Improved Generalization Bounds for Communication Efficient Federated Learning

Federated Learning with Adaptive Local Steps (FedALS)

Peyman Gholami, Hulya Seferoglu

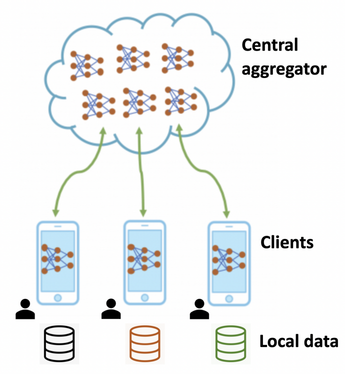
University of Illinois Chicago

July 2024

Table of Contents

Introduction

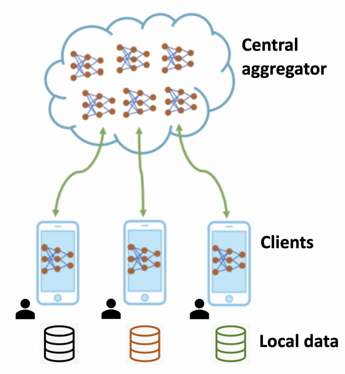
Federated learning



Introduction

Federated learning

- Communication cost: Exchanging models is costly, especially for large models in today's machine learning applications like LLMs.



Introduction

Federated learning

- Communication cost: Exchanging models is costly, especially for large models in today's machine learning applications like LLMs.
- Possible solutions:
  - **Local SGD**
  - Mini-batch SGD

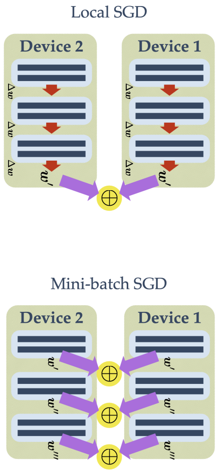


Figure: From Lin, Tao, Sebastian U. Stich and Martin Jaggi. "Don't Use Large Mini-Batches, Use Local SGD."

Federated learning

Purpose of communication:

Federated learning

Purpose of communication:

- Reducing the consensus distance among clients.
- Consensus distance at t: $\frac{1}{K}\sum_{k=1}^{K}\|\hat{\boldsymbol{\theta}}_t - \boldsymbol{\theta}_{k,t}\|^2$, where $\hat{\boldsymbol{\theta}}_t = \frac{1}{K}\sum_{k=1}^{K}\boldsymbol{\theta}_{k,t}$.

Federated learning

Purpose of communication:

- Reducing the consensus distance among clients.
- Consensus distance at t: $\frac{1}{K}\sum_{k=1}^{K}\|\hat{\boldsymbol{\theta}}_t - \boldsymbol{\theta}_{k,t}\|^2$, where $\hat{\boldsymbol{\theta}}_t = \frac{1}{K}\sum_{k=1}^{K}\boldsymbol{\theta}_{k,t}$.
- Helps maintain the overall optimization process on a trajectory toward the global optimum.

Consensus distance

Example:

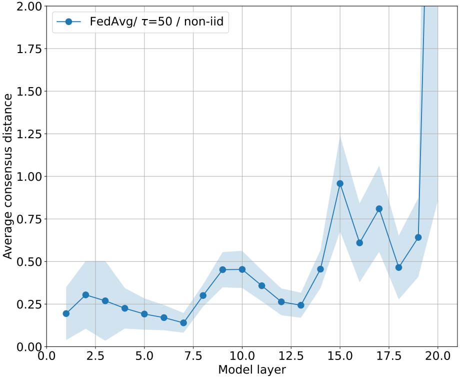
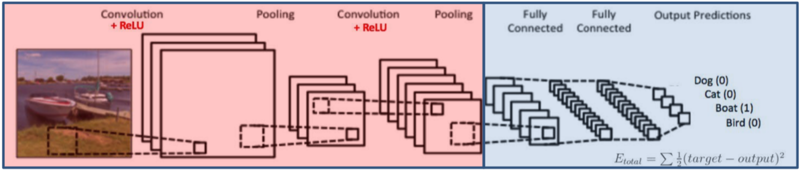


Figure: ResNet-20 on CIFAR-10 with 5 clients with non-iid data distribution over clients (2 classes per client). The early layers responsible for extracting representations exhibit lower levels of consensus distance.

Representation learning



Learning the representation	**Learning classification**
Feature extraction	**Fully connected**
“Backbone” (transferable)	**Head**
Higer Complexity	**Lower Complexity**

Motivation

- The above example indicates that initial layers show higher similarity, so they can be aggregated less frequently.

[1] Sashank J. Reddi et al. "Adaptive Federated Optimization". In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=LkFG3lB13U5; Tao Yu, Eugene Bagdasaryan, and Vitaly Shmatikov. "Salvaging Federated Learning by Local Adaptation". In: *ArXiv* abs/2002.04758 (2020). URL: https://api.semanticscholar.org/CorpusID:211082601.

Motivation

- The above example indicates that initial layers show higher similarity, so they can be aggregated less frequently.
- several empirical studies[1] show that federated learning with multiple local updates per round learns a generalizable representation and is unexpectedly successful in non-iid federated learning.

[1] Sashank J. Reddi et al. "Adaptive Federated Optimization". In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=LkFG3lB13U5; Tao Yu, Eugene Bagdasaryan, and Vitaly Shmatikov. "Salvaging Federated Learning by Local Adaptation". In: *ArXiv* abs/2002.04758 (2020). URL: https://api.semanticscholar.org/CorpusID:211082601.

Motivation

- The above example indicates that initial layers show higher similarity, so they can be aggregated less frequently.
- several empirical studies[1] show that federated learning with multiple local updates per round learns a generalizable representation and is unexpectedly successful in non-iid federated learning.

Motivate us to investigate how local updates affect the generalization of the model.

