-k inexactly by updating w^k for E local mini-batch SGD steps:

$$w^k = w^k - \eta \nabla F_k(w^k)$$

    Send $\Delta_t^k := w_t^k - w_t$ back to Server

  end

  Server aggregates $\{\Delta_t^k\}$:

$$w_{t+1} \leftarrow w_t + \frac{1}{|S_t|} \sum_{k \in S_t} \Delta_t^k$$

end

return w_T

Experiments [10]

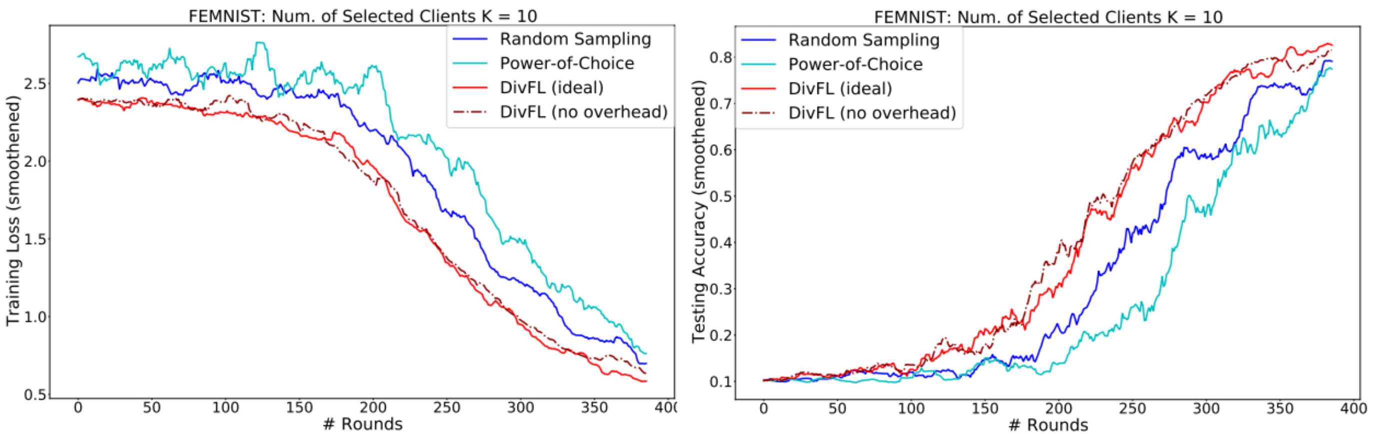


Figure 21: Performance over communication rounds on FedEMNIST

[10] See Appendix C (77) for experimental setting.

Experiments

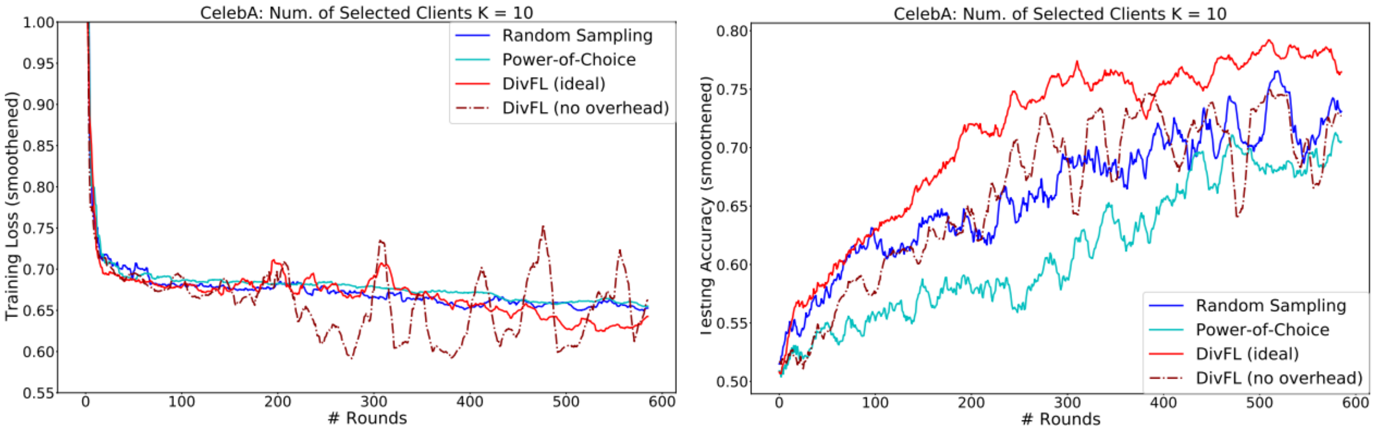


Figure 22: Performance over communication rounds on CelebA dataset

4. Conclusion

Conclusion

Summary

Federated Learning is privacy-preserving machine learning in distributed setting

The performance of federated learning depends on ...

- number of local updates
- local batch size
- data heterogeneity
- total communication rounds
- number of participating clients per round

Bibliography

[1] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.

