Federated Residual Learning

Chen-Yu Wei
University of Southern California
chenyu.wei@usc.edu

Alekh Agarwal **John Langford**
Microsoft Research
{alekha, jcl}@microsoft.com

Abstract

We propose a new federated learning framework[1], Federated Residual Learning, which equips each client with a personalized local model that makes predictions jointly with the server-side shared model. Under this framework, we propose two algorithms FEDRESSGD and FEDRESAVG that improve traditional federated learning algorithms especially when the clients have different data distributions. Besides using personalized local models to better fit client-specific data, we address the slow convergence problem that is typically faced by traditional federated learning algorithms in non-i.i.d. environments. We provide both theoretical justifications and empirical evidence to show the superiority of our algorithms over baselines.

1 Introduction

Federated learning (FL) has become an important paradigm for large-scale distributed learning [1, 6, 9, 12]. A FL system typically involves a central server and a larger number of clients. The main characteristics of FL include: 1) by learning a shared model among all clients, the system can learn a significantly more accurate model than each client could achieve using their own local data; 2) by keeping the data locally and communicating with the server only sporadically, the clients largely reduce the communication cost, and at the same time preserve privacy. The second characteristic distinguishes FL from traditional centralized learning, making it a better framework for communication-limited networks (e.g., low-power sensor networks) or privacy-sensitive applications (e.g., healthcare applications).

The most popular federated learning algorithm is perhaps the FEDAVG algorithm by [9]. FEDAVG proceeds as follows: In every communication round between the server and the clients, the server broadcasts the current model to a randomly-sampled subset of clients. Then each selected client performs *multiple local updates* to the model before returning the *aggregated model update* to the server. Finally, the server updates the model as the *average* over the model updates from the clients.

FEDAVG works quite well when the data distribution is same on all clients. However, it suffers from the *client-drift* problem when the data distributions are different across clients, as identified by [7, 13]. The problem lies in that during the local update phase of the clients, their update directions become less and less aligned when the data distribution are not the same. This is harmful for the averaging phase, making the convergence slow. To address this issue, [7] incorporate *control variates* into their algorithm SCAFFOLD to correct the mismatch between the local update direction and the average update direction across the clients. The idea of control is closely related to the variance-reduction technique used in stochastic optimization, such as SVRG [5].

Although SCAFFOLD alleviates the slow convergence issue in the non-i.i.d. setting, all clients in their system still just learn a *global model* for making predictions. It is questionable whether such a single global model would perform well across the clients when their data distributions are different. Another naïve approach for non-i.i.d. environments is to learn different *local models* for

[1] Future updates of the paper will be provided in https://arxiv.org/abs/2003.12880.

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

each client, and solely use them to make predictions. The obvious drawback of this approach is that the system cannot exploit the similarity of the data across the clients, and thus a client with only few data samples cannot train a good model.

The goal of our work is to combine the benefits of the global model and the client-specific local models in the FL framework. One of our algorithms also incorporate the control-variate technique used by SCAFFOLD to accelerate convergence. Another benefit of our algorithms is that the construction of the local models is very flexible – each client can design their own local model (e.g., deciding the features and the structure of the model), and the server can be totally unaware of the designs. As we will see later, each local model is effectively fitting the *residual* between the ground-truth labels of that client and the prediction of the global model. Therefore, we call our framework Federated Residual Learning.

Our contributions can be summarized as follows: 1) we provide a new formulation for personalized federated learning, which we call Federated Residual Learning, 2) we propose FEDRESSGD that equips the global model with personalized local models, and prove its convergence guarantees, 3) we propose FEDRESAVG, which accelerates FEDRESSGD by incorporating local updates and variance-reduction techniques, and 4) we confirm the theoretical results on simulated and real-world data.

Related work. There have been prior works dealing with the heterogeneity of the data distribution in federated systems [1–4, 8, 10, 12]. However, a fundamental difference between our work and theirs is that their global and local models still operate in the same parameter space, while our framework provides more flexibility in the design of local models, as we will see in Section 2. This also forces us to design new way to perform the training. We are only aware of the concurrent work by [11] that supports global and local model splitting. They demonstrate the efficacy of this scheme through some primary empirical results. Compared to their work, our work provides a much more general framework, and formally establish theoretical results.

2 Problem Setting and Comparison

We consider a federated learning system with one server and N clients. Suppose that the global model is parameterized by $\boldsymbol{w}$, and the local model of client i is parameterized by $\boldsymbol{\theta}_i$. Then our problem is formulated as

$$\underset{\boldsymbol{w},\boldsymbol{\theta}_1,\ldots,\boldsymbol{\theta}_N}{\text{minimize}} \ \frac{1}{N}\sum_{i=1}^{N} L_i(\boldsymbol{w},\boldsymbol{\theta}_i), \tag{1}$$

where $L_i(\boldsymbol{w},\boldsymbol{\theta}_i)$ is the expected loss of client i under global model $\boldsymbol{w}$ and local model $\boldsymbol{\theta}_i$. We use $\mathcal{P}_i$ to denote the distribution of the loss functions of client i, and thus for any fixed $(\boldsymbol{w},\boldsymbol{\theta}_i)$, we have $\mathbb{E}_{\ell\sim\mathcal{P}_i}[\ell(\boldsymbol{w},\boldsymbol{\theta}_i)] = L_i(\boldsymbol{w},\boldsymbol{\theta}_i)$. In general, we allow the loss function ℓ to take any form. A simple example for least-square linear regression problems is

$$\ell(\boldsymbol{w},\boldsymbol{\theta}) = \left(y - \boldsymbol{w}^\top \boldsymbol{x}_g - \boldsymbol{\theta}^\top \boldsymbol{x}_l\right)^2, \tag{2}$$

where y is the label, $\boldsymbol{x}_g$ is the global feature, and $\boldsymbol{x}_l$ is the local feature. Note that $(\boldsymbol{x}_g, \boldsymbol{x}_l, y)$ jointly defines the loss function ℓ, and therefore, $\ell \sim \mathcal{P}_i$ is equivalent to sampling $(\boldsymbol{x}_g, \boldsymbol{x}_l, y)$ from some fixed distribution specific to client i. We allow $\boldsymbol{x}_g$ and $\boldsymbol{x}_l$ to be *same* or *different*, or just having partially overlapping entries. This demonstrates the flexibility of our formulation – it is useful when a client hopes to incorporate some local and special features that are not available at other clients. For non-linear regression, one can replace $\boldsymbol{w}^\top \boldsymbol{x}_g$ and $\boldsymbol{\theta}^\top \boldsymbol{x}_l$ with non-linear functions $f_{\boldsymbol{w}}(\boldsymbol{x}_g)$ and $f_{\boldsymbol{\theta}}(\boldsymbol{x}_l)$ parameterized by $\boldsymbol{w}$ and $\boldsymbol{\theta}$ respectively.

Another example for multi-class linear classification problems is

$$\ell(\boldsymbol{W},\boldsymbol{\Theta}) = \log\left(\frac{\sum_z \exp\left(\boldsymbol{e}_z^\top(\boldsymbol{W}\boldsymbol{x}_g + \boldsymbol{\Theta}\boldsymbol{x}_l)\right)}{\exp\left(\boldsymbol{e}_y^\top(\boldsymbol{W}\boldsymbol{x}_g + \boldsymbol{\Theta}\boldsymbol{x}_l)\right)}\right), \tag{3}$$

where y is the class label, and $\boldsymbol{W}$ and $\boldsymbol{\Theta}$ are matrices whose number of rows is equal to the number of classes. This is the standard logistic loss, but with the logit vector replaced by $\boldsymbol{W}\boldsymbol{x}_g + \boldsymbol{\Theta}\boldsymbol{x}_l$.

Similarly, for non-linear classifiers like neural networks, we can replace $\boldsymbol{W}\boldsymbol{x}_g$ and $\boldsymbol{\Theta}\boldsymbol{x}_l$ with non-linear functions $f_{\boldsymbol{W}}(\boldsymbol{x}_g)$ and $f_{\boldsymbol{\Theta}}(\boldsymbol{x}_l)$ which have vector outputs.

We now compare our problem formulation (1) with other formulations appeared in previous works. The standard FL formulation is

$$\underset{\boldsymbol{w}}{\text{minimize}} \ \frac{1}{N}\sum_{i=1}^{N} L_i(\boldsymbol{w}), \tag{4}$$

which is adopted by FEDAVG [9] and SCAFFOLD [7], even though the latter's goal is to address some issues in the non-i.i.d. setting. Since (4) does not consider personalized models, even if we find its global minimizer $\boldsymbol{w}^*$, the performance can still be arbitrarily bad on some of the clients that are very different from the majority of other clients.

Another line of research considers personalized models with regularization on the distance between the global model and local models:

$$\underset{\boldsymbol{w},\boldsymbol{\theta}_1,\ldots,\boldsymbol{\theta}_N}{\text{minimize}} \ \frac{1}{N}\sum_{i=1}^{N}\left(L_i(\boldsymbol{w}+\boldsymbol{\theta}_i)+\frac{\lambda}{2}\|\boldsymbol{\theta}_i\|^2\right), \qquad \text{s.t.} \quad \sum_{i=1}^{N}\boldsymbol{\theta}_i = \boldsymbol{0}. \tag{5}$$

This formulation is adopted by FEDPROX [8] and L2GD [3][2]. Apart from the regularization term $\|\boldsymbol{\theta}_i\|^2$ and the constraint, their objective function $L_i(\boldsymbol{w}+\boldsymbol{\theta}_i)$ is a special case of our $L_i(\boldsymbol{w},\boldsymbol{\theta}_i)$, and coincides with the linear models (2) and (3) when $\boldsymbol{x}_g = \boldsymbol{x}_l$. However, this formulation restricts that the global model $\boldsymbol{w}$ and the local model $\boldsymbol{\theta}_i$ lie in the same parameter space, which lacks the flexibility for the case $\boldsymbol{x}_g \neq \boldsymbol{x}_l$. Also, it only allows the global and local models to be combined in the parameter space, but not in the label or logit space like the extension of (2) and (3) to the non-linear case.

