JUNE 18-22, 2023
CVPR
VANCOUVER, CANADA

2nd Workshop on Federated Learning for Computer Vision

Federated Learning in Non-IID Settings Aided by Differentially Private Synthetic Data

Huancheng Chen, Haris Vikalo

University of Texas at Austin

Presenter: Huancheng Chen

Outline

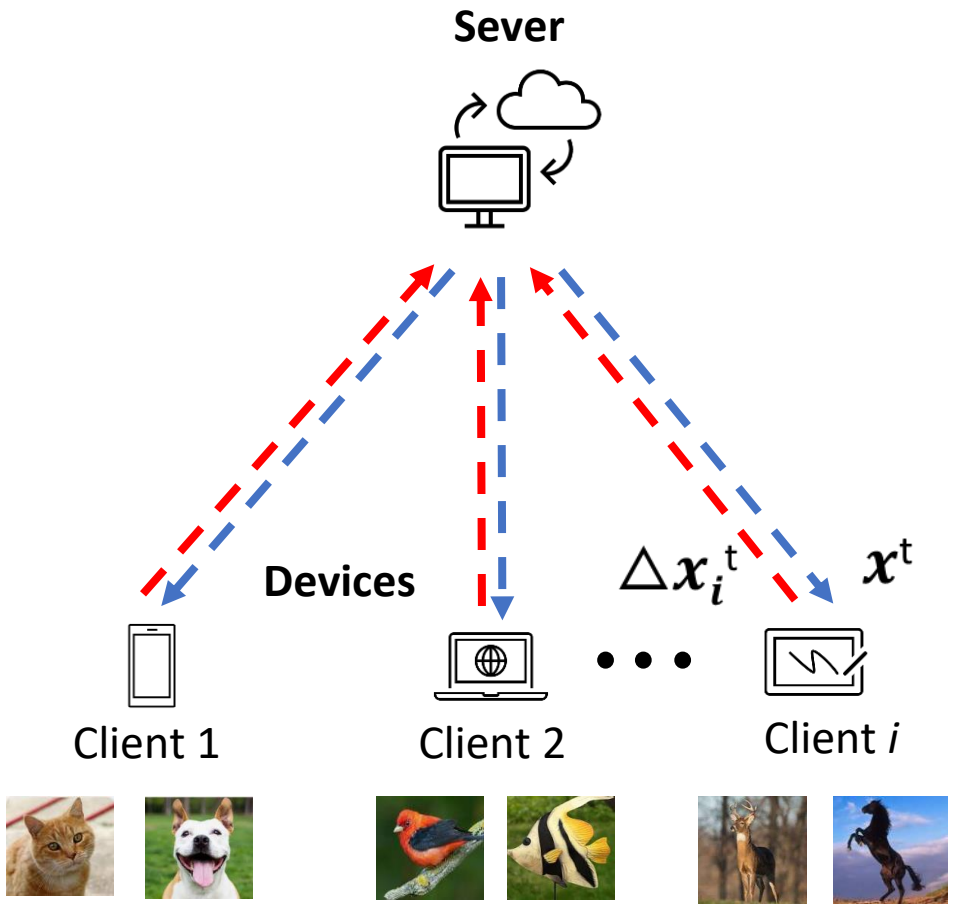
- Background and Motivation
- The Proposed Method
- Experimental Results
- Q&A

Background:

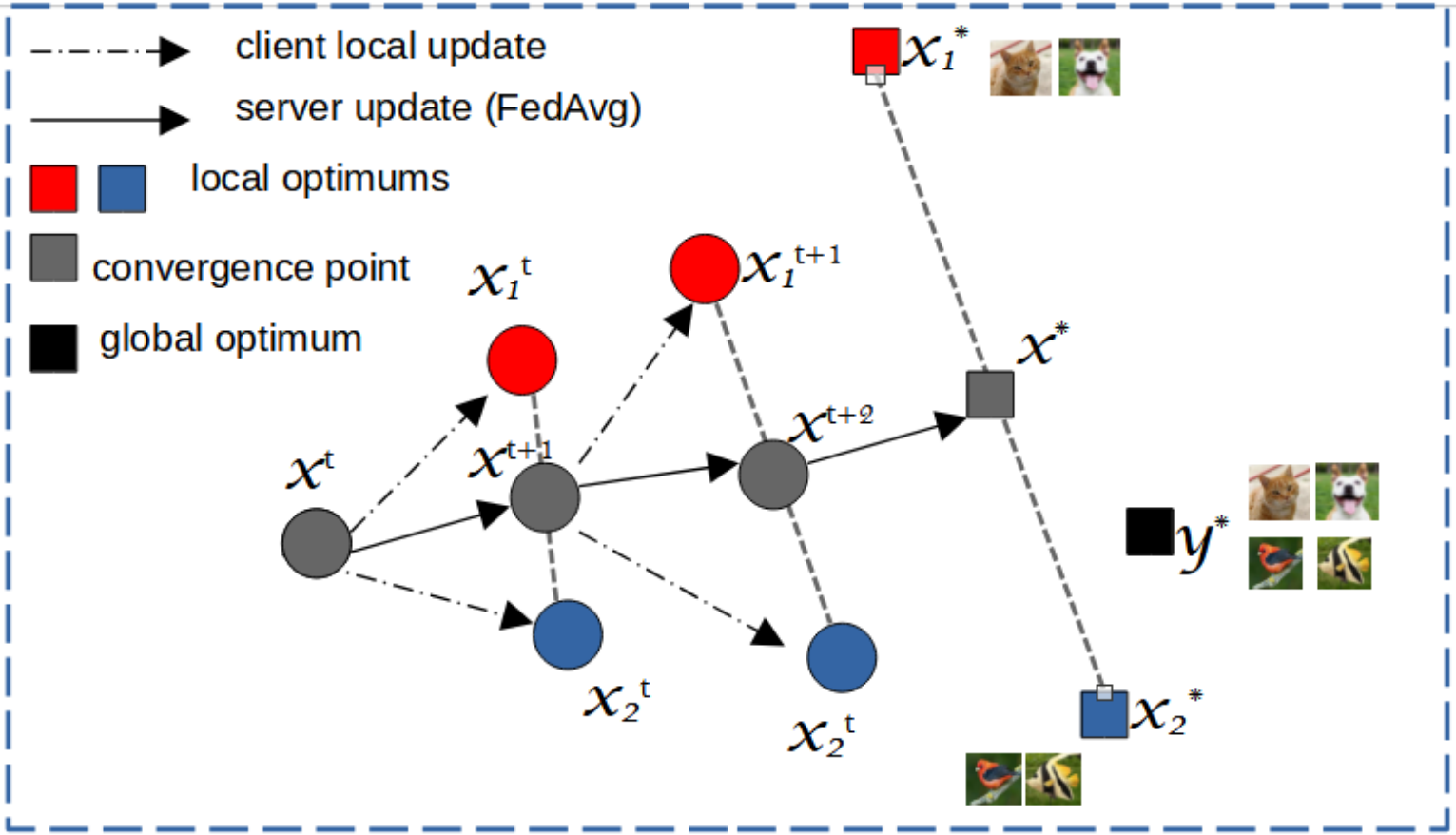
- Federated Learning is a privacy-preserving framework that enables numerous distributed devices to collaboratively train a global model without exchanging their private data.

