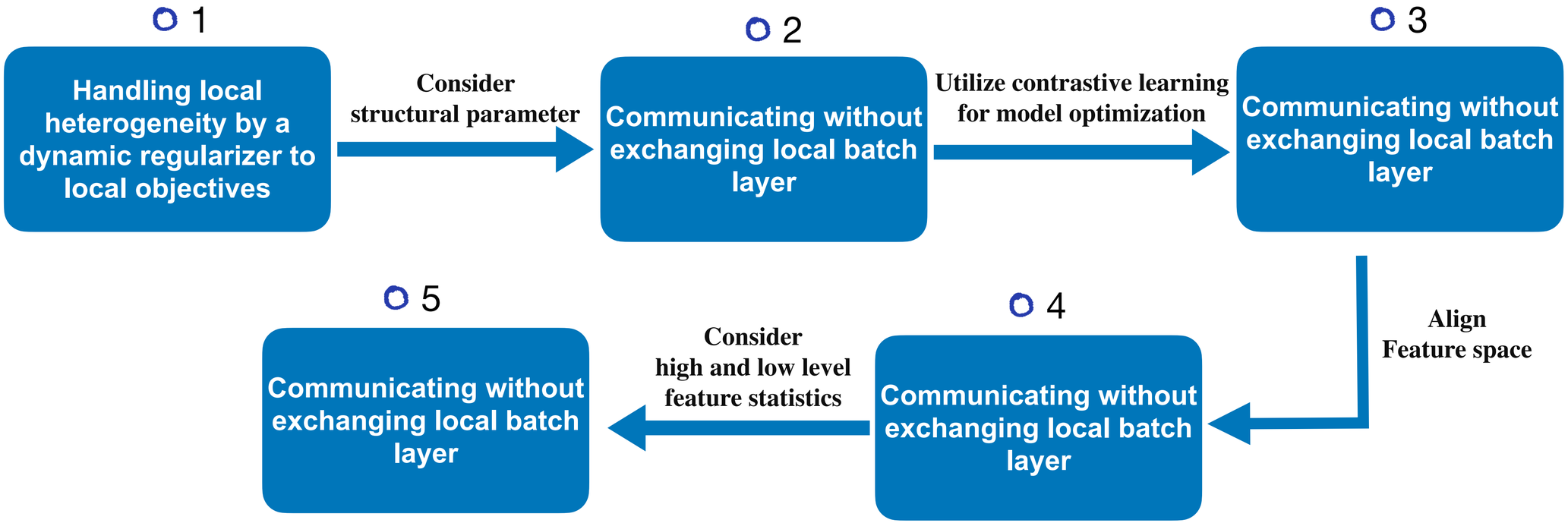
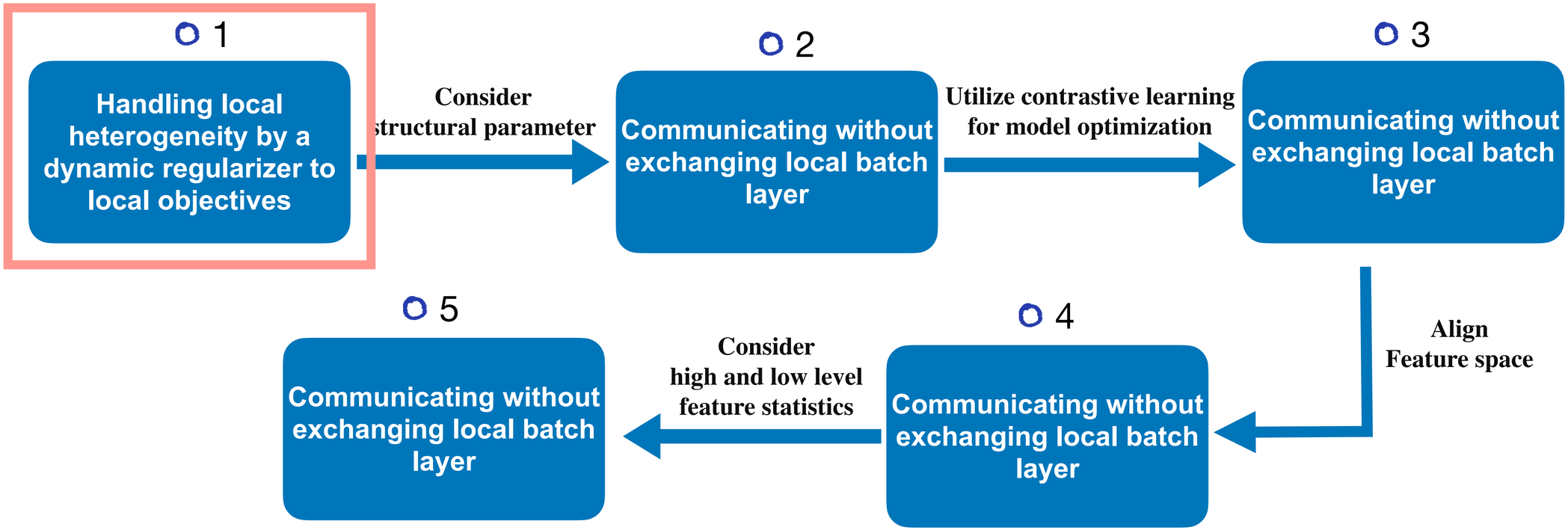
Model in Heterogeneous Federated Environments: Overviews and Methods



Xuqiang Chen 2024 09

Model in Heterogeneous Federated Environments: Overviews and Methods



Federated Optimization in Heterogeneous Networks

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, Virginia Smith

Publish Journal: The Conference on Machine Learning and Systems (*MLsys 2020*)

Publish Time: 2020

Impact Factor: /

Contents

Problem Description

Challenges

Federated Learning is a distributed learning paradigm with **two key challenges** that differentiate it from traditional distributed optimization:

1. **Systems heterogeneity**: significant variability in terms of the systems characteristics on each device in the network；
2. **Statistical heterogeneity**: non-identically distributed data across the network.

- In this work, inspired by FedAvg, we explore a broader framework, **FedProx**, that is capable of handling heterogeneous federated environments while maintaining similar privacy and computational benefits；
- We analyze the convergence behavior of the framework through **a statistical dissimilarity characterization** between local functions, while also taking into account practical systems constraints

Problem Description

Federated learning methods

Federated learning methods (e.g., McMahan et al., 2017; Smith et al., 2017) are designed to handle multiple devices collecting data and a central server coordinating the global learning objective across the network.

$$\min_w \ f(w) = \sum_{k=1}^{N} p_k F_k(w) = \mathbb{E}_k[F_k(w)],$$

where N is the number of devices, $p_k \geq 0$, and $\sum_k p_k = 1$. In general, the local objectives measure the local empirical risk over possibly differing data distributions D_k.

Local objective and local solver

To reduce communication, a ***local objective function*** based on the device's data is used as a surrogate for the global objective function. ***Local solvers*** are used to optimize the local objective functions on each of the selected devices. The key to allowing flexible performance in this scenario is that each of the local objectives can be solved ***inexactly.***

Problem Description

Solve *INEXACTLY*

Definition of the y-inexact

Definition 1 (γ-inexact solution)**.** For a function $h(w; w_0) = F(w) + \frac{\mu}{2}\|w - w_0\|^2$, and $\gamma \in [0, 1]$, we say w^* is a γ-inexact solution of $\min_w h(w; w_0)$ if $\|\nabla h(w^*; w_0)\| \leq \gamma \|\nabla h(w_0; w_0)\|$, where $\nabla h(w; w_0) = \nabla F(w) + \mu(w - w_0)$. Note that a smaller γ corresponds to higher accuracy.

McMahan et al. (2017) show empirically that it is crucial to tune the optimization hyperparameters of FedAvg properly.

the number of local epochs

- more local epochs allow for more local computation and potentially reduced communication
- a larger number of local epochs may lead each device towards the optima of its local objective.

Algorithm 1 Federated Averaging (`FedAvg`)

Input: $K, T, \eta, E, w^0, N, p_k, k = 1, \cdots, N$
for $t = 0, \cdots, T-1$ **do**
  Server selects a subset S_t of K devices at random (each device k is chosen with probability p_k)
  Server sends w^t to all chosen devices
  Each device $k \in S_t$ updates w^t for E epochs of SGD on F_k with step-size η to obtain w_k^{t+1}
  Each device $k \in S_t$ sends w_k^{t+1} back to the server
  Server aggregates the w's as $w^{t+1} = \frac{1}{K}\sum_{k \in S_t} w_k^{t+1}$
end for

Contents

FedProx - Tolerating partial work

In FedProx, we generalize FedAvg by allowing for variable amounts of work to be performed locally across devices based on their available systems resources

Definition 2 (γ_k^t-inexact solution)**.** For a function $h_k(w; w_t) = F_k(w) + \frac{\mu}{2}\|w - w_t\|^2$, and $\gamma \in [0, 1]$, we say w^* is a γ_k^t-inexact solution of $\min_w h_k(w; w_t)$ if $\|\nabla h_k(w^*; w_t)\| \leq \gamma_k^t \|\nabla h_k(w_t; w_t)\|$, where $\nabla h_k(w; w_t) = \nabla F_k(w) + \mu(w - w_t)$. Note that a smaller γ_k^t corresponds to higher accuracy.

FedProx - Proximal term

We propose to add a proximal term to the local subproblem to effectively limit the impact of variable local updates.

$$\min_w h_k(w;\ w^t) = F_k(w) + \frac{\mu}{2}\|w - w^t\|^2 .$$

where w is the global parameter, w^t is the local parameter in t iteration, μ is the trade-off factor.

FedProx

A. It addresses the issue of statistical heterogeneity by restricting the local updates to be closer to the global model without any need to manually set the number of local epochs;

B. It allows for safely incorporating variable amounts of local work resulting from systems heterogeneity.

Algorithm 2 `FedProx` (Proposed Framework)

Input: $K, T, \mu, \gamma, w^0, N, p_k, k = 1, \cdots, N$
for $t = 0, \cdots, T-1$ **do**
  Server selects a subset S_t of K devices at random (each device k is chosen with probability p_k)
  Server sends w^t to all chosen devices
  Each chosen device $k \in S_t$ finds a w_k^{t+1} which is a γ_k^t-inexact minimizer of: $w_k^{t+1} \approx \arg\min_w \ h_k(w;\ w^t) = F_k(w) + \frac{\mu}{2}\|w - w^t\|^2$
  Each device $k \in S_t$ sends w_k^{t+1} back to the server
  Server aggregates the w's as $w^{t+1} = \frac{1}{K}\sum_{k \in S_t} w_k^{t+1}$
end for

Contents

Experiment Results

Data

Synthetic data

We generate samples (X_k, Y_k) according to the model $y = argmax(\text{softmax}(Wx + b))$ $x_k \sim \mathcal{N}(v_k, \Sigma)$,

α controls how much local models differ from each other

β controls how much the local data at each device differs from that of other devices.

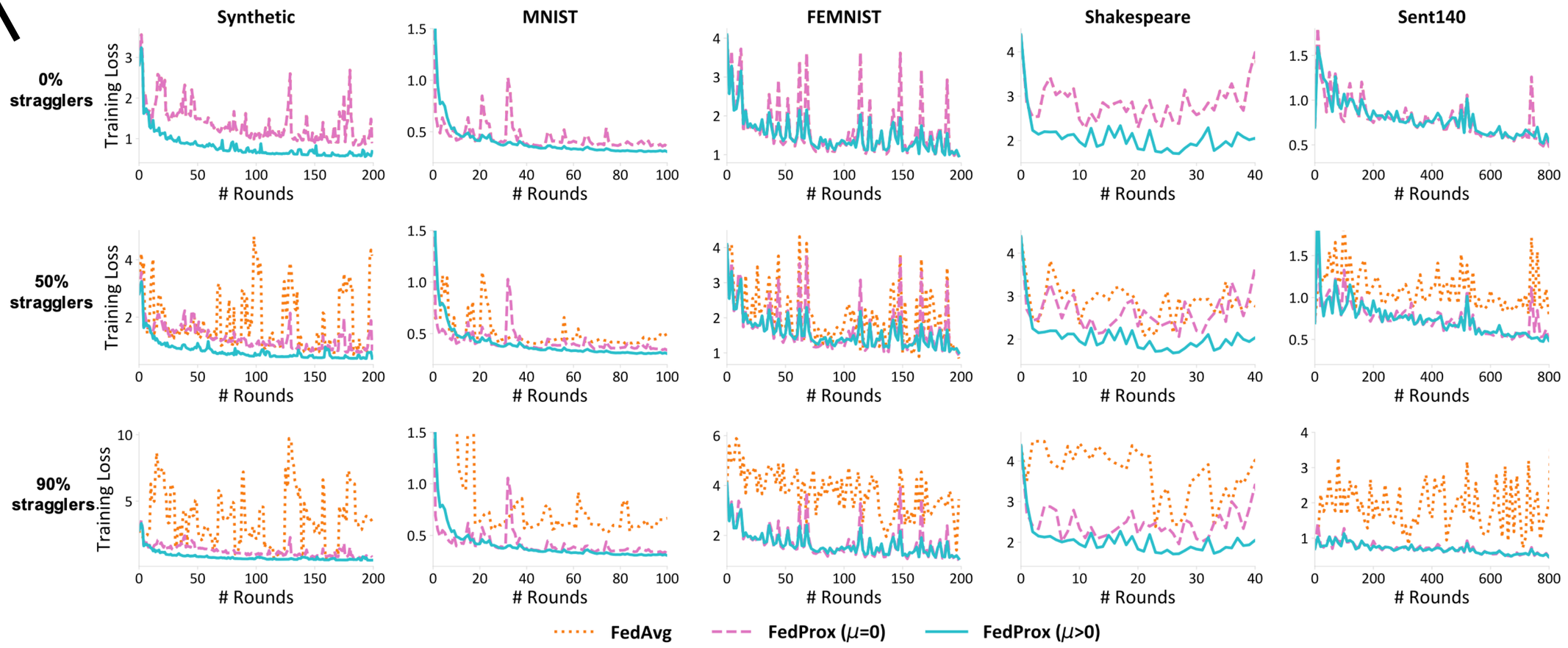
Real world data

Dataset	Devices	Samples	Samples/device	
			mean	stdev
MNIST	1,000	69,035	69	106
FEMNIST	200	18,345	92	159
Shakespeare	143	517,106	3,616	6,808
Sent140	772	40,783	53	32

Experiment Results

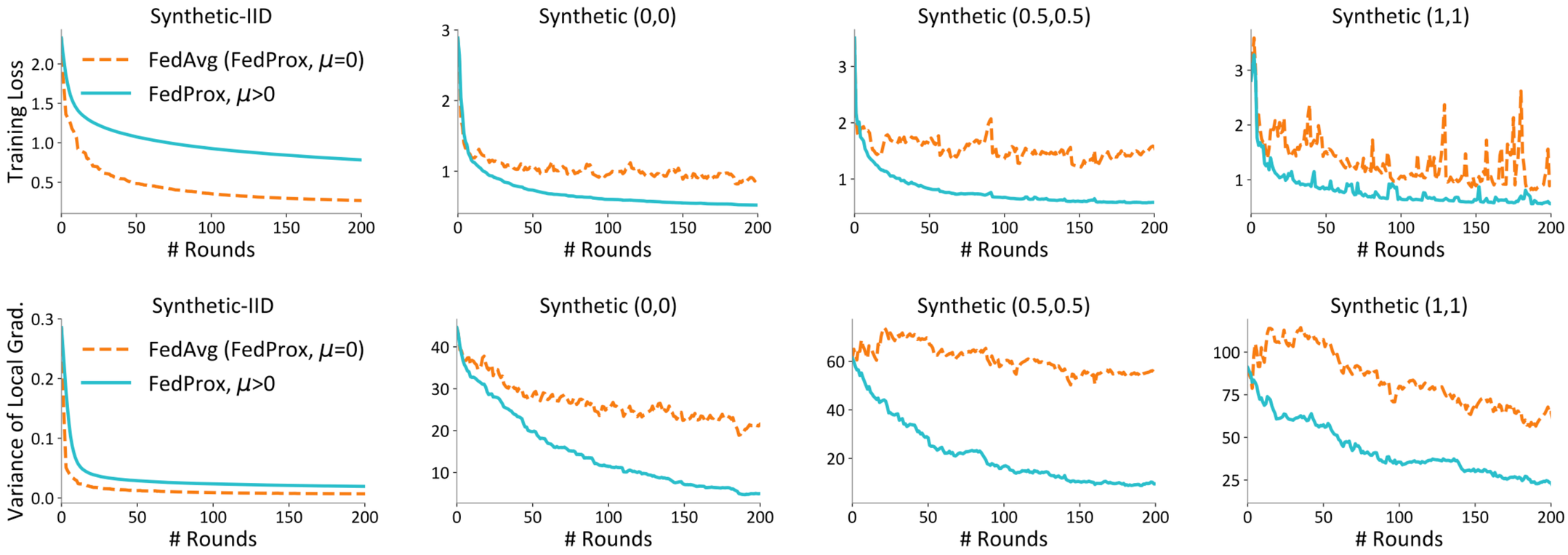
Tolerating partial solutions in the face of systems heterogeneity

assign x number of epochs (chosen uniformly at random between [1, E]) to 0%, 50%, and 90% of the selected devices



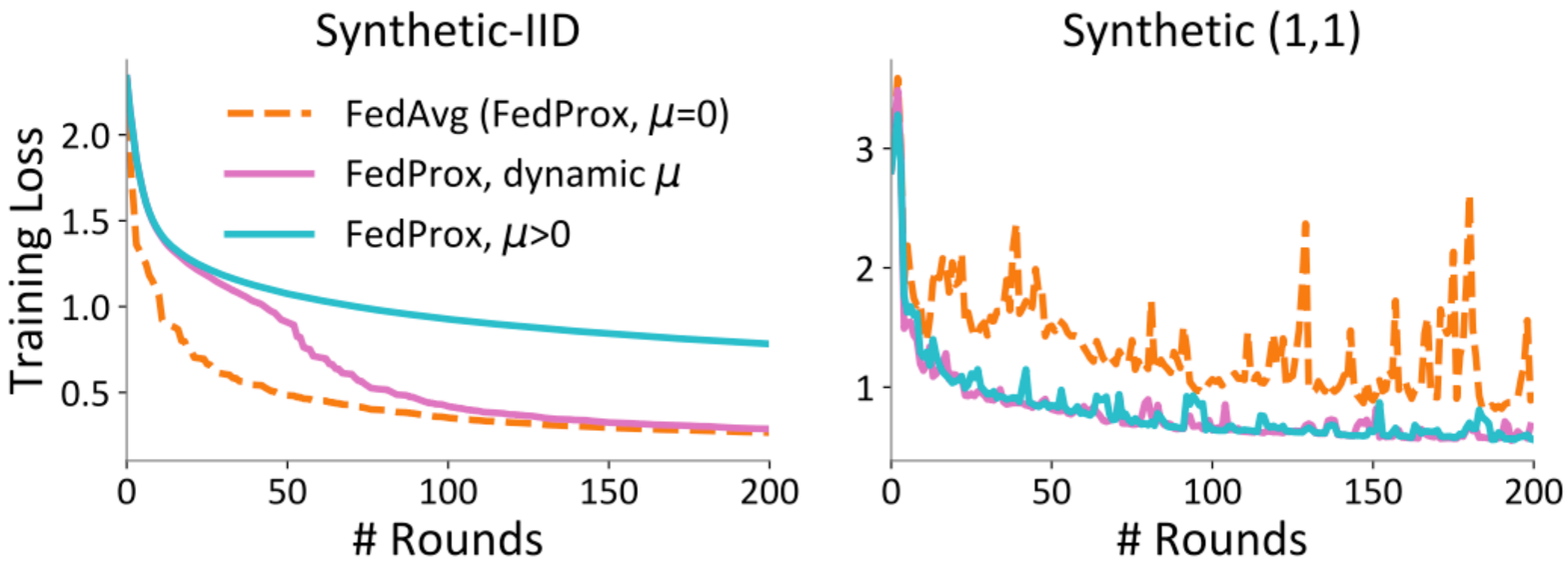
Experiment Results

Effects of Statistical Heterogeneity



Experiment Results

Effectiveness of setting μ adaptively based on the current model performance



Contents

Conclusion

We have proposed **FedProx**, an optimization framework that tackles the systems and statistical heterogeneity inherent in federated networks.

FedProx allows for variable amounts of work to be performed locally across devices, and relies on a proximal term to help stabilize the method.

FedBN: Federated Learning on Non-IID Features via Local Batch Normalization

X Li, M Jiang, X Zhang, M Kamp, Q Dou

Publish Journal: The International Conference on Learning Representations (*ICLR 2021*)

Publish Time: 2021

Impact Factor: /

Contents

Problem Description

Challenges

In most cases, the assumption of **independent and identically distributed samples** across local clients does not hold for federated learning setups.

