Federated Multi-Task Learning under a Mixture of Distributions

Othmane Marfoq[1, 2, 3], Giovanni Neglia[1, 2], Aurélien Bellet[1],
Laetitia Kameni[3], Richard Vidal[3],

[1]Inria and [2]Université Côte d'Azur and [3]Accenture Labs

UNIVERSITÉ
CÔTE D'AZUR

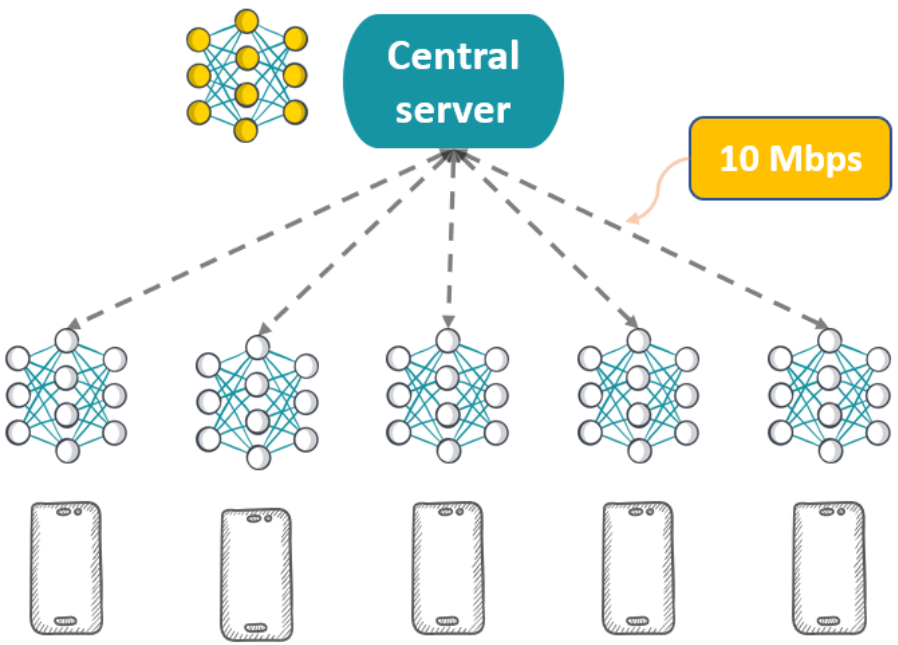
Accenture Labs

Federated Learning

Federated learning (FL) "involves training statistical models over remote devices or siloed data centers, such as mobile phones or hospitals, while keeping data localized". (Li et al. 2020)

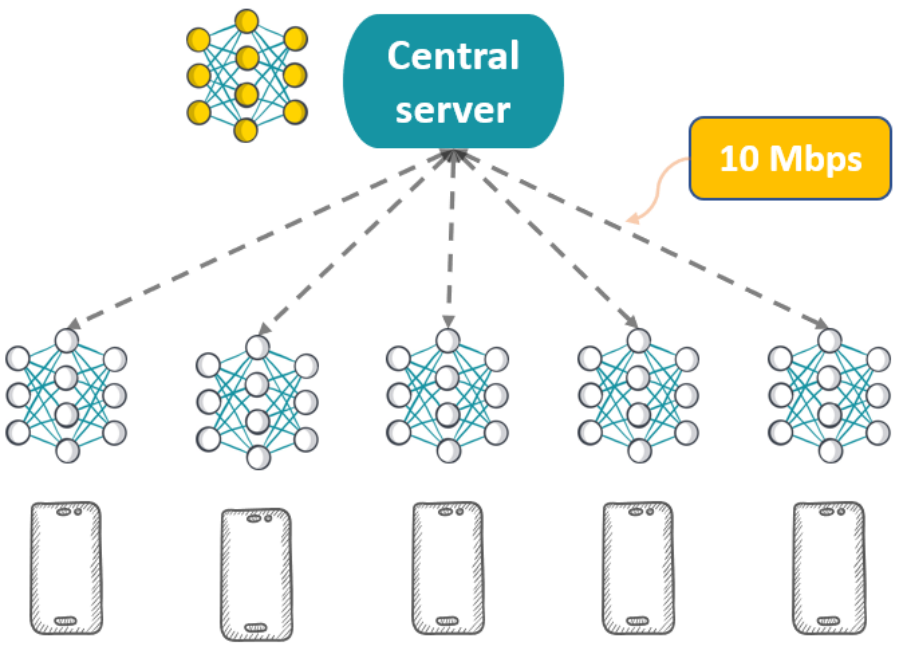
Federated Learning

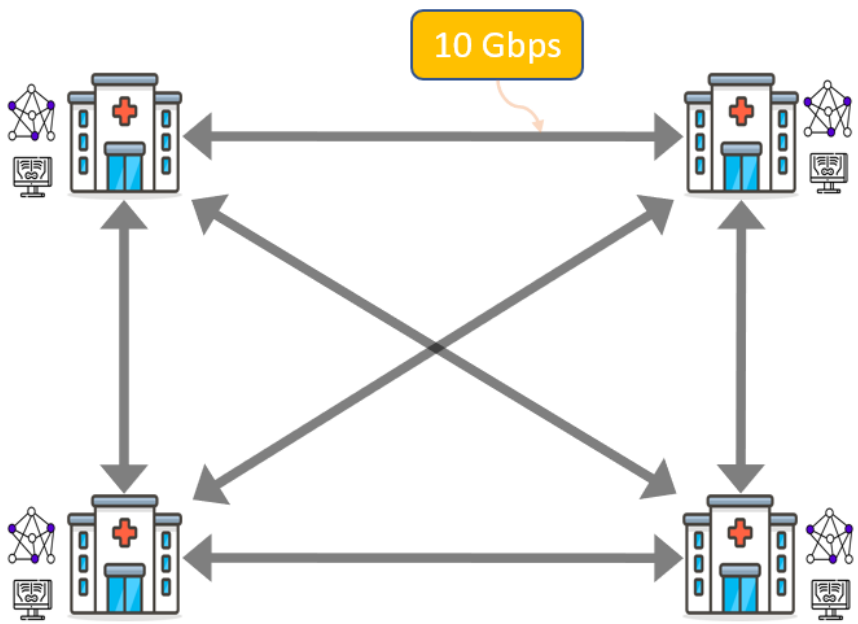
Federated learning (FL) "involves training statistical models over remote devices or siloed data centers, such as mobile phones or hospitals, while keeping data localized". (Li et al. 2020)



Federated Learning

Federated learning (FL) "involves training statistical models over remote devices or siloed data centers, such as mobile phones or hospitals, while keeping data localized". (Li et al. 2020)





Introduction

- A (countable) set $\mathcal{T}$ of classification (or regression) tasks which represent the set of possible clients.