Challenge:

- Training on non-IID data has detrimental effects on the performance of the global model.



Non-IID data causes objective drift: A two-client system example

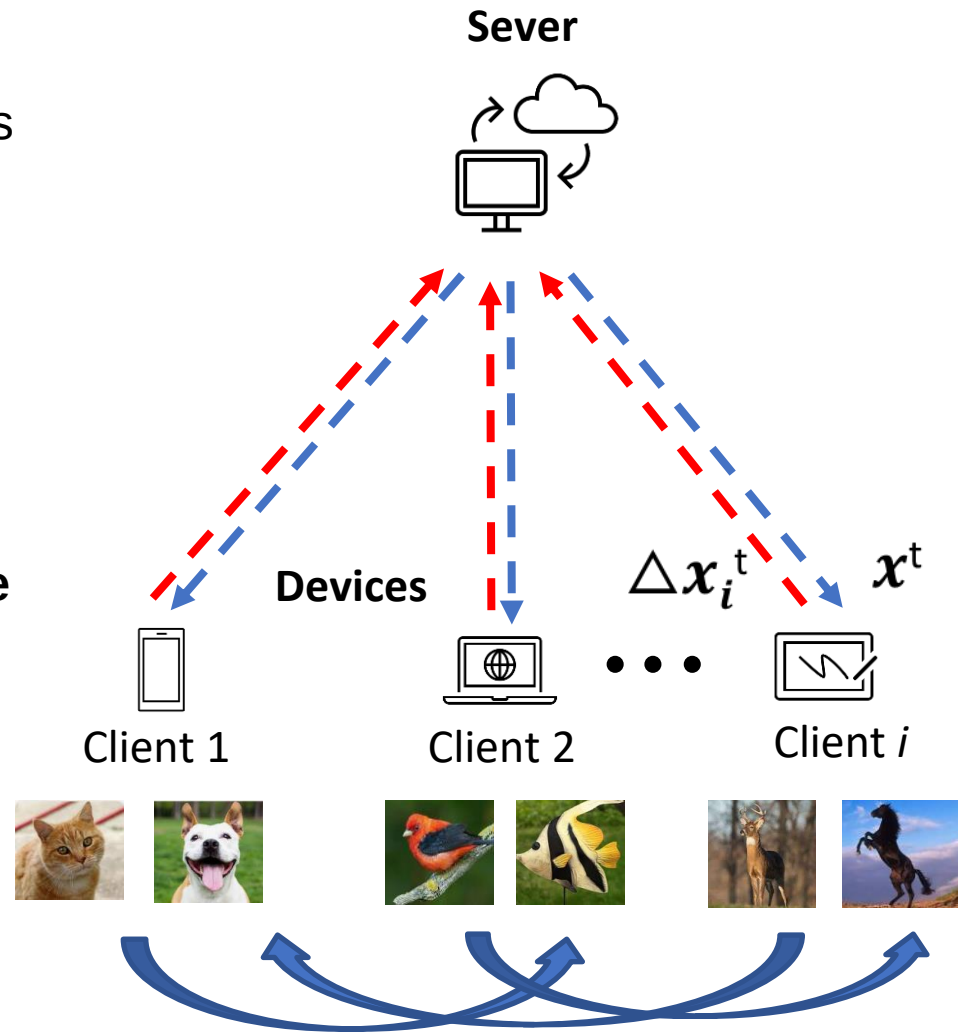


The model trained by FedAvg converges to the weighted average of two local optima corresponding to two different data distribution

- potentially distant from the true global optimum

Motivation

- In a **non-iid** data scenario, the number of samples with different labels available to a client vary significantly from one class to another. For each client, we distinguish between abundant and scarce classes.
- Objective drift problem is caused by **overfitting** in local training. Each client could benefit from **data augmentation** of its scarce classes, accomplished through the assistance by another client.
- We use VAEs to synthesize augmentation data, relying on **class-wise data representations** (extracted locally) as the assisting information that is shared among devices instead of raw data.