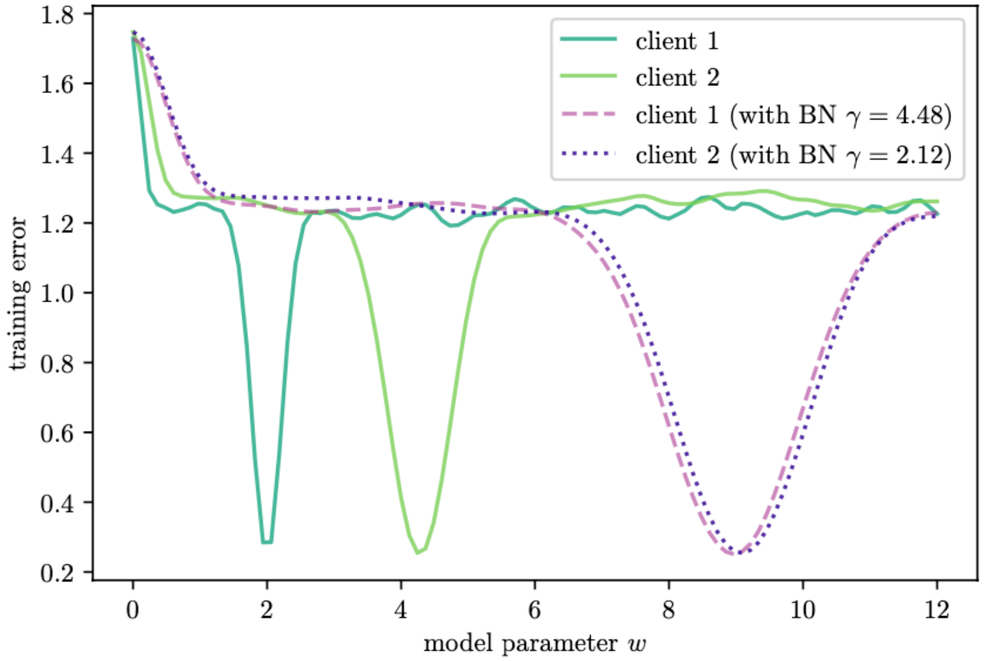
- we focus on the shift in the feature space, which has not yet been explored in the literature. Specifically, we consider that local data deviates in terms of the distribution in feature space, and identify this scenario as **feature shift**.

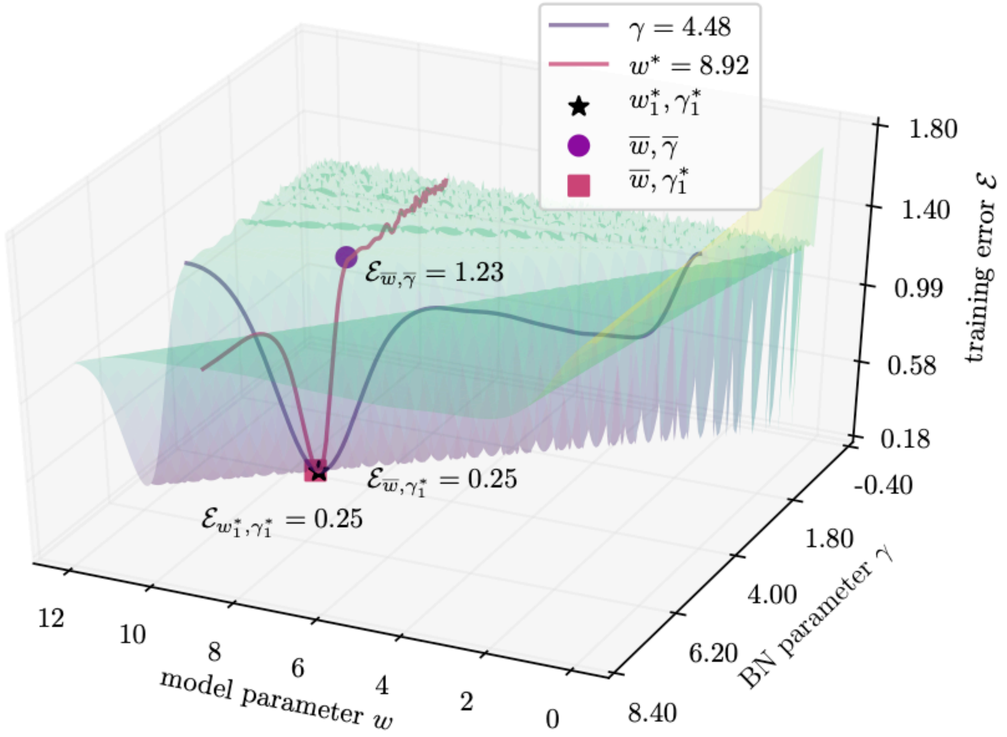
Observation of BN in a FL Toy Example

Problem Description

Observation of BN in a FL Toy Example

First, we illustrate that local batch normalization harmonizes local data distributions. Applying local BN, the local training error surfaces become similar and averaging the models can be beneficial.





Contents

FEDERATED AVERAGING WITH LOCAL BATCH NORMALIZATION

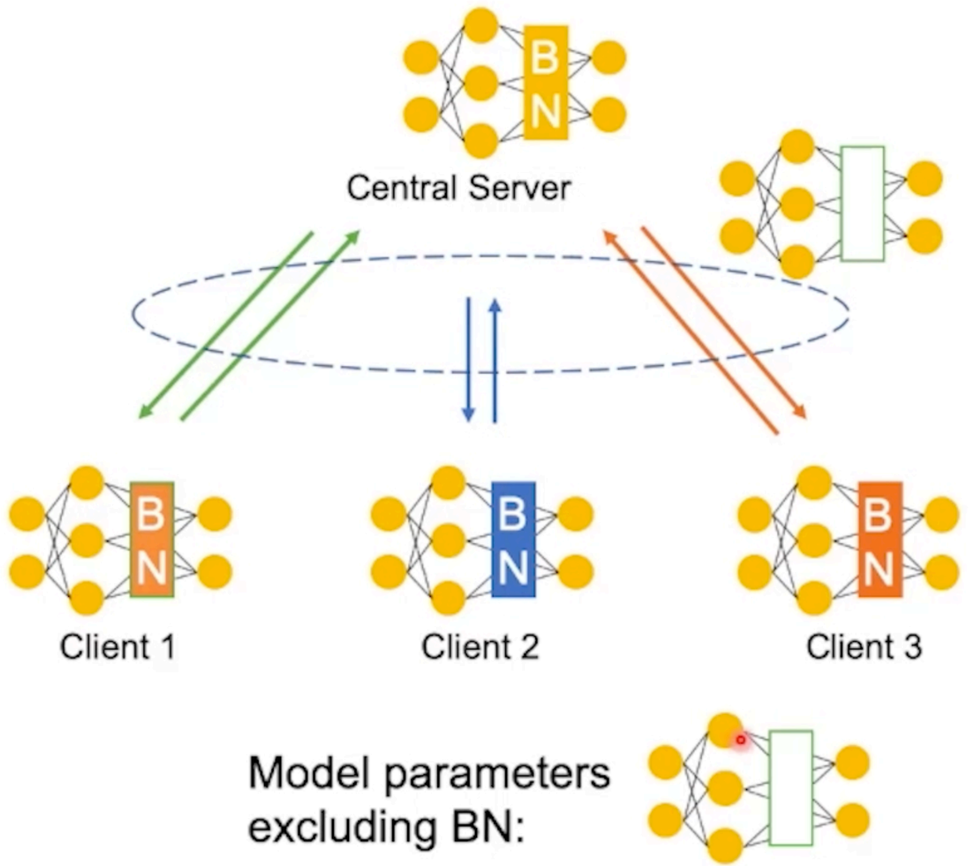
We propose an efficient and effective learning strategy denoted FedBN. Similar to FedAvg, FedBN performs **local updates and averages local models**. However, FedBN assumes local models have BN layers and excludes their parameters from the averaging step.

Algorithm 1 Federated Learning using FedBN

Notations: The user indexed by k, neural network layer indexed by l, initialized model parameters: $w_{0,k}^{(l)}$, local update pace: E, and total optimization round T.

for each round $t = 1, 2, \ldots, T$ **do**
  for each user k and each layer l **do**
    $w_{t+1,k}^{(l)} \leftarrow SGD(w_{t,k}^{(l)})$
  end for
  if $\mathrm{mod}(t, E) = 0$ **then**
    for each user k and each layer l **do**
      if layer l is not BatchNorm **then**
        $w_{t+1,k}^{(l)} \leftarrow \frac{1}{K} \sum_{k=1}^{K} w_{t+1,k}^{(l)}$
      end if
    end for
  end if
end for

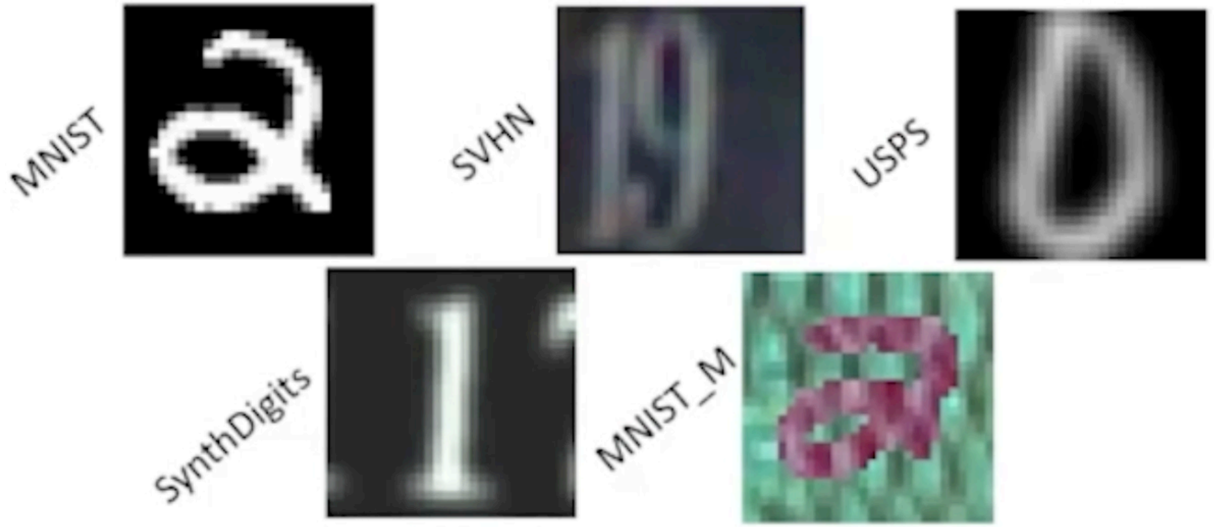
FEDERATED AVERAGING WITH LOCAL BATCH NORMALIZATION



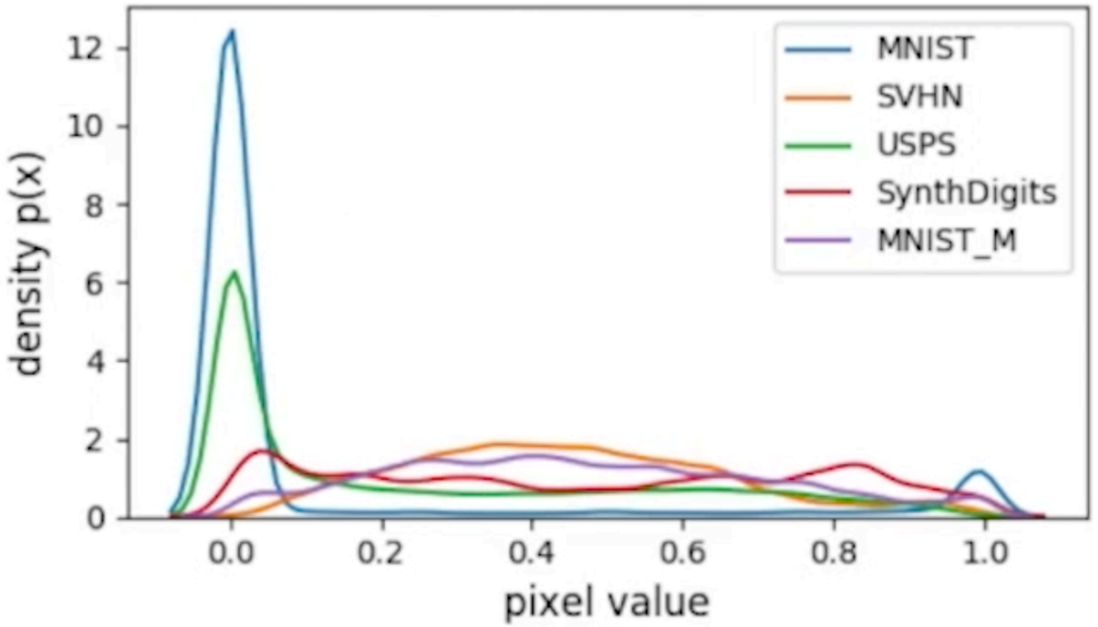
Contents

Experiment Results

Datasets



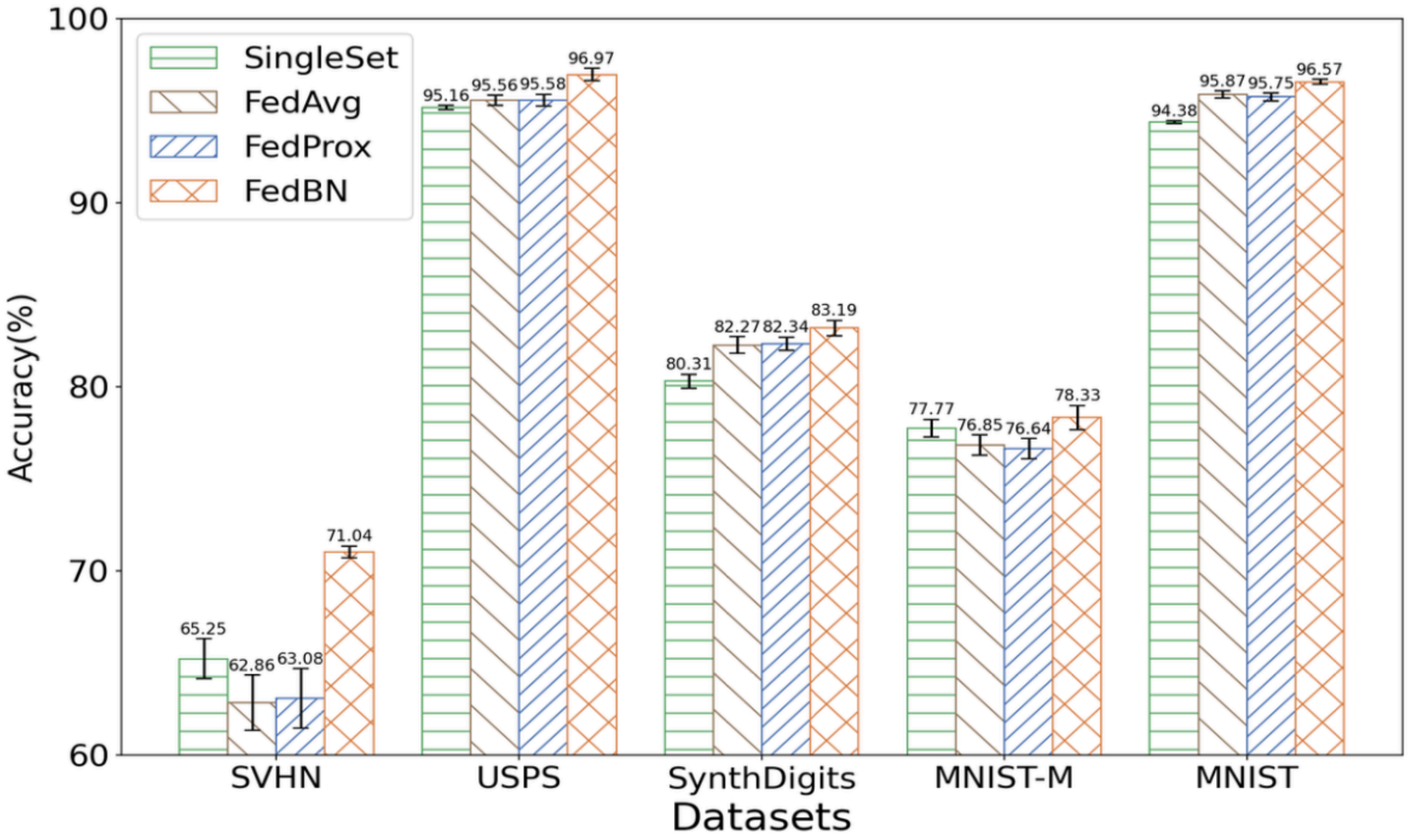
Visualization of Datasets



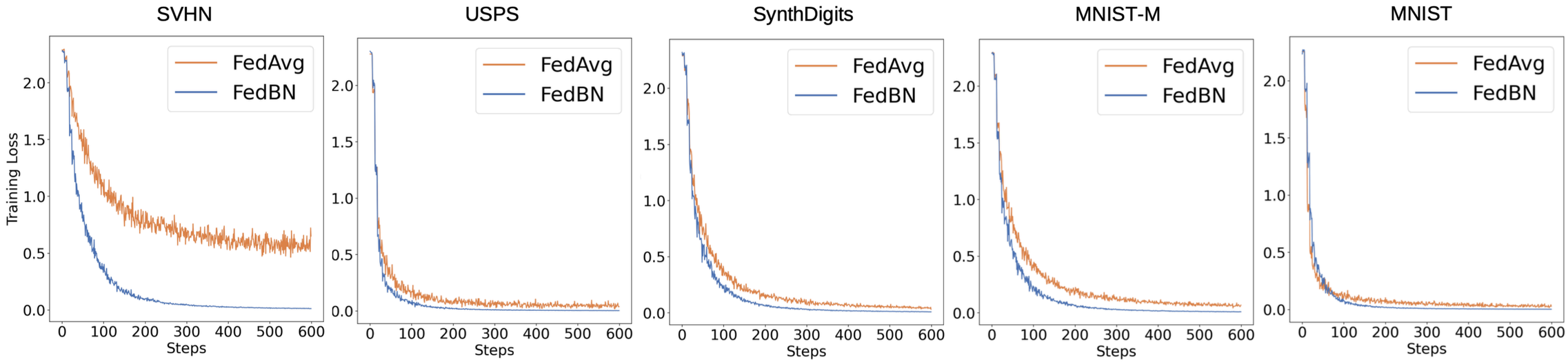
Distribution of the Datasets

Experiment Results

Test Accuracy of Benchmark Datasets

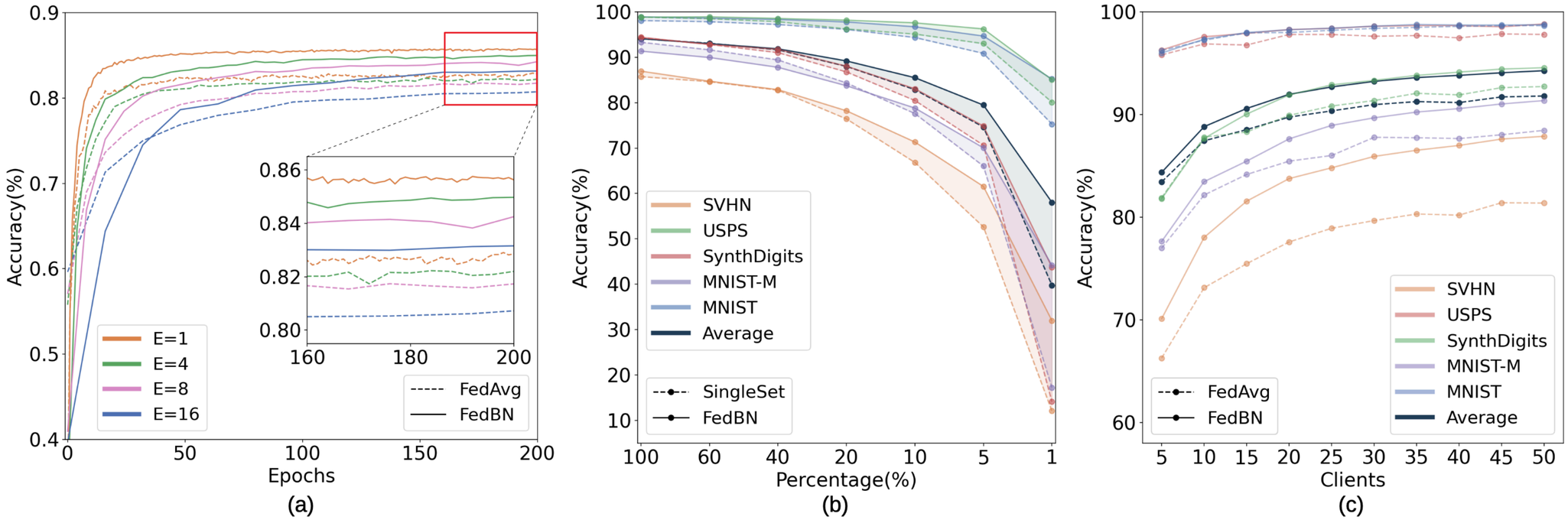


Convergence Rate of Benchmark



Experiment Results

Analysis of Local Updating Epochs, Local Dataset Size,



Model Performance on Real-World Datasets

Method	Caltech-10				DomainNet						ABIDE (medical)			
	A	C	D	W	C	I	P	Q	R	S	NYU	USM	UM	UCLA
SingleSet	54.9 (1.5)	40.2 (1.6)	78.7 (1.3)	86.4 (2.4)	41.0 (0.9)	23.8 (1.2)	36.2 (2.7)	**73.1 (0.9)**	48.5 (1.9)	34.0 (1.1)	58.0 (3.3)	73.4 (2.2)	64.3 (1.4)	57.3 (2.4)
FedAvg	54.1 (1.1)	44.8 (1.0)	66.9 (1.5)	85.1 (2.9)	48.8 (1.9)	24.9 (0.7)	36.5 (1.1)	56.1 (1.6)	46.3 (1.4)	36.6 (2.5)	62.7 (1.7)	73.1 (2.4)	**70.7 (0.5)**	64.7 (0.7)
FedProx	54.2 (2.5)	44.5 (0.5)	65.0 (3.6)	84.4 (1.7)	48.9 (0.8)	24.9 (1.0)	36.6 (1.8)	54.4 (3.1)	47.8 (0.8)	36.9 (2.1)	63.3 (1.0)	73.0 (1.8)	70.5 (1.1)	64.5 (1.2)
FedBN	**63.0 (1.6)**	**45.3 (1.5)**	**83.1 (2.5)**	**90.5 (2.3)**	**51.2 (1.4)**	**26.8 (0.5)**	**41.5 (1.4)**	71.3 (0.7)	**54.8 (0.8)**	**42.1 (1.3)**	**65.6 (1.1)**	**75.1 (1.4)**	68.6 (2.9)	**65.5 (1.0)**

Contents

Model-Contrastive Federated Learning

Qinbin Li, Bingsheng He, Dawn Song

Publish Journal: The Computer Vision and Pattern Recognition Conference (*CVPR 2021*)

Publish Time: 2021

Impact Factor: /

Contents

Challenges

A key challenge in federated learning is the heterogeneity of data distribution on different parties.