Introduction

- A (countable) set $\mathcal{T}$ of classification (or regression) tasks which represent the set of possible clients.
- Data $\mathcal{S}_t = \{s_t^{(i)} \triangleq (\mathbf{x}_t^{(i)},\ y_t^{(i)})\}_{i=1}^{n_t}$ at client t is drawn from a local distribution $\mathcal{D}_t$ over $\mathcal{X} \times \mathcal{Y}$.

Introduction

- A (countable) set $\mathcal{T}$ of classification (or regression) tasks which represent the set of possible clients.
- Data $\mathcal{S}_t = \{s_t^{(i)} \triangleq (\mathbf{x}_t^{(i)},\ y_t^{(i)})\}_{i=1}^{n_t}$ at client t is drawn from a local distribution $\mathcal{D}_t$ over $\mathcal{X} \times \mathcal{Y}$.
- Client t wants to learn hypothesis h_t

$$\underset{h_t \in \mathcal{H}}{\text{minimize}}\, \mathcal{L}_{\mathcal{D}_t}(h_t) \triangleq \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_t}\left[l\left(h_t\left(\mathbf{x}\right), y\right)\right].$$

Introduction

- A (countable) set $\mathcal{T}$ of classification (or regression) tasks which represent the set of possible clients.
- Data $\mathcal{S}_t = \{s_t^{(i)} \triangleq (\mathbf{x}_t^{(i)},\ y_t^{(i)})\}_{i=1}^{n_t}$ at client t is drawn from a local distribution $\mathcal{D}_t$ over $\mathcal{X} \times \mathcal{Y}$.
- Client t wants to learn hypothesis h_t

$$\underset{h_t \in \mathcal{H}}{\text{minimize}}\, \mathcal{L}_{\mathcal{D}_t}(h_t) \triangleq \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_t}\left[l\left(h_t\left(\mathbf{x}\right), y\right)\right].$$

- Having personalized models for each client is a necessity in many FL applications.

Related Work

- *Model agnostic meta-learning* (MAML) based federated *multi-task learning* (MTL).
- Clustered FL.
- Model interpolation: `APFL` and `MAPPER`.
- Federated MTL via task relationships: `MOCHA`, `pFedMe`, `L2SGD` and `FedU`.

Related Work

- *Model agnostic meta-learning* (MAML) based federated *multi-task learning* (MTL).
- Clustered FL.
- Model interpolation: `APFL` and `MAPPER`.
- Federated MTL via task relationships: `MOCHA`, `pFedMe`, `L2SGD` and `FedU`.

  Limitation: restrictive assumptions or complex algorithms.

An impossibility result

Some assumption on the local data distributions $\mathcal{D}_t,\ t \in \mathcal{T}$ are needed for federated learning to be beneficial:

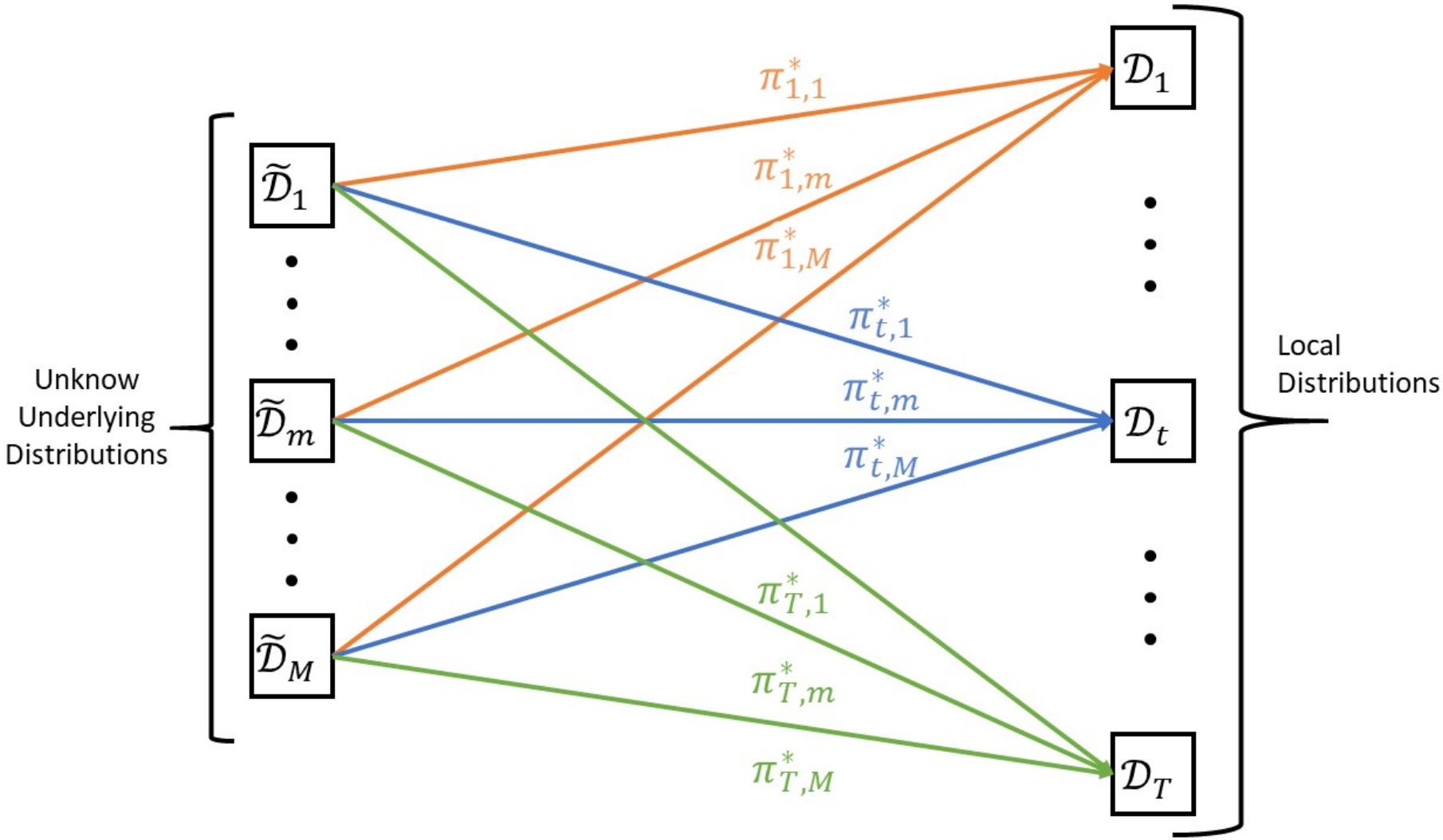
An impossibility result

Some assumption on the local data distributions $\mathcal{D}_t,\ t \in \mathcal{T}$ are needed for federated learning to be beneficial:

- Federated learning with T clients is equivalent to T *semi-supervised learning* (SSL) problems.

- With no assumptions on the data distribution, SSL is impossible. (Ben-David et al. 2008; Darnstädt et al. 2013; Göpfert et al. 2019).

Main assumption



Main assumption

Assumption

There exist M underlying (independent) distributions $\tilde{\mathcal{D}}_m$, $1 \leq m \leq M$, such that for $t \in \mathcal{T}$, $\mathcal{D}_t$ is mixture of the distributions $\{\tilde{\mathcal{D}}_m\}_{m=1}^M$ with weights $\pi_t^* = [\pi_{t1}^*, \ldots, \pi_{tm}^*] \in \Delta^M$, i.e.

$$z_t \sim \mathcal{M}(\pi_t^*), \quad ((x_t, y_t) | z_t = m) \sim \tilde{\mathcal{D}}_m, \quad \forall t \in \mathcal{T}, \tag{1}$$

where $\mathcal{M}(\pi)$ is a multinomial (categorical) distribution with parameters π.

Generalizing Existing Frameworks

The generative model in the mixture assumption extends/covers some popular multi-task/personalized FL formulations in the literature.

Generalizing Existing Frameworks

The generative model in the mixture assumption extends/covers some popular multi-task/personalized FL formulations in the literature.

Example (Clustered Federated Learning)

The mixture assumption recovers this scenario considering $M = C$ and $\pi^*_{tc} = 1$ if task (client) t is in cluster c and $\pi^*_{tc} = 0$ otherwise.

Main Contributions

- Flexible assumption for personalized FL (mixtures of components).
- EM-like learning algorithms with convergence guarantees (both in client-server and fully-decentralized settings).

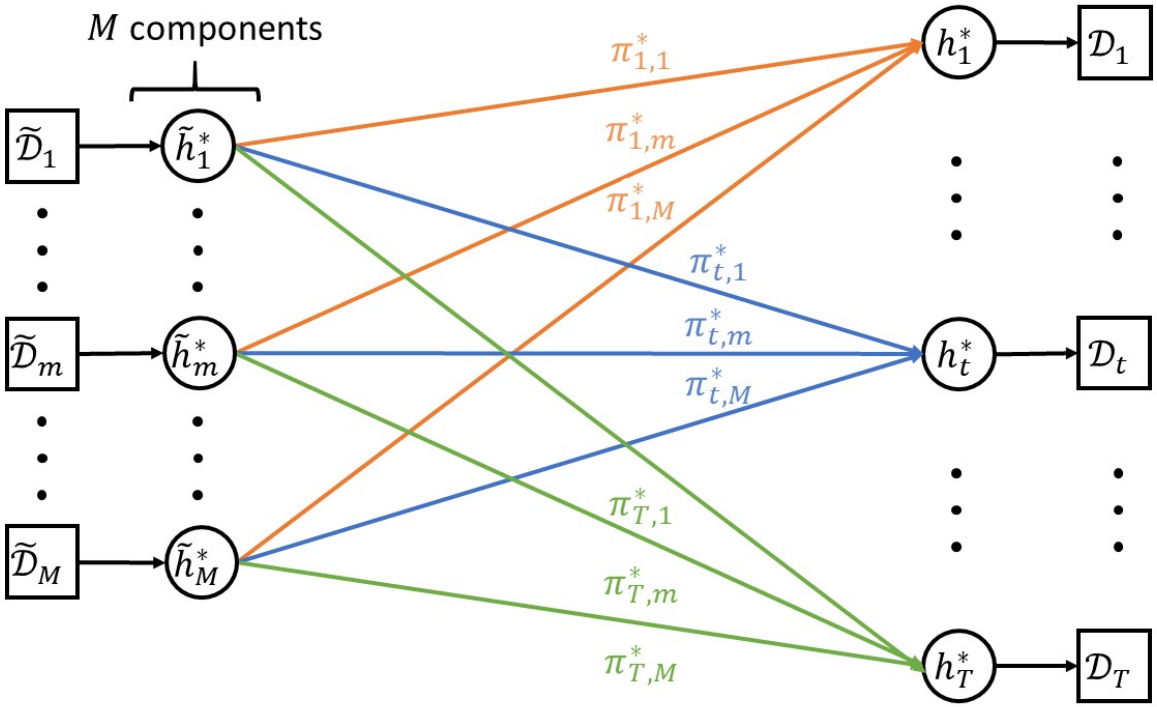
Main Contributions

- Flexible assumption for personalized FL (mixtures of components).
- EM-like learning algorithms with convergence guarantees (both in client-server and fully-decentralized settings).
- More general *federated surrogate optimization* framework.

