Federated Hyperparameter Tuning: Challenges, Baselines, and Connections to Weight-Sharing

Misha Khodak, Renbo Tu, Tian Li, Liam Li, Nina Balcan, Virginia Smith, Ameet Talwalkar

khodak@cmu.edu

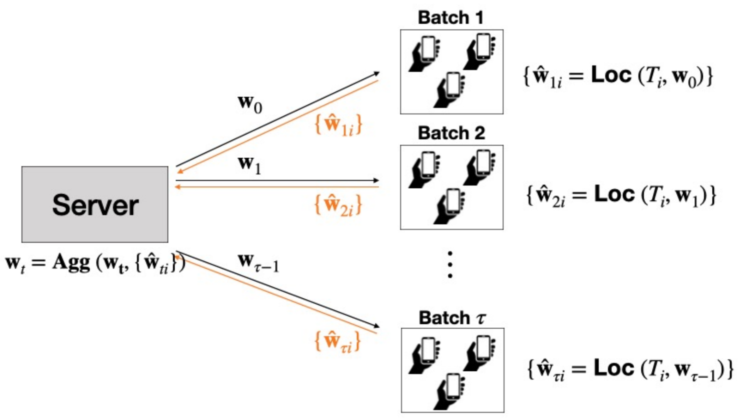
Hewlett Packard Enterprise

Carnegie Mellon University

Optimizing federated hyperparameters

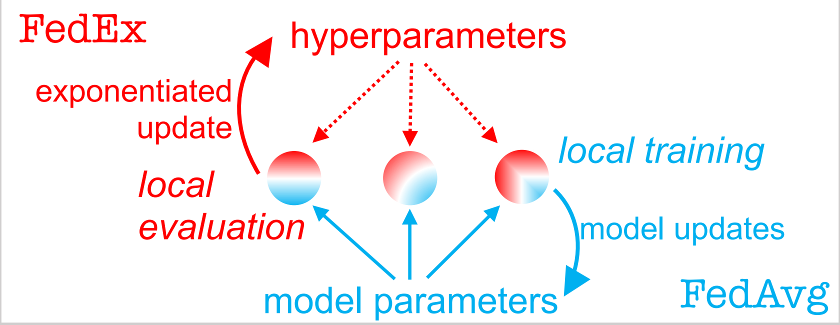
Tuning hyperparameters in federated learning is difficult:

1. Validation data is federated – cannot easily compute validation loss
2. Extreme resource limitations – cannot do many training runs
3. Evaluating personalization – personalized models require extra training to validate



Our contributions:

1. We adapt existing baselines such as random search and successive halving (SHA) to the federated setting and study their limitations
2. We propose a new algorithm called FedEx for tuning local hyperparameters that ameliorates the above challenges



FedEx: Tuning local hyperparameters

Most federated algorithms can be divided into two subroutines:

- $Loc_c(T, \mathbf{w})$ that runs local training (e.g. SGD) on dataset T from initialization $\mathbf{w}$
- $Agg_b(\mathbf{w}, \{\mathbf{w}_i\})$ that aggregates results $\{\mathbf{w}_i\}$ of local training and uses them to update the initialization $\mathbf{w}$

FedEx tunes the hyperparameters b of Agg_b:

- Can be formulated as an application of weight-sharing, a neural architecture search technique, to meta-learning
- Provably tunes the local step-size in the online convex optimization setting
- Applicable to any algorithm with the above structure

- FedAvg
- FedProx
- SCAFFOLD
- Reptile
- FedDyn
- FedPA
- MAML
- Ditto

Input: configurations $c_1, \dots, c_k \in \mathcal{C}$, setting b for Agg_b, schemes for setting step-size η_t and baseline λ_t, total number of steps $\tau \geq 1$

initialize $\theta_1 = \mathbf{1}_k / k$ and shared weights $\mathbf{w}_1 \in \mathbb{R}^d$

for *comm. round* $t = 1, \dots, \tau$ **do**

  for *client* $i = 1, \dots, B$ **do**

    send $\mathbf{w}_t, \theta_t$ to client

    sample $c_{ti} \sim \mathcal{D}_{\theta_t}$

    $\mathbf{w}_{ti} \leftarrow \mathrm{Loc}_{c_{ti}}(T_{ti}, \mathbf{w}_t)$

    send $\mathbf{w}_{ti}, c_{ti}, L_{V_{ti}}(\mathbf{w}_{ti})$ to server

  $\mathbf{w}_{t+1} \leftarrow \mathrm{Agg}_b(\mathbf{w}, \{\mathbf{w}_{ti}\}_{i=1}^B)$

  $\tilde{\nabla}_j \leftarrow \frac{\sum_{i=1}^B |V_{ti}| (L_{V_{ti}}(\mathbf{w}_{ti}) - \lambda_t) 1_{c_{ti}=c_j}}{\theta_{t[j]} \sum_{i=1}^B |V_{ti}|} \;\forall j$

  $\theta_{t+1} \leftarrow \theta_t \odot \exp(-\eta_t \tilde{\nabla})$

  $\theta_{t+1} \leftarrow \theta_{t+1} / \|\theta_{t+1}\|_1$

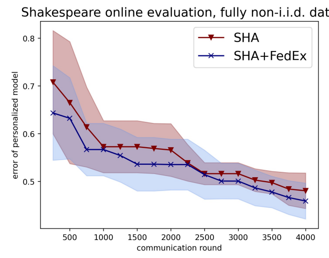
Output: model $\mathbf{w}$, hyperparameter distribution θ

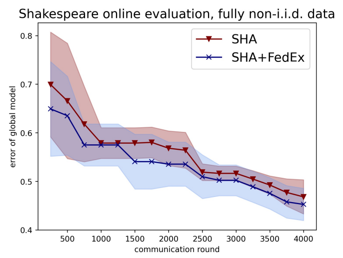
Experimental Results

Evaluations on three standard federated benchmarks

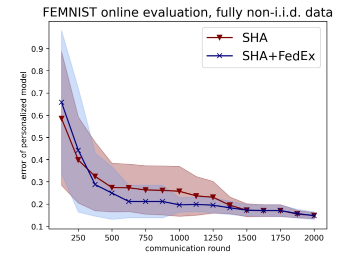
- Adapting SHA is a strong hyperparameter tuning baseline
- FedEx wrapped with SHA leads to consistent improvement in online and final evaluation settings, for both personalization and the global model

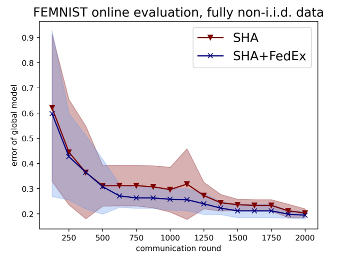
Shakespeare (non-i.i.d.)





FEMNIST (non-i.i.d.)





CIFAR-10 (i.i.d.)