FedDPMS: The Main Ideas

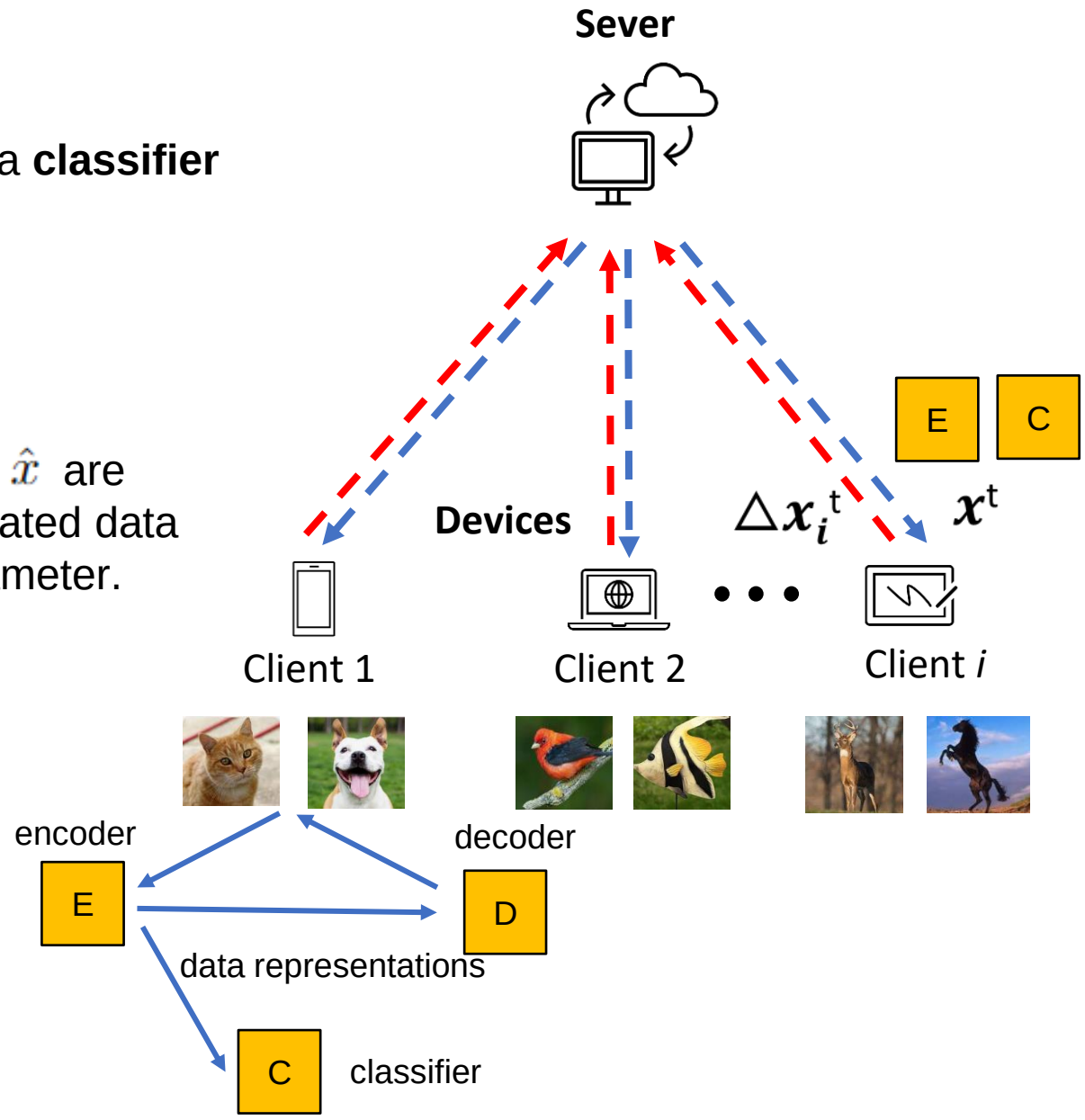
Stage I: Preliminary training

Each client trains a VAE model (consisting of an **encoder**, a **classifier** and a **decoder**) according to the objective

$$\mathcal{L} = \mathrm{CELoss}(\hat{y}, y) + \lambda \mathcal{L}_{\mathrm{VAE}}$$

$$\mathcal{L}_{\mathrm{VAE}} = \mathrm{KLD}(\mathbf{q}, \mathbf{p}) + \mathrm{MSE}(\hat{\mathbf{x}}, \mathbf{x})$$

where y and $\hat{y}$ are true label and predicted label; x and $\hat{x}$ are original data and reconstructed data; **q** and **p** are approximated data distribution and prior distribution of data; λ is a hyper-parameter.

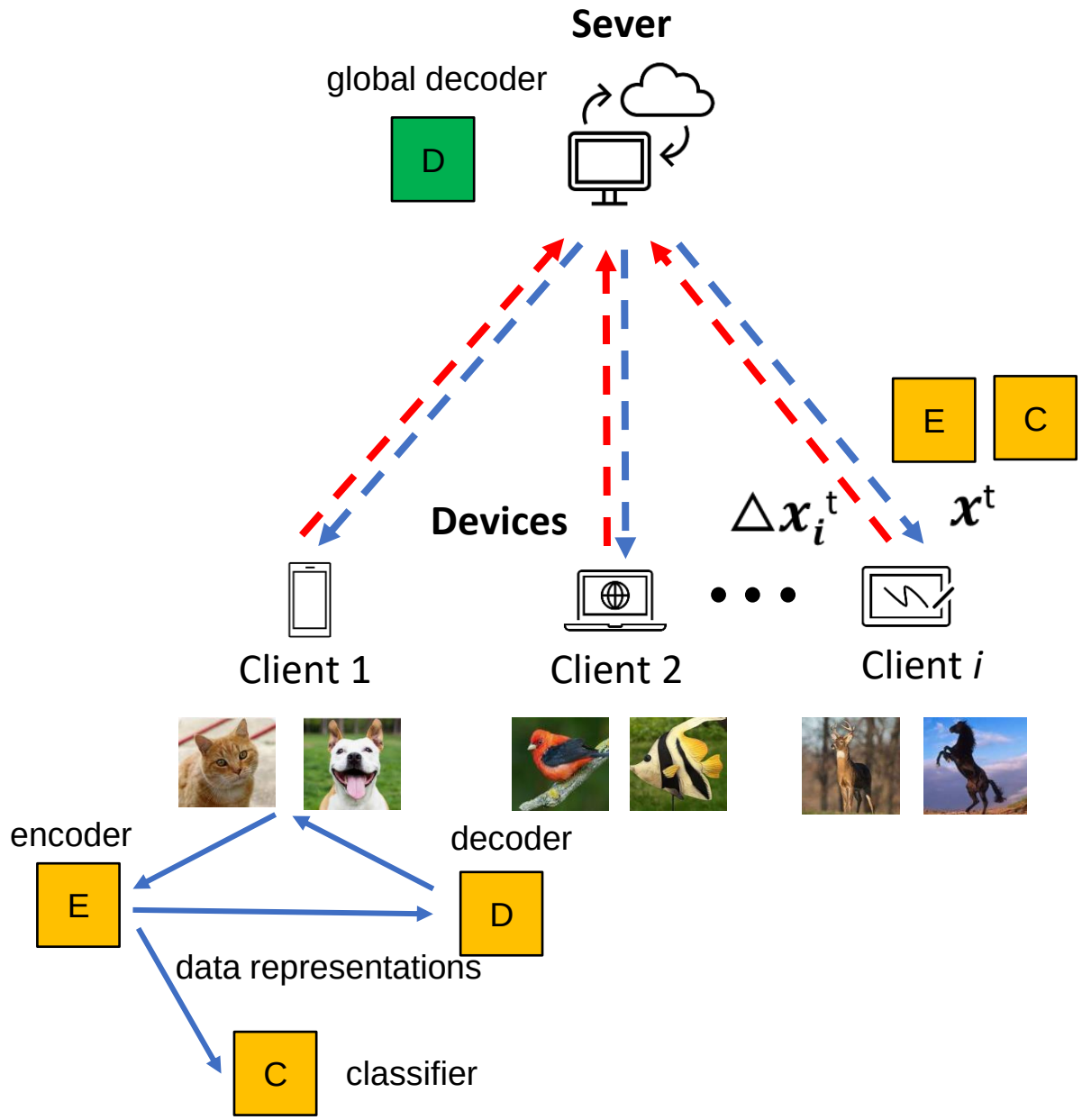


FedDPMS: The Main Ideas

Stage I: Preliminary training:

At the end of the preliminary training, all clients upload their local decoders to the server.