[1] Sashank J. Reddi et al. "Adaptive Federated Optimization". In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=LkFG3lB13U5; Tao Yu, Eugene Bagdasaryan, and Vitaly Shmatikov. "Salvaging Federated Learning by Local Adaptation". In: *ArXiv* abs/2002.04758 (2020). URL: https://api.semanticscholar.org/CorpusID:211082601.

Problem Statement

1. K clients/nodes.

Problem Statement

1. K clients/nodes.
2. Client k has a local dataset $\boldsymbol{S}_k = \{\boldsymbol{z}_{k,1}, \ldots, \boldsymbol{z}_{k,n_k}\}$, where $\boldsymbol{z}_{k,i} = (\boldsymbol{x}_{k,i}, \boldsymbol{y}_{k,i})$ is drawn from a distribution $\mathcal{D}_k$ over $\mathcal{X} \times \mathcal{Y}$.

Problem Statement

1. K clients/nodes.
2. Client k has a local dataset $\boldsymbol{S}_k = \{\boldsymbol{z}_{k,1}, ..., \boldsymbol{z}_{k,n_k}\}$, where $\boldsymbol{z}_{k,i} = (\boldsymbol{x}_{k,i}, \boldsymbol{y}_{k,i})$ is drawn from a distribution $\mathcal{D}_k$ over $\mathcal{X} \times \mathcal{Y}$.
3. Dataset across all nodes is defined as $\boldsymbol{S} = \{\boldsymbol{S}_1, ..., \boldsymbol{S}_K\}$.

Problem Statement

1. K clients/nodes.
2. Client k has a local dataset $\boldsymbol{S}_k = \{\boldsymbol{z}_{k,1}, \ldots, \boldsymbol{z}_{k,n_k}\}$, where $\boldsymbol{z}_{k,i} = (\boldsymbol{x}_{k,i}, \boldsymbol{y}_{k,i})$ is drawn from a distribution $\mathcal{D}_k$ over $\mathcal{X} \times \mathcal{Y}$.
3. Dataset across all nodes is defined as $\boldsymbol{S} = \{\boldsymbol{S}_1, \ldots, \boldsymbol{S}_K\}$.
4. $M_{\boldsymbol{\theta}} = \mathcal{A}(\boldsymbol{S})$ is the output of a possibly stochastic function denoted as $\mathcal{A}(\boldsymbol{S})$, where $M_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$ is the learned model parameterized by $\boldsymbol{\theta}$.

Problem Statement

1. K clients/nodes.
2. Client k has a local dataset $\boldsymbol{S}_k = \{\boldsymbol{z}_{k,1}, ..., \boldsymbol{z}_{k,n_k}\}$, where $\boldsymbol{z}_{k,i} = (\boldsymbol{x}_{k,i}, \boldsymbol{y}_{k,i})$ is drawn from a distribution $\mathcal{D}_k$ over $\mathcal{X} \times \mathcal{Y}$.
3. Dataset across all nodes is defined as $\boldsymbol{S} = \{\boldsymbol{S}_1, ..., \boldsymbol{S}_K\}$.
4. $M_{\boldsymbol{\theta}} = \mathcal{A}(\boldsymbol{S})$ is the output of a possibly stochastic function denoted as $\mathcal{A}(\boldsymbol{S})$, where $M_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$ is the learned model parameterized by $\boldsymbol{\theta}$.
5. Empirical risk on dataset $\boldsymbol{S}$:
$R_{\boldsymbol{S}}(M_{\boldsymbol{\theta}}) = \mathbb{E}_{k \sim \mathcal{K}} R_{\boldsymbol{S}_k}(M_{\boldsymbol{\theta}}) = \mathbb{E}_{k \sim \mathcal{K}} \frac{1}{n_k} \sum_{i=1}^{n_k} l(M_{\boldsymbol{\theta}}, \boldsymbol{z}_{k,i})$

Problem Statement

1. K clients/nodes.
2. Client k has a local dataset $\boldsymbol{S}_k = \{\boldsymbol{z}_{k,1}, ..., \boldsymbol{z}_{k,n_k}\}$, where $\boldsymbol{z}_{k,i} = (\boldsymbol{x}_{k,i}, \boldsymbol{y}_{k,i})$ is drawn from a distribution $\mathcal{D}_k$ over $\mathcal{X} \times \mathcal{Y}$.
3. Dataset across all nodes is defined as $\boldsymbol{S} = \{\boldsymbol{S}_1, ..., \boldsymbol{S}_K\}$.
4. $M_{\boldsymbol{\theta}} = \mathcal{A}(\boldsymbol{S})$ is the output of a possibly stochastic function denoted as $\mathcal{A}(\boldsymbol{S})$, where $M_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$ is the learned model parameterized by $\boldsymbol{\theta}$.
5. Empirical risk on dataset $\boldsymbol{S}$: $R_{\boldsymbol{S}}(M_{\boldsymbol{\theta}}) = \mathbb{E}_{k\sim\mathcal{K}} R_{\boldsymbol{S}_k}(M_{\boldsymbol{\theta}}) = \mathbb{E}_{k\sim\mathcal{K}} \frac{1}{n_k} \sum_{i=1}^{n_k} l(M_{\boldsymbol{\theta}}, \boldsymbol{z}_{k,i})$
6. Population risk: $R(M_{\boldsymbol{\theta}}) = \mathbb{E}_{k\sim\mathcal{K}}\, R_k(M_{\boldsymbol{\theta}}) = \mathbb{E}_{k\sim\mathcal{K}, \boldsymbol{z}\sim\mathcal{D}_k}\, l(M_{\boldsymbol{\theta}}, \boldsymbol{z})$