The additional constraint and the regularization term $\|\boldsymbol{\theta}_i\|^2$ in (5) essentially make the global model $\boldsymbol{w}$ be the center of all personalized models $\{\boldsymbol{w}+\boldsymbol{\theta}_i\}_{i=1}^N$, and restrict that each personalized model be close to each other. These additional regularizations might be helpful when the data distributions of the clients are known to be close to each other, but may be harmful if some clients turn out to be quite distinct from others. Below we provide a toy example to show that our formulation (1) might be a better choice compared to (4) or (5) when the data distributions of the clients are not close to each other.

Example 1. Consider a restaurant recommendation system that is trained in a federated manner, and for simplicity, we only consider two clients. The goal of the recommendation system is to predict the rating of each user based on some features of the restaurants. There are four features: *cleanness*, *price*, *quietness*, *spiciness*, encoded as a four-dimensional vector $\boldsymbol{x}$. Both users like clean restaurants with low price. User A likes to eat light-flavored food in a quiet space, while User B prefers to have spice-flavored food in a loud environment. The expected ratings of the users are thus given by the following:

$$\mu_A(\boldsymbol{x}) = x_1 - x_2 + x_3 - x_4,$$
$$\mu_B(\boldsymbol{x}) = x_1 - x_2 - x_3 + x_4.$$

Assume that the true rating of User i is given by $r_i(\boldsymbol{x}) \sim \mu_i(\boldsymbol{x}) + \mathcal{N}(0,\sigma^2)$ for $i = A, B$, and the prediction loss for a prediction r is defined as $(r - r_i(\boldsymbol{x}))^2$. For simplicity, we also assume that $\boldsymbol{x}$ is i.i.d. drawn from $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_4)$. Then the expected prediction loss given by the best linear models under (1), (4), (5) are σ^2, σ^2+2, and $\sigma^2 + 2\left(\lambda/(2+\lambda)\right)^2$ respectively. We provide the calculation in the appendix.

Notations and assumptions. We denote the gradient of the losses with respect to global parameters and local parameters by

$$\nabla_{\boldsymbol{w}}\ell(\boldsymbol{w},\boldsymbol{\theta}) \triangleq \frac{\partial}{\partial \boldsymbol{w}}\ell(\boldsymbol{w},\boldsymbol{\theta}), \quad \nabla_{\boldsymbol{\theta}}\ell(\boldsymbol{w},\boldsymbol{\theta}) \triangleq \frac{\partial}{\partial \boldsymbol{\theta}}\ell(\boldsymbol{w},\boldsymbol{\theta}),$$

and let $\nabla\ell(\boldsymbol{w},\boldsymbol{\theta}) = (\nabla_{\boldsymbol{w}}\ell(\boldsymbol{w},\boldsymbol{\theta}), \nabla_{\boldsymbol{\theta}}\ell(\boldsymbol{w},\boldsymbol{\theta}))$ be the full gradient.

To derive our theoretical results, we make the following standard assumptions.

[2]The formulation in [3, 8] looks different from (5), but is actually equivalent to it. They write the objective as $\frac{1}{N}\sum_{i=1}^N \left(L_i(\boldsymbol{w}_i) + \frac{\lambda}{2}\|\boldsymbol{w}_i - \overline{\boldsymbol{w}}\|^2\right)$ with $\overline{\boldsymbol{w}} = \frac{1}{N}\sum_{i=1}^N \boldsymbol{w}_i$.

Algorithm 1 FEDAVG

input parameters: η, K
for $r = 1, \ldots, R$ **do**
  communicate $\boldsymbol{w}$ to all clients
  for *all* i *in parallel* **do**
    initialize $\boldsymbol{w}_i \leftarrow \boldsymbol{w}$
    for $k = 1, \ldots, K$ **do**
      sample a mini-batch with average loss ℓ
      update $\boldsymbol{w}_i \leftarrow \boldsymbol{w}_i - \eta \nabla \ell(\boldsymbol{w}_i)$
    send $\boldsymbol{w}_i - \boldsymbol{w}$ to the server
  perform averaging:

$$\boldsymbol{w} \leftarrow \boldsymbol{w} + \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{w}_i - \boldsymbol{w})$$

Algorithm 2 FEDRESNAIVE

input parameters: η, λ, K
for $r = 1, \ldots, R$ **do**
  communicate $\boldsymbol{w}$ to all clients
  for *all* i *in parallel* **do**
    initialize $\boldsymbol{w}_i \leftarrow \boldsymbol{w}$
    for $k = 1, \ldots, K$ **do**
      sample a mini-batch with average loss ℓ
      update $\boldsymbol{w}_i \leftarrow \boldsymbol{w}_i - \eta \nabla_{\boldsymbol{w}} \ell(\boldsymbol{w}_i, \boldsymbol{\theta}_i)$
      update $\boldsymbol{\theta}_i \leftarrow \boldsymbol{\theta}_i - \lambda \nabla_{\boldsymbol{\theta}} \ell(\boldsymbol{w}_i, \boldsymbol{\theta}_i)$
    send $\boldsymbol{w}_i - \boldsymbol{w}$ to the server
  perform averaging for the global parameter:

$$\boldsymbol{w} \leftarrow \boldsymbol{w} + \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{w}_i - \boldsymbol{w})$$

Assumption 1. *The expected loss functions of all clients are convex and γ-smooth, i.e.,*

$$\|\nabla L_i(\boldsymbol{w}, \boldsymbol{\theta}) - \nabla L_i(\boldsymbol{w}', \boldsymbol{\theta}')\| \leq \gamma \left(\|\boldsymbol{w} - \boldsymbol{w}'\| + \|\boldsymbol{\theta} - \boldsymbol{\theta}'\| \right)$$

for any $i, \boldsymbol{w}, \boldsymbol{w}', \boldsymbol{\theta}, \boldsymbol{\theta}'$ (all norms considered in this paper are Euclidean norms).

Assumption 2. *The gradient of the expected loss functions have bounded magnitude, i.e.,*

$$\|\nabla L_i(\boldsymbol{w}, \boldsymbol{\theta})\| \leq G$$

for any $i, \boldsymbol{w}, \boldsymbol{\theta}$.

For a random vector $\boldsymbol{v}$, we use $\mathbb{V}[\boldsymbol{v}]$ to denote its variance defined by $\mathbb{E}[\|\boldsymbol{v} - \mathbb{E}[\boldsymbol{v}]\|^2] = \text{trace}\left(\text{Cov}[\boldsymbol{v}]\right)$. We may use subscripts of $\mathbb{V}$ to explicitly indicate the distribution that $\boldsymbol{v}$ is drawn from.

Assumption 3. *The gradients of the loss functions of all clients have bounded variance:*

$$\mathbb{V}_{\ell \sim \mathcal{P}_i} \left[\nabla \ell(\boldsymbol{w}, \boldsymbol{\theta}) \right] \leq \sigma^2$$

for any $i, \boldsymbol{w}, \boldsymbol{\theta}$, where $\mathcal{P}_i$ is defined in the beginning of Section 2 to denote the distribution of the loss function of client i.

In some of the theorems, we further make the assumption that the expected loss functions $L(\boldsymbol{w}, \{\boldsymbol{\theta}_i\}_{i=1}^N)$ is strongly-convex in $\boldsymbol{w}$ and $\{\boldsymbol{\theta}\}_{i=1}^N$.

3 Algorithms

For simplicity, we study the case without client sampling (see, e.g., [7]). That is, in every round, every client is involved in training. Algorithmically, it is straightforward to extend our algorithms to the case with client sampling. Under this scenario, we first review the standard FEDAVG algorithm, whose pseudocode is presented in Algorithm 1. In each communication round r, the server first communicates the current global model $\boldsymbol{w}$ to the clients. Then each client performs K steps of *local updates* on the global parameter using local samples, and then returns the aggregated update $\boldsymbol{w}_i - \boldsymbol{w}$ to the server (Line 1). Finally, in Line 1, the server takes average over the returned updates, and applies it to the global model, which is then used in the next round.

To extend FEDAVG to the Federated Residual Learning framework that jointly learns a global model and multiple local models, our first attempt is to perform gradient descent on global and local parameters simultaneously, which results in FEDRESNAIVE (Algorithm 2), which also resembles the update rule of [2] (although their global and local models are in the same parameter space). Note that this is a warm-up algorithm that we want to improve, since it has some undesirable properties.

To see the issue of updating the global parameter $\boldsymbol{w}$ and the local parameter $\boldsymbol{\theta}_i$ simultaneously (Line 2 and 2 in Algorithm 2), note that during local updates, the $\boldsymbol{w}_i$ maintained by each client is "temporary", i.e., Line 2 of Algorithm 2 does not reflect the true update of the global parameter $\boldsymbol{w}$. Rather, the true update of $\boldsymbol{w}$ happens in Line 2, when the server takes average over the temporary $\boldsymbol{w}_i$'s. We also note that the gradient with respect to the local parameter $\boldsymbol{\theta}_i$ depends on the choice of the global parameter $\boldsymbol{w}_i$. Therefore, if each client performs simultaneous update on their local copies of $\boldsymbol{w}_i$ and $\boldsymbol{\theta}_i$, their $\boldsymbol{\theta}_i$ would be updated with respect to the wrong (temporary) $\boldsymbol{w}_i$, which will be over-written later in Line 2. Therefore, $\boldsymbol{\theta}_i$ might be updated towards wrong directions. How wrong it might be depends on how different the distributions of the clients are.

Based on the observation above, we propose to always update local parameters under a stable version of global parameter that is synchronized with the server. In the next two subsections, we propose two algorithms that are based on this idea, and provide convergence guarantees that do not have an explicit dependence on the degree of distribution mismatch among the clients.