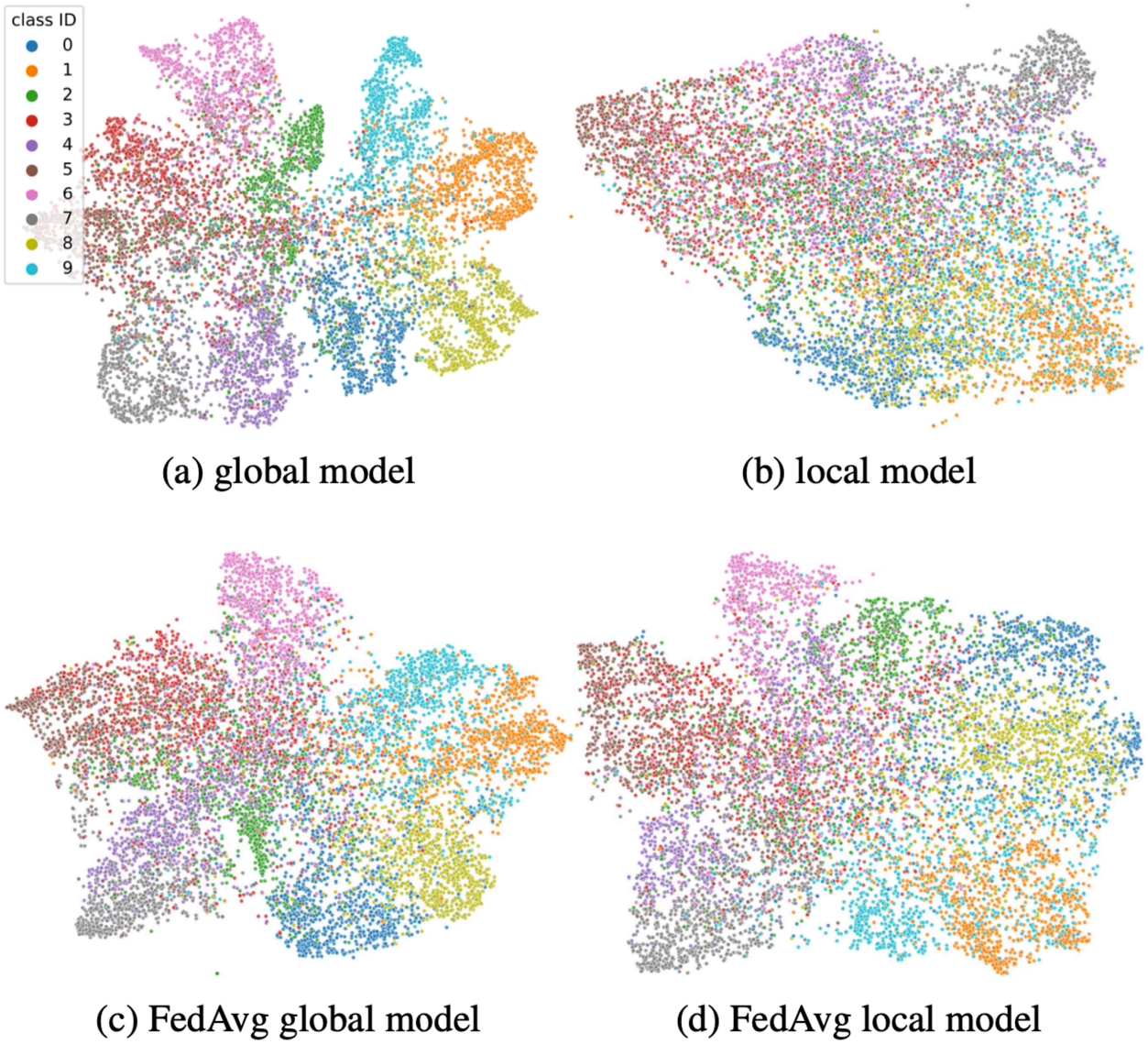
- Suppose there are N parties, denoted $P_1, \ldots, P_N$. Party P_i has a local dataset D_i. Our goal is to learn a machinelearning model w over the dataset $\mathcal{D} \triangleq \bigcup_{i \in [N]} \mathcal{D}^i$ with the help of a central server, while the raw data are not exchanged. The objective is to solve

$$\arg\min_{w} \mathcal{L}(w) = \sum_{i=1}^{N} \frac{|\mathcal{D}^i|}{|\mathcal{D}|} L_i(w),$$

Problem Description

Motivation

MOON is based on an intuitive idea: the model trained on the whole dataset is able to **extract a better feature representation** than the model trained on a skewed subset.



(a) global model (b) local model

(c) FedAvg global model (d) FedAvg local model

Contents

Moon

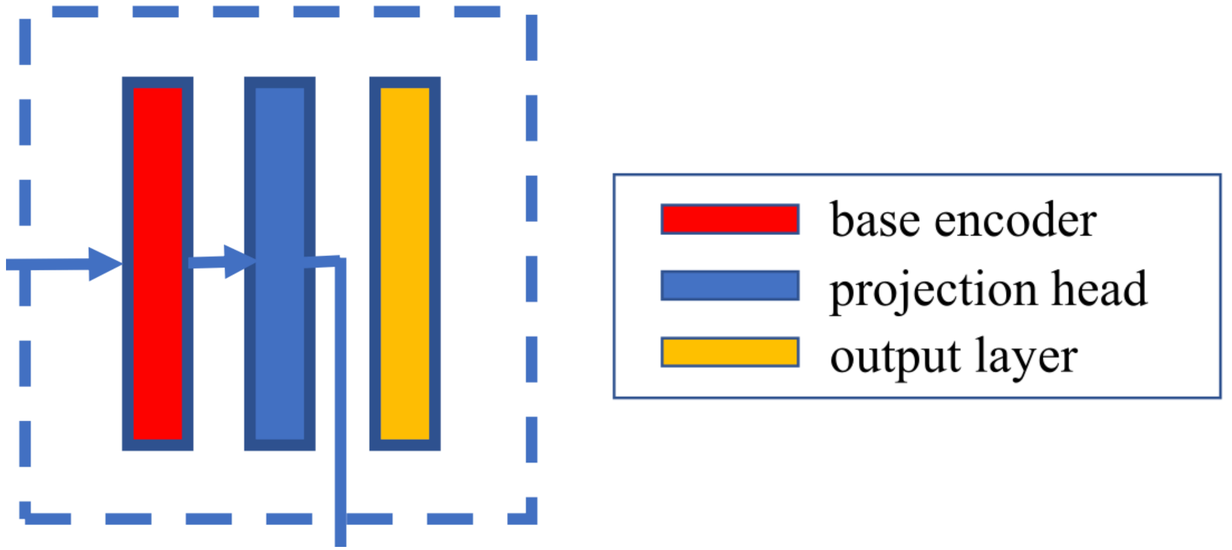
MOON is designed as a simple and effective approach based on FedAvg, only introducing lightweight but novel modifications in the local training phase

decrease the distance between the representation learned by the local model and the representation learned by the global model

increase the distance between the representation learned by the local model and the representation learned by the previous local model

Network Architecture

- Projection head: map the representation to a space with a fixed dimension
- $F_w(\,.\,)$ denotes the whole network, $R_w(\,.\,)$ denotes the network before the output layer.

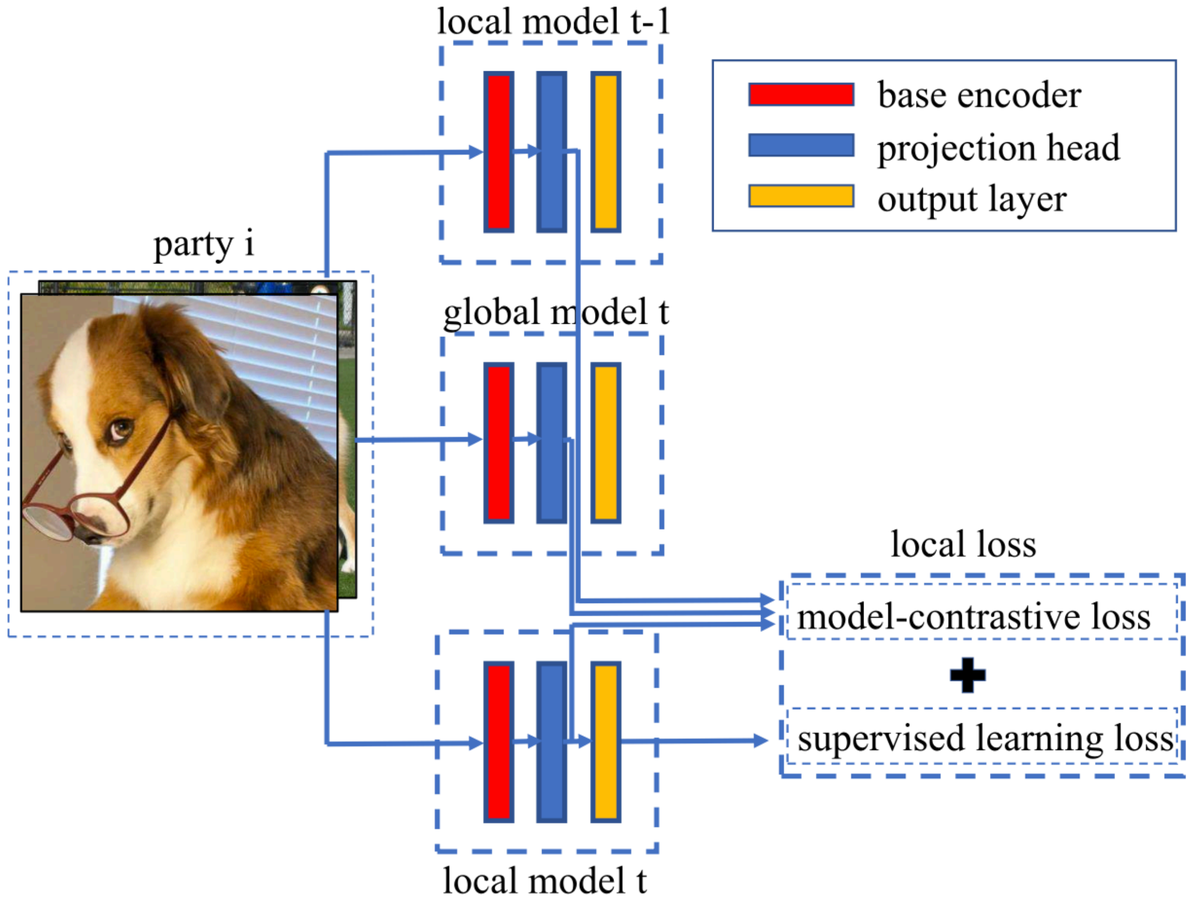


Local Objective

our local loss consists two parts. The first part is a **typical loss term** (e.g., cross-entropy loss) in supervised learning denoted as l_{sup}. The second part is our proposed **model-contrastive loss term** denoted as l_{con}.

$$\ell_{con} = -\log \frac{\exp(\text{sim}(z, z_{glob})/\tau)}{\exp(\text{sim}(z, z_{glob})/\tau) + \exp(\text{sim}(z, z_{prev})/\tau)}$$

$$z_{glob} = R_{w^t}(x) \; z_{prev} = R_{w_i^{t-1}}(x) \; z = R_{w_i^t}(x)$$



Local Objective

The loss of local objective is computed by:

$$\ell = \ell_{sup}(w_i^t; (x, y)) + \mu \ell_{con}(w_i^t; w_i^{t-1}; w^t; x),$$

Algorithm 1: The MOON framework

Input: number of communication rounds T, number of parties N, number of local epochs E, temperature τ, learning rate η, hyper-parameter μ

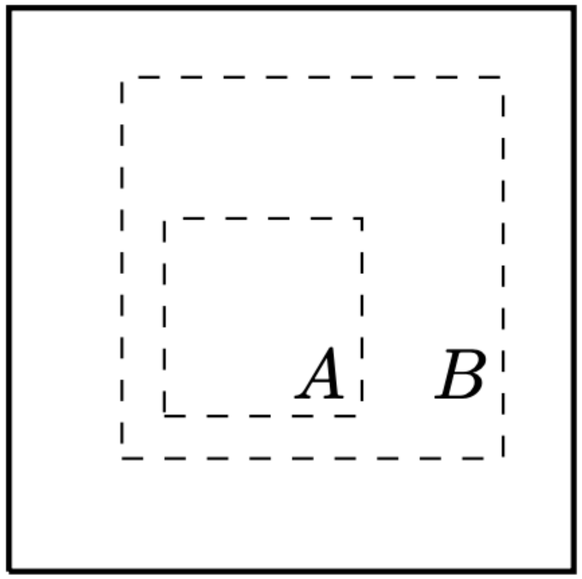
Output: The final model w^T

Server executes:
initialize w^0
for $t = 0, 1, ..., T-1$ **do**
  for $i = 1, 2, ..., N$ **in parallel do**
    send the global model w^t to P_i
    $w_i^t \leftarrow$ **PartyLocalTraining**(i, w^t)
  $w^{t+1} \leftarrow \sum_{k=1}^{N} \frac{|\mathcal{D}^i|}{|\mathcal{D}|} w_k^t$
return w^T

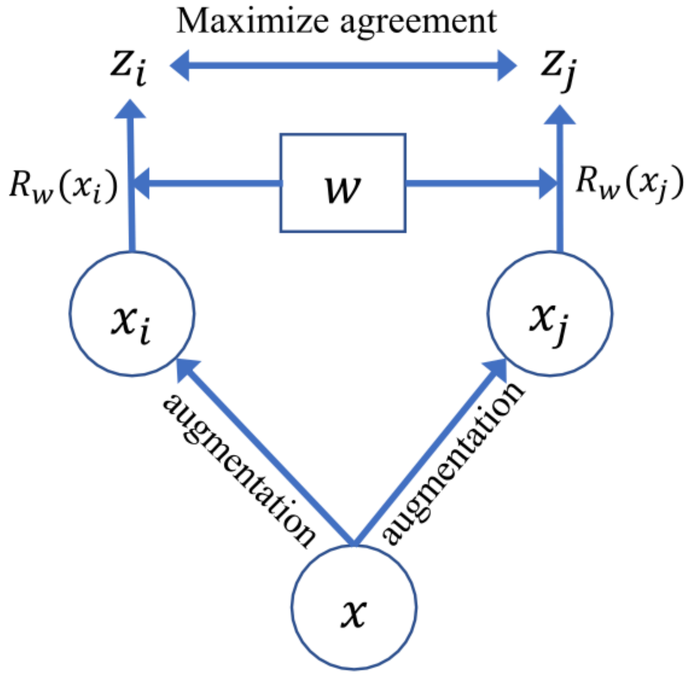
PartyLocalTraining(i, w^t):
$w_i^t \leftarrow w^t$
for epoch $i = 1, 2, ..., E$ **do**
  for each batch $\mathbf{b} = \{x, y\}$ of $\mathcal{D}^i$ **do**
    $\ell_{sup} \leftarrow CrossEntropyLoss(F_{w_i^t}(x), y)$
    $z \leftarrow R_{w_i^t}(x)$
    $z_{glob} \leftarrow R_{w^t}(x)$
    $z_{prev} \leftarrow R_{w_i^{t-1}}(x)$
    $\ell_{con} \leftarrow -\log \frac{\exp(\text{sim}(z, z_{glob})/\tau)}{\exp(\text{sim}(z, z_{glob})/\tau) + \exp(\text{sim}(z, z_{prev})/\tau)}$
    $\ell \leftarrow \ell_{sup} + \mu \ell_{con}$
    $w_i^t \leftarrow w_i^t - \eta \nabla \ell$
return w_i^t to server

Comparison between SimCLR and MOON

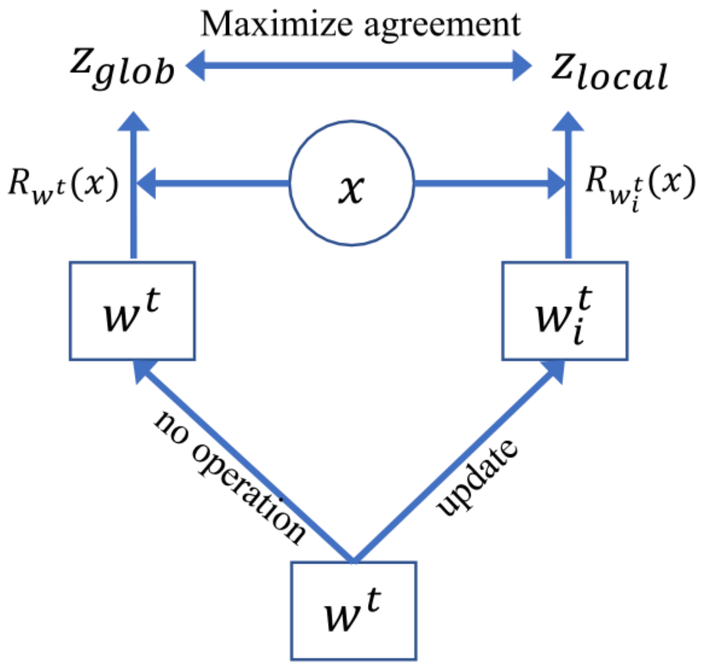
SimCLR maximizes the agreement between representations of different views of the same image



(a) Global and local views.