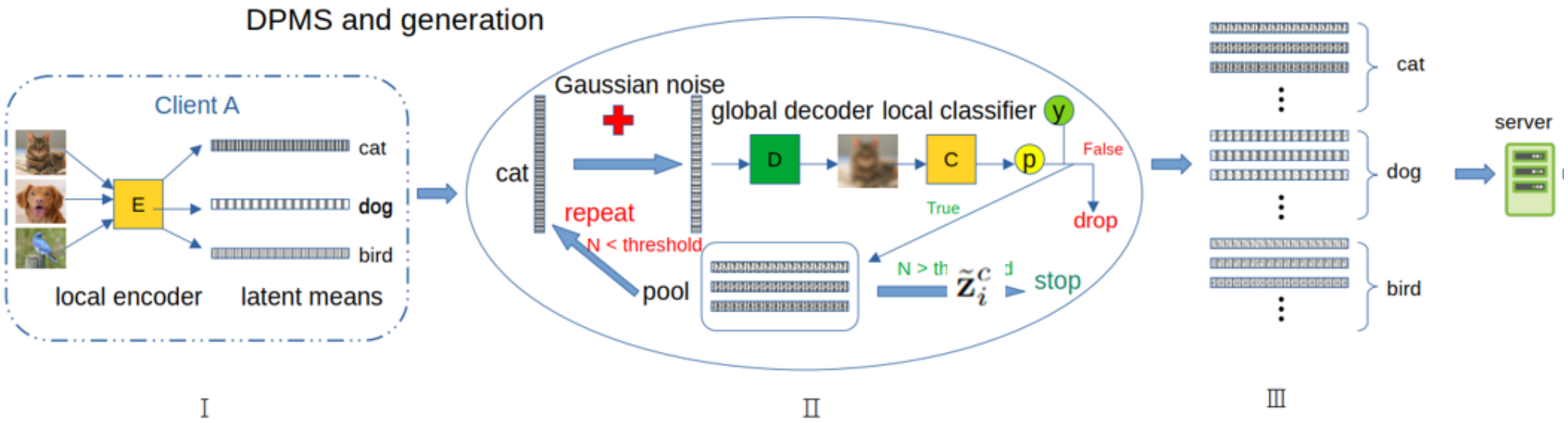
The server aggregates them into the global decoder and broadcast it to all clients.



FedDPMS: The Main Ideas

Stage II: Secondary training

Differential Private Mean Sharing (DPMS):



- Client *i* computes mean of data representation $\mathbf{z}_i^c$ in the abundant class *c*: $\mathbf{z}_i^c = \frac{1}{N_i^c} \sum_{j:y(\mathbf{x}_i^j)=c} E_i(\mathbf{x}_i^j)$
- The client *i* samples DP noise ϵ_i^c under Gaussian mechanism $\mathcal{N}(0, \sigma_f^2)$ and add it to the $\mathbf{z}_i^c$: $\tilde{\mathbf{z}}_i^c = \mathbf{z}_i^c + \epsilon_i^c$
- Utilize the global decoder to reconstruct image $\tilde{\mathbf{x}}_i^c$ with $\tilde{\mathbf{z}}_i^c$ and forward it to the local encoder/classifier. If the prediction is correct, add $\tilde{\mathbf{z}}_i^c$ to the pool. Otherwise, discard $\tilde{\mathbf{z}}_i^c$
- Repeat the above until the number of noise-perturbed data representations in the pool achieves the predetermined quota α . Then the client *i* sends all the data representations in the pool to the server.

FedDPMS: The Main Ideas

Stage II: Secondary training
Addressing privacy concerns

Each client shares the mean of data representations, so the private content of single data sample is preserved after the operation of average

- however, adversaries may be able to infer the data representation of single data through differential attack if two means of data representations computed with **two adjacent datasets** (datasets differing in just one point) are captured:

local dataset $\mathbf{d}$:

$$\mathbf{z}_i^c = \frac{1}{N_i^c} \sum_{j:y(\mathbf{x}_i^j)=c} E_i(\mathbf{x}_i^j)$$

local dataset $\mathbf{d}' = \mathbf{d} \cup \mathbf{x}_i^k$:

$$\mathbf{z}_i^{c\prime} = \frac{1}{N_i^c + 1} \sum_{j \in \mathbf{d}' : y(\mathbf{x_i}^j)=c} E_i(\mathbf{x}_i^j)$$

single data representation can be computed: $E_i(\mathbf{x}_i^k) = N_i^c \mathbf{z}_i^c - (N_i^c + 1)\mathbf{z}_i^{c\prime}$

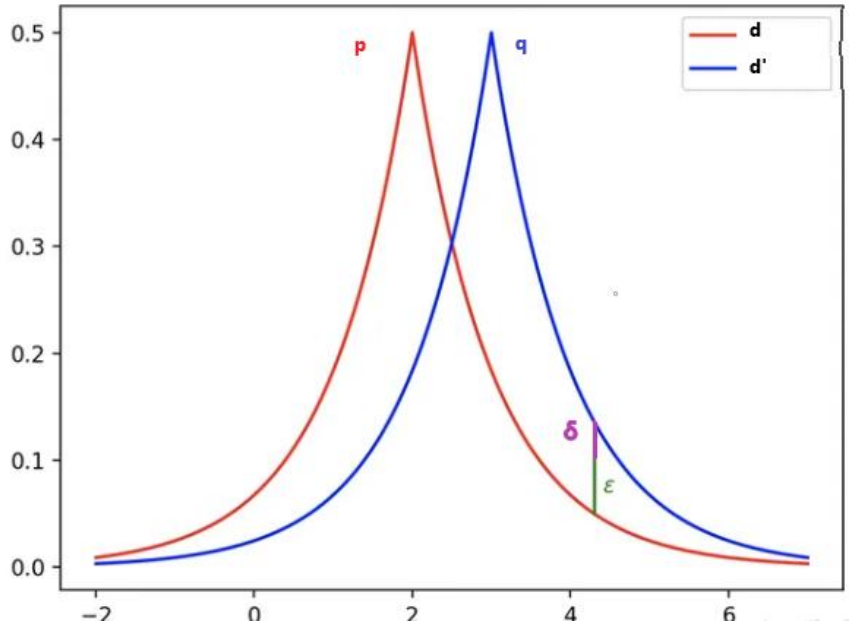
FedDPMS: The Main Ideas

Stage II: Secondary training
Addressing privacy concerns

Definition 1 (Differential Privacy) *A randomized mechanism $\mathcal{M} : \mathcal{D} \to \mathcal{R}$ satisfies (ϵ, δ) differential privacy if for any two adjacent databases $d, d' \in \mathcal{D}$ with only one different sample, and for any subset of the output $S \subseteq \mathcal{R}$, it holds that*

$$\Pr[\mathcal{M}(d) \in S] \leq e^{\epsilon} \Pr[\mathcal{M}(d') \in S] + \delta.$$

The output of the random mechanism $\mathcal{M}$ is a random distribution; ϵ denotes an upper bound on the distance between distributions $\mathcal{M}(d)$ and $\mathcal{M}(d')$ and can be interpreted as the privacy budget, while the relaxing factor δ is the probability that the ε-differential privacy is broken.