Main Contributions

- Flexible assumption for personalized FL (mixtures of components).
- EM-like learning algorithms with convergence guarantees (both in client-server and fully-decentralized settings).
- More general *federated surrogate optimization* framework.
- Higher accuracy and fairness than SOTA algorithms, even for clients not present at training time.

Learning under a mixture model



Proposition (informal)

$$h^*_t = \sum_{m=1}^{M} \breve{\pi}_{tm} h_{\breve{\theta}_m}, \quad \forall t \in \mathcal{T} \tag{2}$$

Learning under a mixture model

- Estimate the parameters $\breve{\Theta}$ and $\breve{\pi}_t,\ 1 \leq t \leq T$, minimizing:

$$f(\Theta, \Pi) \triangleq -\frac{\log p(\mathcal{S}_{1:T}|\Theta, \Pi)}{n} \triangleq -\frac{1}{n}\sum_{t=1}^{T}\sum_{i=1}^{n_t} \log p(s_t^{(i)}|\Theta, \pi_t), \quad (3)$$

Learning under a mixture model

- Estimate the parameters $\breve{\Theta}$ and $\breve{\pi}_t,\ 1 \leq t \leq T$, minimizing:

$$f(\Theta, \Pi) \triangleq -\frac{\log p(\mathcal{S}_{1:T}|\Theta, \Pi)}{n} \triangleq -\frac{1}{n}\sum_{t=1}^{T}\sum_{i=1}^{n_t} \log p(s_t^{(i)}|\Theta, \pi_t), \quad (3)$$

- Use Eq. (4) to get the client predictor for the T clients present at training time.

$$h_t^* = \sum_{m=1}^{M} \breve{\pi}_{tm} h_{\breve{\theta}_m}(\mathbf{x}), \quad \forall t \in \mathcal{T} \quad (4)$$

Expectation-Maximization

A natural approach to solve problem (3) is via the

Expectation-Maximization (EM) algorithm

Expectation-Maximization

A natural approach to solve problem (3) is via the

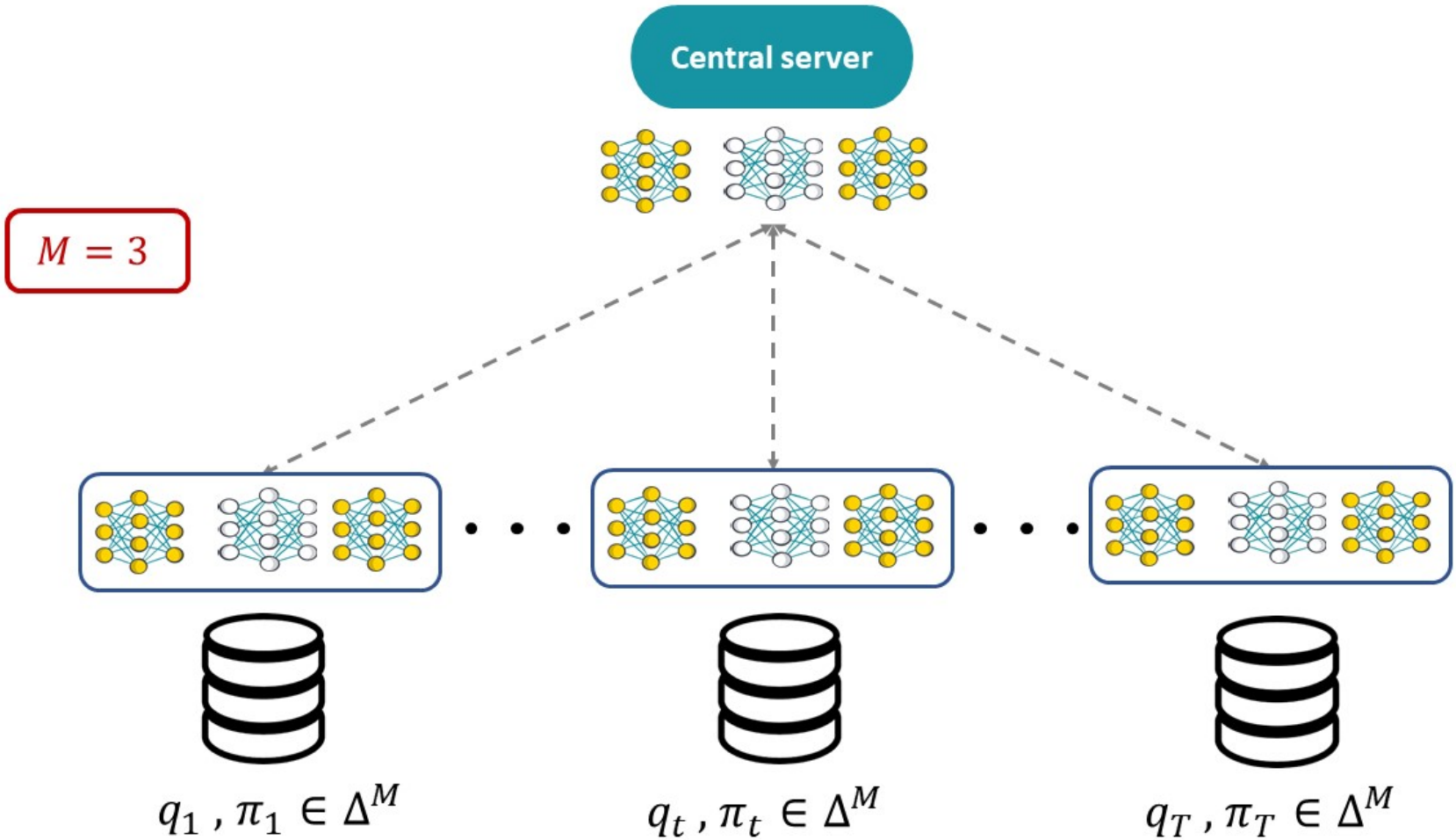
Expectation-Maximization (EM) algorithm

E-step: $$q_t^{k+1}(z_t^{(i)} = m) \propto \pi_{tm}^k \cdot \exp\left(-l(h_{\theta_m^k}(\mathbf{x}_t^{(i)}), y_t^{(i)})\right).$$