SimCLR



MOON

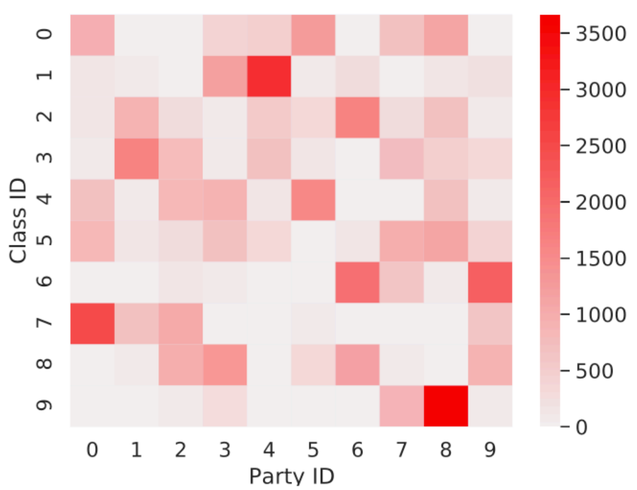
Contents

Dataset and Metrics

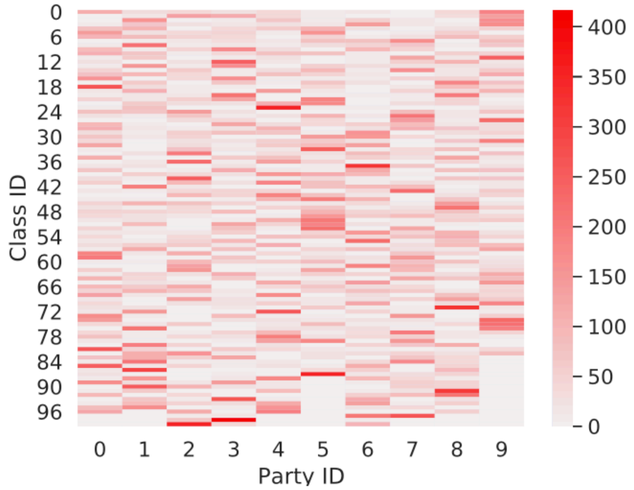
Datasets	Algorithms	Models
CIFAR100	FedAvg	Base CNN
CIFAR10	FedProx	ResNet-50
Tiny-Imagenet	Scaffold	MLP

Experiment Results

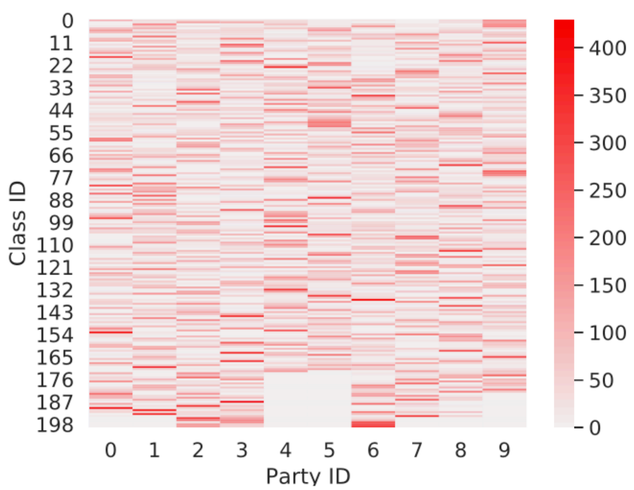
Data distribution of each party using non-IID data partition



(a) CIFAR-10



(b) CIFAR-100



(c) Tiny-Imagenet

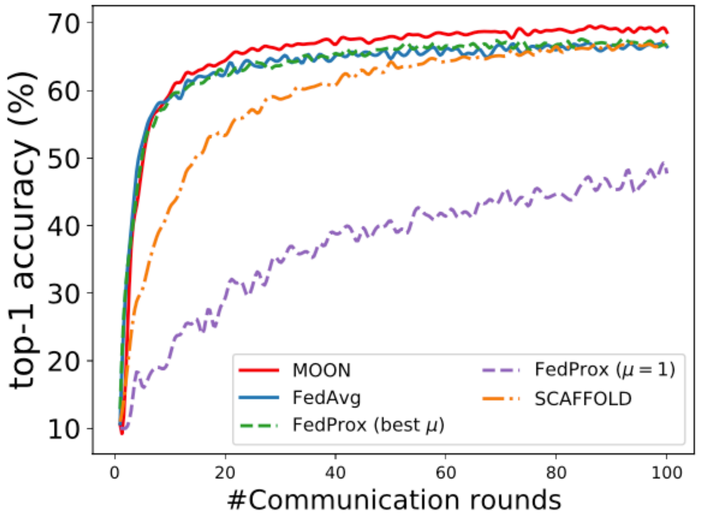
Accuracy and Communication Efficiency

Method	CIFAR-10	CIFAR-100	Tiny-Imagenet
MOON	**69.1%**±0.4%	**67.5%**±0.4%	**25.1%**±0.1%
FedAvg	66.3%±0.5%	64.5% ±0.4%	23.0%±0.1%
FedProx	66.9%±0.2%	64.6%±0.2%	23.2%±0.2%
SCAFFOLD	66.6%±0.2%	52.5% ±0.3%	16.0%±0.2%
SOLO	46.3% ±5.1%	22.3%±1.0%	8.6%±0.4%

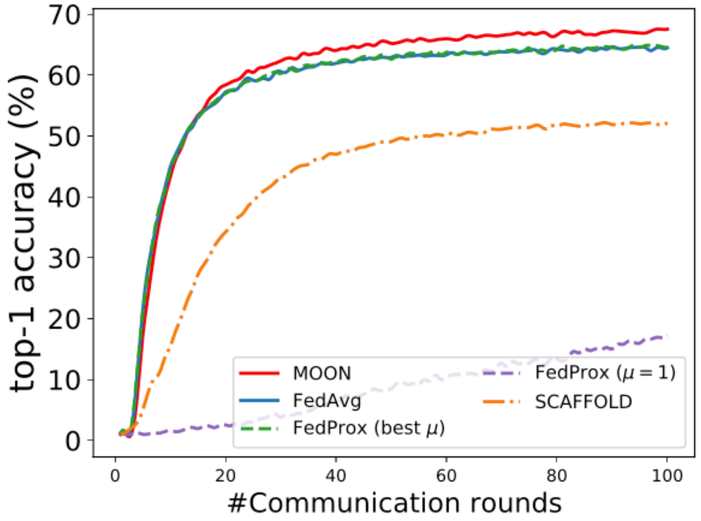
Method	CIFAR-10		CIFAR-100		Tiny-Imagenet	
	#rounds	speedup	#rounds	speedup	#rounds	speedup
FedAvg	100	1×	100	1×	20	1×
FedProx	52	1.9×	75	1.3×	17	1.2×
SCAFFOLD	80	1.3×	\	<1×	\	<1×
MOON	**27**	**3.7**×	**43**	**2.3**×	**11**	**1.8**×

Experiment Results

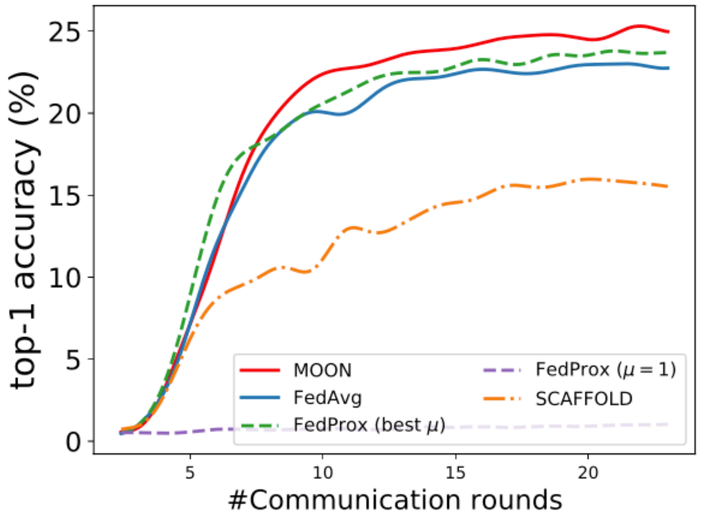
Convergence Rate



(a) CIFAR-10



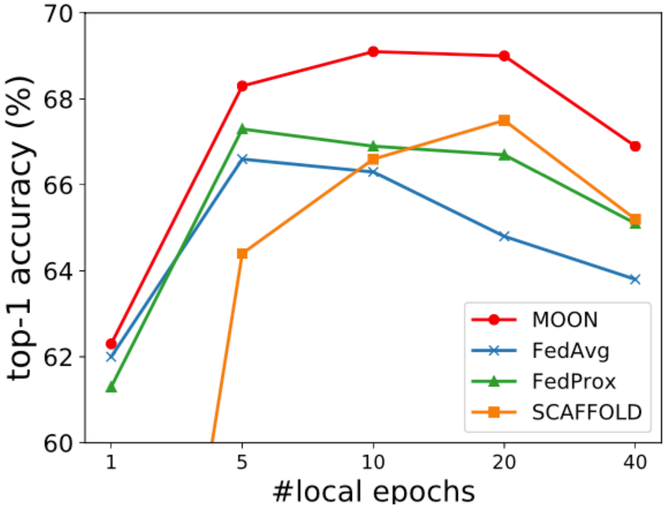
(b) CIFAR-100



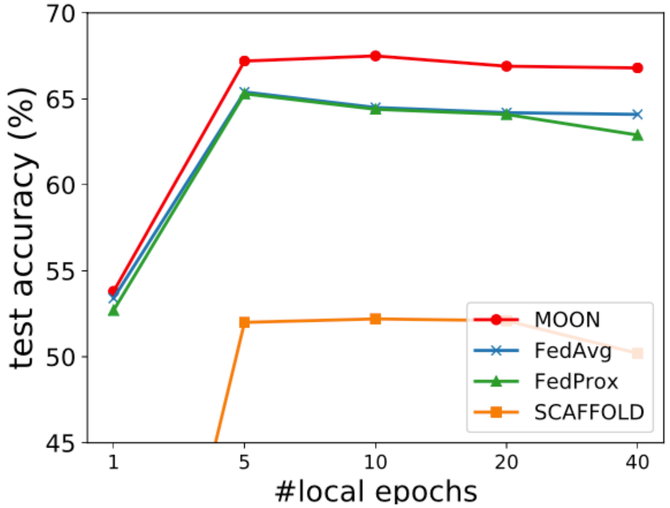
(c) Tiny-Imagenet

Experiment Results

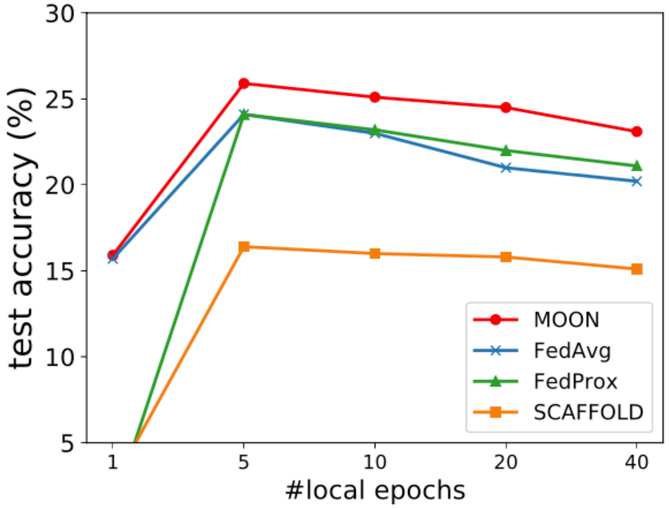
Number of Local Epochs



(a) CIFAR-10



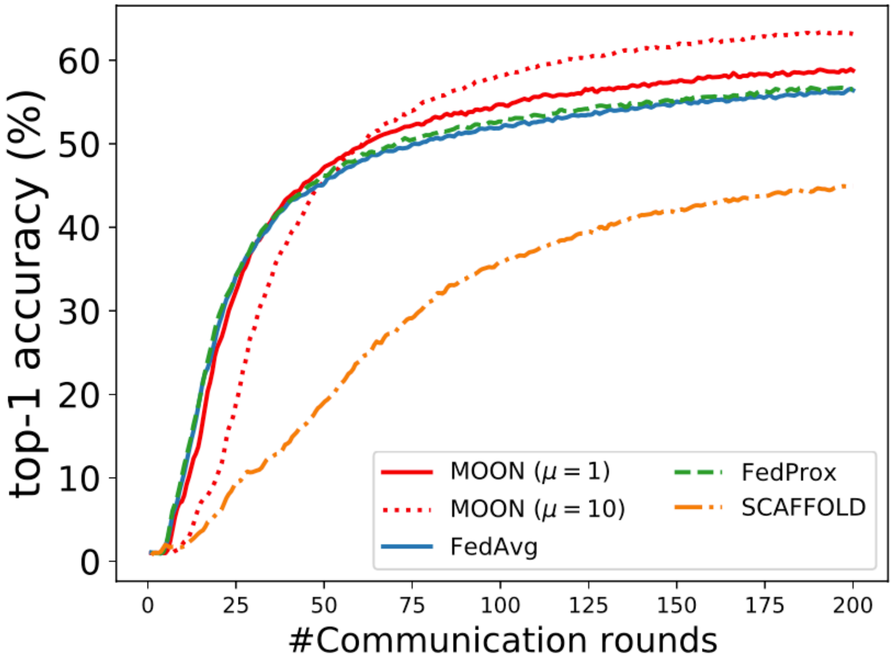
(b) CIFAR-100



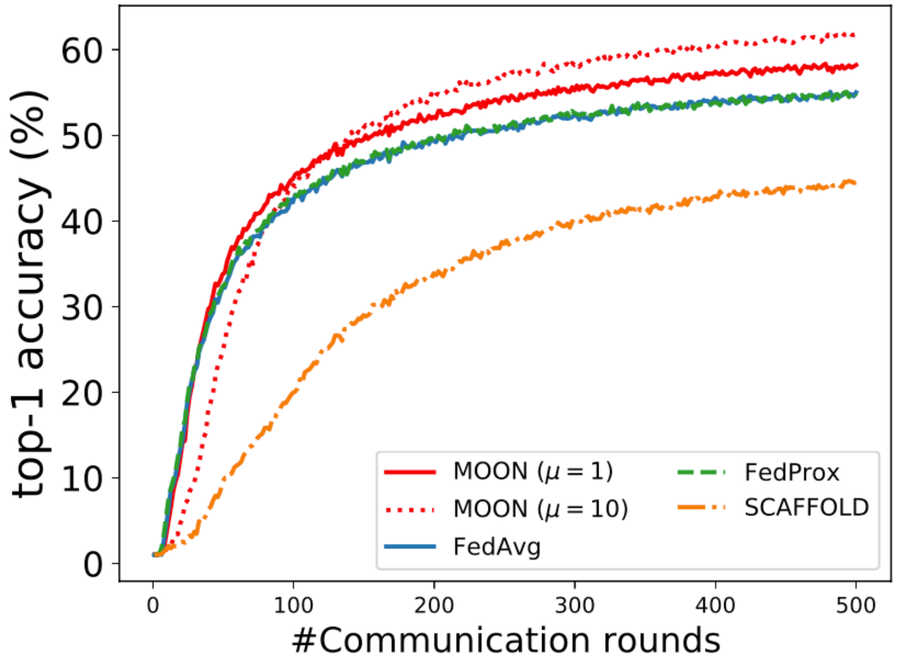
(c) Tiny-Imagenet

Experiment Results

Scalibility



(a) 50 parties



(b) 100 parties (sample fraction=0.2)

Experiment Results

Scalability, Heterogeneity, Loss Function

Method	#parties=50		#parties=100	
	100 rounds	200 rounds	250 rounds	500 rounds
MOON (μ=1)	54.7%	58.8%	54.5%	58.2%
MOON (μ=10)	**58.2%**	**63.2%**	**56.9%**	**61.8%**
FedAvg	51.9%	56.4%	51.0%	55.0%
FedProx	52.7%	56.6%	51.3%	54.6%
SCAFFOLD	35.8%	44.9%	37.4%	44.5%
SOLO	10%±0.9%		7.3%±0.6%	

Method	$\beta = 0.1$	$\beta = 0.5$	$\beta = 5$
MOON	**64.0%**	**67.5%**	**68.0%**
FedAvg	62.5%	64.5%	65.7%
FedProx	62.9%	64.6%	64.9%
SCAFFOLD	47.3%	52.5%	55.0%
SOLO	15.9%±1.5%	22.3%±1%	26.6%±1.4%

second term	CIFAR-10	CIFAR-100	Tiny-Imagenet
none (FedAvg)	66.3%	64.5%	23.0%
ℓ_2 norm	65.8%	66.9%	24.0%
MOON	**69.1%**	**67.5%**	**25.1%**

Contents

Conclusions

We propose model-contrastive learning (MOON), a simple and effective approach for federated learning.

MOON introduces a new learning concept, i.e., contrastive learning in model-level.

As MOON does not require the inputs to be images, it potentially can be applied to non-vision problems.

FedFM: Anchor-Based Feature Matching for Data Heterogeneity in Federated Learning

Rui Ye, Zhenyang Ni, Chenxin Xu, Jianyu Wang, Siheng Chen, YC Eldar

Publish Journal: IEEE Transactions on Signal Processing

Publish Time: 2023

Impact Factor: 4.6

Contents

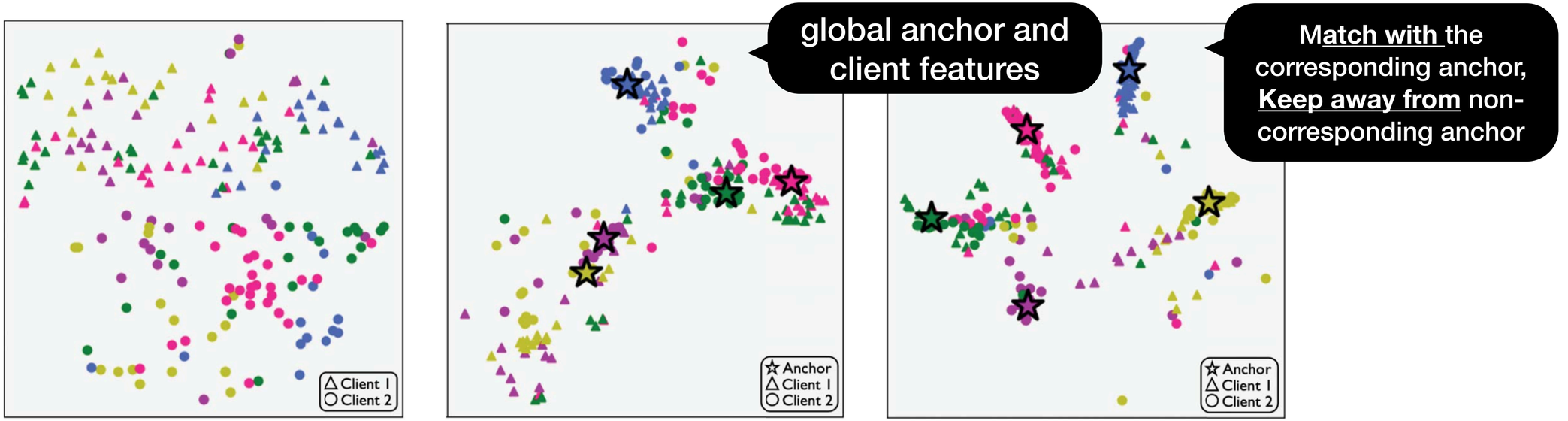
Problem Description

Challenge

Client drift: data might not be independent and identical distributed (IID) across local clients. This may result in large variations in the locally trained models on clients and slow down convergence of the global model.

Contribution

In this article, we propose an **anchor-based Federated Feature Matching** (FedFM) method, the key idea of which is to leverage landmarks shared by all clients to **provide global positioning, promoting a more consistent feature space**.



(a) FedAvg (b) FedFM with ℓ_2 regularization (c) FedFM with CG

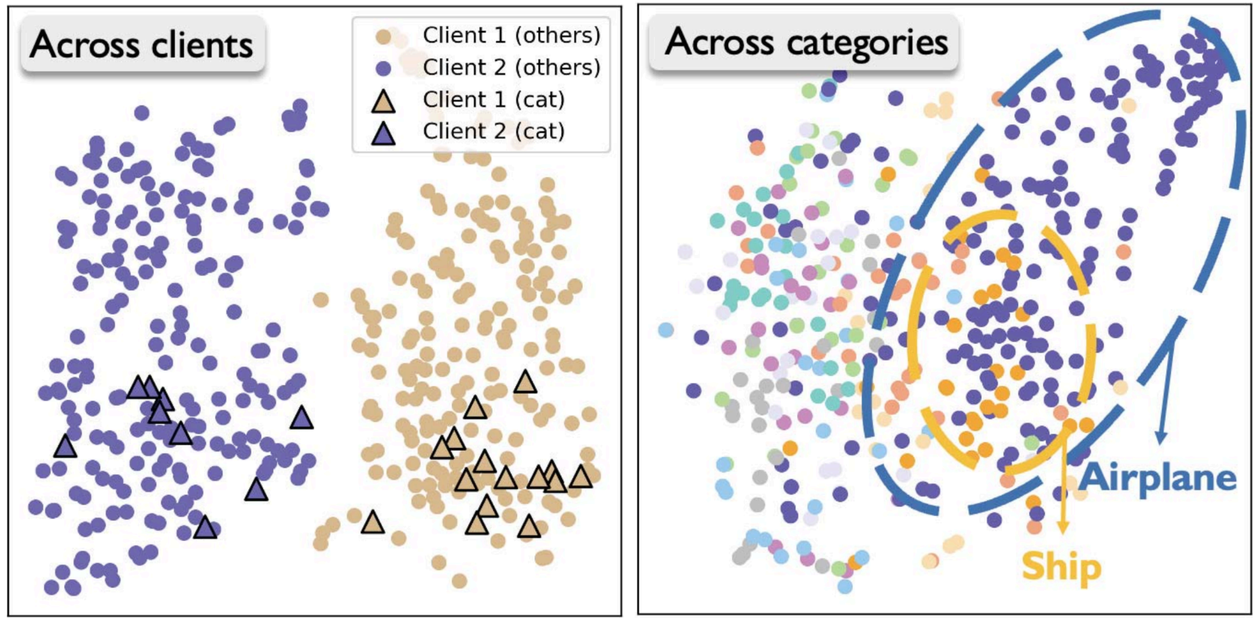
Problem Description

Motivation

Most FL methods do not explore clients' behavior in **feature space**.

We conduct the following FL experiment on CIFAR-10 with two clients.

- Client 1 has 50% of data in cat, and the rest is equally distributed to 9 other categories.