FedDPMS: The Main Ideas

Stage II: Secondary training
Addressing privacy concerns

Definition 2 (Gaussian noise mechanism) *The Gaussian noise mechanism achieving* (ε, δ) *differential privacy for function* $f : \mathcal{D} \to \mathbf{R}$ *is defined as*

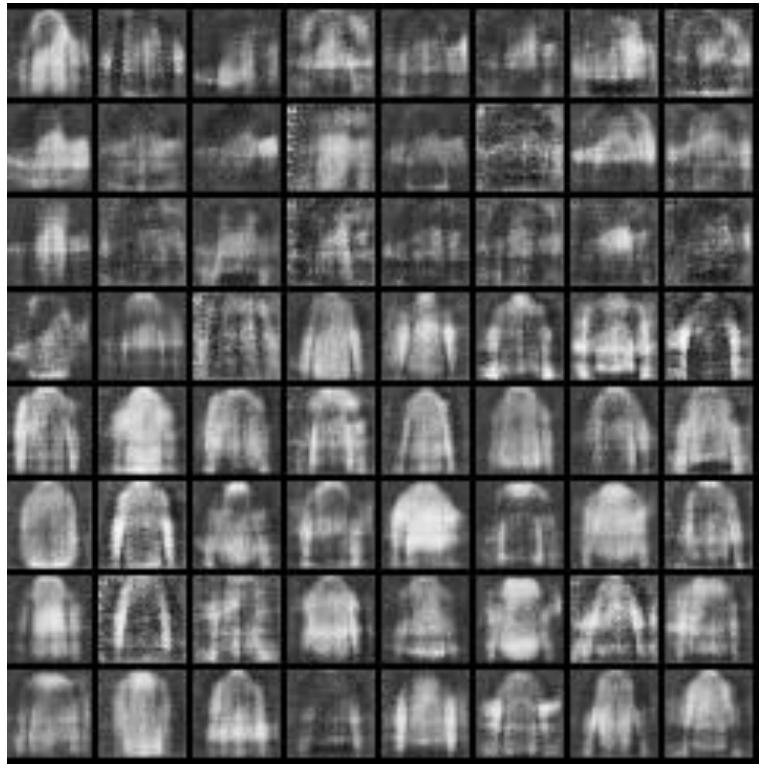
$$\mathcal{M}(d) = f(d) + \mathcal{N}\left(0, S_f^2 \cdot \sigma^2\right),$$

where S_f *denotes the maximum of the absolute distance* $|f(d) - f(d')|$*, and* $\sigma > \sqrt{2 \log \frac{5}{4\delta}}/\varepsilon$*. In other word, if we add a zero-mean Gaussian noise with variance* $S_f^2\sigma^2$ *to the output of* f *and set the privacy budget* ϵ*, the confidence of the resulting mechanism* $\mathcal{M}$ *is* $\delta \geq \frac{4}{5} \exp\left(-(\sigma\epsilon)^2/2\right)$.

In our case, the deterministic function **f** is average operation; sensitive $\mathbf{S_f}$ is $\mathbf{1/\ N_i^c}$;

For $\mathbf{N_i^c}$ = **100**, **σ** set to **4** and **δ** set to **0.01**, one obtains **ε** = **0.5**. Therefore, it is with **99%** confidence that the means of data representations computed on two adjacent datasets would differ by less than **0.5**, which makes them indistinguishable.

Synthetic Images From Noisy Means of Data Representations



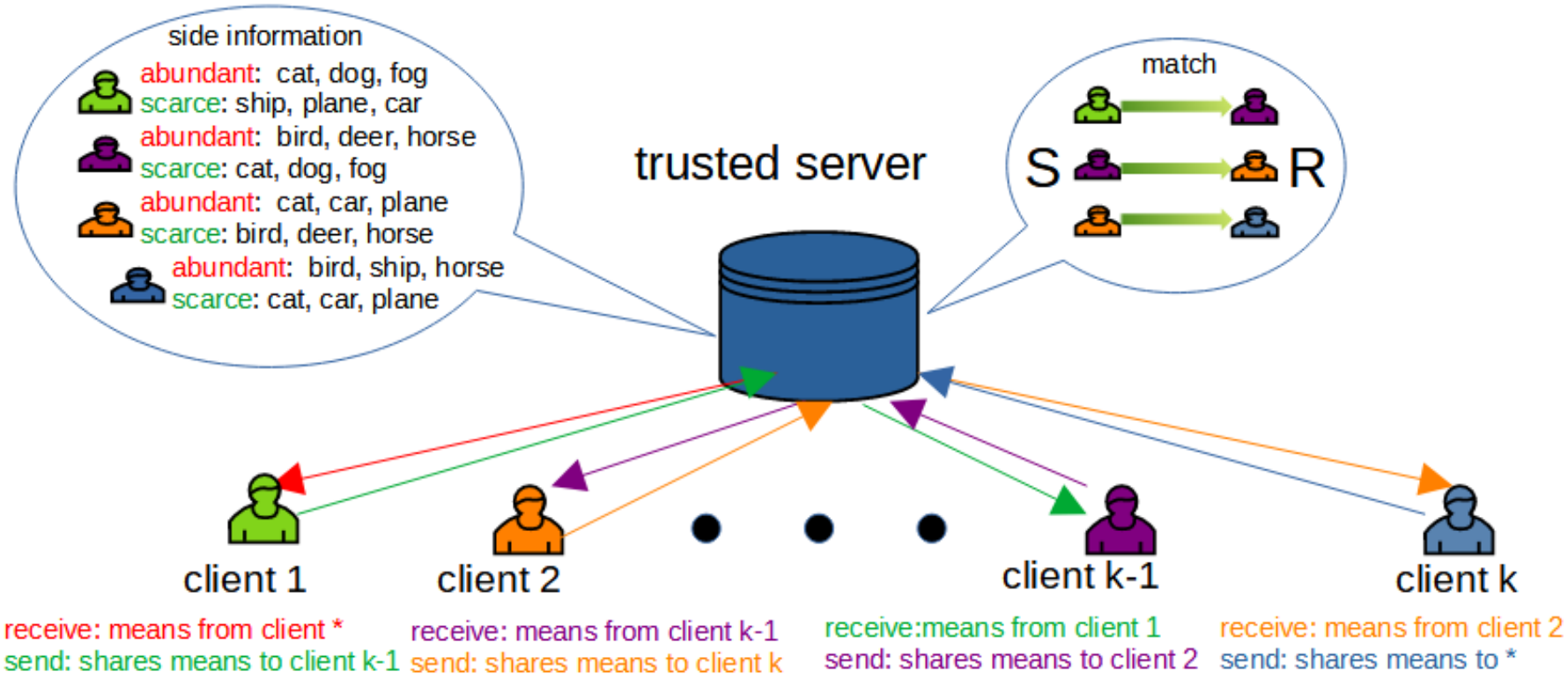
FMNIST

CIFAR10

CIFAR100

FedDPMS: The Main Ideas

Stage II: Secondary training
Client Matching



- The server is given side information about abundant classes and scarce classes of all clients.
- The server matches pairs of clients as receiving client and sending client.

