FEDERATED MULTI-TASK LEARNING
UNDER A MIXTURE OF DISTRIBUTIONS

Aurélien Bellet (Inria)
Joint work with O. Marfoq, G. Neglia (Inria), L. Kameni, R. Vidal (Accenture Labs)

Google Federated Learning Workshop
November 8-10, 2021
• **Personalized models** are a necessity in many Federated Learning (FL) applications

• **Key questions:** how to model the relations between local data distributions? How to design efficient FL algorithms that exploit these relations?
• Local fine-tuning of a global model: [Jiang et al., 2019], [Fallah et al., 2020]...

• Interpolation of global and local model: [Deng et al., 2020], [Mansour et al., 2020]...

  ⇒ works only if local distributions are close from the global distribution

• Clustered FL: [Sattler et al., 2020], [Ghosh et al., 2020]...

  ⇒ no knowledge transfer across clusters
• Multi-task learning via task relationships [Smith et al., 2017], [Vanhaesebrouck et al., 2017] or simpler penalization terms [Hanzely et al., 2020], [Dinh et al., 2020]...

⇒ limited to linear models or lose ability to model complex relationships

• Hypernetworks [Shamsian et al., 2021]

⇒ flexible but potential blow up in the number of parameters

**Overall**: conditions under which users benefit from collaboration are not well understood
1. A *flexible statistical assumption for personalized FL*: local distributions are mixtures of underlying components

2. *Federated EM-like algorithms with convergence guarantees*, both in server-client and fully decentralized settings

3. A general *federated surrogate optimization framework* that can be used to analyze other FL algorithms

4. *Higher accuracy and fairness than SOTA algorithms*, even for users not present at training time
PROBLEM SETTING

• A (countable) set $\mathcal{T}$ of tasks representing the set of possible users

• A data distribution $\mathcal{D}_t$ over $\mathcal{X} \times \mathcal{Y}$ for each user $t \in \mathcal{T}$ with $p_t(x, y)$ the joint density and $p_t(x)$, $p_t(y)$ the marginal densities

• User $t$ wants to learn hypothesis $h_t \in \mathcal{H}$ minimizing the expected risk over $\mathcal{D}_t$:

$$\min_{h_t \in \mathcal{H}} \mathcal{L}_{\mathcal{D}_t}(h_t) = \mathbb{E}_{(x, y) \sim \mathcal{D}_t}[l(h_t(x), y)]$$

• A set of $T$ users $[\mathcal{T}] = \{1, \ldots, T\} \subseteq \mathcal{T}$ participate to the training phase

• Local dataset $\mathcal{S}_t = \{(x_t^{(i)}, y_t^{(i)})\}_{i=1}^{n_t}$ at user $t \in T$ drawn i.i.d. from $\mathcal{D}_t$
• Assume $p_t(x)$ is identical across $t \in T$, but $p_t(y|x)$ can be arbitrarily different

• FL with $T$ users is then equivalent to $T$ semi-supervised learning (SSL) problems

• With no assumptions on the data distribution, SSL does not improve sample complexity [Ben-David et al., 2008, Darnstädt et al., 2013, Göpfert et al., 2019]

⇒ some assumptions on local data distributions are needed for FL to be beneficial
• For any user $t \in \mathcal{T}$, the local distribution $\mathcal{D}_t$ is a mixture of underlying distributions $\tilde{\mathcal{D}}_1, \ldots, \tilde{\mathcal{D}}_M$ defined by weights $\pi^*_t, \ldots, \pi^*_m$.

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 $\pi^*_t = [\pi^*_t, \ldots, \pi^*_m] \in \Delta^M$, i.e.

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

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

PROPOSED ASSUMPTION
• Our assumptions generalize previous personalized FL formulations

• Clustered FL [Sattler et al., 2020, Ghosh et al., 2020] with C clusters: set $M = C$ and $\pi_{tc}^* = 1$ if task (user) $t$ is in cluster $c$ and $\pi_{tc}^* = 0$ otherwise

• We also recover model interpolation [Deng et al., 2020, Mansour et al., 2020] and Fed-MTL with task relationships [Smith et al., 2017, Vanhaesebrouck et al., 2017] as special cases
Proposition (informal)

Let $\tilde{\Theta} = [\tilde{\theta}_1, \ldots, \tilde{\theta}_M]$ and $\tilde{\Pi} = [\tilde{\pi}_1, \ldots, \tilde{\pi}_T]$ be a solution of

$$\arg\min_{\Theta, \Pi} \mathbb{E}_{t \sim D_T} \mathbb{E}_{(x, y) \sim D_t} [-\log p_t(x, y|\Theta, \pi_t)]$$

Then, for any $t \in T$, we have:

$$h^*_t = \sum_{m=1}^{M} \tilde{\pi}_{tm} h_{\tilde{\theta}_m}$$

(1)