(a) Inconsistent features (b) Overlapping features

Contents

Optimization Problem

we propose anchor-based feature matching, which introduces **anchors** to serve as the shared landmarks for aligning all the clients' feature spaces

	Definition
$\mathcal{A}=\{\mathbf{a}_c\}_{c=1}^{C}$	global anchor set
$\mathbf{a}_c$	the anchor of the *cth*
p_k	predefined aggregation
$\Phi_k(\cdot)$	the k*th* client's objective
$F_k(\mathbf{w})$	The task-specific loss
$\lvert\mathcal{B}_k\rvert$	The size of kth client datasets

$$\min_{\mathbf{w},\mathcal{A}} \Phi(\mathbf{w};\mathcal{A}) = \min_{\mathbf{w},\mathcal{A}} \sum_{k=1}^{K} p_k \Phi_k(\mathbf{w};\mathcal{A})$$

$$= \min_{\mathbf{w},\mathcal{A}} \sum_{k=1}^{K} p_k \Big(F_k(\mathbf{w}) + \lambda Q_k(\mathbf{w};\mathcal{A}) \Big),$$

$$Q_k(\mathbf{w};\mathcal{A}) = \sum_{(\mathbf{x},c)\in\mathcal{B}_k} \frac{1}{|\mathcal{B}_k|} q(f_{\text{mid}}(\mathbf{w},\mathbf{x}),\mathcal{A}|c)$$

$$= \sum_{(\mathbf{x},c)\in\mathcal{B}_k} \frac{1}{|\mathcal{B}_k|} \| f_{\text{mid}}(\mathbf{w},\mathbf{x}) - \mathbf{a}_c \|_2^2$$

$$F_k(\mathbf{w}) = \sum_{(\mathbf{x},c)\in\mathcal{B}_k} \ell\left(f_{\text{full}}(\mathbf{w},\mathbf{x}),c\right) / |\mathcal{B}_k|$$

Optimization Problem — optimizing global anchors

Fixing the model parameter at the previous round, w(t), we optimize over the anchor set A by solving

$$\mathcal{A}^{(t)} = \arg\min_{\mathcal{A}} \Phi(\mathbf{w}^{(t)}; \mathcal{A}) = \sum_{k=1}^{K} p_k \Phi_k(\mathbf{w}^{(t)}; \mathcal{A}).$$

The global anchor of the *cth* category has a straightforward closed-form solution as

$$\begin{aligned}\mathbf{a}_c^{(t)} &= \arg\min_{\mathbf{a}} \sum_{k=1}^{K} \sum_{(\mathbf{x},c)\in\mathcal{B}_k} \left\| f_{\text{mid}}(\mathbf{w}^{(t)}, \mathbf{x}) - \mathbf{a}_c \right\|_2^2 \\ &= \frac{1}{\sum_{k=1}^{K} |\mathcal{B}_{k,c}|} \sum_{k=1}^{K} \sum_{(\mathbf{x},c)\in\mathcal{B}_{k,c}} f_{\text{mid}}(\mathbf{w}^{(t)}, \mathbf{x}),\end{aligned}$$

Optimization Problem — optimizing global model

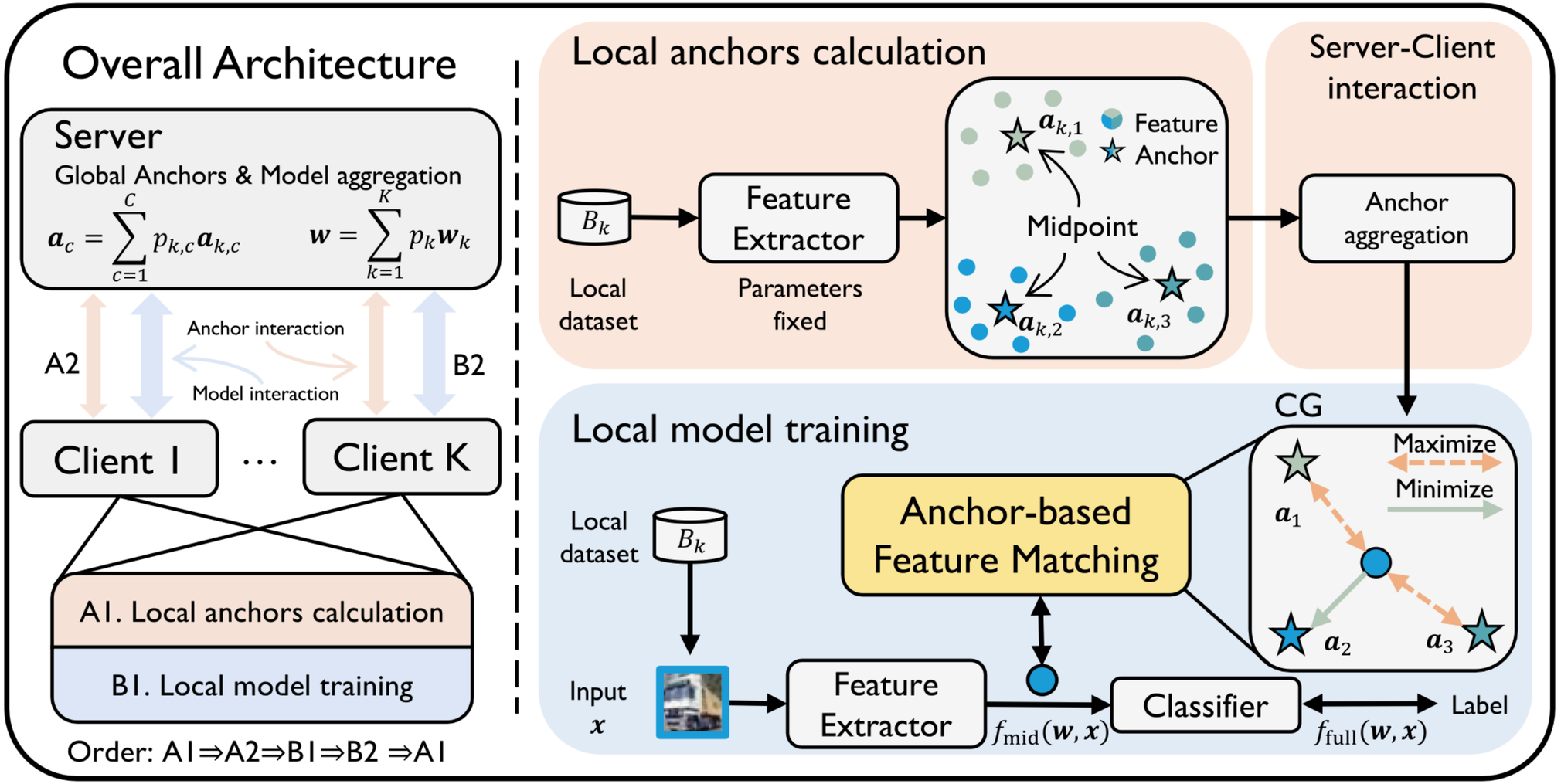
Fixing the global anchors A(t), we optimize over the model parameter w by solving

$$\mathbf{w}^{(t+1)} = \arg\min_{\mathbf{w}} \Phi(\mathbf{w}; \mathcal{A}^{(t)}) = \sum_{k=1}^{K} p_k \Phi_k(\mathbf{w}; \mathcal{A}^{(t)}).$$

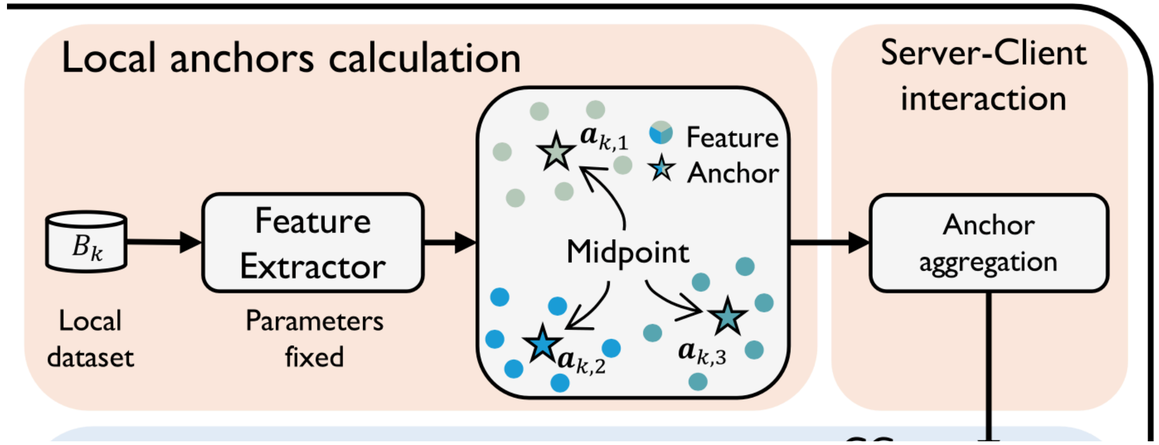
The Parameter Update Function:

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta \sum_{k=1}^{K} p_k \left(\frac{\partial F_k(\mathbf{w})}{\partial \mathbf{w}} + \lambda \frac{\partial Q_k(\mathbf{w}; \mathcal{A}^{(t)})}{\partial \mathbf{w}} \right)$$

Federated Implementation



Federated Implementation — Overview



Step 1: Anchor Update

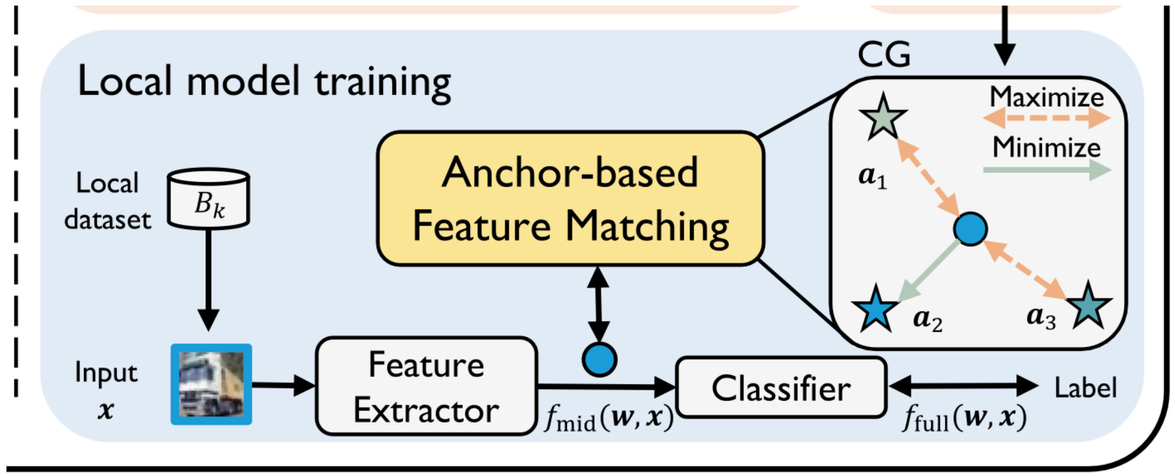
a) Local anchors calculation

$$\mathbf{a}_{k,c}^{(t)} = \frac{1}{|\mathcal{B}_{k,c}|} \sum_{(\mathbf{x},c)\in\mathcal{B}_{k,c}} f_{\text{mid}}(\mathbf{w}^{(t)}, \mathbf{x}) \in \mathbb{R}^d,$$

b) Global anchors aggregation

$$\begin{aligned}\mathbf{a}_c^{(t)} :&= \frac{1}{\sum_{k=1}^{K} |\mathcal{B}_{k,c}|} \sum_{k=1}^{K} |\mathcal{B}_{k,c}| \mathbf{a}_{k,c}^{(t)} \\ &= \frac{1}{\sum_{k=1}^{K} |\mathcal{B}_{k,c}|} \sum_{k=1}^{K} \sum_{(\mathbf{x},c)\in\mathcal{B}_{k,c}} f_{\text{mid}}(\mathbf{w}^{(t)}, \mathbf{x}) \in \mathbb{R}^d.\end{aligned}$$

Federated Implementation — Overview



Step 2: Model Updating

a) Local model training

$$\mathbf{w}_k^{(t,r+1)} = \mathbf{w}_k^{(t,r)} - \eta\left(\frac{\partial F_k(\mathbf{w})}{\partial \mathbf{w}} + \lambda \frac{\partial Q_k(\mathbf{w}; \mathcal{A}^t)}{\partial \mathbf{w}}\right),$$

b) Global model aggregation

$$\mathbf{w}^{(t+1)} \leftarrow \sum_{k=1}^{K} p_k \mathbf{w}_k^{(t,\tau)},$$

Contrastive-Guiding Loss

To address the problem of **overlapping feature** space across categories, we further propose contrastive-guiding (CG) loss to replace the $\ell 2$-based loss in the feature matching term.

⚖ The idea is to force each feature to be **close** to the corresponding anchor while **keeping far away** from non-corresponding anchors.

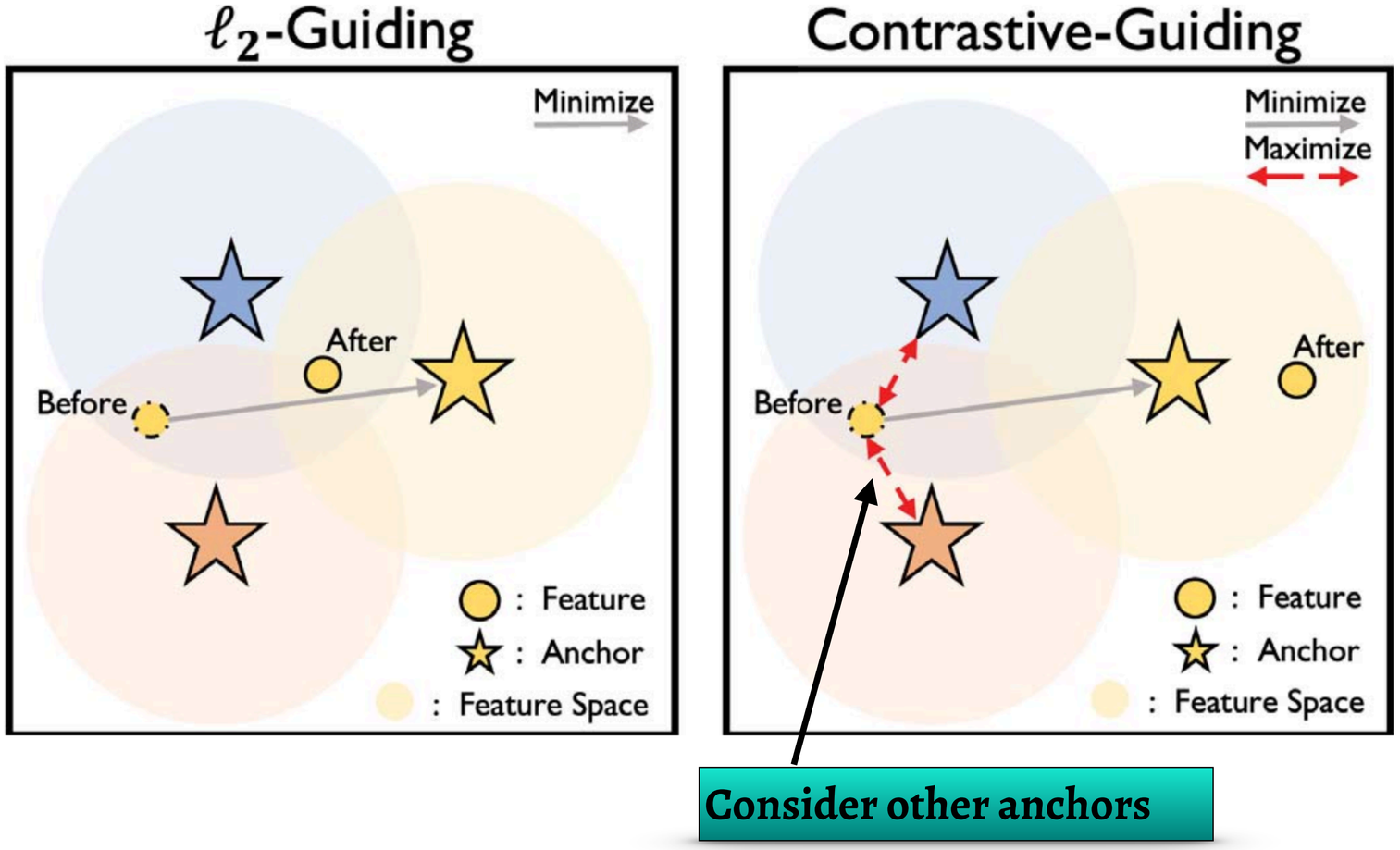
For data sample x, the feature matching loss is

$$q\left(f_{\text{mid}}(\mathbf{w}_k^{(t,r)}, \mathbf{x}), \mathcal{A}^{(t)} | c\right) = \mathcal{L}_{CE}(\mathbf{s}, c),$$

$$s_n = \frac{\exp(\langle \mathbf{a}_n^{(t)}, f_{\text{mid}}(\mathbf{w}_k^{(t,r)}, \mathbf{x})/\alpha \rangle)}{\sum_{i=1}^{C} \exp(\langle \mathbf{a}_i^{(t)}, f_{\text{mid}}(\mathbf{w}_k^{(t,r)}, \mathbf{x})/\alpha \rangle)},$$

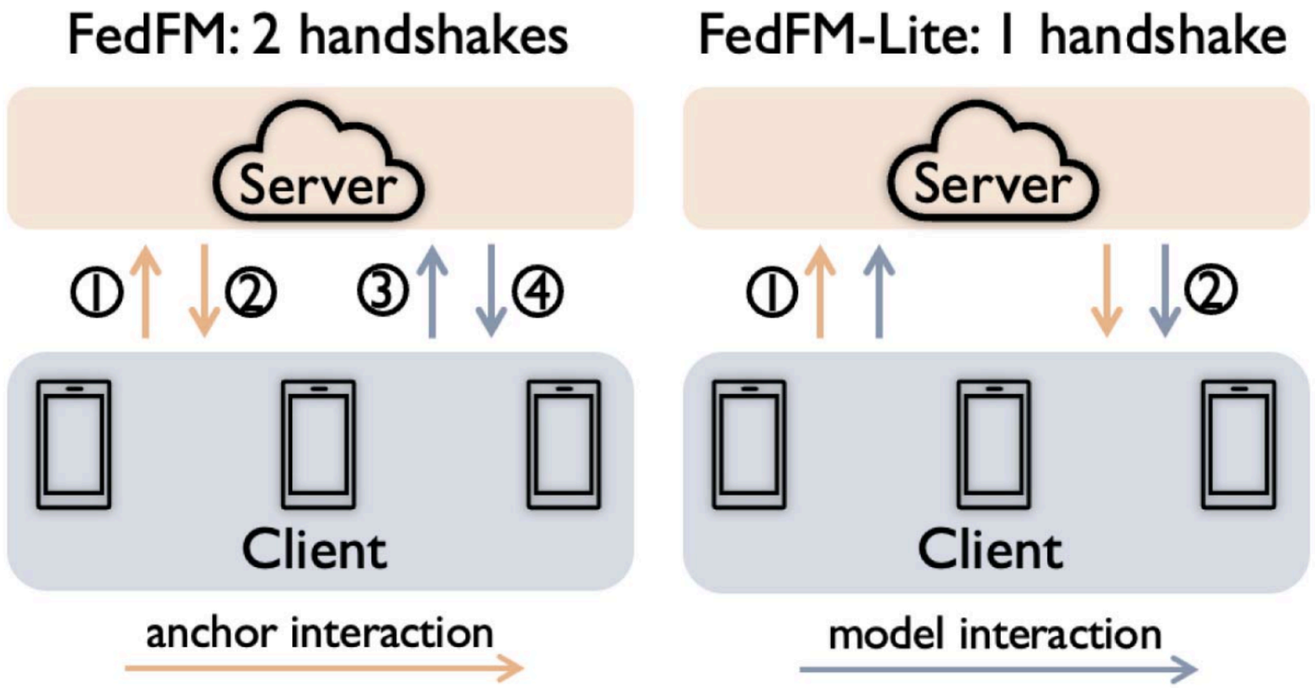
	Definition
$\mathbf{s} = [s_1, s_2, ..., s_C] \in \mathcal{A}^C$	a similarity vector
α	temperature value
$\langle \cdot, \cdot \rangle$	inner product

Contrastive-Guiding Loss



- CG can provide a more precise target
- CG can enlarge the gap across categories

FedFM-Lite



Combining Anchor and Model Communications:

The clients calculate local anchors after completing local model training. Updated model and local anchor are sent to server.

Contents

Experiment Results

Datasets

We conduct experiments on datasets including CIFAR-10, CIFAR-100, CINIC-10.