3.1 FEDRESSGD

Our first algorithm, FEDRESSGD, is an extension of the FEDSGD algorithm [9] to the Federated Residual Learning setting. FEDSGD only serves as a baseline algorithm in [9], because it does not perform multiple steps of local updates on the global parameter, and might converge slow. However, since updating local parameters does not involve synchronization with the server, each client can still quickly find a good model for its own data by training the local model on top of the current version of the global model. Thus, our algorithm can be viewed as a direct remedy for the slow update problem of FEDSGD. The pseudocode is presented in Algorithm 3.

Observe that in Algorithm 3, each client has decoupled updates for the local parameter (in Step 1) and the global parameter (in Step 2): when one is being updated, the other is fixed. Also, the local parameter update (Step 1) is always under a fixed global parameter that has just been synchronized with the server, thus resolving the issue of FEDRESNAIVE. When performing global parameter update (Step 2), each client only provides *one step* gradient, and sends it to the server. This is similar to FEDSGD.

Mathematically, FEDRESSGD is similar to *block coordinate descent*, with two groups of variables $\{\boldsymbol{w}\}$, $\{\boldsymbol{\theta}_i\}_{i=1}^N$ updated in an interleaved manner. The difference is that for the group $\{\boldsymbol{\theta}_i\}_{i=1}^N$ we only sub-sample some of them in each round, and perform multiple steps of updates.

For FEDRESSGD, the following theorem gives its convergence guarantees.

Theorem 1. *Let $\lambda = \frac{1}{N}\eta$. Let $\boldsymbol{w}^r, \boldsymbol{\theta}_i^r$ be the values of the parameters at the beginning of round r ($\boldsymbol{w}^1, \boldsymbol{\theta}_i^1$ are initial parameters), and let*

$$\Delta^{(r)} = \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N L_i(\boldsymbol{w}^r, \boldsymbol{\theta}_i^r)\right] - \frac{1}{N}\sum_{i=1}^N L_i(\boldsymbol{w}^*, \boldsymbol{\theta}_i^*).$$

With proper choices of η, FEDRESSGD *guarantees*

$$\Delta^{(R+1)} = \mathcal{O}\left(\left(\frac{D^4\gamma\sigma^2}{NKR}\right)^{\frac{1}{3}} + \sqrt{\frac{D^2\gamma G}{NR}} + \frac{D^2\gamma}{R}\right)$$

after R communication rounds, where $D = \sup_r \sqrt{\|\boldsymbol{w}^r - \boldsymbol{w}^*\|^2 + \sum_i \|\boldsymbol{\theta}_i^r - \boldsymbol{\theta}_i^*\|^2}$. If we further assume that the loss is β-strongly convex, then FEDRESSGD *guarantees*

$$\Delta^{(R+1)} = \widetilde{\mathcal{O}}\left(\frac{\gamma}{\beta}\frac{\sigma^2}{\beta NKR}\right).$$

Note that unlike previous works about non-i.i.d. FL [2, 8], our bound does not have explicit dependence on the degree of distribution mismatch among the clients.

3.2 FEDRESAVG

Although FEDRESSGD is theoretically sound, one might still hope to accelerate the convergence of the global model. To improve it, we perform multiple steps of updates on the global parameter in

Algorithm 3 FEDRESSGD

input parameters: η, λ, K
for $r = 1, \ldots, R$ **do**
  communicate $\boldsymbol{w}$ to all clients
  for *all* i *in parallel* **do**
    ▷ Step 1. update local parameters
    for $k = 1, \ldots, K$ **do**
      sample a mini-batch with average loss ℓ
      update $\boldsymbol{\theta}_i \leftarrow \boldsymbol{\theta}_i - \lambda \nabla_{\boldsymbol{\theta}} \ell(\boldsymbol{w}, \boldsymbol{\theta}_i)$
    ▷ Step 2. update global parameters
    sample a large batch with average loss ℓ
    (large batch size $= K \times$ mini-batch size)
    assign $\boldsymbol{w}_i \leftarrow \boldsymbol{w} - \eta K \nabla_{\boldsymbol{w}} \ell(\boldsymbol{w}, \boldsymbol{\theta}_i)$
    send $\boldsymbol{w}_i - \boldsymbol{w}$ to the server
  perform averaging for the global parameter:

$$\boldsymbol{w} \leftarrow \boldsymbol{w} + \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{w}_i - \boldsymbol{w})$$

Algorithm 4 FEDRESAVG

input parameters: η, λ, K, α
for $r = 1, \ldots, R$ **do**
  communicate $\boldsymbol{w}$ to all clients
  for *all* i *in parallel* **do**
    ▷ Step 1. update local parameters
    for $k = 1, \ldots, K$ **do**
      sample a mini-batch with average loss ℓ
      update $\boldsymbol{\theta}_i \leftarrow \boldsymbol{\theta}_i - \lambda \nabla_{\boldsymbol{\theta}} \ell(\boldsymbol{w}, \boldsymbol{\theta}_i)$
    ▷ Step 2. update global parameters
    initialize $\boldsymbol{w}_i \leftarrow \boldsymbol{w}$
    for $k = 1, \ldots, K$ **do**
      sample a mini-batch with average loss ℓ
      compute $\boldsymbol{g}_i^{(k)} \triangleq \nabla_{\boldsymbol{w}} \ell(\boldsymbol{w}_i, \boldsymbol{\theta}_i)$
      update $\boldsymbol{w}_i \leftarrow \boldsymbol{w}_i - \eta \left(\boldsymbol{g}_i^{(k)} - \boldsymbol{c}_i + \boldsymbol{c} \right)$
    $\boldsymbol{c}_i \leftarrow \begin{cases} \mathbf{0} & \text{(Option I)} \\ \frac{1}{K} \sum_{i=1}^{K} \boldsymbol{g}_i^{(k)} & \text{(Option II)} \end{cases}$
    send $\boldsymbol{w}_i - \boldsymbol{w}, \boldsymbol{c}_i$ to the server
  perform averaging for the global parameter:

$$\boldsymbol{w} \leftarrow \boldsymbol{w} + \frac{\alpha}{N} \sum_{i=1}^{N} (\boldsymbol{w}_i - \boldsymbol{w}), \quad \boldsymbol{c} \leftarrow \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{c}_i$$

one communication round, like how FEDAVG improves over FEDSGD. The resulted algorithm is FEDRESAVG (Algorithm 4).

We provide two versions of it (specified by Option I and II in Line 4 of Algorithm 4). For Option I, we simply replace the single update step in FEDRESSGD by multiple updates (Line 4-4). However, since the global parameter is not synchronized during the local updates, the problem of *client-drift* emerges, and the convergence rate is provably deteriorated when the data is not i.i.d. across the clients [7].

To address this problem, we adopt the idea of SCAFFOLD that uses *control variates* to correct the distribution mismatch. This results in the Option II of Algorithm 4. To see how the control variates resolve the issue, notice that the best update direction of the global parameter is $\frac{1}{N} \sum_{j=1}^{N} \nabla_{\boldsymbol{w}} L_j(\boldsymbol{w}, \boldsymbol{\theta}_j)$. With Option II, since $\boldsymbol{g}_i \approx \nabla_{\boldsymbol{w}} L_i(\boldsymbol{w}, \boldsymbol{\theta}_i)$, $\boldsymbol{c}_i \approx \nabla_{\boldsymbol{w}} L_i(\boldsymbol{w}, \boldsymbol{\theta}_i)$, and $\boldsymbol{c} \approx \frac{1}{N} \sum_{j=1}^{N} \nabla_{\boldsymbol{w}} L_i(\boldsymbol{w}, \boldsymbol{\theta}_j)$, we have $\boldsymbol{g}_i^{(k)} - \boldsymbol{c}_i + \boldsymbol{c} \approx \frac{1}{N} \sum_{j=1}^{N} \nabla_{\boldsymbol{w}} L_j(\boldsymbol{w}, \boldsymbol{\theta}_j)$, matching the update direction we want.

For FEDRESAVG with control variates (i.e., Option II), we have the following guarantee for its convergence.

Theorem 2. *We use the same notations as in Theorem 1. With proper choices of η, for general γ-smooth losses,* FEDRESAVG *guarantees*

$$\Delta^{(R+1)} = \mathcal{O}\left(\sqrt[3]{\frac{D^4 \gamma \sigma^2}{NKR}} \right)$$

after R communication rounds. If we further assume that the loss is β-strongly convex, then FEDRESAVG *guarantees*

$$\Delta^{(R+1)} = \widetilde{\mathcal{O}}\left(\frac{\gamma}{\beta} \frac{\sigma^2}{\beta NKR} \right).$$

Remark. In Theorem 2, we only show the dominant terms (in terms of R) in the bounds. For details on the lower-order terms, please check the appendix. We also note that these dominant terms match those in Theorem 1. However, as observed in previous works [7, 9], although it is challenging to show the theoretical benefits of local updates, it usually performs better in experiments.

References

[1] Fei Chen, Mi Luo, Zhenhua Dong, Zhenguo Li, and Xiuqiang He. Federated meta-learning with fast convergence and efficient communication. *CoRR*, abs/1802.07876, 2018.

[2] Yuyang Deng, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. Adaptive personalized federated learning. *arXiv preprint arXiv:2003.13461*, 2020.

[3] Filip Hanzely and Peter Richtárik. Federated learning of a mixture of global and local models. *arXiv preprint arXiv:2002.05516*, 2020.

[4] Yihan Jiang, Jakub Konečnỳ, Keith Rush, and Sreeram Kannan. Improving federated learning personalization via model agnostic meta learning. *arXiv preprint arXiv:1909.12488*, 2019.

[5] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in neural information processing systems*, pages 315–323, 2013.

[6] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.

[7] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank J Reddi, Sebastian U Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for on-device federated learning. *arXiv preprint arXiv:1910.06378*, 2019.

[8] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *arXiv preprint arXiv:1812.06127*, 2018.

[9] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282, 2017.

[10] Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. Agnostic federated learning. In *ICML*, 2019.