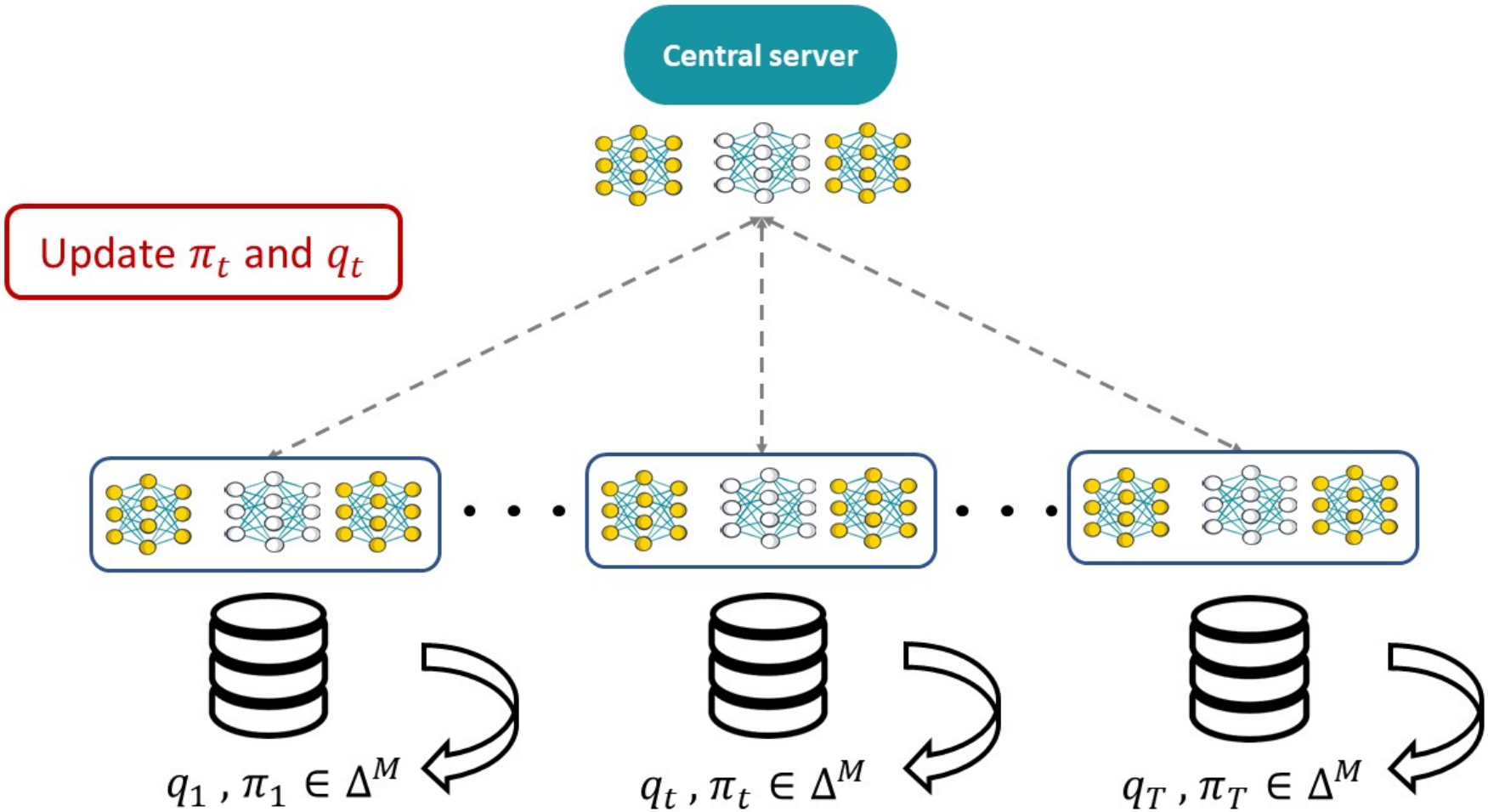
M-step: $$\pi_{tm}^{k+1} = \frac{\sum_{i=1}^{n_t} q_t^{k+1}(z_t^{(i)} = m)}{n_t},$$

$$\theta_m^{k+1} \in \arg\min_{\theta \in \mathbb{R}^d} \sum_{t=1}^{T} \sum_{i=1}^{n_t} q_t^{k+1}(z_t^{(i)} = m) \cdot l\big(h_\theta(\mathbf{x}_t^{(i)}), y_t^{(i)}\big).$$