[2] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[3] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. *arXiv preprint arXiv:1907.02189*, 2019.

[4] Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and H Brendan McMahan. Adaptive federated optimization. *arXiv preprint arXiv:2003.00295*, 2020.

[5] Zachary Charles, Zachary Garrett, Zhouyuan Huo, Sergei Shmulyian, and Virginia Smith. On large-cohort training for federated learning. *arXiv preprint arXiv:2106.07820*, 2021.

[6] Jack Goetz, Kshitiz Malik, Duc Bui, Seungwhan Moon, Honglei Liu, and Anuj Kumar. Active federated learning. *arXiv preprint arXiv:1909.12641*, 2019.

[7] Yann Fraboni, Richard Vidal, Laetitia Kameni, and Marco Lorenzi. Clustered sampling: Low-variance and improved representativity for clients selection in federated learning. In *International Conference on Machine Learning*, pages 3407–3416. PMLR, 2021.

[8] Yae Jee Cho, Jianyu Wang, and Gauri Joshi. Client selection in federated learning: Convergence analysis and power-of-choice selection strategies. *arXiv preprint arXiv:2010.01243*, 2020.

[9] Ravikumar Balakrishnan, Tian Li, Tianyi Zhou, Nageen Himayat, Virginia Smith, and Jeff Bilmes. Diverse client selection for federated learning via submodular maximization. In *International Conference on Learning Representations*, 2021.

[10] Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, Amin Karbasi, Jan Vondrák, and Andreas Krause. Lazier than lazy greedy. *CoRR*, abs/1409.7938, 2014.

[11] Sebastian Caldas, Peter Wu, Tian Li, Jakub Konečný, H. Brendan McMahan, Virginia Smith, and Ameet Talwalkar. LEAF: A benchmark for federated settings. *CoRR*, abs/1812.01097, 2018.

Q & A

Appendix A

Experimental setting of FedAVG

MNIST

- Task: image classification
- Model: **MLP**(with 2-hidden layers with 200 units each using ReLu activations), **CNN**(with two 5x5 convolution layers (the first with 32 channels, the second with 64, each followed with 2x2 maxpooling),a fully connected layer with 512 units and ReLu activation, and a final softmax output layer(1,663,370 total parameters)
- Partition: **IID**(balanced), **Non-IID**(by dividing the data it into 200 shards of size 300, and assign each of 100 clients 2 shards, then most clients only have examples of two digits.)

Appendix B

Experimental setting of FedOpt, LargeCohort

Dataset	Clients (train/test)	Examples (train/test)	Model
CIFAR100	500/100	50,000/10,000	ResNet-18 w. GN
FedEMNIST	3,400/3,400	671,585/77,483	2-CNN w. dropout, max-pooling, 2 fc layers
ShakeSpeare	715/715	16,068/2,356	2-LSTM
Stack Overflow	342,477/204,088	135,818,730 /16,586,035	1-LSTM

Hyperparameters	Values	Hyperparameters	Values
E	1	η_c, η_s	$\{10^i \| -3 \leq i \leq 1\}$
M	50	B	20, 20, 4, 32
T	1,500		

Appendix C

Experimental setting of DivFL

FedEMNIST dataset

- Total 500 clients where each client contains 3 out of 10 lowercase handwritten characters.
- Task: image classification with 62 classes.
- Model: CNN with two 5x5-convolutional and 2x2-maxpooling (with a stride of 2) layers followed by a dense layer with 128 activations.

CelebA dataset

- Total 515 clients (Leaf [11] base).
- Task: image binary classification (whether it's smiling or not).
- Model: CNN with 4 3x3-convolutional and 2x2-maxpooling layers followed by a dense layer.

Appendix D.1

submodular function

A function $f: 2^V \to \mathbb{R}$ assigns a subset $A \subseteq V$ a utility value $f(A)$,

$$f(A \cup \{i\}) - f(A) \geq f(B \cup \{i\}) - f(B) \tag{38}$$

for any $A \subseteq B \subseteq V$ and $i \in V \backslash B$.

We can regard $f(A \cup \{i\}) - f(A)$ as the marginal gain of adding a new element i to A.

STOCHASTIC-GREEDY algorithm [10]

Algorithm 1 STOCHASTIC-GREEDY

Input: $f : 2^V \to \mathbb{R}_+, k \in \{1, \ldots, n\}$.
Output: A set $A \subseteq V$ satisfying $|A| \leq k$.

1: $A \leftarrow \emptyset$.
2: **for** $(i \leftarrow 1;\ i \leq k;\ i \leftarrow i + 1)$ **do**
3: $\quad R \leftarrow$ a random subset obtained by sampling s random elements from $V \setminus A$.
4: $\quad a_i \leftarrow \text{argmax}_{a \in R} \Delta(a|A)$.
5: $\quad A \leftarrow A \cup \{a_i\}$
6: **return** A.