Federated Learning with Heterogeneous Users: A Superquantile Optimization Approach


March 14 @ INFORMS

Krishna Pillutla,

Yassine Laguel, Jérôme Malick, Zaid Harchaoui
% of World Population That Uses a Smartphone

Image Credit: Business Wire
Federated Learning

Machine learning has moved from the data centers to edge devices

Challenges:
- Communication efficiency
- Statistical heterogeneity
- Privacy of user data
Machine learning has moved from the data centers to edge devices.

**Challenges:**

- Communication efficiency
- Statistical heterogeneity
- Privacy of user data
Outline

- Background
- Distributional Robustness with Simplicial-FL
- Algorithm & Convergence Guarantees
- Numerical Results
Usual Approach to Federated Learning

Objective

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} F_i(w)
\]

where

\[
F_i(w) = \mathbb{E}_{z \sim p_i} [f(w; z)]
\]

loss on client \(i\)

[McMahan et al. AISTATS (2017), Kairouz et al. (2021)]
Usual Approach to Federated Learning

The **FedAvg Algorithm** [McMahan et al. (2017)]:

*Step 1 of 3: Server broadcasts global model to sampled clients*

Parallel Gradient Distribution [Mangasarian. SICON (1995)]
Iterative Parameter Mixing [McDonald et al. ACL (2009)]
BMUF [Chen & Huo. ICASSP (2016)]
Local SGD [Stich. ICLR (2019)]
Usual Approach to Federated Learning

The FedAvg Algorithm [McMahan et al. (2017)]:

Step 1 of 3: Server broadcasts global model to sampled clients

Step 2 of 3: Clients perform some local SGD steps on their local data

Parallel Gradient Distribution [Mangasarian. SICON (1995)]
Iterative Parameter Mixing [McDonald et al. ACL (2009)]

BMUF [Chen & Huo. ICASSP (2016)]
Local SGD [Stich. ICLR (2019)]
Usual Approach to Federated Learning

**The FedAvg Algorithm** [McMahan et al. (2017)]:

**Step 1 of 3:** Server broadcasts global model to sampled clients

**Step 2 of 3:** Clients perform some local SGD steps on their local data

**Step 3 of 3:** Aggregate client updates securely

---

Parallel Gradient Distribution [Mangasarian. SICON (1995)]
Iterative Parameter Mixing [McDonald et al. ACL (2009)]

BMUF [Chen & Huo. ICASSP (2016)]
Local SGD [Stich. ICLR (2019)]
Usual Approach to Federated Learning

**The FedAvg Algorithm** [McMahan et al. (2017)]:

*Step 1 of 3: Server broadcasts global model to sampled clients*

*Step 2 of 3: Clients perform some local SGD steps on their local data*

*Step 3 of 3: Aggregate client updates securely*

---

Parallel Gradient Distribution [Mangasarian. SICON (1995)]
Iterative Parameter Mixing [McDonald et al. ACL (2009)]

BMUF [Chen & Huo. ICASSP (2016)]
Local SGD [Stich. ICLR (2019)]
Outline

- Background
- **Distributional Robustness with Simplicial-FL**
- Algorithm & Convergence Guarantees
- Numerical Results
Global model is trained on *average distribution* across clients (ERM)
Global model is deployed on *individual* clients.
Global model is deployed on *individual* clients

Train-test mismatch!
Global model is deployed on *individual* clients.
Our Approach: minimize the tail error directly!
Our Approach: minimize the tail error directly!

Simplicial-FL Objective:

$$\min_w S_\theta \left( (F_1(w), \cdots, F_n(w)) \right)$$

Superquantile | Conditional Value at Risk

\[ S_\theta(Z) = \mathbb{E}[Z \mid Z > Q_\theta(Z)] \]

[Rockafellar & Uryasev (2002)]
Distributional robustness

**Dual expression**

\[
\mathcal{S}_\theta(x_1, \ldots, x_n) = \max \left\{ \sum_i \pi_i x_i : \pi_i \geq 0, \sum_i \pi_1 = 1, \pi_i \leq p_i / \theta \right\}
\]

Assuming a new test client with mixture distribution \( p_\pi = \sum_i \pi_i p_i \),