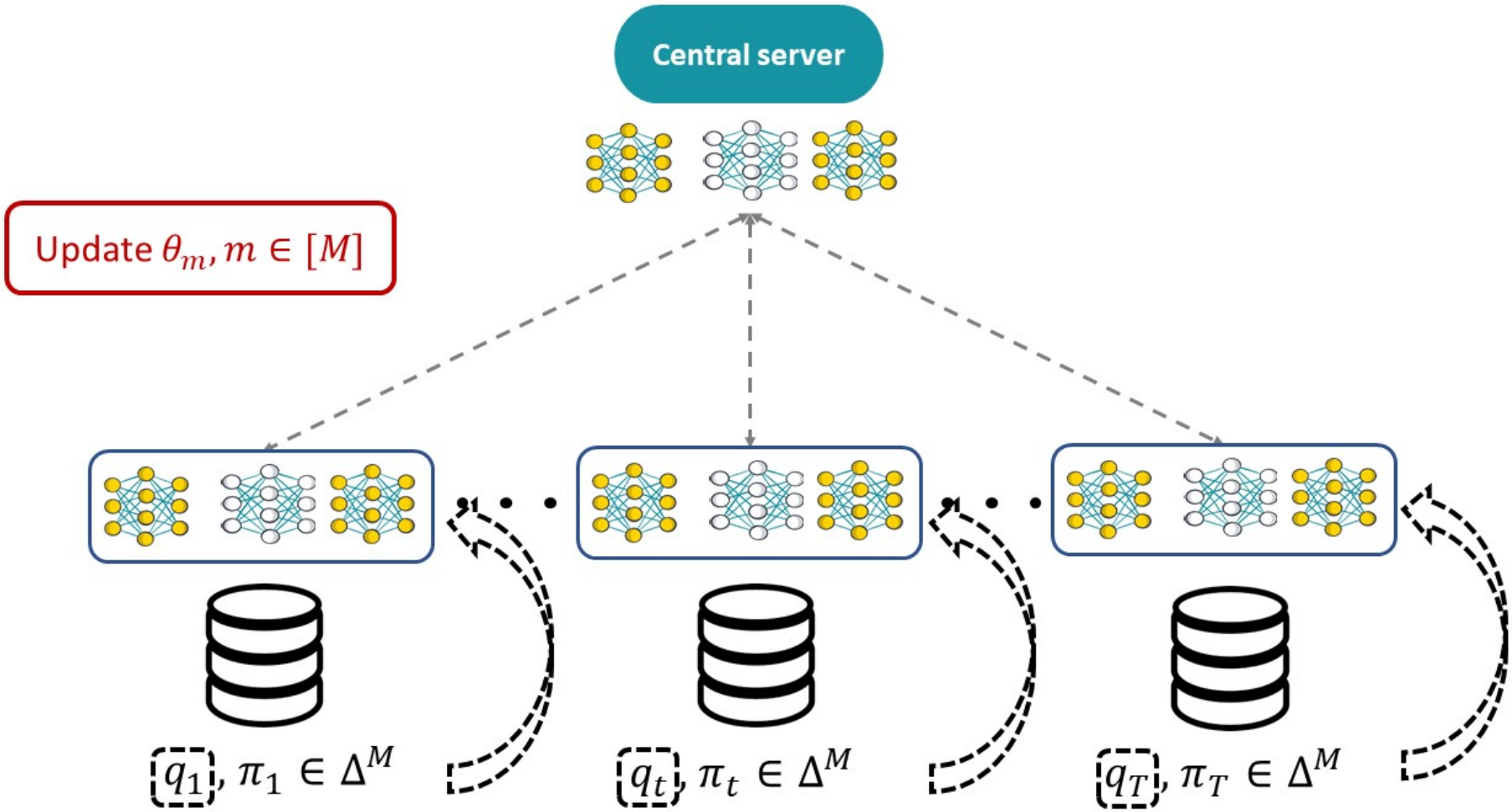
Federated Expectation-Maximization



Federated Expectation-Maximization



Federated Expectation-Maximization



Federated Expectation Maximization

Theorem

Under Assumptions 1–3 and some other mild assumptions, when clients use SGD as local solver with learning rate $\eta = \frac{a_0}{\sqrt{K}}$*, after a large enough number of communication rounds* K*,* `FedEM`*'s iterates satisfy:*

$$\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\left\|\nabla_{\Theta} f\left(\Theta^k, \Pi^k\right)\right\|_F^2 \leq \mathcal{O}\left(\frac{1}{\sqrt{K}}\right),$$

$$\frac{1}{K}\sum_{k=1}^{K}\Delta_{\Pi} f(\Theta^k, \Pi^k) \leq \mathcal{O}\left(\frac{1}{K^{3/4}}\right),$$

where the expectation is over the random batches samples, and

$$\Delta_{\Pi} f(\Theta^k, \Pi^k) \triangleq f\left(\Theta^k, \Pi^k\right) - f\left(\Theta^k, \Pi^{k+1}\right) \geq 0.$$

Fully Decentralized Settings

Theorem (Informal)

In the same setting of the previous theorem and under an additional mild assumption on the connectivity of the communication graph, `D-FedEM`*'s individual estimates* $(\Theta_t^k)_{1 \leq t \leq T}$ *converge to a common value* $\bar{\Theta}^k$*. Moreover,* $\bar{\Theta}^k$ *and* Π^k *converge to a stationary point of* f*.*

Surrogate Federated Optimization

- `FedEM` can be seen as a particular instance of a more general framework that we call *federated surrogate optimization*.

Surrogate Federated Optimization

- `FedEM` can be seen as a particular instance of a more general framework that we call *federated surrogate optimization*.
- This framework minimizes an objective function $\sum_{t=1}^{T} \omega_t f_t(\mathbf{u}, \mathbf{v}_t)$
- Each client $t \in [T]$ can compute a partial first order surrogate of f_t.

Experiments

Dataset	Local	FedAvg	FedProx	FedAvg+	clustered FL	pFedMe	FedEM (Ours)
FEMNIST	71.0 / 57.5	78.6 / 63.9	78.9 / 64.0	75.3 / 53.0	73.5 / 55.1	74.9 / 57.6	**79.9** / **64.8**
EMNIST	71.9 / 64.3	82.6 / 75.0	83.0 / 75.4	83.1 / 75.8	82.7 / 75.0	83.3 / 76.4	**83.5** / **76.6**
CIFAR10	70.2 / 48.7	78.2 / 72.4	78.0 / 70.8	82.3 / 70.6	78.6 / 71.2	81.7 / 73.6	**84.3** / **78.1**
CIFAR100	31.5 / 19.9	40.9 / 33.2	41.0 / 33.2	39.0 / 28.3	41.5 / 34.1	41.8 / 32.5	**44.1** / **35.0**
Shakespeare	32.0 / 16.6	**46.7** / 42.8	45.7 / 41.9	40.0 / 25.5	46.6 / 42.7	41.2 / 36.8	**46.7** / **43.0**
Synthetic	65.7 / 58.4	68.2 / 58.9	68.2 / 59.0	68.9 / 60.2	69.1 / 59.0	69.2 / 61.2	**74.7** / **66.7**

Table: Test accuracy: average across clients / bottom decile.