FedDPMS: The Main Ideas

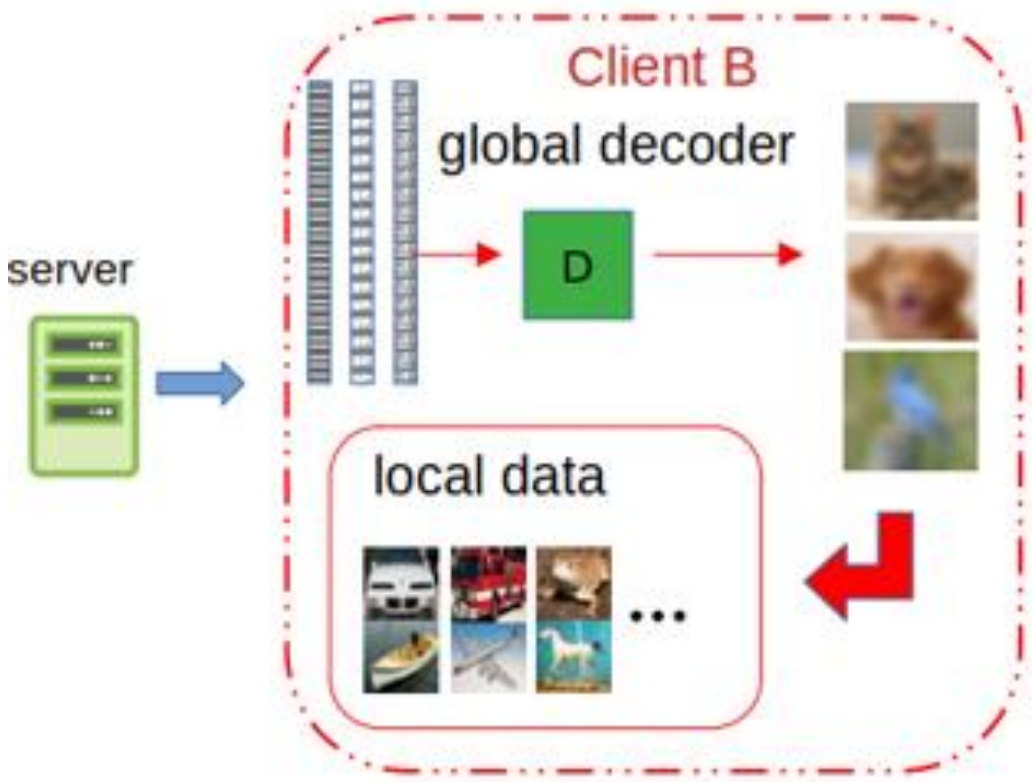
Stage II: Secondary training

Data Synthesis

- After matching, the server sends the corresponding means of data representations to the receiving client.
- The receiving client utilizes the global decoder to reconstruct images and merges them into the original local dataset.
- Through the above steps, the classes in the local dataset become richer.

Model Training

- After data synthesis, all clients may **delete** all parameters of local and global decoders to free up memory.
- In the following training, clients utilize the augmented local dataset to train local models (encoder/classifier).



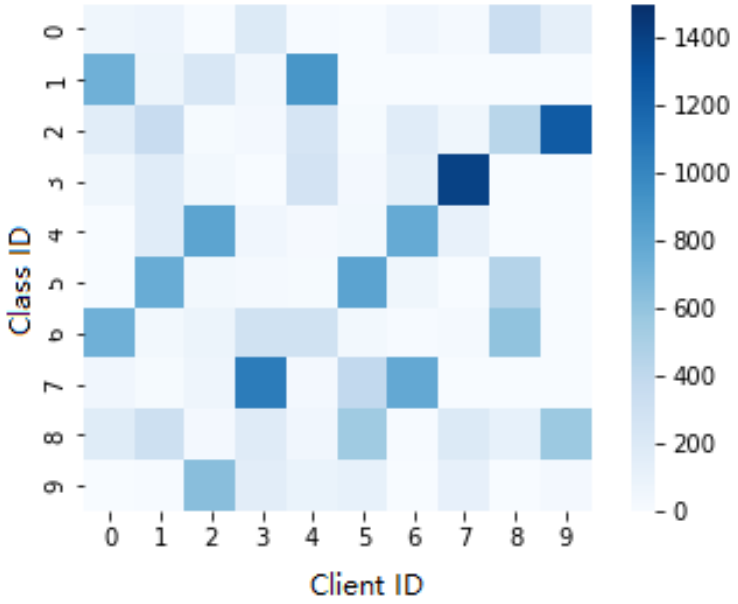
Experimental Results

Dataset: FMNIST, CIFAR10, CIFAR100

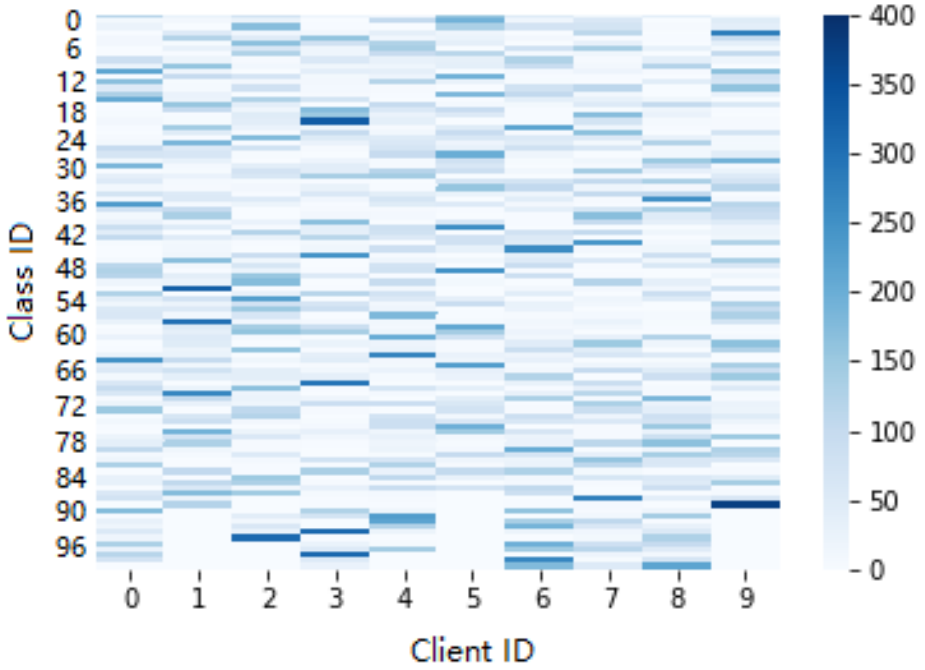
Baselines: FedAvg, FedProx, FedMix, Moon