[11] Daniel Peterson, Pallika Kanani, and Virendra J Marathe. Private federated learning with domain adaptation. *International Workshop on Federated Learning for User Privacy and Data Confidentiality*, 2019.

[12] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S Talwalkar. Federated multi-task learning. In *Advances in Neural Information Processing Systems*, pages 4424–4434, 2017.

[13] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*, 2018.

A Auxiliary Lemmas

Lemma 1. *For any γ-smooth convex function f and any $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}$ in its domain,*

$$f(\boldsymbol{a}) - f(\boldsymbol{b}) \leq (\boldsymbol{a} - \boldsymbol{b}) \cdot \nabla f(\boldsymbol{c}) + \frac{\gamma}{2}\|\boldsymbol{a} - \boldsymbol{c}\|^2.$$

Lemma 2. *For any β-strongly convex function f with minimizer $\boldsymbol{x}^*$,*

$$f(\boldsymbol{x}) - f(\boldsymbol{x}^*) \leq \frac{1}{2\beta}\left\|\nabla f(\boldsymbol{x})\right\|^2.$$

Proof. By the definition of strongly convexity, we have

$$f(\boldsymbol{x}') \geq f(\boldsymbol{x}) + \langle \nabla f(\boldsymbol{x}), \boldsymbol{x}' - \boldsymbol{x}\rangle + \frac{\beta}{2}\|\boldsymbol{x}' - \boldsymbol{x}\|^2$$

Using $\langle \boldsymbol{u}, \boldsymbol{v}\rangle + \frac{\beta}{2}\|\boldsymbol{v}\|^2 \geq -\frac{1}{2\beta}\|\boldsymbol{u}\|^2$, we see that the right-hand side above is lower bounded by

$$f(\boldsymbol{x}) - \frac{1}{2\beta}\left\|\nabla f(\boldsymbol{x})\right\|^2.$$

Since this holds for any $\boldsymbol{x}'$, we let $\boldsymbol{x}' = \boldsymbol{x}^*$ and get $f(\boldsymbol{x}^*) \geq f(\boldsymbol{x}) - \frac{1}{2\beta}\left\|\nabla f(\boldsymbol{x})\right\|^2$, which finishes the proof. □

Lemma 3. *Let $\{B_r\}_{r=1,2,\ldots}$ be a non-negative sequence that satisfies the following:*

$$B_{r+1} \leq B_r - c(B_r)^2 + d$$

for some $c, d > 0$ such that $B_r \leq \frac{1}{2c}$ for all r. Then $B_r \leq \frac{1}{cr} + \sqrt{\frac{d}{c}}$ for all $r \geq 1$.

Proof. We use induction to prove this. When $r = 1$, the bound holds since $B_1 \leq \frac{1}{2c} \leq \frac{1}{cr}$.

Suppose $B_k \leq \frac{1}{ck} + \sqrt{\frac{d}{c}}$ for all $k \leq r$. Notice that the function $g(B_r) \triangleq B_r - c(B_r)^2$ is increasing in B_r since $g'(B_r) = 1 - 2cB_r \geq 0$ by the assumption that $B_r \leq \frac{1}{2c}$. Then by induction, we have

$$\begin{aligned}
B_{r+1} &\leq \frac{1}{cr} + \sqrt{\frac{d}{c}} - c\left(\frac{1}{cr} + \sqrt{\frac{d}{c}}\right)^2 + d \\
&= \frac{1}{cr} + \sqrt{\frac{d}{c}} - \frac{1}{cr^2} - \frac{2}{r}\sqrt{\frac{d}{c}} - d + d \\
&\leq \frac{1}{c}\left(\frac{1}{r} - \frac{1}{r^2}\right) + \sqrt{\frac{d}{c}} \\
&\leq \frac{1}{c(r+1)} + \sqrt{\frac{d}{c}}.
\end{aligned}$$

□

Lemma 4. *Let $\{B_r\}_{r=1,2,\ldots}$ be a non-negative sequence that satisfies the following:*

$$B_{r+1} \leq B_r - cB_r + d$$

for some $0 < c < 1$ and $d > 0$. Then $B_r \leq B_1\left(1-c\right)^{r-1} + \frac{d}{c}$.

Proof. We use induction to prove this. The case of $r = 1$ is clear. Suppose that $B_k \leq B_1(1-c)^{k-1} + \frac{d}{c}$ for all $k \leq r$. Then

$$B_{r+1} \leq (1-c)\left(B_1(1-c)^{r-1} + \frac{d}{c}\right) + d = B_1(1-c)^r + \frac{d}{c}.$$

□

B Detailed Calculation for Example 1

Under the Federated Residual Learning framework (1), an optimal solution is $\boldsymbol{w} = (1, 1, 0, 0), \boldsymbol{\theta}_A = (0, 0, 1, -1), \boldsymbol{\theta}_B = (0, 0, -1, 1)$. Since this perfectly predict the mean of each user's utility, the squared loss is simply the variance, i.e., σ^2.

Under the single global model framework (4), the optimal model is $\boldsymbol{w} = (1, 1, 0, 0)$. For each client, the mean error is $\|(1, 1, 0, 0) - (1, 1, 1, -1)\| = \sqrt{2}$. The square of the mean error would be added to the expected loss, which results in the expected prediction error $\sigma^2 + 2$.

Under the personalized model with proximal term framework (5), the optimal solution of $\boldsymbol{\theta}_A$ is of the form $(0, 0, \theta, -\theta)$. θ determined by the solution of the following:

$$\min_{\theta}(1 - \theta)^2 + \frac{\lambda}{2}\theta^2.$$

Solving this, we get $\theta = \frac{2}{2+\lambda}$. Therefore, the mean error is $(0, 0, 1 - \theta, -1 + \theta) = (0, 0, \frac{\lambda}{2+\lambda}, \frac{-\lambda}{2+\lambda})$ for user A. The expected prediction error would then be $\sigma^2 + 2(\frac{\lambda}{2+\lambda})^2$.

C Analysis for FEDRESSGD

In this section, we provide detailed analysis for Theorem 1. The analysis for Theorem 2 as well as experiments are deferred to the appendix.

Lemma 5. *Consider a specific communication round r in Algorithm 3. Let the global parameter be $\boldsymbol{w}$, and the local parameters be $\{\boldsymbol{\theta}_i\}_{i=1}^N$ at the beginning of communication round r. Also, let the local parameters be $\{\boldsymbol{\theta}_i'\}_{i=1}^N$ at the end of Step 1. If $\lambda \leq \frac{1}{\gamma}$, then conditioned on all history before round r, we have*

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i'\right)\right] - \frac{1}{N}\sum_{i=1}^{N} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i\right) \\
&\leq \frac{-\lambda K}{4N}\sum_{i=1}^{N}\left\|\nabla_{\boldsymbol{\theta}} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i\right)\right\|^2 + \frac{\gamma\lambda^2\sigma^2 K}{2} + \frac{\gamma^2\lambda^3 G^2 K^2}{2}.
\end{aligned}$$

Proof. We first consider a fixed i. Let $\boldsymbol{\theta}^{[0]} = \boldsymbol{\theta}_i$ and $\boldsymbol{\theta}^{[K]} = \boldsymbol{\theta}_i'$, and let $\boldsymbol{\theta}^{[k]}$ be the $\boldsymbol{\theta}_i$ at the end of the k-th for-loop in Step 1. Consider the update of client i's local parameter from $\boldsymbol{\theta}^{[k]}$ to $\boldsymbol{\theta}^{[k+1]}$, taking expectation conditioned on the history up to the k-th for-loop:

$$\begin{aligned}
&\mathbb{E}\left[L_i\left(\boldsymbol{w}, \boldsymbol{\theta}^{[k+1]}\right)\right] - L_i\left(\boldsymbol{w}, \boldsymbol{\theta}^{[k]}\right) \\
&\leq -\lambda\mathbb{E}_{\ell\sim\mathcal{P}_i}\left[\left\langle\nabla_{\boldsymbol{\theta}}\ell\left(\boldsymbol{w}, \boldsymbol{\theta}^{[k]}\right), \nabla_{\boldsymbol{\theta}} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}^{[k]}\right)\right\rangle\right] \\
&\quad + \frac{\gamma\lambda^2}{2}\mathbb{E}_{\ell\sim\mathcal{P}_i}\left[\left\|\nabla_{\boldsymbol{\theta}}\ell(\boldsymbol{w}, \boldsymbol{\theta}^{[k]})\right\|^2\right] && \text{(by Lemma 1)} \\
&\leq -\left(\lambda - \frac{\gamma\lambda^2}{2}\right)\left\|\nabla_{\boldsymbol{\theta}} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}^{[k]}\right)\right\|^2 \\
&\quad + \frac{\gamma\lambda^2}{2}\mathbb{V}_{\ell\sim\mathcal{P}_i}\left[\nabla_{\boldsymbol{\theta}}\ell\left(\boldsymbol{w}, \boldsymbol{\theta}^{[k]}\right)\right] && \text{(because } \mathbb{E}_{\ell\sim\mathcal{P}_i}[\nabla_{\boldsymbol{\theta}}\ell\left(\boldsymbol{w}, \boldsymbol{\theta}^{[k]}\right)] = \nabla_{\boldsymbol{\theta}} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}^{[k]}\right)) \\
&\leq -\frac{\lambda}{2}\left\|\nabla_{\boldsymbol{\theta}} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}^{[k]}\right)\right\|^2 + \frac{\gamma\lambda^2\sigma^2}{2} \\
&\leq -\frac{\lambda}{4}\left\|\nabla_{\boldsymbol{\theta}} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}^{[0]}\right)\right\|^2 + \frac{\gamma\lambda^2\sigma^2}{2} \\
&\quad + \frac{\lambda}{2}\left\|\nabla_{\boldsymbol{\theta}} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}^{[k]}\right) - \nabla_{\boldsymbol{\theta}} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}^{[0]}\right)\right\|^2 && \text{(using } \|\boldsymbol{a} + \boldsymbol{b}\|^2 \leq 2\|\boldsymbol{a}\|^2 + 2\|\boldsymbol{b}\|^2) \\
&\leq -\frac{\lambda}{4}\left\|\nabla_{\boldsymbol{\theta}} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i\right)\right\|^2 + \frac{\gamma\lambda^2\sigma^2}{2} + \frac{\lambda}{2}\left(\gamma\lambda G K\right)^2. \\
& && (\|\boldsymbol{\theta}^{[k]} - \boldsymbol{\theta}^{[0]}\| \leq \lambda K\|\sup\nabla_{\boldsymbol{\theta}} L_i(\boldsymbol{w}, \boldsymbol{\theta})\| \leq \lambda G K)
\end{aligned}$$