Problem Statement

1. K clients/nodes.
2. Client k has a local dataset $\boldsymbol{S}_k = \{\boldsymbol{z}_{k,1}, \dots, \boldsymbol{z}_{k,n_k}\}$, where $\boldsymbol{z}_{k,i} = (\boldsymbol{x}_{k,i}, \boldsymbol{y}_{k,i})$ is drawn from a distribution $\mathcal{D}_k$ over $\mathcal{X} \times \mathcal{Y}$.
3. Dataset across all nodes is defined as $\boldsymbol{S} = \{\boldsymbol{S}_1, \dots, \boldsymbol{S}_K\}$.
4. $M_{\boldsymbol{\theta}} = \mathcal{A}(\boldsymbol{S})$ is the output of a possibly stochastic function denoted as $\mathcal{A}(\boldsymbol{S})$, where $M_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$ is the learned model parameterized by $\boldsymbol{\theta}$.
5. Empirical risk on dataset $\boldsymbol{S}$:
   $R_{\boldsymbol{S}}(M_{\boldsymbol{\theta}}) = \mathbb{E}_{k\sim\mathcal{K}} R_{\boldsymbol{S}_k}(M_{\boldsymbol{\theta}}) = \mathbb{E}_{k\sim\mathcal{K}} \frac{1}{n_k} \sum_{i=1}^{n_k} l(M_{\boldsymbol{\theta}}, \boldsymbol{z}_{k,i})$
6. Population risk: $R(M_{\boldsymbol{\theta}}) = \mathbb{E}_{k\sim\mathcal{K}} R_k(M_{\boldsymbol{\theta}}) = \mathbb{E}_{k\sim\mathcal{K}, \boldsymbol{z}\sim\mathcal{D}_k} l(M_{\boldsymbol{\theta}}, \boldsymbol{z})$
7. Generalization error for dataset $\boldsymbol{S}$ and function $\mathcal{A}(\boldsymbol{S})$:
   $\Delta_{\mathcal{A}}(\boldsymbol{S}) = R(\mathcal{A}(\boldsymbol{S})) - R_{\boldsymbol{S}}(\mathcal{A}(\boldsymbol{S}))$

Problem Statement

1. K clients/nodes.
2. Client k has a local dataset $\boldsymbol{S}_k = \{\boldsymbol{z}_{k,1}, \ldots, \boldsymbol{z}_{k,n_k}\}$, where $\boldsymbol{z}_{k,i} = (\boldsymbol{x}_{k,i}, \boldsymbol{y}_{k,i})$ is drawn from a distribution $\mathcal{D}_k$ over $\mathcal{X} \times \mathcal{Y}$.
3. Dataset across all nodes is defined as $\boldsymbol{S} = \{\boldsymbol{S}_1, \ldots, \boldsymbol{S}_K\}$.
4. $M_{\boldsymbol{\theta}} = \mathcal{A}(\boldsymbol{S})$ is the output of a possibly stochastic function denoted as $\mathcal{A}(\boldsymbol{S})$, where $M_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$ is the learned model parameterized by $\boldsymbol{\theta}$.
5. Empirical risk on dataset $\boldsymbol{S}$: $R_{\boldsymbol{S}}(M_{\boldsymbol{\theta}}) = \mathbb{E}_{k\sim\mathcal{K}} R_{\boldsymbol{S}_k}(M_{\boldsymbol{\theta}}) = \mathbb{E}_{k\sim\mathcal{K}} \frac{1}{n_k} \sum_{i=1}^{n_k} l(M_{\boldsymbol{\theta}}, \boldsymbol{z}_{k,i})$
6. Population risk: $R(M_{\boldsymbol{\theta}}) = \mathbb{E}_{k\sim\mathcal{K}} R_k(M_{\boldsymbol{\theta}}) = \mathbb{E}_{k\sim\mathcal{K}, \boldsymbol{z}\sim\mathcal{D}_k} l(M_{\boldsymbol{\theta}}, \boldsymbol{z})$
7. Generalization error for dataset $\boldsymbol{S}$ and function $\mathcal{A}(\boldsymbol{S})$: $\Delta_{\mathcal{A}}(\boldsymbol{S}) = R(\mathcal{A}(\boldsymbol{S})) - R_{\boldsymbol{S}}(\mathcal{A}(\boldsymbol{S}))$
8. Expected generalization: $\mathbb{E}_{\boldsymbol{S}} \Delta_{\mathcal{A}}(\boldsymbol{S}) = \mathbb{E}_{\{\boldsymbol{S}_k \sim \mathcal{D}_k^{n_k}\}_{k=1}^K} \Delta_{\mathcal{A}}(\boldsymbol{S})$

One-Round Generalization Bound

Theorem

Let $l(M_{\boldsymbol{\theta}}, \boldsymbol{z})$ be μ-strongly convex and L-smooth in $M_{\boldsymbol{\theta}}$, $M_{\boldsymbol{\theta}_k} = \mathcal{A}_k(\boldsymbol{S}_k)$ represents the model obtained from Empirical Risk Minimization (ERM) algorithm on local dataset $\boldsymbol{S}_k$, i.e., $M_{\boldsymbol{\theta}_k} = \arg\min_M \sum_{i=1}^{n_k} l(M, \boldsymbol{z}_{k,i})$, and $M_{\hat{\boldsymbol{\theta}}} = \mathcal{A}(\boldsymbol{S})$ is the model after one round of FedAvg ($\hat{\boldsymbol{\theta}} = \mathbb{E}_{k\sim\mathcal{K}}\, \boldsymbol{\theta}_k$). Then, the expected generalization error is

$$\mathbb{E}_{\boldsymbol{S}}\Delta_{\mathcal{A}}(\boldsymbol{S}) \le \mathbb{E}_{k\sim\mathcal{K}}\Bigg[\frac{L\mathcal{K}(k)^2}{\mu} \underbrace{\mathbb{E}_{\boldsymbol{S}_k}\Delta_{\mathcal{A}_k}(\boldsymbol{S}_k)}_{\text{Expected local generalization}} + 2\sqrt{\frac{L}{\mu}}\mathcal{K}(k)\Big(\underbrace{\mathbb{E}_{\boldsymbol{S}}\,\delta_{k,\mathcal{A}}(\boldsymbol{S})}_{\text{Expected non-iidness}} \underbrace{\mathbb{E}_{\boldsymbol{S}_k}\Delta_{\mathcal{A}_k}(\boldsymbol{S}_k)}_{\text{Expected local generalization}}\Big)^{\frac{1}{2}}\Bigg], \tag{1}$$

where $\delta_{k,\mathcal{A}}(\boldsymbol{S}) = R_{\boldsymbol{S}_k}(\mathcal{A}(\boldsymbol{S})) - R_{\boldsymbol{S}_k}(\mathcal{A}_k(\boldsymbol{S}_k))$ indicates the level of non-iidness at client k for function $\mathcal{A}$ on dataset $\boldsymbol{S}$.