Three data heterogeneity:

A. the category distributions of clients follow a Dirichlet distribution Dir10(β)
B. each client has several dominant categories (with much more data samples) while we keep the dataset size of each client the same
C. each client has no data sample from several categories

Compared Methods

FedAvg, FedAvgM, FedProx, SCAFFOLD, FedDyn, FedNova, Moon

Quantitative Analysis

Method	CIFAR-10			CINIC-10			CIFAR-100		
	NIID-1	NIID-2	Memory	NIID-1	NIID-2	Memory	NIID-1	NIID-2	Memory
FedAvg [1]	66.69 ±0.69	69.47 ±0.48	11,182	55.96 ±0.16	58.56 ±0.22	11,182	62.16 ±0.04	62.33 ±0.27	23,705
FedAvgM [9]	66.85 ±0.42	67.87 ±0.17	11,182	56.15 ±0.45	58.79 ±0.30	11,182	61.23 ±0.12	61.30 ±0.27	23,705
FedProx [6]	66.99 ±0.26	69.42 ±0.38	22,364	55.58 ±0.13	58.32 ±0.11	22,364	61.96 ±0.05	62.20 ±0.28	47,410
SCAFFOLD [7]	69.91 ±0.54	71.48 ±0.23	22,364	58.60 ±0.27	60.78 ±0.32	22,364	67.32 ±0.29	67.24 ±0.03	47,410
FedDyn [16]	68.32 ±0.34	67.63 ±0.16	22,364	56.71 ±0.50	59.92 ±0.15	22,364	43.41 ±0.54	46.44 ±0.87	47,410
FedNova [19]	66.80 ±0.81	69.45 ±0.49	11,182	55.67 ±0.24	58.63 ±0.22	11,182	62.35 ±0.20	62.31 ±0.26	23,705
MOON [20]	67.74 ±0.30	71.09 ±0.22	33,546	57.25 ±0.07	59.28 ±0.03	33,546	62.56 ±0.22	62.99 ±0.13	71,115
FedFM (Ours)	**72.89**±0.22	**74.52**±0.21	11,187	**62.56**±0.40	**65.75**±0.46	11,187	**71.48**±0.25	**72.13**±0.45	23,909

Method	Missing 1	Missing 2	Missing 3	Missing 5	Missing 7	Memory	Bandwidth
FedAvg [1]	70.54 ±0.22	70.50 ±0.24	69.87 ±0.30	67.25 ±0.54	59.52 ±0.59	11,182	11,182
FedAvgM [9]	70.02 ±0.40	69.93 ±0.57	69.34 ±0.37	67.04 ±0.47	57.08 ±0.66	11,182	11,182
FedProx [6]	71.16 ±0.42	70.72 ±0.35	69.82 ±0.23	67.25 ±0.54	58.58 ±0.23	22,364	11,182
SCAFFOLD [7]	72.67 ±0.39	72.94 ±0.30	72.60 ±0.22	71.43 ±0.05	64.28 ±0.60	22,364	22,364
FedDyn [16]	67.43 ±0.51	67.76 ±0.64	67.78 ±0.28	69.53 ±0.59	64.75 ±0.30	22,364	11,182
FedNova [19]	70.56 ±0.25	70.48 ±0.23	70.05 ±0.15	67.56 ±0.52	59.66 ±0.42	11,182	11,182
MOON [20]	72.64 ±0.25	72.21 ±0.22	71.57 ±0.23	68.86 ±0.27	57.80 ±1.02	33,546	11,182
FedFM (Ours)	**75.97**±0.44	**75.84**±0.23	**75.04**±0.29	**73.23**±0.35	**65.24**±0.52	11,187	11,187

Qualitative Analysis

(a) FedAvg [1]

(b) FedAvgM [9]

(c) FedProx [6]

(d) SCAFFOLD [7]

(e) FedDyn [16]

(f) FedNova [19]

(g) MOON [20]

(h) **FedFM (ours)**

Qualitative Analysis

For more comprehensive comparisons, we also evaluate **the quality of feature space:**

normalized mutual information (NMI) :

silhouette score (SS)

Metric	FedAvg [1]	FedAvgM [9]	FedProx [6]	SCAFFOLD [7]	FedDyn [16]	FedNova [19]	MOON [20]	**FedFM (ours)**
NMI	0.413	0.411	0.397	0.432	0.485	0.416	0.481	**0.557**
SS	0.036	0.038	0.006	0.056	0.136	0.049	0.068	**0.173**

Experiment Results

Resource Costs

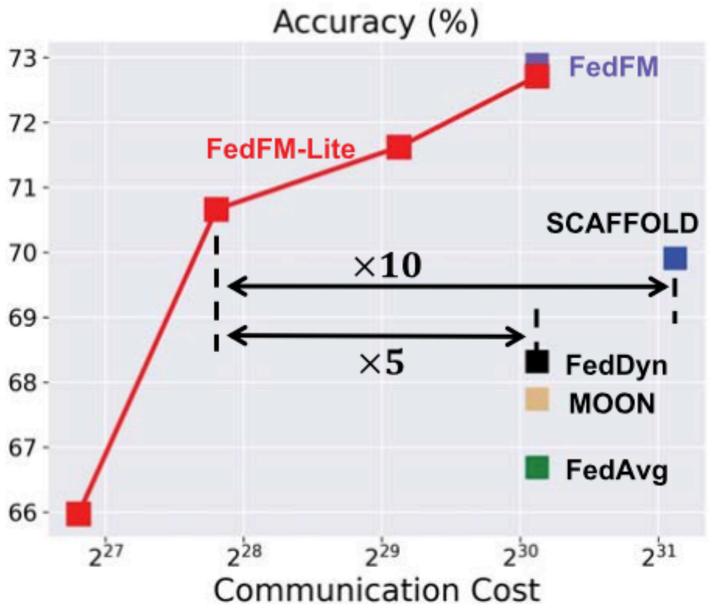
K	20	30	50	100
FedAvg [1]	58.48 (474 / 474)	54.46 (711 / 711)	50.20 (1,185 / 1,185)	41.41 (2,370 / 2,370)
FedAvgM [9]	58.36 (474 / 474)	54.48 (711 / 711)	52.86 (1,185 / 1,185)	46.72 (2,370 / 2,370)
FedProx [6]	58.27 (948 / 474)	54.50 (1,422 / 711)	50.55 (2,370 / 1,185)	40.62 (4,740 / 2,370)
SCAFFOLD [7]	64.65 (948 / 948)	61.82 (1,422 / 1,422)	56.71 (2,370 / 2,370)	47.70 (4,740 / 4,740)
FedDyn [16]	41.90 (948 / 474)	41.13 (1,422 / 711)	39.30 (2,370 / 1,185)	31.21 (4,740 / 2,370)
FedNova [19]	58.01 (474 / 474)	53.83 (711 / 711)	50.34 (1,185 / 1,185)	42.61 (2,370 / 2,370)
MOON [20]	57.63 (1,422 / 474)	52.71 (2,133 / 711)	47.84 (3,555 / 1,185)	38.45 (7,110 / 2,370)
FedFM (ours)	**69.49** (478 / 478)	**67.70** (717 / 717)	**64.22** (1,195 / 1,195)	**55.38** (2,390 / 2,390)

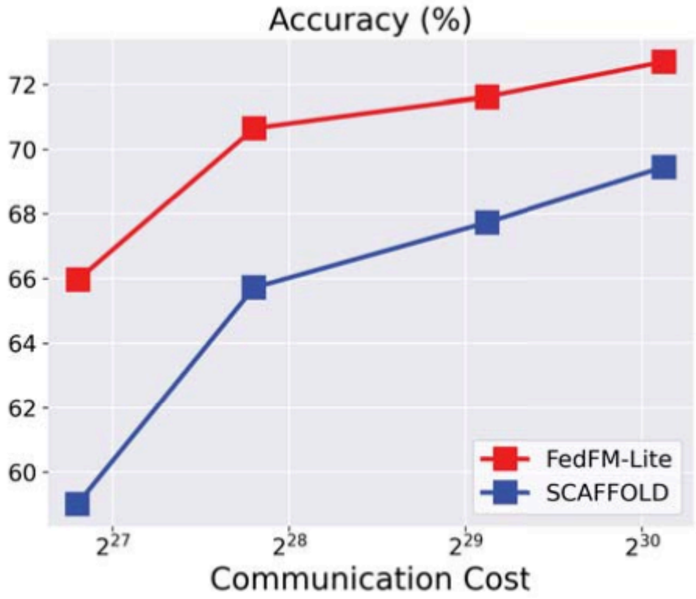
Accuracy(Memory/Bandwidth cost)

- FEDFM TAKES ONLY **0.86%** MORE **RESOURCEOVERHEAD** TO ACHIEVE **13.97%** HIGHER **ACCURACY** THAN FEDAVG; (K=100)
- FEDFM ACHIEVES **7.68%** HIGHER ACCURACY WITH ONLY **HALF THE MEMORY** AND BANDWIDTH COSTS COMPARED WITH SCAFFOLD

Experiment Results

Resources Costs

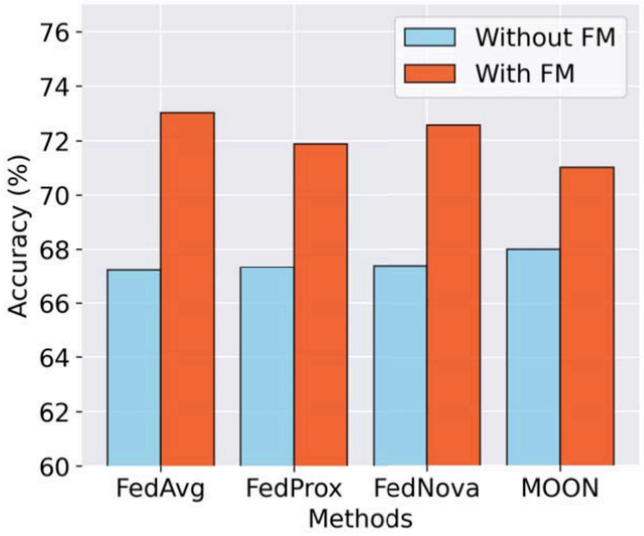




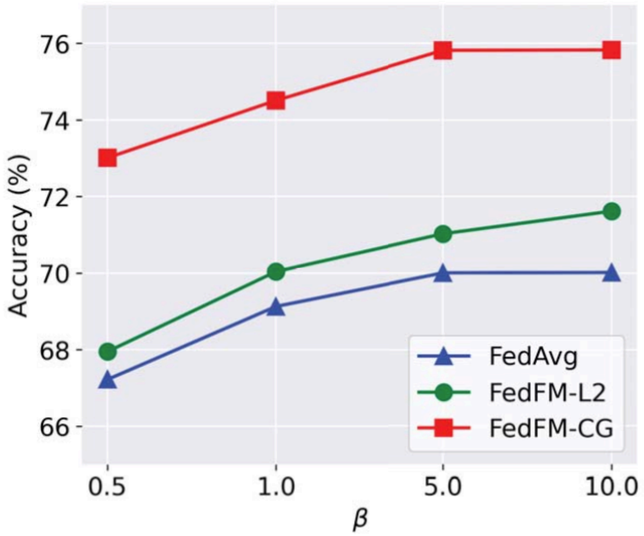
Large Scale Dataset (FEMNIST)

Model	FedAvg	SCAFFOLD	**FedFM (ours)**
ResNet18-v1	79.93	67.61	**81.62**
ResNet18-v2	81.85	74.63	**83.05**

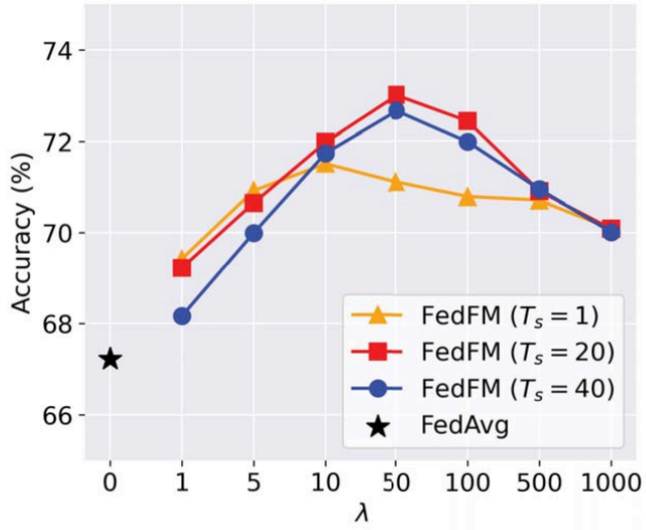
Ablation Study



(a) Modularity of FM



(b) ℓ_2-Guiding and CG



(c) T_s and λ

Experiment Results

Effects of the Weight of Feature Matching

Method	FedAvg [1]	FedFM (Weighted)	FedFM (Uniform)
Accuracy	66.69 ±0.69	72.89 ±0.22	72.87 ±0.26

Effects of Temperature

Dataset	CIFAR-10				CIFAR-100			
α	0.01	0.1	1.0	10.0	0.01	0.1	1.0	10.0
Acc	70.44	72.41	71.46	69.17	68.10	71.41	68.24	65.05

Contents

Conclusion

Facing statistical data heterogeneity, there are two unsatisfying phenomena in feature space (inconsistent features across clients and overlapping features across categories) for existing federated learning methods.

Motivated by this, we propose an **anchor-based federated feature matching** (FedFM) method, which utilizes shared anchors to guide feature learning at multiple local models, promoting a consistent feature space.

Feature matching can also be applied to multimodality (e.g., image and text) tasks to align both features of image and text.

Federated Feature Augmentation and Alignment

Tianfei Zhou, Ye Yuan, Binglu Wang, Ender Konukoglu

Publish Journal: IEEE Transactions on Pattern Analysis and Machine Intelligence

Publish Time: 2024

Impact Factor: 20.8

Contents

Problem Description

Challenge

The inherent **non-independent and identically distributed** (non-i.i.d.) nature of data distribution among clients results in significant degradation of the acquired model.

In this study, we concentrate on the **feature-level shift**, which is prevalent in numerous real-world scenarios.

FedFA+ Algorithm.

We address heterogeneous FL based on the exploration of **feature statistics**.

- **FedFA^{l}** enables more broadly exploration of the feature space.
- **FedFA^{h}** forces the consistency of augmented features across clients.

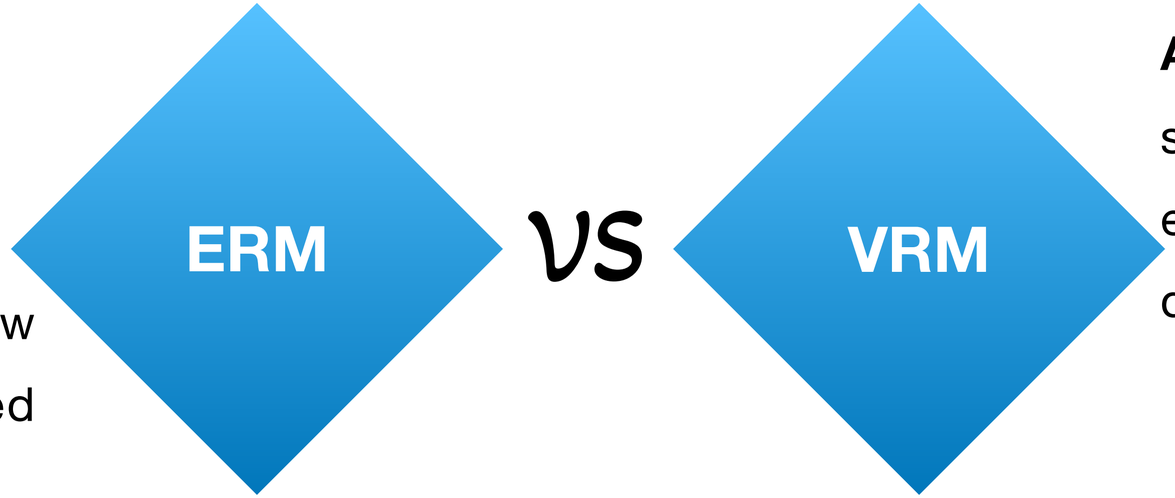
Contents

Method Design

Federated Feature Augmentation (FedFA)

DrawBack:

- Strongly depends on how well each approximated local distribution.



A vicinity distribution: novel virtual samples can be generated to enlarge the support of the local data distribution

$$\mathbb{V}_m(\hat{x}_i, \hat{y}_i | x_i, y_i)$$

approximation

$$\mathbb{P}_m \leftarrow \mathbb{P}_m^v = {}^1/_{N_m} \sum_{i=1}^{N_m} \mathbb{V}_m(\hat{x}_i, \hat{y}_i | x_i, y_i).$$

m: the index of clients

Method Design

Federated Feature Augmentation (FedFA$^{\int}$) : Probabilistic First-/Second-Order Statistic Modeling

FedFA$^{\int}$ belongs to the family of **label-preserving** feature augmentation.

- it estimates a vicinity distribution V_m^k at each convolutional layer h^k for client m;
- It performs implicit feature augmentation by manipulating **channel-wise feature statistics;**

$$\mu_m^k = \frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W} \boldsymbol{X}_m^{k,(h,w)} \quad \in \mathbb{R}^{B\times C},$$

$$\sigma_m^k = \sqrt{\frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W}(\boldsymbol{X}_m^{k,(h,w)} - \mu_m^k)^2} \quad \in \mathbb{R}^{B\times C},$$

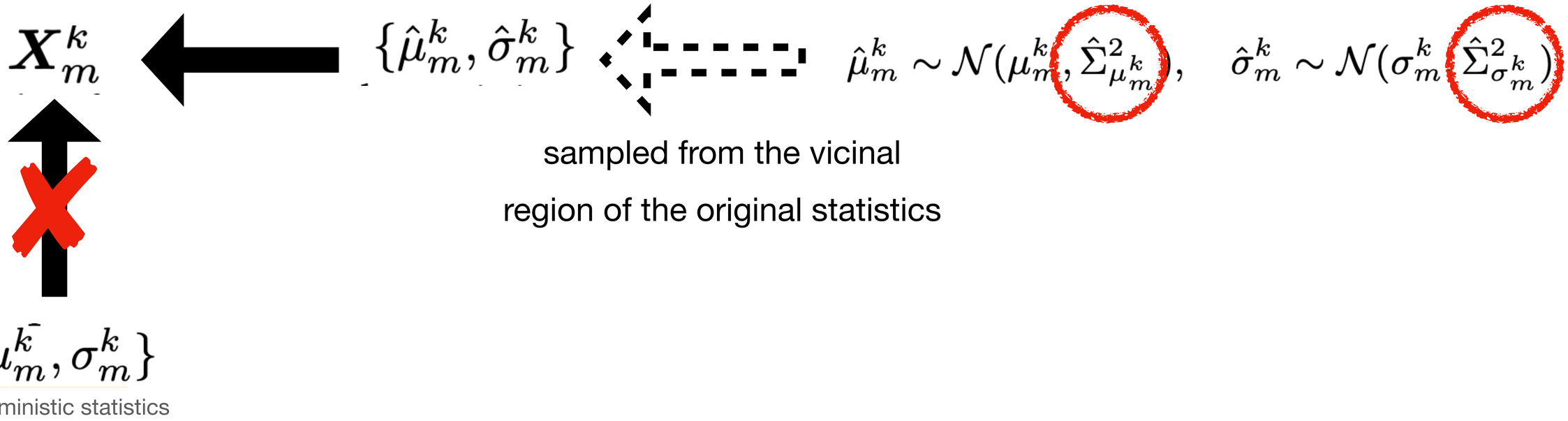
	Definition
H x W	Spatial size
C	Channel number
X_m^k	Input
Y_m^k	Ouput

m: the index of clients

Method Design

Federated Feature Augmentation (FedFA) : Probabilistic First-/Second-Order Statistic Modeling

- Feature statistics can shift across different clients;
- FedFA explicitly captures such shift via **probabilistic modeling;**



m: the index of clients, k: the index of convolutional layer

Federated Feature Augmentation (FedFA[ʃ])

Client-specific statistic variance estimation

Client-sharing Statistic Variance Estimation

Adaptive Variance Fusion

We compute client-specific variances of feature based on the information within each mini-batch

$$\Sigma^2_{\mu^k_m} = \frac{1}{B}\sum_{b=1}^{B}(\mu_m^{k,(b)} - \mathbb{E}_B[\mu_m^k])^2 \quad \in \mathbb{R}^C,$$
$$\Sigma^2_{\sigma^k_m} = \frac{1}{B}\sum_{b=1}^{B}(\sigma_m^{k,(b)} - \mathbb{E}_B[\sigma_m^k])^2 \quad \in \mathbb{R}^C,$$

where $\mu_m^{k,(b)} \in \mathbb{R}^C$ and $\sigma_m^{k,(b)} \in \mathbb{R}^C$ represent the feature mean and standard deviation of the b-th image.

Method Design

Federated Feature Augmentation (FedFA)

Client-specific statistic variance estimation

Client-sharing Statistic Variance Estimation

Adaptive Variance Fusion

We further estimate client-sharing feature statistic variances taking information of all clients into account. $\bar{\mu}^k = [\bar{\mu}_1^k, \ldots, \bar{\mu}_M^k] \in \mathbb{R}^{M \times C}$ and $\bar{\sigma}^k = [\bar{\sigma}_1^k, \ldots, \bar{\sigma}_M^k] \in \mathbb{R}^{M \times C}$ denote the collections of accumulated features stat.

$$\Sigma^2_{\mu^k} = \frac{1}{M} \sum_{m=1}^{M} (\bar{\mu}_m^k - \mathbb{E}_M[\bar{\mu}^k])^2 \quad \in \mathbb{R}^C,$$
$$\Sigma^2_{\sigma^k} = \frac{1}{M} \sum_{m=1}^{M} (\bar{\sigma}_m^k - \mathbb{E}_M[\bar{\sigma}^k])^2 \quad \in \mathbb{R}^C.$$

Modulate client-sharing estimations with *Student's t-distribution*:

$$\gamma^{(j)}_{\mu^k} = \frac{C(1 + 1/\Sigma^{2,(j)}_{\mu^k})^{-1}}{\sum_{c=1}^{C}(1 + 1/\Sigma^{2,(c)}_{\mu^k})^{-1}} \quad \in \mathbb{R},$$
$$\gamma^{(j)}_{\sigma^k} = \frac{C(1 + 1/\Sigma^{2,(j)}_{\sigma^k})^{-1}}{\sum_{c=1}^{C}(1 + 1/\Sigma^{2,(c)}_{\sigma^k})^{-1}} \quad \in \mathbb{R},$$

Modulated variances

The channels with large values are assigned with much higher importance in γ_{μ^k} than other channels.