Summing the above over $k = 0, \ldots, K-1$, and taking expectation, we get

$$\begin{aligned}
&\mathbb{E}\left[L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i'\right)\right] - L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i\right) \\
&\quad \leq -\frac{\lambda K}{4}\left\|\nabla_{\boldsymbol{\theta}} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i\right)\right\|^2 + \frac{\gamma \lambda^2 \sigma^2 K}{2} + \frac{\gamma^2 \lambda^3 G^2 K^3}{2}.
\end{aligned}$$

Taking average over i finishes the proof. □

Lemma 6. *Consider a specific communication round r in Algorithm 3. Let the global parameter be $\boldsymbol{w}$, and the local parameters be $\{\boldsymbol{\theta}_i'\}_{i=1}^N$ at the beginning of Step 2. Also, let the global parameters be $\boldsymbol{w}'$ at the end of Step 2. If $\eta \leq \frac{1}{\gamma K}$, then conditioned on all history before Step 2, we have*

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N L_i(\boldsymbol{w}', \boldsymbol{\theta}_i')\right] - \frac{1}{N}\sum_{i=1}^N L_i(\boldsymbol{w}, \boldsymbol{\theta}_i') \\
&\leq -\frac{\eta K}{4}\left\|\frac{1}{N}\sum_{i=1}^N \nabla_{\boldsymbol{w}} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i'\right)\right\|^2 + \frac{\gamma \eta^2 \sigma^2 K}{2N} + \frac{\gamma^2 \eta \lambda^2 G^2 K^3}{2}.
\end{aligned}$$

Proof. For every i, k, we have

$$\begin{aligned}
&L_i(\boldsymbol{w}', \boldsymbol{\theta}_i') - L_i(\boldsymbol{w}, \boldsymbol{\theta}_i') \\
&\leq \langle \boldsymbol{w}' - \boldsymbol{w}, \nabla L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i'\right)\rangle + \frac{\gamma}{2}\left\|\boldsymbol{w}' - \boldsymbol{w}\right\|^2 \qquad \text{(Lemma 1)}
\end{aligned}$$

Averaging the above over $i = 1, \ldots, N$:

$$\begin{aligned}
&\frac{1}{N}\sum_{i=1}^N L_i(\boldsymbol{w}', \boldsymbol{\theta}_i') - \frac{1}{N}\sum_{i=1}^N L_i(\boldsymbol{w}, \boldsymbol{\theta}_i') \\
&\leq \left\langle \boldsymbol{w}' - \boldsymbol{w}, \frac{1}{N}\sum_{i=1}^N \nabla L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i'\right)\right\rangle + \frac{\gamma}{2}\left\|\boldsymbol{w}' - \boldsymbol{w}\right\|^2.
\end{aligned}$$

Now notice that $\boldsymbol{w}' - \boldsymbol{w} = -\frac{\eta K}{N}\sum_{i=1}^N \nabla_{\boldsymbol{w}} \ell_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i'\right)$, where ℓ_i is the average loss of of the large batch specified in Line 3 of Algorithm 3 that corresponds to client i. Conditioned on history before Step 2, we have $\mathbb{E}[\boldsymbol{w}' - \boldsymbol{w}] = -\frac{\eta K}{N}\sum_{i=1}^N \nabla_{\boldsymbol{w}} L_i(\boldsymbol{w}, \boldsymbol{\theta}_i')$. Taking expectation (conditioned on history before Step 2) on the last expression, we get

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N L_i(\boldsymbol{w}', \boldsymbol{\theta}_i')\right] - \frac{1}{N}\sum_{i=1}^N L_i(\boldsymbol{w}, \boldsymbol{\theta}_i') \\
&\leq -\eta K\left\|\frac{1}{N}\sum_{i=1}^N \nabla_{\boldsymbol{w}} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i'\right)\right\|^2 \\
&\qquad + \frac{\gamma \eta^2 K^2}{2}\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^N \nabla_{\boldsymbol{w}} \ell_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i'\right)\right\|^2\right] \\
&\leq \left(-\eta K + \frac{\gamma \eta^2 K^2}{2}\right)\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^N \nabla_{\boldsymbol{w}} L_i\left(\boldsymbol{w}_i, \boldsymbol{\theta}_i'\right)\right\|^2\right] \\
&\qquad + \frac{\gamma \eta^2 K^2}{2}\mathbb{V}\left[\frac{1}{N}\sum_{i=1}^N \nabla_{\boldsymbol{w}} \ell_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i'\right)\right] \\
&\leq \frac{-\eta K}{2}\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^N \nabla_{\boldsymbol{w}} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i'\right)\right\|^2\right]
\end{aligned}$$

$$+\frac{\gamma\eta^2K^2}{2}\times\frac{\sigma^2}{NK} \tag{6}$$

where the last inequality is because the variance of each $\nabla\ell_i$ is upper bounded by $\frac{\sigma^2}{K}$ for that ℓ_i is an average over K mini-batches; also, ℓ_i are independently sampled for different i.

We continue to bound (6). It can be upper bounded by

$$\begin{aligned}
&\frac{-\eta K}{4}\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w},\boldsymbol{\theta}_i\right)\right\|^2\right]+\frac{\gamma\eta^2\sigma^2K}{2N}\\
&\quad+\frac{\eta K}{2}\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^{N}\left(\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w},\boldsymbol{\theta}_i\right)-\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w},\boldsymbol{\theta}_i'\right)\right)\right\|^2\right]\\
&\leq\frac{-\eta K}{4}\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w},\boldsymbol{\theta}_i\right)\right\|^2\right]+\frac{\gamma\eta^2\sigma^2K}{2N}\\
&\quad+\frac{\eta K}{2}\times(\gamma\lambda GK)^2.
\end{aligned}$$

This finishes the proof. □

Proof of Theorem 1. By Lemma 5 and 6, with the choice of $\lambda=\frac{\eta}{N}$, we have

$$\begin{aligned}
&\Delta^{(r+1)}-\Delta^{(r)} &(7)\\
&=\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}L_i(\boldsymbol{w}^{r+1},\boldsymbol{\theta}_i^{r+1})-\frac{1}{N}\sum_{i=1}^{N}L_i(\boldsymbol{w}^r,\boldsymbol{\theta}_i^r)\right]\\
&\leq-\frac{\eta K}{4}\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla_{\boldsymbol{w}}L_i(\boldsymbol{w}^r,\boldsymbol{\theta}_i^r)\right\|^2\right]\\
&\quad-\frac{\eta K}{4}\times\frac{1}{N^2}\sum_{i=1}^{N}\mathbb{E}\left[\left\|\nabla_{\boldsymbol{\theta}}L_i(\boldsymbol{w}^r,\boldsymbol{\theta}_i^r)\right\|^2\right]\\
&\quad+\frac{\gamma\eta^2\sigma^2K}{N}+\frac{\gamma^2\eta^3G^2K^3}{N^2}. &(8)
\end{aligned}$$

Notice that

$$\begin{aligned}
&\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla L_i(\boldsymbol{w},\boldsymbol{\theta}_i)\right\|^2\\
&=\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla_{\boldsymbol{w}}L_i(\boldsymbol{w},\boldsymbol{\theta}_i)\right\|^2+\frac{1}{N^2}\sum_{i=1}^{N}\left\|\nabla_{\boldsymbol{\theta}}L_i(\boldsymbol{w},\boldsymbol{\theta}_i)\right\|^2.
\end{aligned}$$

Combining this with (8), rearranging, and telescoping on $\Delta^{(r+1)}-\Delta^{(r)}$, we get

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{R}\sum_{r=1}^{R}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla L_i(\boldsymbol{w}^r,\boldsymbol{\theta}_i^r)\right\|^2\right]\\
&\leq\frac{4\Delta^{(1)}}{\eta KR}+\frac{4\gamma\eta\sigma^2}{N}+\frac{4\gamma^2\eta^2G^2K^2}{N^2}
\end{aligned}$$

Optimize η with the constraint $\eta\leq\frac{1}{\gamma K}$, we get the first bound.

For general convex losses, by convexity, we have

$$\Delta^{(r)}\leq\left(\frac{1}{N}\sum_{i=1}^{N}\nabla L_i(\boldsymbol{w}^r,\boldsymbol{\theta}_i^r)\right)^\top\begin{bmatrix}\boldsymbol{w}^r-\boldsymbol{w}^*\\\boldsymbol{\theta}_1^r-\boldsymbol{\theta}_1^*\\\cdots\\\boldsymbol{\theta}_N^r-\boldsymbol{\theta}_N^*\end{bmatrix}$$

$$\leq \left\| \frac{1}{N} \sum_{i=1}^{N} \nabla L_i(\boldsymbol{w}^r, \boldsymbol{\theta}_i^r) \right\| D,$$

which, when combined with (8), gives the recursion formula:

$$\Delta^{(r+1)} \leq \Delta^{(r)} - \frac{\eta K}{4D^2} \left(\Delta^{(r)}\right)^2 + \frac{\gamma \eta^2 \sigma^2 K}{N} + \frac{\gamma^2 \eta^3 G^2 K^3}{N^2}.$$

Apply Lemma 3, we get

$$\Delta^{(R+1)} = \mathcal{O}\left(\frac{D^2}{\eta R K} + \sqrt{\frac{D^2 \gamma \eta \sigma^2}{N}} + \frac{D \gamma \eta G K}{N} \right)$$

Choosing the best η we get the second bound in the theorem.