Data Partitions:

Generating clients' data partitions with Dirichlet Distribution with a concentration parameter $\boldsymbol{\beta}$. The proportion $\mathbf{p}_c$ of samples with label **c** among **m** clients is drawn from: $\mathbf{p}_c \sim Dir_m(\boldsymbol{\beta})$

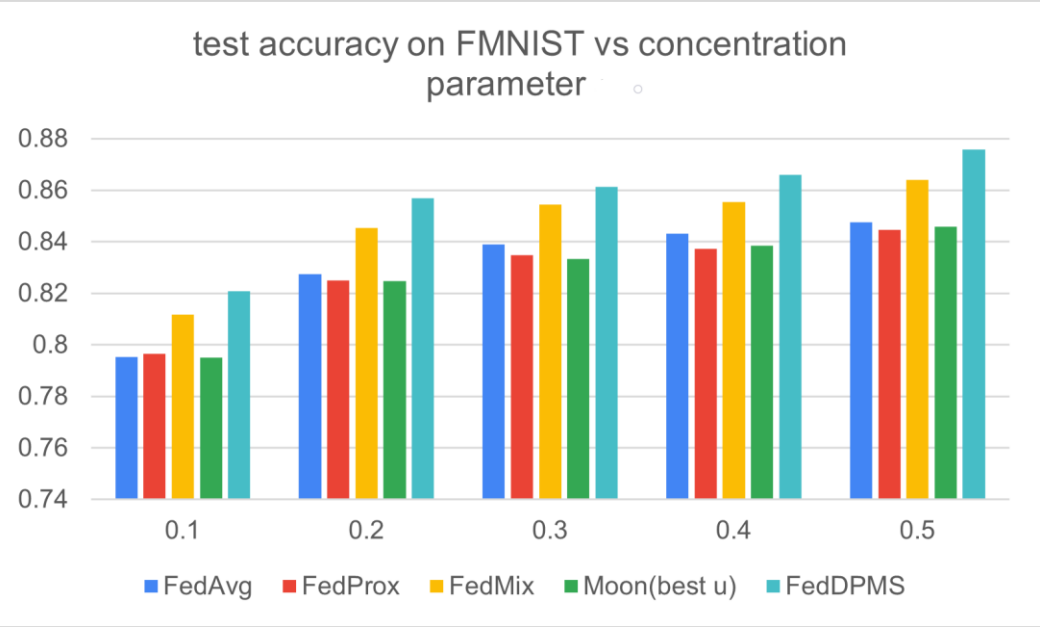


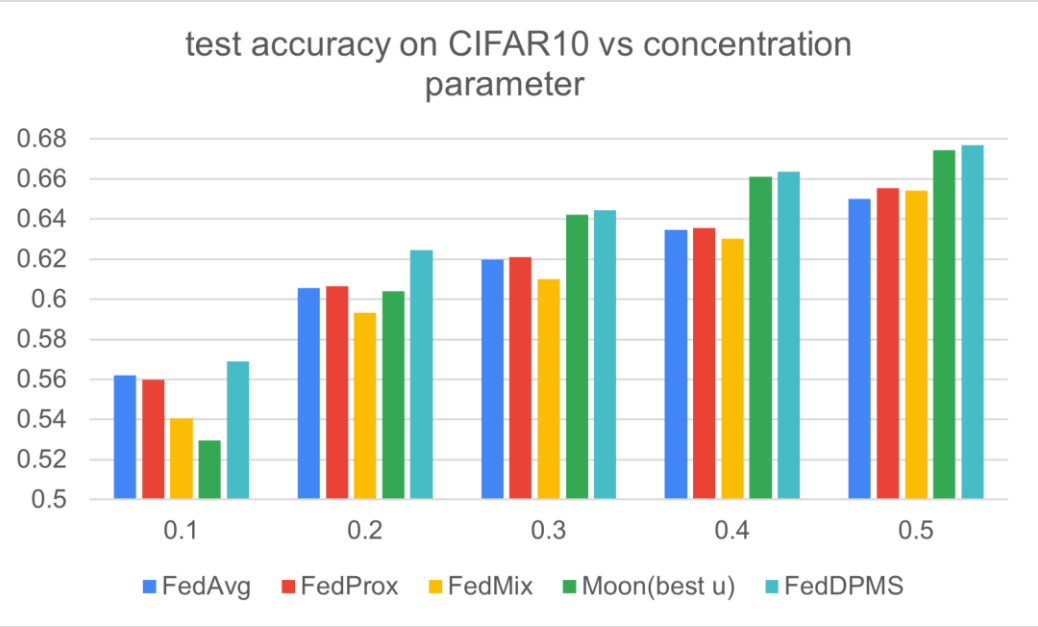
CIFAR10 training set is sampled into 10 partitions with $\boldsymbol{\beta}$ = 0.5