One-Round Generalization Bound

$$\mathbb{E}_{\boldsymbol{S}} \Delta_{\mathcal{A}}(\boldsymbol{S}) \leq \mathbb{E}_{k\sim\mathcal{K}} \Bigg[\frac{L\mathcal{K}(k)^2}{\mu} \underbrace{\mathbb{E}_{\boldsymbol{S}_k} \Delta_{\mathcal{A}_k}(\boldsymbol{S}_k)}_{\text{Expected local generalization}} + 2\sqrt{\frac{L}{\mu}}\mathcal{K}(k)\Big(\underbrace{\mathbb{E}_{\boldsymbol{S}}\, \delta_{k,\mathcal{A}}(\boldsymbol{S})}_{\text{Expected non-iidness}} \underbrace{\mathbb{E}_{\boldsymbol{S}_k} \Delta_{\mathcal{A}_k}(\boldsymbol{S}_k)}_{\text{Expected local generalization}} \Big)^{\frac{1}{2}} \Bigg],$$

- iid: $\mathcal{K}(k)^2$ (enhancement compared to the state of the art[2])

[2] Leighton Pate Barnes, Alex Dytso, and H. Vincent Poor. "Improved Information-Theoretic Generalization Bounds for Distributed, Federated, and Iterative Learning". In: *Entropy* 24 (2022). URL: https://api.semanticscholar.org/CorpusID:246634528.

One-Round Generalization Bound

$$\mathbb{E}_{\boldsymbol{S}} \Delta_{\mathcal{A}}(\boldsymbol{S}) \leq \mathbb{E}_{k\sim\mathcal{K}} \Bigg[\frac{L\mathcal{K}(k)^2}{\mu} \underbrace{\mathbb{E}_{\boldsymbol{S}_k} \Delta_{\mathcal{A}_k}(\boldsymbol{S}_k)}_{\text{Expected local generalization}} + 2\sqrt{\frac{L}{\mu}}\mathcal{K}(k) \Bigg(\underbrace{\mathbb{E}_{\boldsymbol{S}}\, \delta_{k,\mathcal{A}}(\boldsymbol{S})}_{\text{Expected non-iidness}} \underbrace{\mathbb{E}_{\boldsymbol{S}_k} \Delta_{\mathcal{A}_k}(\boldsymbol{S}_k)}_{\text{Expected local generalization}} \Bigg)^{\frac{1}{2}} \Bigg],$$

- iid: $\mathcal{K}(k)^2$ (enhancement compared to the state of the art[2])
- noniid: the expected generalization error bound does not necessarily decrease with averaging.

[2] Leighton Pate Barnes, Alex Dytso, and H. Vincent Poor. "Improved Information-Theoretic Generalization Bounds for Distributed, Federated, and Iterative Learning". In: *Entropy* 24 (2022). URL: https://api.semanticscholar.org/CorpusID:246634528.

One-Round Generalization Bound

$$\mathbb{E}_{\boldsymbol{S}} \Delta_{\mathcal{A}}(\boldsymbol{S}) \leq \mathbb{E}_{k\sim\mathcal{K}} \left[\frac{L\mathcal{K}(k)^2}{\mu} \underbrace{\mathbb{E}_{\boldsymbol{S}_k} \Delta_{\mathcal{A}_k}(\boldsymbol{S}_k)}_{\text{Expected local generalization}} + 2\sqrt{\frac{L}{\mu}}\mathcal{K}(k) \left(\underbrace{\mathbb{E}_{\boldsymbol{S}}\, \delta_{k,\mathcal{A}}(\boldsymbol{S})}_{\text{Expected non-iidness}} \underbrace{\mathbb{E}_{\boldsymbol{S}_k} \Delta_{\mathcal{A}_k}(\boldsymbol{S}_k)}_{\text{Expected local generalization}} \right)^{\frac{1}{2}} \right],$$

- iid: $\mathcal{K}(k)^2$ (enhancement compared to the state of the art[2])
- noniid: the expected generalization error bound does not necessarily decrease with averaging.

FedAvg works well in iid setup.

[2] Leighton Pate Barnes, Alex Dytso, and H. Vincent Poor. "Improved Information-Theoretic Generalization Bounds for Distributed, Federated, and Iterative Learning". In: *Entropy* 24 (2022). URL: https://api.semanticscholar.org/CorpusID:246634528.

One-Round Generalization Bound

$$\mathbb{E}_{\boldsymbol{S}}\Delta_{\mathcal{A}}(\boldsymbol{S}) \leq \mathbb{E}_{k\sim\mathcal{K}}\Bigg[\frac{L\mathcal{K}(k)^2}{\mu}\underbrace{\mathbb{E}_{\boldsymbol{S}_k}\Delta_{\mathcal{A}_k}(\boldsymbol{S}_k)}_{\text{Expected local generalization}} + 2\sqrt{\frac{L}{\mu}}\mathcal{K}(k)\Big(\underbrace{\mathbb{E}_{\boldsymbol{S}}\,\delta_{k,\mathcal{A}}(\boldsymbol{S})}_{\text{Expected non-iidness}}\underbrace{\mathbb{E}_{\boldsymbol{S}_k}\Delta_{\mathcal{A}_k}(\boldsymbol{S}_k)}_{\text{Expected local generalization}}\Big)^{\frac{1}{2}}\Bigg],$$

Partial client participation:

One-Round Generalization Bound

$$\mathbb{E}_{\boldsymbol{S}} \Delta_{\mathcal{A}}(\boldsymbol{S}) \leq \mathbb{E}_{k\sim\mathcal{K}} \Bigg[\frac{L\mathcal{K}(k)^2}{\mu} \underbrace{\mathbb{E}_{\boldsymbol{S}_k} \Delta_{\mathcal{A}_k}(\boldsymbol{S}_k)}_{\text{Expected local generalization}} + 2\sqrt{\frac{L}{\mu}}\mathcal{K}(k)\Bigg(\underbrace{\mathbb{E}_{\boldsymbol{S}}\, \delta_{k,\mathcal{A}}(\boldsymbol{S})}_{\text{Expected non-iidness}} \underbrace{\mathbb{E}_{\boldsymbol{S}_k} \Delta_{\mathcal{A}_k}(\boldsymbol{S}_k)}_{\text{Expected local generalization}} \Bigg)^{\frac{1}{2}} \Bigg],$$

Partial client participation:

Case I: Sampling $\hat{K}$ clients with replacement based on distribution $\mathcal{K}$, followed by averaging the local models with equal weights. $(\mathcal{K}(k) \rightarrow \frac{1}{\hat{K}})$

One-Round Generalization Bound

$$\mathbb{E}_{\boldsymbol{S}}\Delta_{\mathcal{A}}(\boldsymbol{S}) \leq \mathbb{E}_{k\sim\mathcal{K}}\Bigg[\frac{L\mathcal{K}(k)^2}{\mu} \underbrace{\mathbb{E}_{\boldsymbol{S}_k}\Delta_{\mathcal{A}_k}(\boldsymbol{S}_k)}_{\text{Expected local generalization}}$$

$$+2\sqrt{\frac{L}{\mu}}\mathcal{K}(k)\Bigg(\underbrace{\mathbb{E}_{\boldsymbol{S}}\,\delta_{k,\mathcal{A}}(\boldsymbol{S})}_{\text{Expected non-iidness}}\underbrace{\mathbb{E}_{\boldsymbol{S}_k}\Delta_{\mathcal{A}_k}(\boldsymbol{S}_k)}_{\text{Expected local generalization}}\Bigg)^{\frac{1}{2}}\Bigg],$$

Partial client participation:

Case I: Sampling $\hat{K}$ clients with replacement based on distribution $\mathcal{K}$, followed by averaging the local models with equal weights. $(\mathcal{K}(k) \to \frac{1}{\hat{K}})$

Case II: Sampling $\hat{K}$ clients without replacement uniformly at random, then performing weighted averaging of local models. Here, the weight of client k is rescaled to $\frac{\mathcal{K}(k)K}{\hat{K}}$. $(\mathcal{K}(k) \to \frac{\mathcal{K}(k)K}{\hat{K}})$

$R-$Round Generalization Bound

- $Z_{k,r} = \bigcup \{\mathcal{B}_{k,r,t}\}_{t=0}^{\tau-1}$, where $\mathcal{B}_{k,r,t}$ is the batch of samples used in local step t of round r in node k. τ is the duration of one reound.

$R-$Round Generalization Bound

- $Z_{k,r} = \bigcup \{\mathcal{B}_{k,r,t}\}_{t=0}^{\tau-1}$, where $\mathcal{B}_{k,r,t}$ is the batch of samples used in local step t of round r in node k. τ is the duration of one reound.
- Empirical risk: $\frac{1}{R}\sum_{r=1}^{R} \mathbb{E}_{k\sim\mathcal{K}}\left[\frac{1}{|Z_{k,r}|}\sum_{i\in Z_{k,r}} l(M_{\hat{\boldsymbol{\theta}}_r}, \boldsymbol{z}_{k,i})\right]$

$R-$Round Generalization Bound

- $Z_{k,r} = \bigcup\{\mathcal{B}_{k,r,t}\}_{t=0}^{\tau-1}$, where $\mathcal{B}_{k,r,t}$ is the batch of samples used in local step t of round r in node k. τ is the duration of one reound.
- Empirical risk: $\frac{1}{R}\sum_{r=1}^{R}\mathbb{E}_{k\sim\mathcal{K}}\left[\frac{1}{|Z_{k,r}|}\sum_{i\in Z_{k,r}} l(M_{\hat{\boldsymbol{\theta}}_r}, \boldsymbol{z}_{k,i})\right]$
- Generalization error:
$$\Delta_{FedAvg}(\boldsymbol{S}) = \frac{1}{R}\sum_{r=1}^{R}\mathbb{E}_{k\sim\mathcal{K}}\left[\mathbb{E}_{\boldsymbol{z}\sim\mathcal{D}_k} l(M_{\hat{\boldsymbol{\theta}}_r}, \boldsymbol{z}) - \frac{1}{|Z_{k,r}|}\sum_{i\in Z_{k,r}} l(M_{\hat{\boldsymbol{\theta}}_r}, \boldsymbol{z}_{k,i})\right]$$

$R-$Round Generalization Bound

- $Z_{k,r} = \bigcup \{\mathcal{B}_{k,r,t}\}_{t=0}^{\tau-1}$, where $\mathcal{B}_{k,r,t}$ is the batch of samples used in local step t of round r in node k. τ is the duration of one reound.
- Empirical risk: $\frac{1}{R}\sum_{r=1}^{R} \mathbb{E}_{k\sim\mathcal{K}}\left[\frac{1}{|Z_{k,r}|}\sum_{i\in Z_{k,r}} l(M_{\hat{\boldsymbol{\theta}}_r}, \boldsymbol{z}_{k,i})\right]$
- Generalization error: $\Delta_{FedAvg}(\boldsymbol{S}) = \frac{1}{R}\sum_{r=1}^{R} \mathbb{E}_{k\sim\mathcal{K}}\left[\mathbb{E}_{\boldsymbol{z}\sim\mathcal{D}_k} l(M_{\hat{\boldsymbol{\theta}}_r}, \boldsymbol{z}) - \frac{1}{|Z_{k,r}|}\sum_{i\in Z_{k,r}} l(M_{\hat{\boldsymbol{\theta}}_r}, \boldsymbol{z}_{k,i})\right]$
- Bounded gradient variance: $\frac{1}{|Z_{k,r}|}\sum_{i\in Z_{k,r}} \|\nabla l(M, \boldsymbol{z}_{k,i}) - \frac{1}{|Z_{k,r}|}\sum_{i\in Z_{k,r}} \nabla l(M, \boldsymbol{z}_{k,i})\|^2 \leq \sigma^2$.