For β-strongly convex losses, by Lemma 2, we have

$$\Delta^{(r)} \leq \frac{1}{2\beta} \left\| \frac{1}{N} \sum_{i=1}^{N} \nabla L_i(\boldsymbol{w}^r, \boldsymbol{\theta}_i^r) \right\|^2.$$

Again, using this in (8), we get

$$\Delta^{(r+1)} \leq \Delta^{(r)} - \frac{\eta \beta K}{2} \Delta^{(r)} + \frac{\gamma \eta^2 \sigma^2 K}{N} + \frac{\gamma^2 \eta^3 G^2 K^3}{N^2}$$

Using Lemma 4 and choosing the best η, we get the third bound in the theorem. □

D Analysis for FEDRESAVG

Lemma 7. *Consider a specific communication round r in Algorithm 4. Let the global parameter be $\boldsymbol{w}$, and the local parameters be $\{\boldsymbol{\theta}_i\}_{i=1}^N$ at the beginning of communication round r. Also, let the local parameters be $\{\boldsymbol{\theta}_i'\}_{i=1}^N$ at the end of Step 1. If $\lambda \leq \frac{1}{\gamma}$, then conditioned on all history before round r, we have*

$$\mathbb{E}\left[\frac{1}{N} \sum_{i=1}^{N} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i'\right) \right] - \frac{1}{N} \sum_{i=1}^{N} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i\right) \leq -\frac{\lambda K}{4} \times \frac{1}{N} \sum_{i=1}^{N} \left\| \nabla_{\boldsymbol{\theta}} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i\right) \right\|^2 + \frac{\gamma \lambda^2 \sigma^2 K + \gamma^2 \lambda^3 G^2 K^2}{2}.$$

Proof. The proof is same as that of Lemma 5, which we present in Section C. □

Lemma 8. *Consider a specific communication round r in Algorithm 4. Let the global parameter be $\boldsymbol{w}$, and the local parameters be $\{\boldsymbol{\theta}_i\}_{i=1}^N$ at the beginning of Step 1. Also, let the global parameters be $\boldsymbol{w}'$, and the local parameters be $\{\boldsymbol{\theta}_i'\}_{i=1}^N$ at the end of Step 2. If $\eta \leq \frac{1}{96\gamma\alpha K}$, then conditioned on all history before Step 2, we have*

$$\mathbb{E}\left[\frac{1}{N} \sum_i L_i(\boldsymbol{w}', \boldsymbol{\theta}_i') - \frac{1}{N} \sum_i L_i(\boldsymbol{w}, \boldsymbol{\theta}_i') \right]$$
$$\leq -\frac{\alpha \eta K}{4} \mathbb{E}\left[\left\| \frac{1}{N} \sum_{i=1}^{N} \nabla_{\boldsymbol{w}} L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i\right) \right\|^2 \right] + \frac{\gamma \alpha^2 \eta^2 \sigma^2 K}{N} + \gamma^2 \alpha \eta^3 G^2 K^3 + \frac{\gamma^2 \eta \lambda^2 G^2 K^3}{N} + \frac{\gamma}{K} \mathbb{E}\left[\frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{K} \left\| \boldsymbol{w}_i^{[k]} - \boldsymbol{w} \right\|^2 \right]$$

where $\boldsymbol{w}_i^{[k]}$ denotes the $\boldsymbol{w}_i$ at the beginning of the k-th for-loop in Step 2.

Proof. Let $\boldsymbol{w}_i^{[1]} = \boldsymbol{w}$ and $\boldsymbol{w}_i^{[K+1]}$ be the $\boldsymbol{w}_i$ maintained by client i at the end of Step 2, and $\boldsymbol{w}_i^{[k]}$ be the $\boldsymbol{w}_i$ at the beginning of the k-th for-loop in Step 2. Then for every i, k, we have

$$\mathbb{E}\left[L_i(\boldsymbol{w}', \boldsymbol{\theta}_i') - L_i(\boldsymbol{w}, \boldsymbol{\theta}_i') \right]$$

$$\leq \mathbb{E}\left[\left\langle \boldsymbol{w}' - \boldsymbol{w}, \nabla L_i\left(\boldsymbol{w}_i^{[k]}, \boldsymbol{\theta}_i'\right)\right\rangle + \frac{\gamma}{2}\left\|\boldsymbol{w}' - \boldsymbol{w}_i^{[k]}\right\|^2\right] \qquad \text{(Lemma 1)}$$
$$\leq \mathbb{E}\left[\left\langle \boldsymbol{w}' - \boldsymbol{w}, \nabla L_i\left(\boldsymbol{w}_i^{[k]}, \boldsymbol{\theta}_i'\right)\right\rangle + \gamma\left\|\boldsymbol{w}' - \boldsymbol{w}\right\|^2 + \gamma\left\|\boldsymbol{w} - \boldsymbol{w}_i^{[k]}\right\|^2\right]$$

Averaging the above over $k = 1, \ldots, K$, we get

$$\mathbb{E}\left[L_i(\boldsymbol{w}', \boldsymbol{\theta}_i') - L_i(\boldsymbol{w}, \boldsymbol{\theta}_i')\right] \leq \frac{1}{K}\mathbb{E}\left[\left\langle \boldsymbol{w}' - \boldsymbol{w}, \sum_{k=1}^{K}\nabla L_i\left(\boldsymbol{w}_i^{[k]}, \boldsymbol{\theta}_i'\right)\right\rangle\right] + \gamma\mathbb{E}\left[\left\|\boldsymbol{w}' - \boldsymbol{w}\right\|^2\right] + \frac{\gamma}{K}\mathbb{E}\left[\sum_{k=1}^{K}\left\|\boldsymbol{w}_i^{[k]} - \boldsymbol{w}\right\|^2\right]$$

Further averaging over $i = 1, \ldots, N$:

$$\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}L_i(\boldsymbol{w}', \boldsymbol{\theta}_i') - \frac{1}{N}\sum_{i=1}^{N}L_i(\boldsymbol{w}, \boldsymbol{\theta}_i')\right]$$
$$\leq \frac{1}{KN}\mathbb{E}\left[\left\langle \boldsymbol{w}' - \boldsymbol{w}, \sum_{k=1}^{K}\sum_{i=1}^{N}\nabla L_i\left(\boldsymbol{w}_i^{[k]}, \boldsymbol{\theta}_i'\right)\right\rangle\right] + \gamma\mathbb{E}\left[\left\|\boldsymbol{w}' - \boldsymbol{w}\right\|^2\right] + \frac{\gamma}{KN}\mathbb{E}\left[\sum_{k=1}^{K}\sum_{i=1}^{N}\left\|\boldsymbol{w}_i^{[k]} - \boldsymbol{w}\right\|^2\right].$$

Let $\ell_i^{[k]}$ be the loss of the k-th mini-batch of client i (i.e., the loss received in the k-th for-loop in Step 2). Then $\boldsymbol{w}' - \boldsymbol{w} = -\frac{\alpha\eta}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}\left(\nabla_{\boldsymbol{w}}\ell_i^{[k]}\left(\boldsymbol{w}_i^{[k]}, \boldsymbol{\theta}_i'\right) - \boldsymbol{c}_i + \boldsymbol{c}\right) = -\frac{\alpha\eta}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}\nabla_{\boldsymbol{w}}\ell_i^{[k]}\left(\boldsymbol{w}_i^{[k]}, \boldsymbol{\theta}_i'\right)$. Thus the last expression can be further upper bounded by

$$-\frac{\alpha\eta}{KN^2}\mathbb{E}\left[\left\|\sum_{i=1}^{N}\sum_{k=1}^{K}\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w}_i^{[k]}, \boldsymbol{\theta}_i'\right)\right\|^2\right]$$
$$+\frac{\gamma\alpha^2\eta^2}{N^2}\mathbb{E}\left[\left\|\sum_{i=1}^{N}\sum_{k=1}^{K}\nabla_{\boldsymbol{w}}\ell_i^{[k]}\left(\boldsymbol{w}_i^{[k]}, \boldsymbol{\theta}_i'\right)\right\|^2\right] + \frac{\gamma}{KN}\mathbb{E}\left[\sum_{i=1}^{N}\sum_{k=1}^{K}\left\|\boldsymbol{w}_i^{[k]} - \boldsymbol{w}\right\|^2\right]$$
$$\leq \left(-\frac{\alpha\eta}{KN^2} + \frac{\gamma\alpha^2\eta^2}{N^2}\right)\mathbb{E}\left[\left\|\sum_{i=1}^{N}\sum_{k=1}^{K}\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w}_i^{[k]}, \boldsymbol{\theta}_i'\right)\right\|^2\right]$$
$$+\frac{\gamma\alpha^2\eta^2}{N^2}\mathbb{V}\left[\sum_{i=1}^{N}\sum_{k=1}^{K}\nabla_{\boldsymbol{w}}\ell_i^{[k]}\left(\boldsymbol{w}_i^{[k]}, \boldsymbol{\theta}_i'\right)\right] + \frac{\gamma}{KN}\mathbb{E}\left[\sum_{i=1}^{N}\sum_{k=1}^{K}\left\|\boldsymbol{w}_i^{[k]} - \boldsymbol{w}\right\|^2\right].$$
$$\leq -\frac{\alpha\eta}{2KN^2}\mathbb{E}\left[\left\|\sum_{i=1}^{N}\sum_{k=1}^{K}\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w}_i^{[k]}, \boldsymbol{\theta}_i'\right)\right\|^2\right] + \frac{\gamma\alpha^2\eta^2}{N^2} \times KN\sigma^2 + \frac{\gamma}{KN}\mathbb{E}\left[\sum_{i=1}^{N}\sum_{k=1}^{K}\left\|\boldsymbol{w}_i^{[k]} - \boldsymbol{w}\right\|^2\right]$$
$$\leq -\frac{\alpha\eta}{4KN^2}\mathbb{E}\left[\left\|\sum_{i=1}^{N}\sum_{k=1}^{K}\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i\right)\right\|^2\right] + \frac{\gamma\alpha^2\eta^2\sigma^2K}{N} + \frac{\gamma}{KN}\mathbb{E}\left[\sum_{i=1}^{N}\sum_{k=1}^{K}\left\|\boldsymbol{w}_i^{[k]} - \boldsymbol{w}\right\|^2\right]$$
$$+\frac{\alpha\eta}{2KN^2}\mathbb{E}\left[\left\|\sum_{i=1}^{N}\sum_{k=1}^{K}\left(\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i\right) - \nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w}_i^{[k]}, \boldsymbol{\theta}_i'\right)\right)\right\|^2\right]$$
$$\leq -\frac{\alpha\eta K}{4}\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i\right)\right\|^2\right] + \frac{\gamma\alpha^2\eta^2\sigma^2K}{N} + \frac{\alpha\eta}{2KN^2} \times (KN\gamma\eta GK)^2$$
$$+\frac{\alpha\eta}{2KN^2} \times N \times (K\gamma\lambda GK)^2 + \frac{\gamma}{KN}\mathbb{E}\left[\sum_{i=1}^{N}\sum_{k=1}^{K}\left\|\boldsymbol{w}_i^{[k]} - \boldsymbol{w}\right\|^2\right]$$
$$\leq -\frac{\alpha\eta K}{4}\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w}, \boldsymbol{\theta}_i\right)\right\|^2\right] + \frac{\gamma\alpha^2\eta^2\sigma^2K}{N} + \gamma^2\alpha\eta^3G^2K^3 + \frac{\gamma^2\eta\lambda^2G^2K^3}{N} + \frac{\gamma}{K}\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}\left\|\boldsymbol{w}_i^{[k]} - \boldsymbol{w}\right\|^2\right]$$