Federated Feature Augmentation (FedFA)

Client-specific statistic variance estimation

Client-sharing Statistic Variance Estimation

Adaptive Variance Fusion

To incorporate this information for local learning, we weight the client-specific statistic variances $\{\Sigma^2_{\mu^k_m}, \Sigma^2_{\sigma^k_m}\}$ by $\{\gamma_{\mu^k}, \gamma_{\sigma^k}\}$.

$$\hat{\Sigma}^2_{\mu^k_m} = (\gamma_{\mu^k} + 1) \odot \Sigma^2_{\mu^k_m} \quad \in \mathbb{R}^C,$$
$$\hat{\Sigma}^2_{\sigma^k_m} = (\gamma_{\sigma^k} + 1) \odot \Sigma^2_{\sigma^k_m} \quad \in \mathbb{R}^C,$$

Where $\odot$ denotes the Hadamard product .

$$\gamma^{(j)}_{\mu^k} = \frac{C(1 + 1/\Sigma^{2,(j)}_{\mu^k})^{-1}}{\sum_{c=1}^{C}(1 + 1/\Sigma^{2,(c)}_{\mu^k})^{-1}} \quad \in \mathbb{R},$$
$$\gamma^{(j)}_{\sigma^k} = \frac{C(1 + 1/\Sigma^{2,(j)}_{\sigma^k})^{-1}}{\sum_{c=1}^{C}(1 + 1/\Sigma^{2,(c)}_{\sigma^k})^{-1}} \quad \in \mathbb{R},$$

Modulated variances

The channels with large values are assigned with much higher importance in γ_{μ^k} than other channels.

Federated Feature Augmentation Layer

We design a federated feature augmentation (FFA) layer to synthesize novel feature $\hat{X}_m^k$

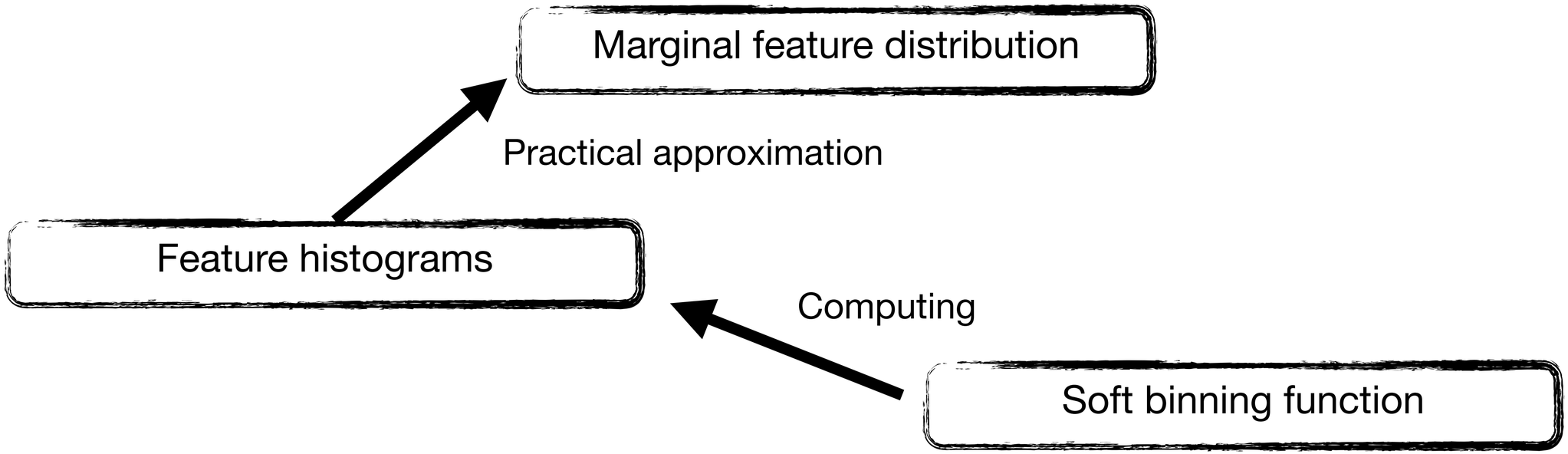
$$\hat{\boldsymbol{X}}_m^k = \mathrm{FFA}(\boldsymbol{X}_m^k) = \hat{\sigma}_m^k \frac{\boldsymbol{X}_m^k - \mu_m^k}{\sigma_m^k} + \hat{\mu}_m^k,$$

$$\text{where} \quad \hat{\mu}_m^k \sim \mathcal{N}(\mu_m^k, \hat{\Sigma}_{\mu_m^k}^2), \quad \hat{\sigma}_m^k \sim \mathcal{N}(\sigma_m^k, \hat{\Sigma}_{\sigma_m^k}^2).$$

Federated Feature Alignment (FedFA$^{\hbar}$)

Marginal Feature Distribution Approximation by High-Order Statistics

We align marginal feature distribution across different clients.



Federated Feature Alignment (FedFA^{h})

Soft binning function

We approximately parameterize p using a histogram with L normalized bin counts $\boldsymbol{\pi}_m^{z_c} = [\pi_m^{z_c,1}, \cdots, \pi_m^{z_c,L}]$.

$$\boldsymbol{\pi}_m^{z_c} = \sum_{i=1}^{N_m} \frac{\Phi(z_c^i)}{N_m}$$

Where Φ is the softmax layer, z_c^i is the transform feature.

$$\boldsymbol{\pi}_m = [\boldsymbol{\pi}_m^{z_1}; \boldsymbol{\pi}_m^{z_2}; \ldots; \boldsymbol{\pi}_m^{z_C}] \quad \in [0,1]^{LC}.$$

High-Order Feature Statistic Alignment

Global histogram:

$$\boldsymbol{\pi} = \frac{\sum_{m=1}^{M} \boldsymbol{\pi}_m}{M}$$

π is distributed back to each client to guide local model learning in the next round.

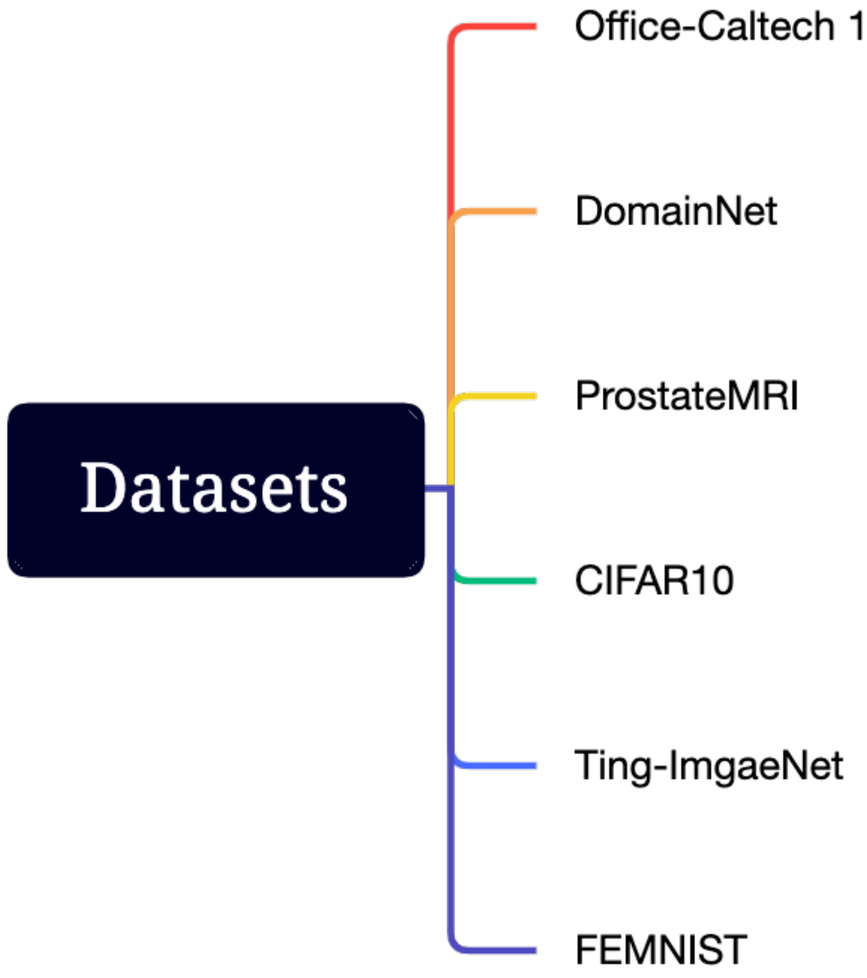
$$\mathcal{L}_m^{\text{FedFA}^h} = \frac{1}{2}(D_{\text{KL}}(\boldsymbol{\pi}_m^b \| \boldsymbol{\pi}) + D_{\text{KL}}(\boldsymbol{\pi} \| \boldsymbol{\pi}_m^b))$$

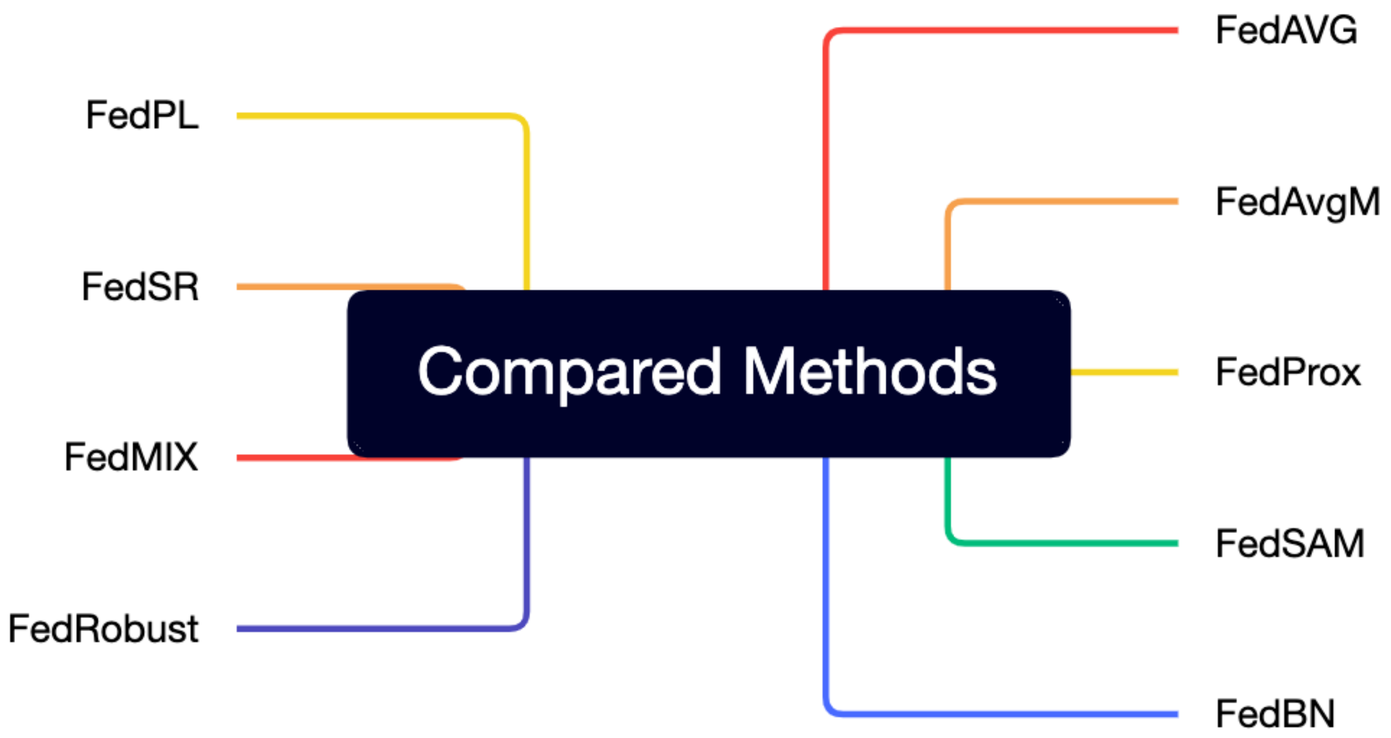
Training Loss

$$\mathcal{L} \triangleq \frac{1}{M} \sum_{m \in [M]} \mathcal{L}_m^{\text{FedFA}^l} + \lambda \mathcal{L}_m^{\text{FedFA}^h}.$$

Contents

Datasets and compared methods





FedFA-R(andom): perturbs feature statistics based on Gaussian distribution with a same standard deviation for all channels

FedFA-C(lient): performs augmentation based only on client-specific variances

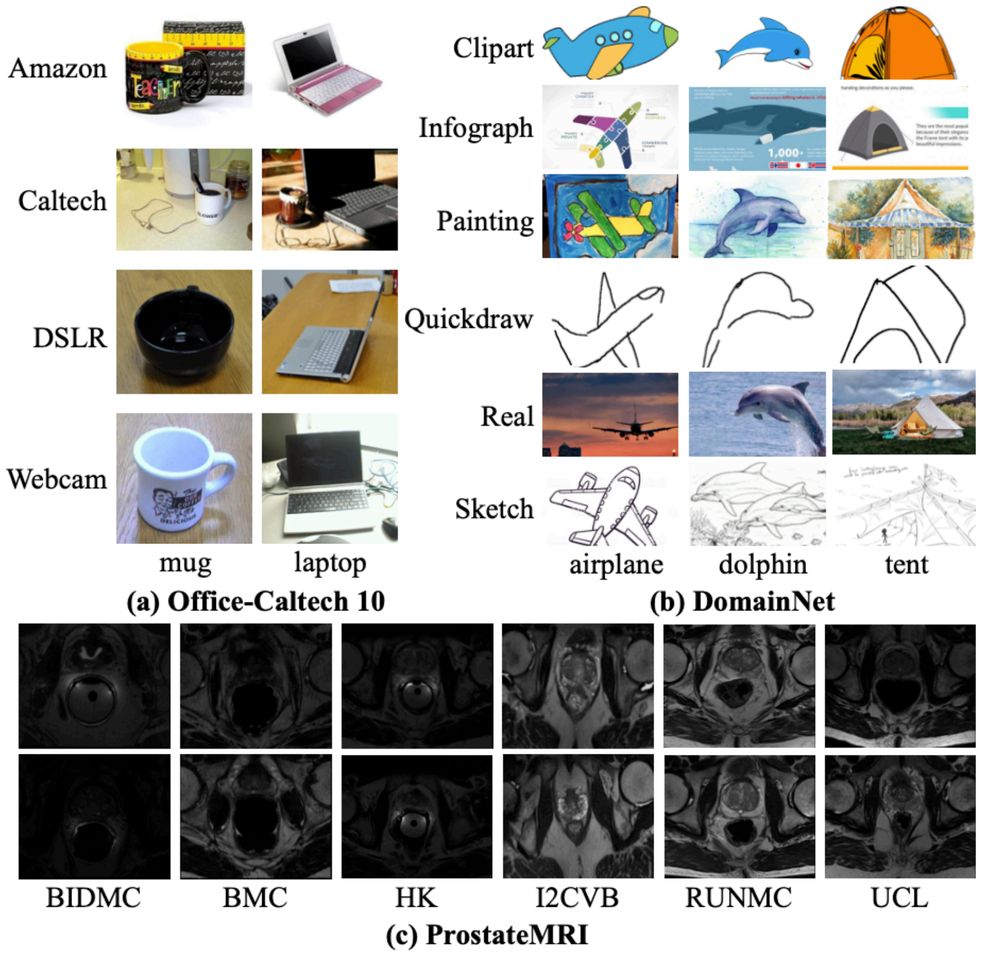
Experiment Setting

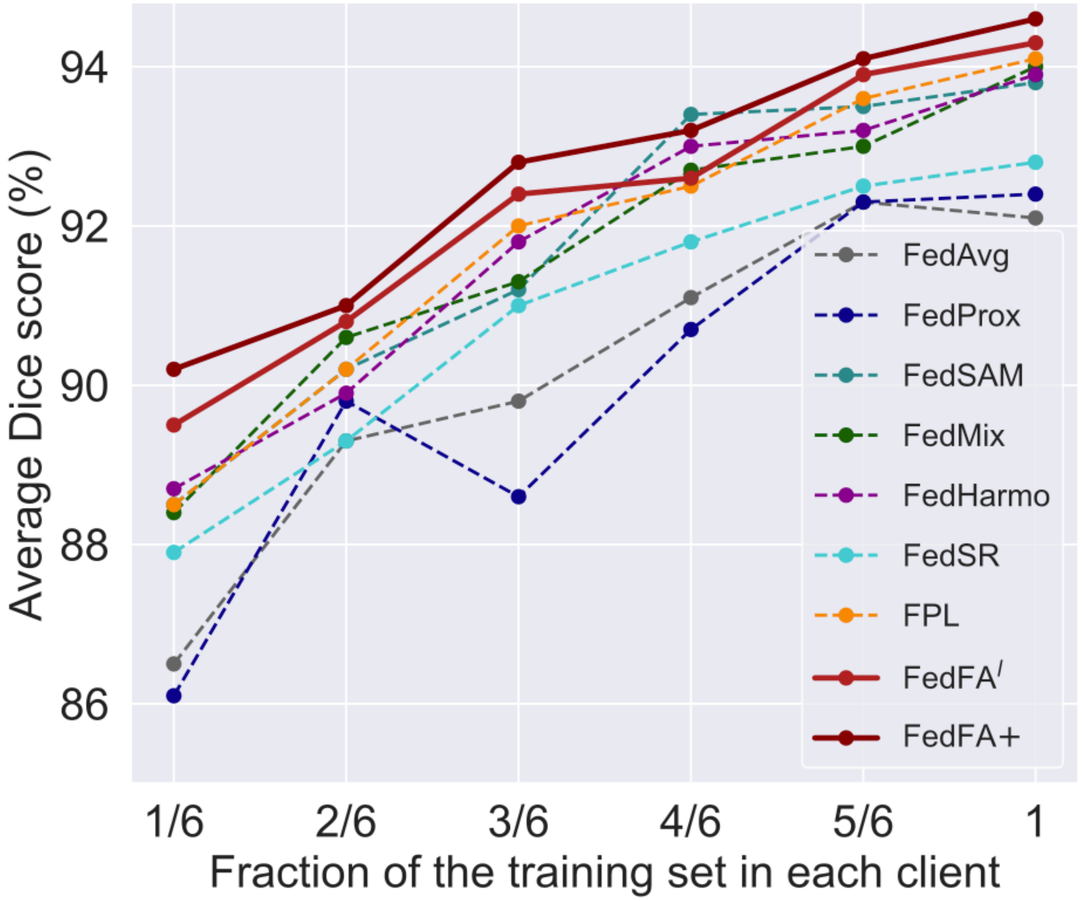
Hyper-parameters	Office	DomainNet	ProstateMRI	CIFAR-10	Tiny-ImageNet	FEMNIST
federation-aware configuration						
# Rounds	400	400	500	100	500	1000
# Epochs per round	1	1	1	10	5	5
# Clients	4	6	6	100	100	3400
# Categories	10	10	2	10	200	62
Participation rate	1.0	1.0	1.0	0.1	0.05	0.015
local client training configuration						
Network	AlxeNet	AlexNet	U-Net	ResNet	ResNet	ResNet
Optimizer	SGD	SGD	Adam	SGD	SGD	SGD
Local batch size	32	32	16	10	20	32
Local learning rate	1e-2	1e-2	1e-4	1e-1	1e-1	5e-2

Split	Office-Caltech 10 [96]				DomainNet [97]						ProstateMRI [98]					
	Amazon	Caltech	DSLR	Webcam	Clipart	Infograph	Painting	Quickdraw	Real	Sketch	BIDMC	HK	I2CVB	BMC	RUNMC	UCL
`train`	459	538	75	141	672	840	791	1280	1556	708	156	94	280	230	246	105
`val`	307	360	50	95	420	525	494	800	972	442	52	31	93	76	82	35
`test`	192	225	32	59	526	657	619	1000	1217	554	52	31	93	76	82	35