$R-$Round Generalization Bound

Theorem

Let $l(M_{\boldsymbol{\theta}}, \boldsymbol{z})$ be μ-strongly convex and L-smooth in $M_{\boldsymbol{\theta}}$. Local models at round r are calculated by doing τ local steps and the gradient variance is bounded by σ^2. The aggregated model at round r is $M_{\hat{\boldsymbol{\theta}}_r}$ is obtained by performing FedAvg and where the data points used in round r (i.e., $Z_{k,r}$) are sampled without replacement. Then the average generalization error bound is

$$\frac{1}{R}\sum_{r=1}^{R}\mathbb{E}_{k\sim\mathcal{K}}\left[\frac{2L\mathcal{K}(k)^2}{\mu}A+\sqrt{\frac{8L}{\mu}}\mathcal{K}(k)(AB)^{\frac{1}{2}}\right], \tag{2}$$

where $A=\tilde{O}\left(\sqrt{\frac{\mathcal{C}(M_{\boldsymbol{\theta}})}{|Z_{k,r}|}}+\frac{\sigma^2}{\mu\tau}+\frac{L}{\mu}\right)$,

$B=\tilde{O}\left(\mathbb{E}_{\{Z_{k,r}\}_{k=1}^{K}}\delta_{k,\mathcal{A}}(\{Z_{k,r}\}_{k=1}^{K})+\frac{\sigma^2}{\mu\tau}+\frac{L}{\mu}\right)$, *and $\mathcal{C}(M_{\boldsymbol{\theta}})$ shows the complexity of the model class of $M_{\boldsymbol{\theta}}$.*

R-Round Generalization Bound

$$\mathbb{E}_{\boldsymbol{S}}\, \Delta_{FedAvg}(\boldsymbol{S}) \leq \frac{1}{R}\sum_{r=1}^{R} \mathbb{E}_{k\sim\mathcal{K}} \left[\frac{2L\mathcal{K}(k)^2}{\mu} A + \sqrt{\frac{8L}{\mu}} \mathcal{K}(k)(AB)^{\frac{1}{2}} \right]$$

$$A = \tilde{O}\left(\sqrt{\frac{\mathcal{C}(M_{\boldsymbol{\theta}})}{|Z_{k,r}|}} + \frac{\sigma^2}{\mu\tau} + \frac{L}{\mu} \right), B = \tilde{O}\left(\mathbb{E}\, \delta_{k,\mathcal{A}}(\{Z_{k,r}\}_{k=1}^{K}) + \frac{\sigma^2}{\mu\tau} + \frac{L}{\mu} \right)$$

Representation learning Interpretation:

R-Round Generalization Bound

$$\mathbb{E}_{\boldsymbol{S}}\, \Delta_{FedAvg}(\boldsymbol{S}) \leq \frac{1}{R}\sum_{r=1}^{R} \mathbb{E}_{k\sim\mathcal{K}}\left[\frac{2L\mathcal{K}(k)^2}{\mu}A + \sqrt{\frac{8L}{\mu}\mathcal{K}(k)(AB)^{\frac{1}{2}}}\right]$$

$$A = \tilde{O}\left(\sqrt{\frac{\mathcal{C}(M_{\boldsymbol{\theta}})}{|Z_{k,r}|}} + \frac{\sigma^2}{\mu\tau} + \frac{L}{\mu}\right), B = \tilde{O}\left(\mathbb{E}\,\delta_{k,\mathcal{A}}(\{Z_{k,r}\}_{k=1}^{K}) + \frac{\sigma^2}{\mu\tau} + \frac{L}{\mu}\right)$$

Representation learning Interpretation:

$M_{\boldsymbol{\theta}}(\boldsymbol{x}) = (M_{\phi} \circ M_{\boldsymbol{h}})(\boldsymbol{x}) = M_{\boldsymbol{h}}(M_{\phi}(\boldsymbol{x}))$, $\mathcal{C}(M_{\boldsymbol{h}}) \ll \mathcal{C}(M_{\phi})$

R-Round Generalization Bound

$$\mathbb{E}_{\boldsymbol{S}}\, \Delta_{FedAvg}(\boldsymbol{S}) \leq \frac{1}{R}\sum_{r=1}^{R} \mathbb{E}_{k\sim\mathcal{K}} \left[\frac{2L\mathcal{K}(k)^2}{\mu} A + \sqrt{\frac{8L}{\mu}\mathcal{K}(k)(AB)^{\frac{1}{2}}} \right]$$

$$A = \tilde{O}\left(\sqrt{\frac{\mathcal{C}(M_{\boldsymbol{\theta}})}{|Z_{k,r}|}} + \frac{\sigma^2}{\mu\tau} + \frac{L}{\mu} \right), B = \tilde{O}\left(\mathbb{E}\,\delta_{k,\mathcal{A}}(\{Z_{k,r}\}_{k=1}^{K}) + \frac{\sigma^2}{\mu\tau} + \frac{L}{\mu} \right)$$

Representation learning Interpretation:

$M_{\boldsymbol{\theta}}(\boldsymbol{x}) = (M_\phi \circ M_{\boldsymbol{h}})(\boldsymbol{x}) = M_{\boldsymbol{h}}(M_\phi(\boldsymbol{x}))$, $\mathcal{C}(M_{\boldsymbol{h}}) \ll \mathcal{C}(M_\phi)$

> Our key intuition in this paper is that we can *reduce the aggregation frequency of M_ϕ, which leads to a larger τ and $|Z_{k,r}|$, hence smaller generalization error bound.*