□

In the next lemma, we use similar notations as in the proof of Lemma 8, but with an additional superscript to indicate the communication round: $\boldsymbol{w}^r$ is the $\boldsymbol{w}$ in the beginning of round r; $\boldsymbol{w}_i^{r,[k]}$ is the $\boldsymbol{w}_i$ of client i in the beginning of the k-th for-loop. Also, we use $\boldsymbol{c}_i^r$ and $\boldsymbol{c}^r$ to denote the $\boldsymbol{c}_i$ and $\boldsymbol{c}$ used in round r. $\boldsymbol{\theta}_i'^r$ denotes the $\boldsymbol{\theta}_i$ at the beginning of Step 2 in round r. $\ell_i^{r,[k]}$ denotes the loss received by client i in round r in the k-th for-loop (Line 4-4 of Algorithm 4).

Lemma 9. *For any client i and $k \in \{1, \ldots, K\}$,*

$$\mathbb{E}\left[\left\|\boldsymbol{w}_i^{r,[k]} - \boldsymbol{w}^r\right\|^2\right] \leq \mathcal{O}\left(\eta^2\sigma^2 K + \gamma^2(\eta+\lambda)^4 N^2 G^2 K^4\right) + 6\eta^2 K^2 \mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^N \nabla_{\boldsymbol{w}} L_i\left(\boldsymbol{w}^{r-1}, \boldsymbol{\theta}_i'^{r-1}\right)\right\|^2\right]$$

Proof. By the update rule of Algorithm 4, we have for any i and k,

$$\begin{aligned}
&\mathbb{E}\left[\left\|\boldsymbol{w}_i^{r,[k+1]} - \boldsymbol{w}^r\right\|^2\right] \\
&= \eta^2 k^2 \mathbb{E}\left[\left\|\frac{1}{k}\sum_{h=1}^k \left(\nabla_{\boldsymbol{w}} \ell_i^{r,[h]}\left(\boldsymbol{w}_i^{r,[h]}, \boldsymbol{\theta}_i'^r\right) - \boldsymbol{c}_i^r + \boldsymbol{c}^r\right)\right\|^2\right] \\
&\leq 6\eta^2 k^2 \mathbb{E}\left[\left\|\frac{1}{k}\sum_{h=1}^k \nabla_{\boldsymbol{w}} \ell_i^{r,[h]}\left(\boldsymbol{w}_i^{r,[h]}, \boldsymbol{\theta}_i'^r\right) - \frac{1}{k}\sum_{h=1}^k \nabla_{\boldsymbol{w}} L_i\left(\boldsymbol{w}_i^{r,[h]}, \boldsymbol{\theta}_i'^r\right)\right\|^2\right] && (\textbf{term}_1) \\
&\quad + 6\eta^2 k^2 \mathbb{E}\left[\left\|\frac{1}{k}\sum_{h=1}^k \nabla_{\boldsymbol{w}} L_i\left(\boldsymbol{w}_i^{r,[h]}, \boldsymbol{\theta}_i'^r\right) - \frac{1}{K}\sum_{h=1}^K \nabla_{\boldsymbol{w}} L_i\left(\boldsymbol{w}_i^{r-1,[h]}, \boldsymbol{\theta}_i'^{r-1}\right)\right\|^2\right] && (\textbf{term}_2) \\
&\quad + 6\eta^2 k^2 \mathbb{E}\left[\left\|\frac{1}{K}\sum_{h=1}^K \nabla_{\boldsymbol{w}} L_i\left(\boldsymbol{w}_i^{r-1,[h]}, \boldsymbol{\theta}_i'^{r-1}\right) - \boldsymbol{c}_i^r\right\|^2\right] && (\textbf{term}_3) \\
&\quad + 6\eta^2 k^2 \mathbb{E}\left[\left\|\boldsymbol{c}^r - \frac{1}{KN}\sum_{h=1}^K\sum_{i=1}^N \nabla_{\boldsymbol{w}} L_i\left(\boldsymbol{w}_i^{r-1,[h]}, \boldsymbol{\theta}_i'^{r-1}\right)\right\|^2\right] && (\textbf{term}_4) \\
&\quad + 6\eta^2 k^2 \mathbb{E}\left[\left\|\frac{1}{KN}\sum_{h=1}^K\sum_{i=1}^N \nabla_{\boldsymbol{w}} L_i\left(\boldsymbol{w}_i^{r-1,[h]}, \boldsymbol{\theta}_i'^{r-1}\right) - \frac{1}{N}\sum_{i=1}^N \nabla_{\boldsymbol{w}} L_i\left(\boldsymbol{w}^{r-1}, \boldsymbol{\theta}_i'^{r-1}\right)\right\|^2\right] && (\textbf{term}_5) \\
&\quad + 6\eta^2 k^2 \mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^N \nabla_{\boldsymbol{w}} L_i\left(\boldsymbol{w}^{r-1}, \boldsymbol{\theta}_i'^{r-1}\right)\right\|^2\right], && (\textbf{term}_6)
\end{aligned}$$

where we use $\|\boldsymbol{a}_1 + \boldsymbol{a}_2 + \cdots + \boldsymbol{a}_6\|^2 \leq 6\left(\|\boldsymbol{a}_1\|^2 + \cdots + \|\boldsymbol{a}_6\|^2\right)$.

For the $\textbf{term}_3$ above, note that

$$\boldsymbol{c}_i^r = \sum_{h=1}^K \nabla_{\boldsymbol{w}} \ell_i^{r-1,[h]}\left(\boldsymbol{w}_i^{r-1,[h]}, \boldsymbol{\theta}_i'^{r-1}\right),$$

so $\textbf{term}_3$ is equal to $6\eta^2k^2$ times the variance of $\boldsymbol{c}_i^r$, which is upper bounded by $6\eta^2k^2 \times \frac{\sigma^2}{K} \leq 6\eta^2\sigma^2 K$. Similarly, $\textbf{term}_4 \leq 6\eta^2k^2 \times \frac{\sigma^2}{KN} \leq \frac{6\eta^2\sigma^2 K}{N}$. Similarly, $\textbf{term}_1$ is $6\eta^2k^2$ times the variance of $\frac{1}{k}\sum_{h=1}^k \nabla_{\boldsymbol{w}}\left(\boldsymbol{w}_i^{[h]}, \boldsymbol{\theta}_i'^r\right)$, which is upper bounded by $6\eta^2k^2 \times \frac{\sigma^2}{k} \leq 6\eta^2\sigma^2 K$.

To bound $\mathbf{term}_2$, notice that

$$\begin{aligned}
&\left\|\frac{1}{k}\sum_{h=1}^{k}\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w}_i^{r,[h]},\boldsymbol{\theta}_i'^{r}\right)-\frac{1}{K}\sum_{h=1}^{K}\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w}_i^{r-1,[h]},\boldsymbol{\theta}_i'^{r-1}\right)\right\| \\
&\leq \max_{h,h'}\left\|L_i\left(\boldsymbol{w}_i^{r,[h]},\boldsymbol{\theta}_i'^{r}\right)-\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w}_i^{r-1,[h']},\boldsymbol{\theta}_i'^{r-1}\right)\right\| \\
&\leq \gamma\max_{h,h'}\left(\left\|\boldsymbol{w}_i^{r,[h]}-\boldsymbol{w}^{r}\right\|+\left\|\boldsymbol{w}^{r}-\boldsymbol{w}^{r-1}\right\|+\left\|\boldsymbol{w}^{r-1}-\boldsymbol{w}_i^{r-1,[h']}\right\|+\left\|\boldsymbol{\theta}_i'^{r}-\boldsymbol{\theta}_i'^{r-1}\right\|\right) \\
&\leq \gamma\left(\eta GK+\eta NGK+\eta GK+\lambda GK\right) \\
&\leq 4\gamma(\eta+\lambda)NGK.
\end{aligned}$$

Therefore, $\mathbf{term}_2 \leq 6\eta^2K^2\times 16\gamma^2(\eta+\lambda)^2N^2G^2K^2 = \mathcal{O}\left(\gamma^2(\eta+\lambda)^4N^2G^2K^4\right)$. Similarly, $\mathbf{term}_5 \leq 6\eta^2K^2\times\left(\gamma\max_{h,h'}\left\|\boldsymbol{w}_i^{r-1,[h]}-\boldsymbol{w}_i^{r,[h']}\right\|\right)^2 = \mathcal{O}\left(\gamma^2(\eta+\lambda)^4N^2G^2K^4\right)$.