Experiments

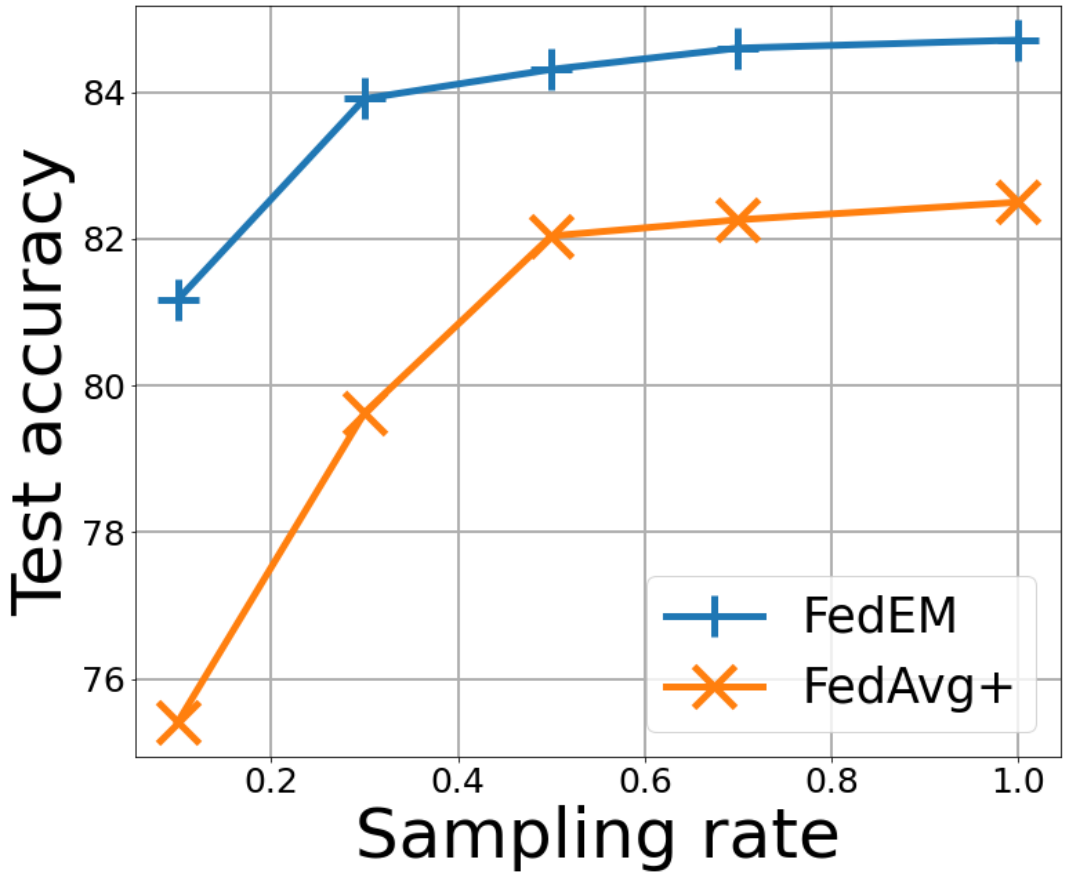


Figure: Effect of client sampling rate on the test accuracy for CIFAR10.

Experiments

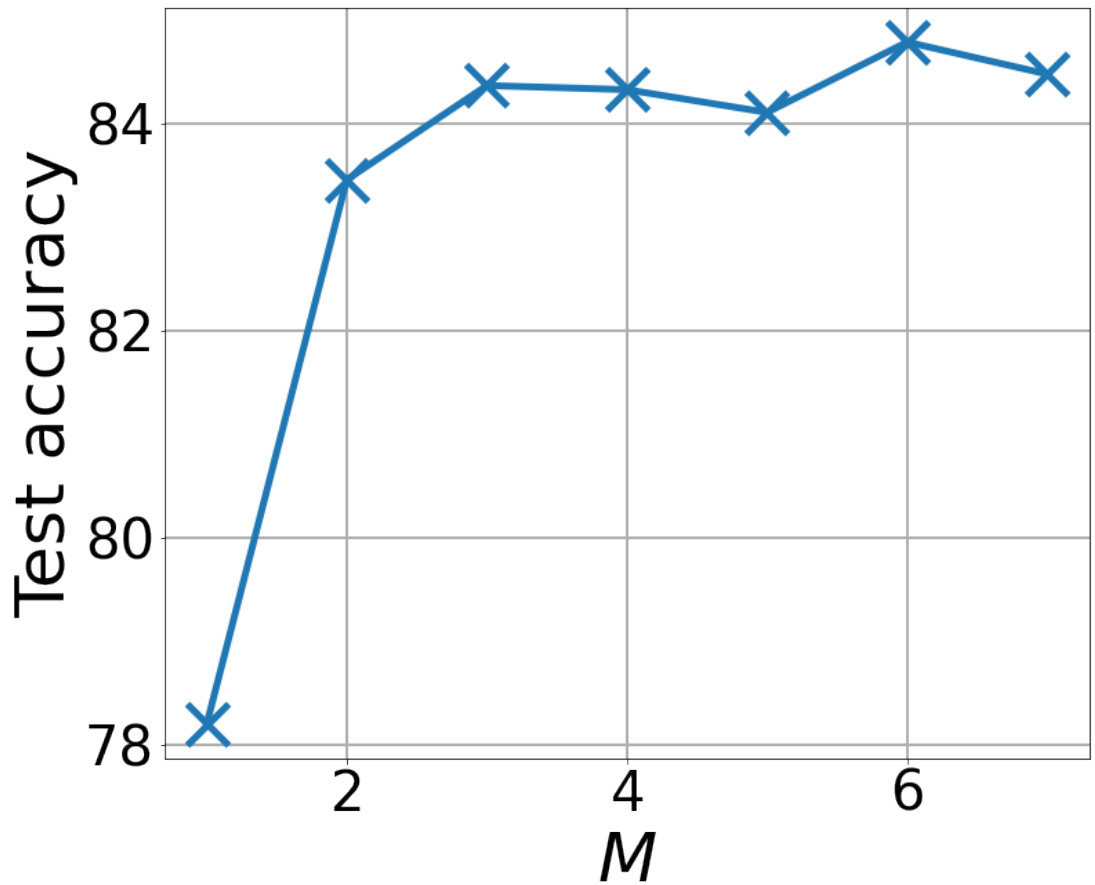


Figure: Effect of number of mixture components M on the test accuracy

Experiments

Dataset	FedAvg	FedAvg+	FedEM
FEMNIST	78.3 (80.9)	74.2 (84.2)	**79.1** (81.5)
EMNIST	83.4 (82.7)	83.7 (92.9)	**84.0** (83.3)
CIFAR10	77.3 (77.5)	80.4 (80.5)	**85.9** (90.7)
CIFAR100	41.1 (42.1)	36.5 (55.3)	**47.5** (46.6)
Shakespeare	**46.7** (47.1)	40.2 (93.0)	**46.7** (46.6)
Synthetic	68.6 (70.0)	69.1 (72.1)	**73.0** (74.1)

Table: Average test accuracy across **clients unseen at training** (train accuracy in parenthesis).