R-Round Generalization Bound

$$\mathbb{E}_{\boldsymbol{S}}\, \Delta_{FedAvg}(\boldsymbol{S}) \leq \frac{1}{R}\sum_{r=1}^{R} \mathbb{E}_{k\sim\mathcal{K}}\left[\frac{2L\mathcal{K}(k)^2}{\mu}A + \sqrt{\frac{8L}{\mu}}\mathcal{K}(k)(AB)^{\frac{1}{2}}\right]$$

$$A = \tilde{O}\left(\sqrt{\frac{\mathcal{C}(M_{\boldsymbol{\theta}})}{|Z_{k,r}|}} + \frac{\sigma^2}{\mu\tau} + \frac{L}{\mu}\right), B = \tilde{O}\left(\mathbb{E}\,\delta_{k,\mathcal{A}}(\{Z_{k,r}\}_{k=1}^{K}) + \frac{\sigma^2}{\mu\tau} + \frac{L}{\mu}\right)$$

Representation learning Interpretation:

$M_{\boldsymbol{\theta}}(\boldsymbol{x}) = (M_{\phi} \circ M_{\boldsymbol{h}})(\boldsymbol{x}) = M_{\boldsymbol{h}}(M_{\phi}(\boldsymbol{x}))$, $\mathcal{C}(M_{\boldsymbol{h}}) \ll \mathcal{C}(M_{\phi})$

> Our key intuition in this paper is that we can *reduce the aggregation frequency of M_{ϕ}, which leads to a larger τ and $|Z_{k,r}|$, hence smaller generalization error bound.*

Aggregation frequency of M_{ϕ} cannot be reduced arbitrarily, as it would increase the empirical risk. (convergence rate: $O\left(\frac{\tau}{T} + \left(\frac{\tau}{T}\right)^{\frac{2}{3}} + \frac{1}{\sqrt{T}}\right)$ $, T = \tau R)$

FedALS: Federated Learning with Adaptive Local Steps

- Main idea: to maintain a uniform generalization error across both components (M_{ϕ} and $M_{\boldsymbol{h}}$).

Algorithm 1 FedALS

Input: Initial model $\{\boldsymbol{\theta}_{k,1,0} = [\boldsymbol{\phi}_{k,1,0}, \boldsymbol{h}_{k,1,0}]\}_{k=1}^{K}$, Learning rate η, number of local steps for the head model τ, adaptation coefficient α.

for Round r in $1, \ldots, R$ **do**
  for Node k in $1, \ldots, K$ **in parallel do**
    for Local step t in $0, \ldots, \tau - 1$ **do**
      Sample the batch $\mathcal{B}_{k,r,t}$ from $\mathcal{D}_k$.
      $\boldsymbol{\theta}_{k,r,t+1} = \boldsymbol{\theta}_{k,r,t} - \frac{\eta}{|\mathcal{B}_{k,r,t}|} \sum_{i \in \mathcal{B}_{k,r,t}} \nabla l(M_{\boldsymbol{\theta}_{k,r,t}}, \boldsymbol{z}_{k,i})$
      if $\mod(r\tau + t, \tau) = 0$ **then**
        $\boldsymbol{h}_{k,r,t} \leftarrow \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{h}_{k,r,t}$
      else if $\mod(r\tau + t, \alpha\tau) = 0$ **then**
        $\boldsymbol{\phi}_{k,r,t} \leftarrow \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{\phi}_{k,r,t}$
    $\boldsymbol{\theta}_{k,r+1,0} = \boldsymbol{\theta}_{k,r,\tau}$
return $\hat{\boldsymbol{\theta}}_R = \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{\theta}_{k,R,\tau}$

FedALS: Federated Learning with Adaptive Local Steps

- Main idea: to maintain a uniform generalization error across both components (M_{ϕ} and $M_{\boldsymbol{h}}$).
- Introduce parameter $\alpha = \frac{\tau_{M_{\phi}}}{\tau_{M_{\boldsymbol{h}}}}$ as an adaptation coefficient.

Algorithm 1 FedALS

Input: Initial model $\{\boldsymbol{\theta}_{k,1,0} = [\boldsymbol{\phi}_{k,1,0}, \boldsymbol{h}_{k,1,0}]\}_{k=1}^{K}$, Learning rate η, number of local steps for the head model τ, adaptation coefficient α.

for Round r in $1, ..., R$ **do**
  for Node k in $1, ..., K$ **in parallel do**
    for Local step t in $0, ..., \tau - 1$ **do**
      Sample the batch $\mathcal{B}_{k,r,t}$ from $\mathcal{D}_k$.
      $\boldsymbol{\theta}_{k,r,t+1} = \boldsymbol{\theta}_{k,r,t} - \frac{\eta}{|\mathcal{B}_{k,r,t}|} \sum_{i \in \mathcal{B}_{k,r,t}} \nabla l(M_{\boldsymbol{\theta}_{k,r,t}}, \boldsymbol{z}_{k,i})$
      if $\mod(r\tau + t, \tau) = 0$ **then**
        $\boldsymbol{h}_{k,r,t} \leftarrow \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{h}_{k,r,t}$
      else if $\mod(r\tau + t, \alpha\tau) = 0$ **then**
        $\boldsymbol{\phi}_{k,r,t} \leftarrow \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{\phi}_{k,r,t}$
    $\boldsymbol{\theta}_{k,r+1,0} = \boldsymbol{\theta}_{k,r,\tau}$
return $\hat{\boldsymbol{\theta}}_R = \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{\theta}_{k,R,\tau}$

Generalization Upper Bound

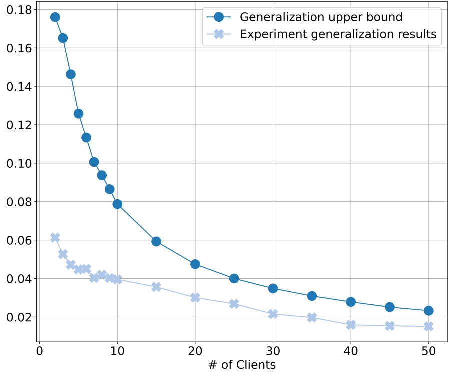
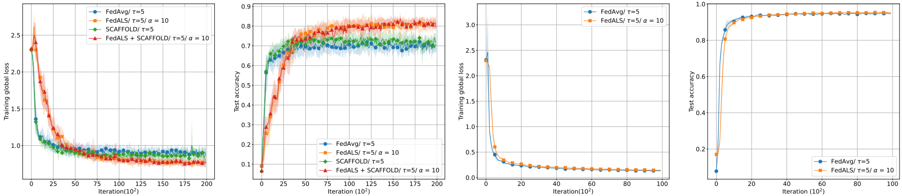


Figure: Generalization error and its upper bound derived in this work. The model is a logistic regression on a synthetic dataset generated from a multivariate normal distribution.