Simplicial-FL objective is equivalent to:

\[
\min_w \max_{\pi: \pi \leq (n\theta)^{-1}} \mathbb{E}_{z \sim p_\pi} [f(w; z)]
\]

*Worst-case over a family of distributions*
Outline

• Background

• Distributional Robustness with Simplicial-FL

• Algorithm & Convergence Guarantees

• Numerical Results
Optimization

Simplicial-FL Objective:

\[ F_\theta(w) = \mathcal{S}_\theta \left( \left( F_1(w), \ldots, F_n(w) \right) \right) \]

Challenges:

- Superquantile is nonsmooth

- Superquantile is nonlinear (unbiased stochastic gradients not possible)
ERM Algorithm (FedAvg):

$$\min_w \frac{1}{n} \sum_{i=1}^{n} F_i(w)$$

Step 1 of 3: Server samples $m$ clients and broadcasts global model

Simplicial-FL Algorithm:

$$\min_w \min_{\theta} \left( \left( F_1(w), \ldots, F_n(w) \right) \right)$$

Step 1 of 3: Server samples $m$ clients and broadcasts global model
ERM Algorithm (FedAvg):

$$\min_w \frac{1}{n} \sum_{i=1}^{n} F_i(w)$$

*Step 2 of 3: Clients perform $\tau$ local SGD steps on their local data*

Simplicial-FL Algorithm:

$$\min_w \mathbb{S}_\theta\left( (F_1(w), \ldots, F_n(w)) \right)$$

*Step 2 of 3: Clients perform $\tau$ local SGD steps on their local data*
Step 3 of 3: Aggregate updates contributed by \textit{all clients}

\[ \min_{w} \frac{1}{n} \sum_{i=1}^{n} F_i(w) \]

\textbf{ERM Algorithm (FedAvg)}:

\textbf{Simplicial-FL Algorithm}:

\[ \min_{w} \mathbb{S}_\theta\left( (F_1(w), \ldots, F_n(w)) \right) \]

Step 3 of 3: Aggregate updates contributed by \textit{tail clients} only
Convergence (Non-convex)

**Nonsmooth**: Subdifferential from the chain rule

\[
\partial F_\theta(w) \ni \sum_{i=1}^{n} \pi^*_i \nabla F_i(w) \quad \text{where} \quad \pi^* \in \arg \max_{\pi \in \mathcal{P}_\theta} \sum_i \pi_i F_i(w)
\]
Convergence (Non-convex)

**Nonsmooth:** Subdifferential from the chain rule

\[
\partial F_\theta(w) \ni \sum_{i=1}^{n} \pi_i^* \nabla F_i(w) \quad \text{where} \quad \pi^* \in \arg\max_{\pi \in \mathcal{P}_\theta} \sum_i \pi_i F_i(w)
\]

**Nonlinear:** We optimize a surrogate

\[
\overline{F}_\theta(w) = \mathbb{E}_{S: |S|=m} \left[ S_\theta \left( \left( F_i(w) : i \in S \right) \right) \right]
\]
Theorem [P., Laguel, Malick, Harchaoui]

Suppose each $F_i$ is $L$-smooth and $G$-Lipschitz.

Then, Simplicial-FL satisfies the convergence guarantee:

$$\mathbb{E}\left\|\Phi_{\theta}^{2L}(w_t)\right\|^2 \leq \sqrt{\frac{\Delta_0LG^2}{t}} + (1 - \tau)^{1/3} \left(\frac{\Delta_0LG}{t}\right)^{2/3} + \frac{\Delta_0L}{t}$$

$t$: #comm. rounds
$	au$: #local update steps
$\Delta_0$: initial error

$$\Phi_{\theta}^\mu(w) = \inf_y \left\{ \Phi_{\theta}(y) + \frac{\mu}{2} \|y - w\|^2 \right\}$$

Moreau envelope of $F_{\theta}$ | well defined for $\mu > L$
**Convergence (strongly convex)**

**Nonsmooth**: Consider the smoothing

\[ F_{\theta}^{\nu}(w) = \max_{\pi \in \mathcal{P}_\theta} \left\{ \sum_i \pi_i F_i(w) - \nu \sum_i \pi_i \log \pi_i \right\} \]