- We can estimate $\tilde{\Theta}$ and $\tilde{\Pi}$ by minimizing

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

- For a user $t'$ not seen at training time: learn $\pi_{t'}$ in a single shot, and use (1)
CENTRALIZED EXPECTATION-MAXIMIZATION

- Natural approach: Expectation-Maximization (EM) algorithm
- We denote by $q_t$ the distribution over the latent variables $z_t^{(i)}$
- **E-step:** $q_t^{k+1}(z_t^{(i)} = m) \propto \pi_{tm}^k \cdot \exp \left( -l(h_{\theta_m^k}(\vec{x}_t^{(i)}), y_t^{(i)}) \right)$
- **M-step:**

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

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

$M = 3$

$q_1, \pi_1 \in \Delta^M$
$q_t, \pi_t \in \Delta^M$
$q_T, \pi_T \in \Delta^M$
FEDERATED EXPECTATION-MAXIMIZATION

Update $\pi_t$ and $q_t$

$q_1, \pi_1 \in \Delta^M$

$q_t, \pi_t \in \Delta^M$

$q_T, \pi_T \in \Delta^M$
Update $\theta_m, m \in [M]$
Theorem (Informal)

With local SGD as the local solver, the iterates of FedEM satisfy:

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

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

where the expectation is over the random batches samples, and

\[ \Delta_{\Pi} f(\Theta^k, \Pi^k) \triangleq f(\Theta^k, \Pi^k) - f(\Theta^k, \Pi^{k+1}) \geq 0. \]
FedEM can be seen as a particular instance of a more general framework that we call federated surrogate optimization, extending the centralized framework of [Mairal, 2013]. This framework minimizes an objective function of the form $\sum_{t=1}^{T} \omega_t f_t (\vec{u}, \vec{v}_t)$. Each user $t \in [T]$ can compute a partial first order surrogate of $f_t$. Our framework can be used to analyze the convergence of other FL algorithms, such as pFedMe [Dinh et al., 2020] (see paper for details).
## EXPERIMENTS

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>FEMNIST</td>
<td>71.0 / 57.5</td>
<td>78.6 / 63.9</td>
<td>78.9 / 64.0</td>
<td>75.3 / 53.0</td>
<td>73.5 / 55.1</td>
<td>74.9 / 57.6</td>
<td>79.9 / 64.8</td>
</tr>
<tr>
<td>EMNIST</td>
<td>71.9 / 64.3</td>
<td>82.6 / 75.0</td>
<td>83.0 / 75.4</td>
<td>83.1 / 75.8</td>
<td>82.7 / 75.0</td>
<td>83.3 / 76.4</td>
<td>83.5 / 76.6</td>
</tr>
<tr>
<td>CIFAR10</td>
<td>70.2 / 48.7</td>
<td>78.2 / 72.4</td>
<td>78.0 / 70.8</td>
<td>82.3 / 70.6</td>
<td>78.6 / 71.2</td>
<td>81.7 / 73.6</td>
<td>84.3 / 78.1</td>
</tr>
<tr>
<td>CIFAR100</td>
<td>31.5 / 19.9</td>
<td>40.9 / 33.2</td>
<td>41.0 / 33.2</td>
<td>39.0 / 28.3</td>
<td>41.5 / 34.1</td>
<td>41.8 / 32.5</td>
<td>44.1 / 35.0</td>
</tr>
<tr>
<td>Shakespeare</td>
<td>32.0 / 16.6</td>
<td><strong>46.7</strong> / 42.8</td>
<td>45.7 / 41.9</td>
<td>40.0 / 25.5</td>
<td>46.6 / 42.7</td>
<td>41.2 / 36.8</td>
<td><strong>46.7</strong> / 43.0</td>
</tr>
<tr>
<td>Synthetic</td>
<td>65.7 / 58.4</td>
<td>68.2 / 58.9</td>
<td>68.2 / 59.0</td>
<td>68.9 / 60.2</td>
<td>69.1 / 59.0</td>
<td>69.2 / 61.2</td>
<td><strong>74.7</strong> / 66.7</td>
</tr>
</tbody>
</table>

**Table 1:** Test accuracy: average across users / bottom decile.

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

**Table 2:** Average test accuracy across users unseen at training (train accuracy in parenthesis).
**Figure 1:** Effect of local dataset size on the average test accuracy across unseen users for CIFAR100.
Figure 2: Effect of user sampling rate on the test accuracy for CIFAR10.
Figure 3: Effect of number of mixture components $M$ on the test accuracy for CIFAR10.
THANK YOU FOR YOUR ATTENTION!

TO APPEAR AT NEURIPS 2021

ARXIV LINK: https://arxiv.org/abs/2108.10252

CODE: https://github.com/omarfoq/fedem
In COLT.

Unlabeled data does provably help.
In STACS.

Adaptive personalized federated learning.

Personalized Federated Learning with Moreau Envelopes.
In NeurIPS.

In NeurIPS.

An efficient framework for clustered federated learning.
In NeurIPS.