FedALS Experimental Results

Image classification (ϕ : the convolutional layers of ResNet, $\boldsymbol{h}$ the final dense layers)

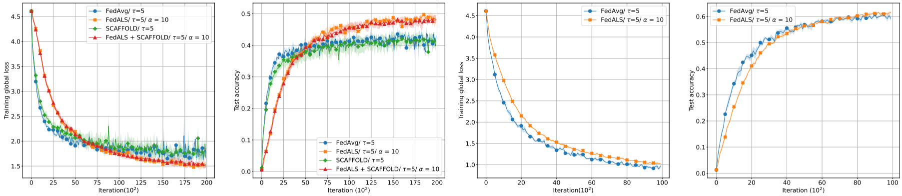


(a) non-iid (b) iid

Figure: Training ResNet-20 on SVHN.

FedALS Experimental Results

Image classification (ϕ : the convolutional layers of ResNet, $\boldsymbol{h}$ the final dense layers)

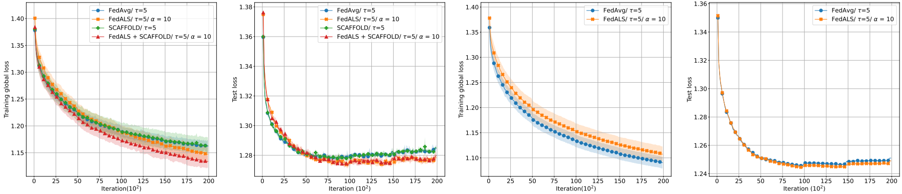


(a) non-iid (b) iid

Figure: Training ResNet-20 on CIFAR-100.

FedALS Experimental Results

LLM fine-tuning (ϕ : first 10 Transformer layers, $\boldsymbol{h}$ the final 2 Transformer layers)



(a) non-iid (b) iid

Figure: Fine-tuning OPT-125M on MultiNLI.

The Role of α and Communication Overhead

$$\alpha = \frac{\tau_{M_{\phi}}}{\tau_{M_{\boldsymbol{h}}}}$$

Table: The accuracy and communication overhead per client after training ResNet-20 in non-iid setting with $\tau = 5$ and variable α.

Value of $\boldsymbol{\alpha}$	Dataset		# of communicated parameters
	SVHN	CIFAR-10	
1	0.7010 ± 0.0330	0.4651 ± 0.0071	$2.344B$
5	0.8107 ± 0.0278	0.5201 ± 0.0302	$0.473B$
10	**0.8117 ± 0.0214**	**0.5224 ± 0.0365**	$0.239B$
25	0.7201 ± 0.1565	0.3814 ± 0.0641	$0.099B$
50	0.6377 ± 0.0520	0.2853 ± 0.0641	$0.052B$
100	0.5837 ± 0.0715	0.2817 ± 0.032	$0.029B$

Different Combinations of $\phi, \boldsymbol{h}$

$$\boldsymbol{\theta} = [\phi, \boldsymbol{h}] = [\text{first } \mathbf{L} \text{ layers, rest of the layers}]$$

Table: Different Combinations of $\phi, \boldsymbol{h}$ for training ResNet-20 in non-iid setting with $\tau = 5$, $\alpha = 10$.

Value of L	Dataset		
	SVHN	CIFAR-10	CIFAR-100
20	0.6991 ± 0.0160	0.4383 ± 0.0423	0.4781 ± 0.0123
16	$\mathbf{0.7112 \pm 0.0471}$	$\mathbf{0.4687 \pm 0.0111}$	$\mathbf{0.4782 \pm 0.0087}$
12	0.6760 ± 0.0474	0.4125 ± 0.0283	0.4249 ± 0.0143
8	0.6381 ± 0.0428	0.3779 ± 0.03451	0.4085 ± 0.0094
4	0.6339 ± 0.0446	0.3730 ± 0.0310	0.4183 ± 0.0108
1	0.6058 ± 0.0197	0.4013 ± 0.0308	0.3880 ± 0.0305

Conclusion

- Characterized generalization error bound federated learning in terms of local generalization and non-iidness.

Conclusion

- Characterized generalization error bound federated learning in terms of local generalization and non-iidness.
- Showed that less frequent aggregations, hence more local updates leads to a more generalizable model.

Conclusion

- Characterized generalization error bound federated learning in terms of local generalization and non-iidness.
- Showed that less frequent aggregations, hence more local updates leads to a more generalizable model.
- This insight led us to develop FedALS algorithm by increasing local steps for the initial layers of a deep learning model while doing more averaging for the final layers.

Thank You!

References I

- Barnes, Leighton Pate, Alex Dytso, and H. Vincent Poor. "Improved Information-Theoretic Generalization Bounds for Distributed, Federated, and Iterative Learning". In: *Entropy* 24 (2022). URL: https://api.semanticscholar.org/CorpusID:246634528.
- Reddi, Sashank J. et al. "Adaptive Federated Optimization". In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=LkFG3lB13U5.
- Yu, Tao, Eugene Bagdasaryan, and Vitaly Shmatikov. "Salvaging Federated Learning by Local Adaptation". In: *ArXiv* abs/2002.04758 (2020). URL: https://api.semanticscholar.org/CorpusID:211082601.