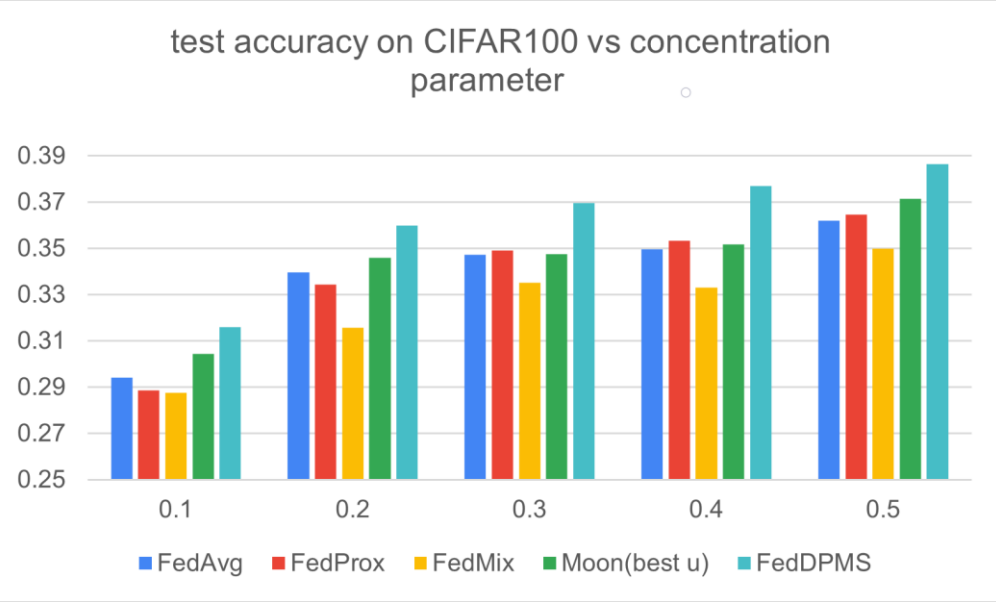


CIFAR100 training set is sampled into 50 partitions with $\boldsymbol{\beta}$ = 0.5

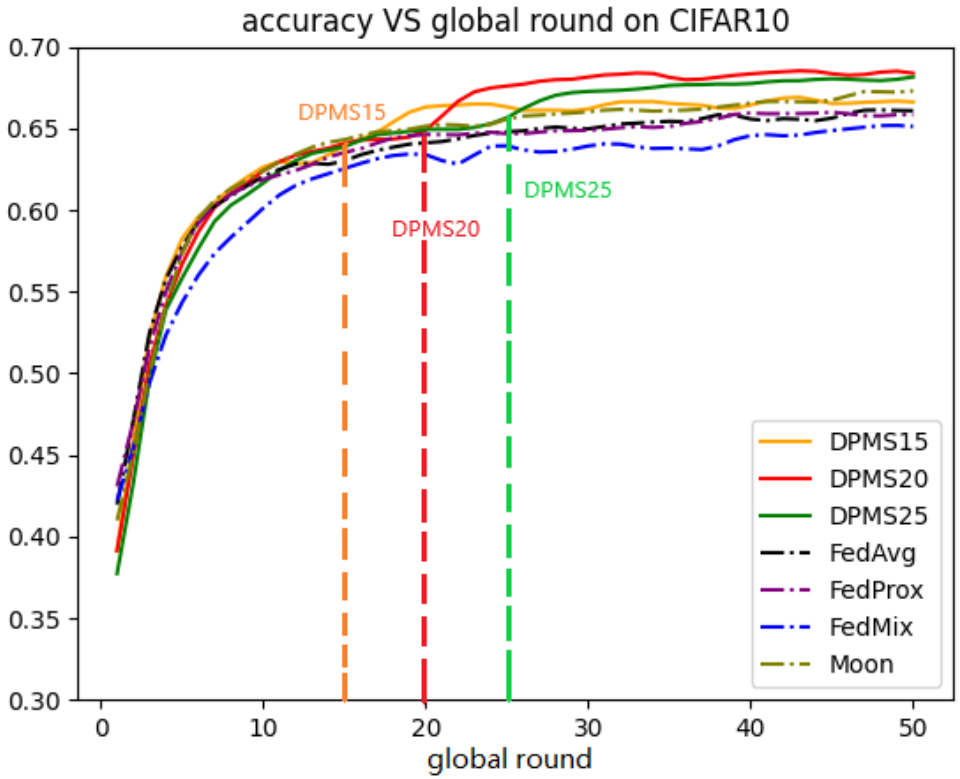
Experimental Results

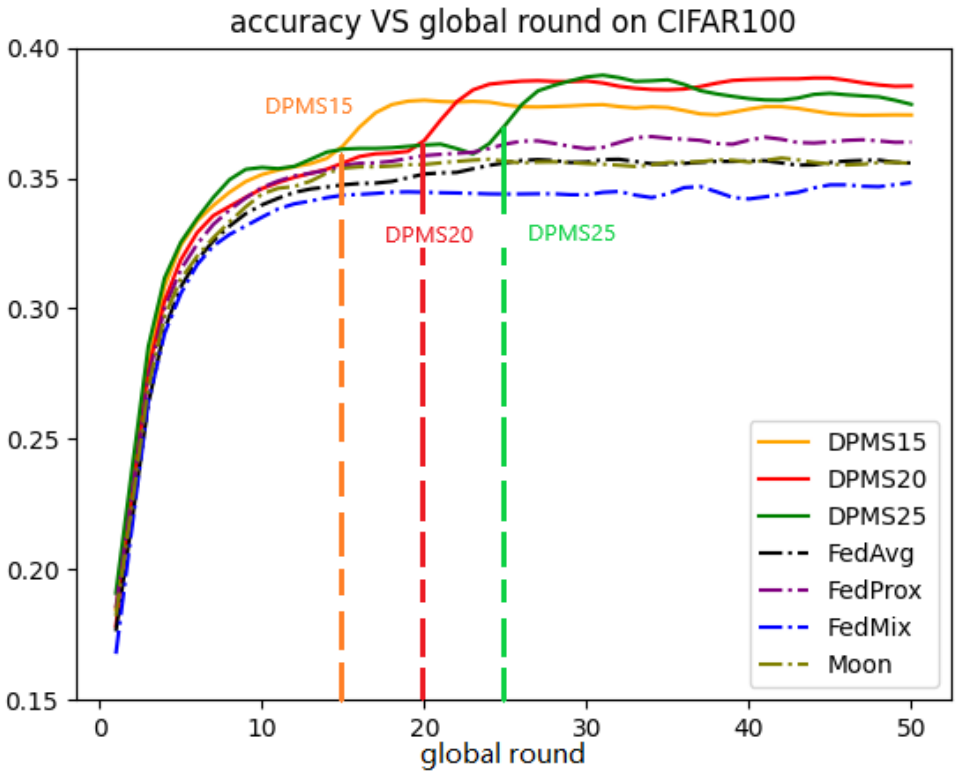






Experimental Results





The synthetic images are only a small fraction of the augmented dataset, playing a role of regularizer in the secondary training

- no need for the generator to synthesize high-quality images
- no need for more rounds for preliminary training is not necessary improve the performance

Conclusions and Future Work

- FedDPMS enables data augmentation in non-IID Federated Learning by sharing class-wise data representations
- Data augmentation effectively improves performance of non-IID Federated Learning with only a minor additional computation and memory overhead
- To promote privacy, we relied on differential privacy concepts
- As part of the future work, we will investigate settings in which privacy concerns may require further privacy-protection mechanisms

Thanks for your attention!

Questions?