Experiment Results

Segmentation performance on ProstateMRI





Experiment Results

Feature visualization

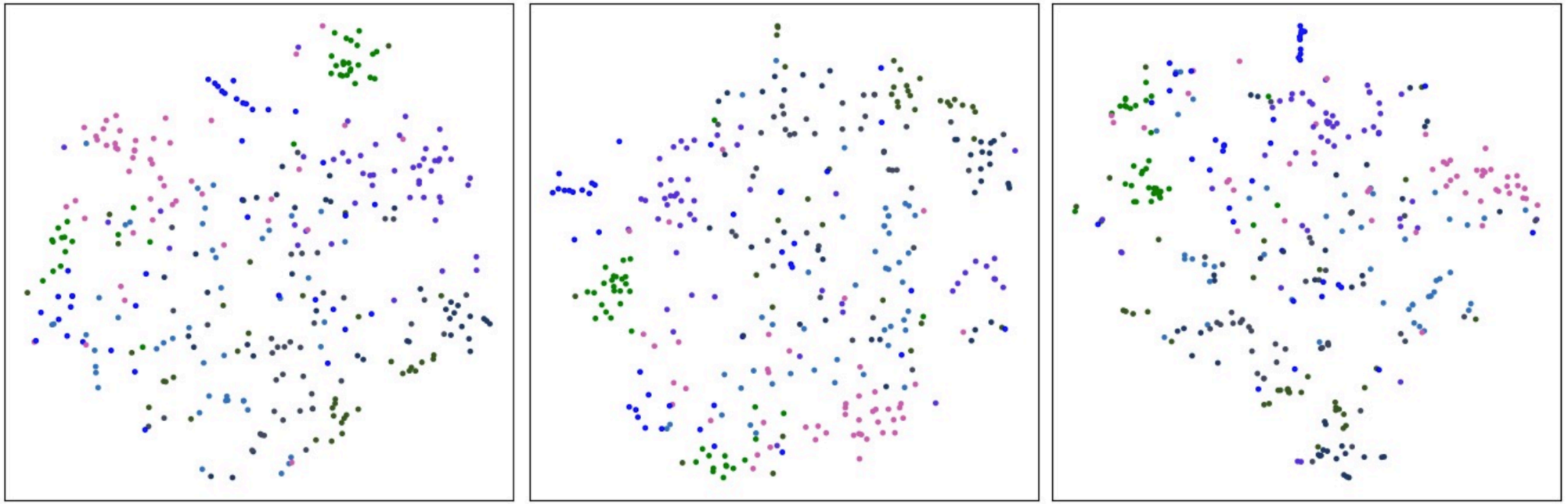


Image classification results

Algorithm	Office-Caltech 10 [96]					DomainNet [97]						
	A	C	D	W	Average	C	I	P	Q	R	S	Average
FedAvg [AISTAT17] [5]	84.4	66.7	75.0	88.1	78.5	71.5	33.2	57.8	76.5	72.9	65.2	62.8
FedAvgM [arXiv19] [102]	85.9	64.0	71.9	94.9	79.2	79.8	33.3	58.8	72.6	72.8	66.1	62.5
FedProx [MLSys20] [42]	84.9	64.0	78.1	88.1	78.8	70.9	32.9	61.2	74.1	71.1	67.9	63.0
FedRobust [NeurIPS20] [15]	82.3	64.0	81.3	93.2	80.2	70.9	32.9	60.7	75.7	72.6	68.5	63.6
FedBN [ICLR20] [14]	82.3	63.6	81.2	94.9	80.5	72.4	32.7	64.3	74.0	69.9	70.8	64.0
FedMix [ICLR21] [28]	81.7	63.1	81.3	93.2	79.8	75.9	34.1	61.7	73.8	69.4	70.6	64.3
FedSAM [ICML22] [18]	81.7	63.1	50.0	81.4	69.1	60.1	30.1	53.0	64.8	61.9	47.3	52.9
FedSR [NeurIPS22] [19]	86.7	65.3	79.4	88.1	79.9	73.1	32.8	60.3	73.0	73.7	68.2	63.5
FPL [CVPR23] [23]	86.9	68.8	81.0	94.5	82.8	74.5	35.6	61.3	76.4	75.4	71.1	65.7
FedFA+	90.5	68.8	90.9	88.5	**84.8**	78.6	37.5	60.2	78.5	74.1	75.9	**67.5**
FedFA+ (Swin-T)	91.8	71.0	91.6	89.1	**85.9**	80.3	40.6	62.0	79.8	75.8	77.1	**69.3**

Algorithm	BIDMC (32)	HK (32)	I2CVB (46)	BMC (38)	RUNMC (41)	UCL (32)	Average
FedAvg [AISTAT17] [5]	81.2	90.8	86.1	84.0	91.0	86.2	86.5
FedAvgM [arXiv19] [102]	80.3	91.6	88.2	82.2	91.2	86.5	86.7
FedProx [MLSys20] [42]	82.8	89.1	89.8	79.5	89.8	85.6	86.1
FedRobust [NeurIPS20] [15]	81.7	91.3	91.5	88.5	89.4	84.2	87.7
FedBN [ICLR20] [14]	88.9	92.3	90.6	88.1	87.6	85.4	88.8
FedMix [ICLR21] [28]	86.3	91.6	89.6	88.1	89.8	85.2	88.4
FedSAM [ICML22] [18]	82.7	92.5	91.8	83.6	92.6	88.1	88.5
FedSR [NeurIPS22] [19]	80.1	92.8	88.9	84.0	92.0	89.5	87.9
FedHarmo [AAAI22] [16]	86.7	91.6	92.7	84.2	92.5	84.6	88.7
FPL [CVPR23] [23]	82.4	91.6	91.3	85.4	93.7	86.5	88.5
FedFA+	89.1	92.8	90.1	89.0	91.9	88.4	**90.2**
FedFA+ (Swin-T)	89.8	93.1	92.2	90.3	92.7	89.6	**91.3**

Experiment Results

Heterogeneous Environments

Algorithm	CIFAR-10 [99]		Tiny-ImageNet [40]		FEMNIST [38]
	Dir (0.6)	Dir (0.3)	Dir (0.6)	Dir (0.3)	
FedAvg [AISTAT17] [5]	73.3	69.2	33.9	31.8	78.5
FedAvgM [arXiv19] [102]	73.4	69.1	36.3	34.2	78.8
FedProx [MLSys20] [42]	74.0	69.5	34.3	32.3	78.4
FedBN [ICLR20] [14]	73.7	69.8	35.6	33.5	79.1
FedRobust [NeurIPS20] [15]	74.9	70.5	35.8	33.1	79.3
FedSAM [ICML22] [18]	74.3	70.0	37.2	35.4	79.2
FedMix [ICLR21] [28]	75.5	70.7	36.8	34.9	79.7
FedFA+	**77.3**	**72.6**	**38.8**	**36.2**	**80.5**
FedFA+ (Swin-T)	**81.6**	**76.1**	**39.6**	**37.1**	**81.6**

Analysis of essential components in FedFA+

Variant	Office [96]	DomainNet [97]	ProstateMRI [98]	CIFAR-10 [99] Dir (0.6)	CIFAR-10 [99] Dir (0.3)	Tiny-ImageNet [40] Dir (0.6)	Tiny-ImageNet [40] Dir (0.3)	FEMNIST [38]
FedAvg [5]	78.5	62.8	86.5	73.3	69.2	33.9	31.8	78.5
FedFA^l	83.1	66.5	89.5	76.3	71.9	36.5	34.9	79.6
FedFA^h	78.9	63.0	87.0	74.2	69.9	35.0	33.1	79.0
FedFA+	**84.8**	**67.5**	**90.2**	**77.3**	**72.6**	38.8	36.2	**80.5**
FedAvg (Swin-T) [5]	80.1	64.1	87.5	77.1	72.8	35.0	32.7	79.8
FedFA^l (Swin-T)	84.5	67.7	90.6	80.1	75.1	37.3	35.4	81.1
FedFA^h (Swin-T)	81.1	64.5	88.1	78.0	73.6	35.8	34.0	80.4
FedFA+ (Swin-T)	**85.9**	**69.3**	**91.3**	**81.6**	**76.1**	39.6	37.1	**81.6**

Impacts of different normalization methods

Normalization	Office [96]	DomainNet [97]	ProstateMRI [98]
GroupNorm	82.6	66.0	88.6
LayerNorm	82.8	66.7	89.1
BatchNorm (*default*)	83.1	66.5	89.5

Comparison between FedFAC and FedFAR

Variant	Office [96]	DomainNet [97]	ProstateMRI [98]
FedAvg [5]	78.5	62.8	86.5
FedFA-R	78.6	61.0	86.1
FedFA-C	79.5	63.7	87.8
FedFA^l	**83.1**	**66.5**	**89.5**

Efficacy of FedFa against conventional augmentation tech.

Algorithm	Office [96]	DomainNet [97]	ProstateMRI [98]
FedAvg [5]	78.5	62.8	86.5
Mixup [52]	79.2	63.4	87.0
M-Mixup [67]	79.6	63.5	87.6
MixStyle [34]	79.9	64.1	88.5
MoEx [35]	80.2	64.6	88.3
DAC-SC [108]	82.1	65.2	88.9
FedFA^l	**83.1**	**66.5**	**89.5**
FedFA^l+Mixup [52]	83.7	67.0	89.9
FedFA^l+M-Mixup [67]	83.6	66.9	90.2
FedFA^l+MixStyle [34]	84.0	67.2	90.2
FedFA^l+MoEx [35]	83.9	67.0	90.1
FedFA^l+DAC-SC [108]	84.5	67.7	90.8

Ablation Study

Analysis of different fusion strategies

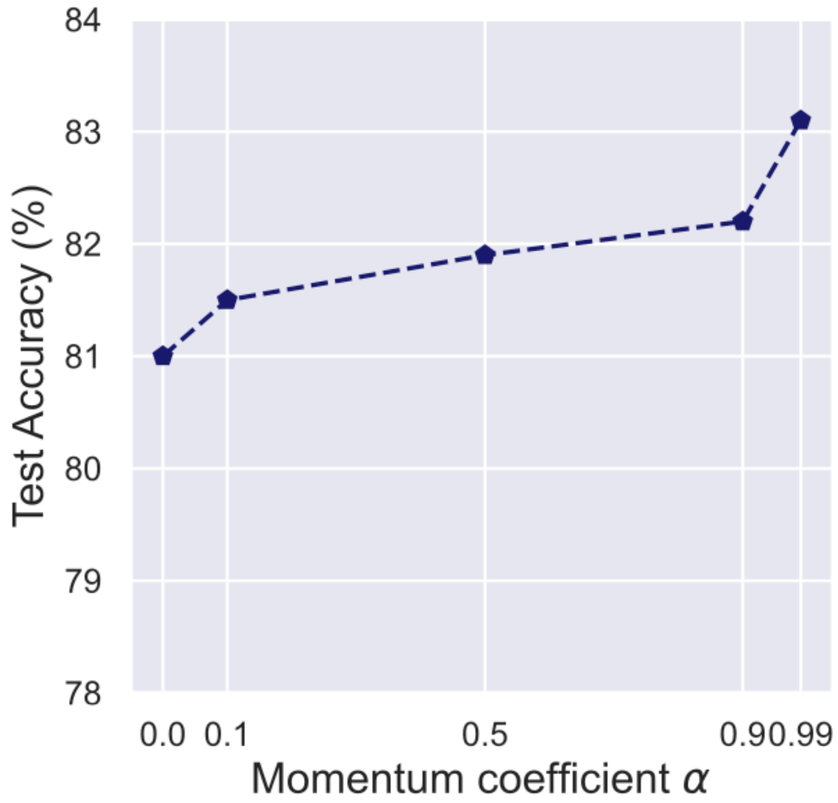
Variant	Office [96]	DomainNet [97]	ProstateMRI [98]
Direct Fusion	80.6	64.1	86.9
Adaptive Fusion	83.1	66.5	89.5

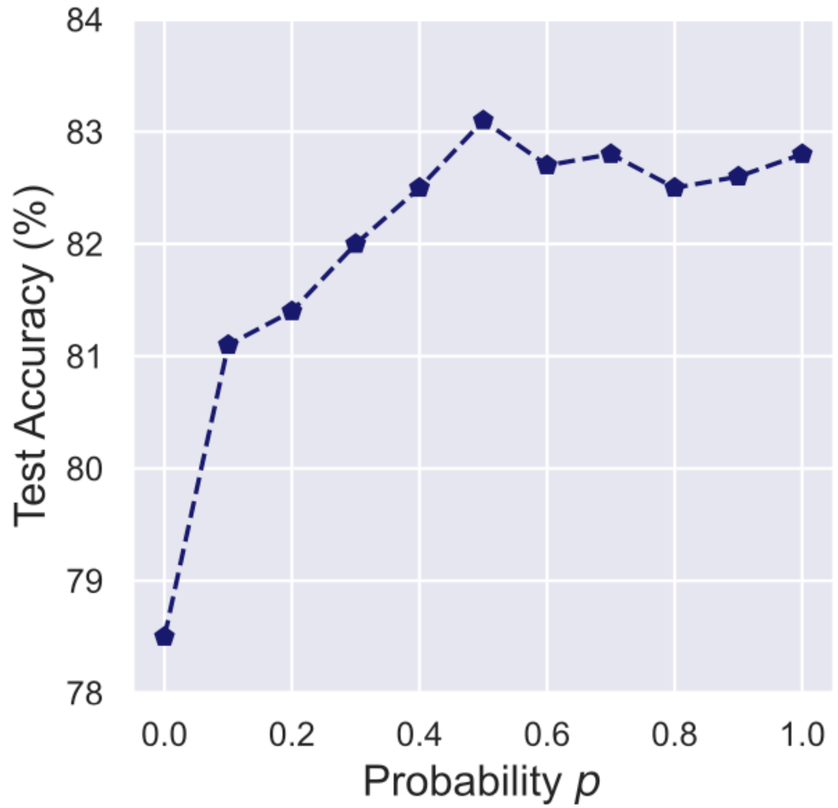
Different architecture using FFA

Variant	Office [96] (AlexNet)	DomainNet [97] (AlexNet)	DomainNet [97] (R50)	DomainNet [97] (Dense100)	ProstateMRI [98] (U-Net)
FedAvg	78.5	62.8	74.8	70.7	86.5
$\{1\}$	78.8	63.5	75.3	71.0	88.5
$\{1,2\}$	80.0	63.9	75.6	71.5	88.6
$\{1,2,3\}$	80.0	64.0	75.9	71.7	89.0
$\{1,2,3,4\}$	80.6	64.3	76.1	72.0	88.8
$\{1,2,3,4,5\}$	**83.1**	**66.5**	**77.2**	**73.4**	**89.5**
$\{2,3,4,5\}$	81.6	65.2	76.7	72.8	88.6
$\{1,2,4,5\}$	82.0	65.8	76.9	72.9	88.8
$\{3,4,5\}$	78.4	63.8	75.8	71.6	87.0
$\{4,5\}$	79.4	64.7	75.6	71.5	85.9
$\{5\}$	79.2	64.6	75.7	71.3	86.3
$\{1,5\}$	80.4	65.5	76.1	71.9	88.5
$\{2,3,4\}$	79.5	64.3	75.5	71.4	88.8
$\{2,3\}$	78.7	64.0	75.3	71.2	88.5
$\{3,4\}$	78.3	63.2	75.1	70.9	86.5
$\{3\}$	78.0	63.1	74.9	70.8	86.5

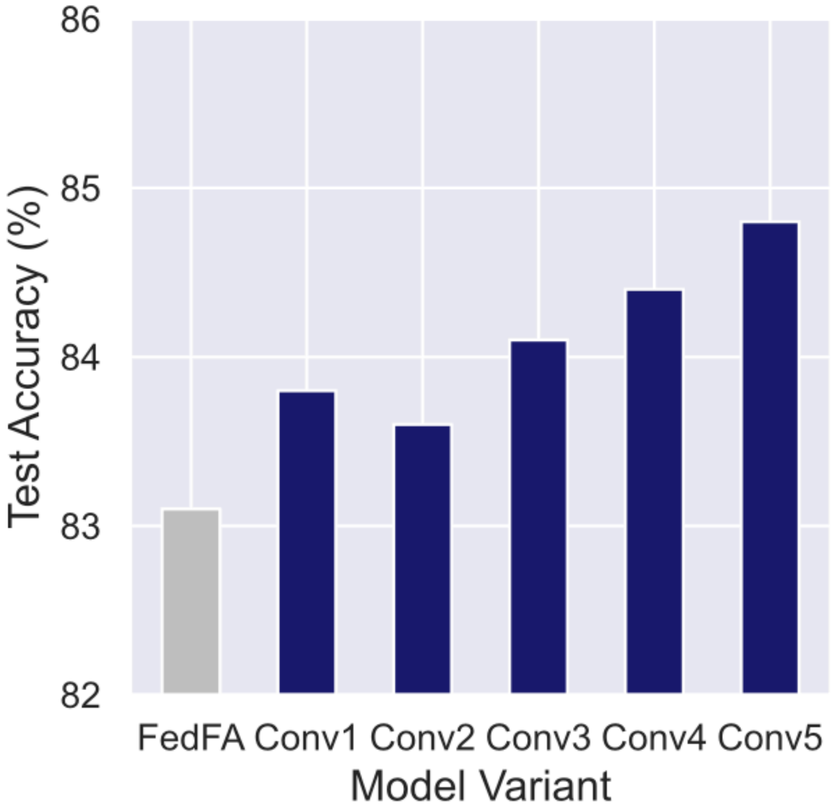
Experiment Results

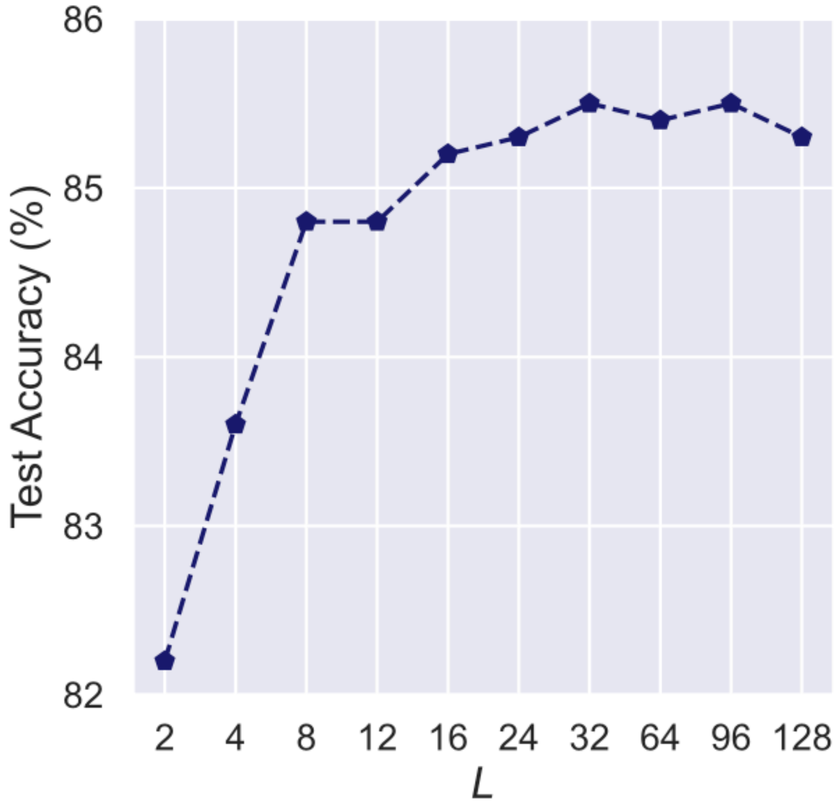
Hyper-parameter analysis





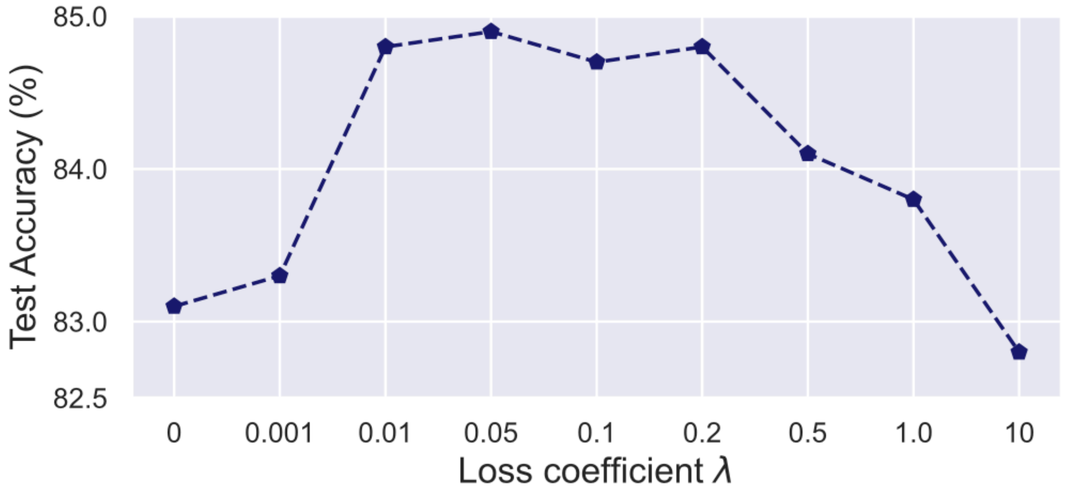
Ablation Study of Federated Feature Alignment

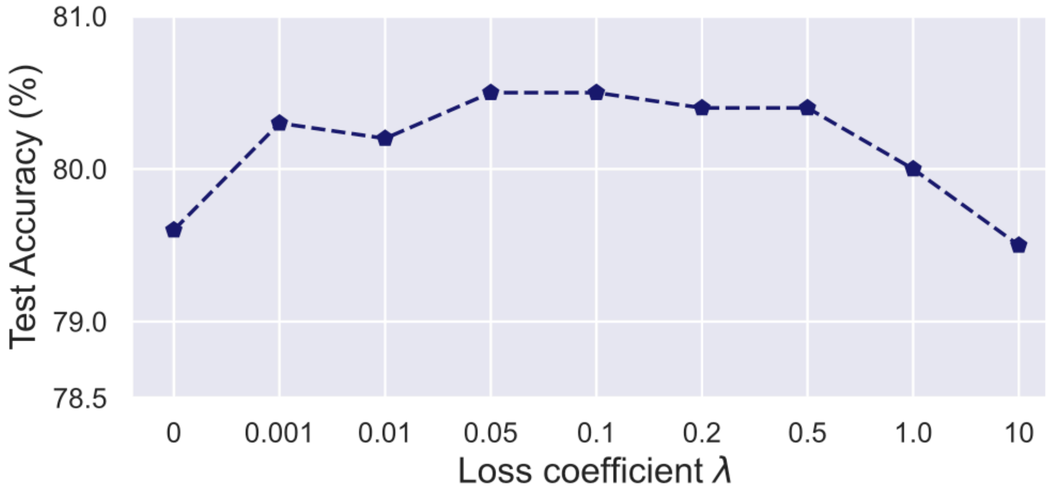




Experiment Results

Hyper-parameter analysis





Contents

Conclusion

In this work, we present **FedFA+** for addressing heterogeneity in federated learning. It is unique in exploiting **statistical information** of latent features.

Federated feature augmentation (FedFA$^{\ell}$), which models low-order feature statistics;

Federated feature alignment (FedFA^{h}), which approximates latent features using higher-order statistics and aligns them across clients to reduce discrepancies between clients.