Experiments

Table: Test accuracy comparison across different tasks. For each method, the best test accuracy is reported. For `FedEM` we run only $\frac{K}{M}$ rounds, where K is the total number of rounds for other methods–$K = 80$ for Shakespeare and $K = 200$ for all other datasets–and $M = 3$ is the number of components used in `FedEM`.

Dataset	Local	`FedAvg`	`FedProx`	`FedAvg+`	`Clustered FL`	`pFedMe`	`FedEM` (Ours)
FEMNIST	71.0	**78.6**	78.6	75.3	73.5	74.9	74.0
EMNIST	71.9	82.6	82.7	83.1	82.7	**83.3**	82.7
CIFAR10	70.2	78.2	78.0	82.3	78.6	81.7	**82.5**
CIFAR100	31.5	41.0	40.9	39.0	41.5	41.8	**42.0**
Shakespeare	32.0	**46.7**	45.7	40.0	46.6	41.2	43.8
Synthetic	65.7	68.2	68.2	68.9	69.1	69.2	**73.2**

Experiments

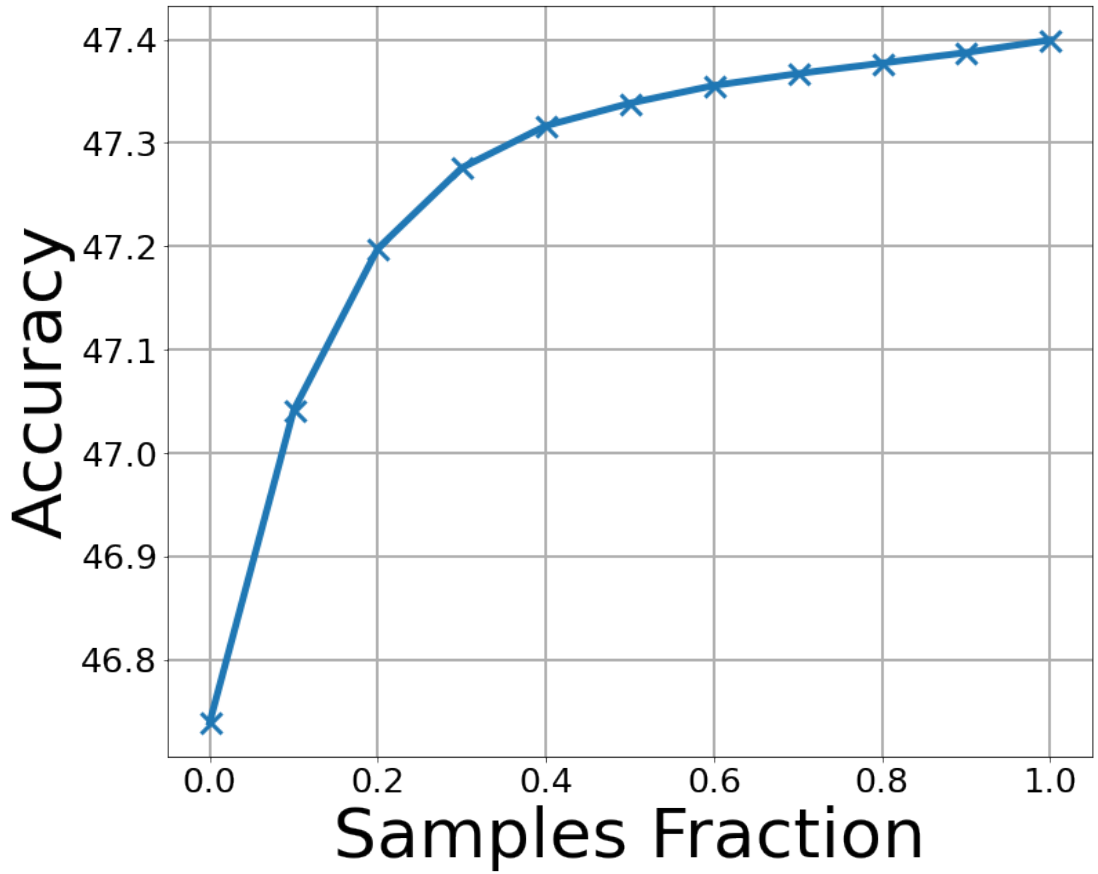


Figure: Effect of the number of samples on the average test accuracy across clients unseen at training.

Conclusion

Thank you for your attention

Project link: `https://github.com/omarfoq/FedEM`

Email: `othmane.marfoq@inria.fr`

References I

- Ben-David, Shai, Tyler Lu, and D. Pál (2008). "Does Unlabeled Data Provably Help? Worst-case Analysis of the Sample Complexity of Semi-Supervised Learning". In: *COLT*.
- Darnstädt, Malte, H. U. Simon, and Balázs Szörényi (2013). "Unlabeled Data Does Provably Help". In: *STACS*.
- Göpfert, Christina et al. (2019). "When can unlabeled data improve the learning rate?" In: *Conference on Learning Theory*. PMLR, pp. 1500–1518.
- Li, Tian et al. (2020). "Federated learning: Challenges, methods, and future directions". In: *IEEE Signal Processing Magazine* 37.3, pp. 50–60.