Strongly convex

Neg. entropy

\[ \nabla F_{\theta}(w) = \sum_{i=1}^{n} [\pi_v]_i^* \nabla F_i(w) \]

where

\[ \pi_v^* = \arg \max_{\pi \in \mathcal{P}_\theta} \left\{ \sum_i \pi_i F_i(w) - \nu \sum_i \pi_i \log \pi_i \right\} \]

Infimal convolution smoothing [Nesterov. Math. Prog. (2005), Beck & Teboulle. SIOPT (2012)]
**Convergence (strongly convex)**

**Nonsmooth**: Consider the smoothing

\[
F^\nu_\theta(w) = \max_{\pi \in \mathcal{P}_\theta} \left\{ \sum_i \pi_i F_i(w) - \nu \sum_i \pi_i \log \pi_i \right\}
\]

\[
\nabla F_\theta(w) = \sum_{i=1}^n [\pi^*_i] \nabla F_i(w)
\]

where

\[
\pi^*_i = \arg \max_{\pi \in \mathcal{P}_\theta} \left\{ \sum_i \pi_i F_i(w) - \nu \sum_i \pi_i \log \pi_i \right\}
\]

\[
\pi^* = \arg \max_{\pi \in \mathcal{P}_\theta} \left\{ \sum_i \pi_i F_i(w) \right\}
\]

Infimal convolution smoothing [Nesterov. Math. Prog. (2005), Beck & Teboulle. SIOPT (2012)]
**Theorem** [P., Laguel, Malick, Harchaoui]

Suppose each $F_i$ is $L$-smooth and $G$-Lipschitz, and add a regularization $\frac{\lambda}{2}\|w\|^2$.

Then, Simplicial-FL satisfies the convergence guarantee:

$$\mathbb{E} \left[ F_{\theta}(w_t) - F^* \right] \leq \lambda \Delta_0 \exp \left( -\frac{t}{\sqrt{2}\kappa^3} \right) + \frac{G^2}{\lambda T} + \frac{G^2\kappa^2}{\lambda T^2}$$

$t$: #comm. rounds  
$\tau$: #local update steps  
$\Delta_0$: initial error

$\kappa = L/\lambda$ is the condition number
Outline

• Background

• Distributional Robustness with Simplicial-FL

• Algorithm & Convergence Guarantees

• Numerical Results
Experiments on EMNIST
Objective: Misclassification Error

<table>
<thead>
<tr>
<th>Method</th>
<th>Mean Value</th>
<th>90th Percentile</th>
</tr>
</thead>
<tbody>
<tr>
<td>FedAvg</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simplicial-FL</td>
<td>(\theta = 0.8)</td>
<td>(\theta = 0.5)</td>
</tr>
</tbody>
</table>

![Graphs showing the mean and 90th percentile of train loss and test misclassification error for FedAvg and Simplicial-FL with different values of \(\theta\).]
Experiments on EMNIST

Histogram of errors

- Simplicial-FL has the smallest 90th percentile error
- Simplicial-FL is competitive on the mean error
Distributionally robust learning in PyTorch

```python
import torch.nn.functional as F
from sqwash import reduce_supercquantile

for x, y in dataloader:
    y_hat = model(x)
    batch_losses = F.cross_entropy(y_hat, y, reduction='none')  # must set `reduction='none'`
    loss = reduce_supercquantile(batch_losses, superquantile_tail_fraction=0.5)  # Additional line
    loss.backward()  # Proceed as usual from here
...```

Install: `pip install sqwash`

Papers

Federated Learning with Heterogeneous Devices: A Superquantile Optimization Approach.
Under Review (arXiv 2112.09429)

A Superquantile Approach to Federated Learning with Heterogeneous Devices.

Superquantiles at Work: Machine Learning Applications and Efficient (Sub)gradient Computation.
Yassine Laguel, Krishna Pillutla, Jérôme Malick, Zaid Harchaoui.
Set-Valued and Variational Analysis (2021).

Code for experiments: https://github.com/krishnap25/simplicial-fl