Combining all the above, we get

$$\left\|\boldsymbol{w}_i^{r,[k]}-\boldsymbol{w}^{r}\right\|^2 \leq \mathcal{O}\left(\eta^2\sigma^2K+\gamma^2(\eta+\lambda)^4N^2G^2K^4\right)+6\eta^2K^2\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w}^{r-1},\boldsymbol{\theta}_i'^{r-1}\right)\right\|^2\right]$$

for all k, finishing the proof.

□

Theorem 3. ***(Restatement of Theorem 2)*** *Let $\boldsymbol{w}^r,\boldsymbol{\theta}_i^r$ be the values of the parameters at the beginning of round r, and let*

$$\Delta^{(r)} = \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}L_i\left(\boldsymbol{w}^r,\boldsymbol{\theta}_i^r\right)\right]-\frac{1}{N}\sum_{i=1}^{N}L_i\left(\boldsymbol{w}^*,\boldsymbol{\theta}_i^*\right).$$

Then for γ-smooth convex losses, Algorithm 4 guarantees

$$\mathbb{E}\left[\frac{1}{R}\sum_{r=1}^{R}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla L_i\left(\boldsymbol{w}^r,\boldsymbol{\theta}_i^r\right)\right\|^2\right] = \mathcal{O}\left(\sqrt{\frac{\gamma\sigma^2\Delta^{(1)}}{NKR}}\right)$$

and

$$\Delta^{(R+1)} = \mathcal{O}\left(\left(\frac{D^4\gamma\sigma^2}{NKR}\right)^{\frac{1}{3}}\right).$$

with proper choices of η. For β-strongly convex losses, Algorithm 4 guarantees

$$\Delta^{(R+1)} = \mathcal{O}\left(\frac{\gamma}{\beta}\frac{\sigma^2}{\beta NKR}\right).$$

Proof. Combining Lemma 7, 8, and 9, and picking $\alpha=\sqrt{N}$, we get

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}L_i\left(\boldsymbol{w}^{r+1},\boldsymbol{\theta}_i^{r+1}\right)-\frac{1}{N}\sum_{i=1}^{N}L_i\left(\boldsymbol{w}^r,\boldsymbol{\theta}_i^r\right)\right] \\
&\leq -\frac{\lambda NK}{4}\times\frac{1}{N^2}\sum_{i=1}^{N}\left\|\nabla_{\boldsymbol{\theta}}L_i\left(\boldsymbol{w},\boldsymbol{\theta}_i\right)\right\|^2+\frac{\gamma\lambda^2\sigma^2K+\gamma^2\lambda^3G^2K^2}{2} \\
&\quad-\frac{\eta\sqrt{N}K}{4}\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w},\boldsymbol{\theta}_i'\right)\right\|^2\right]+19\gamma\eta^2\sigma^2K+\sqrt{N}\gamma^2\eta^3G^2K^3+192\gamma^3\eta^4N^2G^2K^4 \\
&\quad+6\gamma\eta^2K^2\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla_{\boldsymbol{w}}L_i\left(\boldsymbol{w}^{r-1},\boldsymbol{\theta}_i^{r-1}\right)\right\|^2\right]. \qquad (9)
\end{aligned}$$

Below we pick $\lambda = \frac{1}{\sqrt{N}}\eta$, and notice that

$$\left\|\nabla\left(\frac{1}{N}\sum_{i=1}^{N} L_i(\boldsymbol{w}^r, \boldsymbol{\theta}_i^r)\right)\right\|^2 = \mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla_{\boldsymbol{w}} L_i\left(\boldsymbol{w}^{r-1}, \boldsymbol{\theta}_i^{r-1}\right)\right\|^2\right] + \frac{1}{N^2}\sum_{i=1}^{N}\mathbb{E}\left[\left\|\nabla_{\boldsymbol{\theta}} L_i(\boldsymbol{w}^r, \boldsymbol{\theta}_i^r)\right\|^2\right]$$

Recall that

$$\Delta^{(r)} = \frac{1}{N}\sum_{i=1}^{N} L_i\left(\boldsymbol{w}^r, \boldsymbol{\theta}_i^r\right) - \frac{1}{N}\sum_{i=1}^{N} L_i\left(\boldsymbol{w}^*, \boldsymbol{\theta}_i^*\right),$$

and define

$$\delta^{(r)} = 6\gamma\eta^2 K^2 \mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla L_i\left(\boldsymbol{w}^{r-1}, \boldsymbol{\theta}_i^{r-1}\right)\right\|^2\right].$$

Then by (9), we have

$$\Delta^{(r+1)} - \left(\Delta^{(r)} + \delta^{(r)}\right) \leq -\frac{\eta\sqrt{N}K}{4}\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i}\nabla L_i\left(\boldsymbol{w}^r, \boldsymbol{\theta}_i^r\right)\right\|^2\right] + \mathcal{O}\left(\gamma\eta^2\sigma^2 K + \sqrt{N}\gamma^2\eta^3 G^2 K^3 + \gamma^3\eta^4 N^2 G^2 K^4\right),$$

implying that

$$\begin{aligned}
&\left(\Delta^{(r+1)} + 2\delta^{(r+1)}\right) - \left(\Delta^{(r)} + \delta^{(r)}\right) \\
&\leq -\left(\frac{\eta\sqrt{N}K}{4} - 12\gamma\eta^2 K^2\right)\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i}\nabla L_i\left(\boldsymbol{w}^r, \boldsymbol{\theta}_i^r\right)\right\|^2\right] + \mathcal{O}\left(\gamma\eta^2\sigma^2 K + \sqrt{N}\gamma^2\eta^3 G^2 K^3 + \gamma^3\eta^4 N^2 G^2 K^4\right) \\
&\leq -\frac{\eta\sqrt{N}K}{8}\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i}\nabla L_i\left(\boldsymbol{w}^r, \boldsymbol{\theta}_i^r\right)\right\|^2\right] + \mathcal{O}\left(\gamma\eta^2\sigma^2 K + \sqrt{N}\gamma^2\eta^3 G^2 K^3 + \gamma^3\eta^4 N^2 G^2 K^4\right).
\end{aligned} \tag{10}$$

Therefore,

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{R}\sum_{r=1}^{R}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla L_i\left(\boldsymbol{w}^r, \boldsymbol{\theta}_i^r\right)\right\|^2\right] \\
&\leq \frac{8}{\eta\sqrt{N}KR}\sum_{r=1}^{R}\left(\left(\Delta^{(r)} + \delta^{(r)}\right) - \left(\Delta^{(r+1)} + 2\delta^{(r+1)}\right)\right) + \frac{8}{\eta\sqrt{N}K} \times \mathcal{O}\left(\gamma\eta^2\sigma^2 K + \sqrt{N}\gamma^2\eta^3 G^2 K^3 + \gamma^3\eta^4 N^2 G^2 K^4\right) \\
&\leq \frac{8\Delta^{(1)}}{\eta\sqrt{N}KR} + \frac{48\gamma\eta^2 K^2}{\eta\sqrt{N}KR}\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla L_i\left(\boldsymbol{w}^1, \boldsymbol{\theta}_i^1\right)\right\|^2\right] + \mathcal{O}\left(\frac{\gamma\eta\sigma^2}{\sqrt{N}} + \gamma^2\eta^2 G^2 K^2 + \gamma^3\eta^3 N^{\frac{3}{2}} G^2 K^3\right) \\
&\leq \frac{8\Delta^{(1)}}{\eta\sqrt{N}KR} + \frac{1}{2R}\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla L_i\left(\boldsymbol{w}^1, \boldsymbol{\theta}_i^1\right)\right\|^2\right] + \mathcal{O}\left(\frac{\gamma\eta\sigma^2}{\sqrt{N}} + \gamma^2\eta^2 G^2 K^2 + \gamma^3\eta^3 N^{\frac{3}{2}} G^2 K^3\right).
\end{aligned}$$

Moving the second term above to the left-hand side, and picking the best η (under the constraint $\eta \leq \frac{1}{96\gamma\alpha K} = \frac{1}{96\gamma\sqrt{N}K}$), we get

$$\begin{aligned}
\mathbb{E}\left[\frac{1}{R}\sum_{r=1}^{R}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla L_i\left(\boldsymbol{w}^r, \boldsymbol{\theta}_i^r\right)\right\|^2\right] &= \mathcal{O}\left(\frac{\Delta^{(1)}}{\eta\sqrt{N}KR} + \frac{\gamma\eta\sigma^2}{\sqrt{N}} + \gamma^2\eta^2 G^2 K^2 + \gamma^3\eta^3 N^{\frac{3}{2}} G K^3\right) \\
&= \mathcal{O}\left(\sqrt{\frac{\gamma\sigma^2\Delta^{(1)}}{NKR}} + \left(\frac{\Delta^{(1)}\gamma G}{\sqrt{N}R}\right)^{\frac{2}{3}} + \left(\frac{\Delta^{(1)}\gamma G^{\frac{2}{3}}}{R}\right)^{\frac{3}{4}} + \frac{\gamma\Delta^{(1)}}{R}\right).
\end{aligned}$$

For general convex losses, by convexity, we have

$$\Delta^{(r)} \leq \left(\frac{1}{N}\sum_{i=1}^{N}\nabla L_i(\boldsymbol{w}^r, \boldsymbol{\theta}_i^r)\right)^\top \begin{bmatrix} \boldsymbol{w}^r - \boldsymbol{w}^* \\ \boldsymbol{\theta}_1^r - \boldsymbol{\theta}_1^* \\ \cdots \\ \boldsymbol{\theta}_N^r - \boldsymbol{\theta}_N^* \end{bmatrix}$$
$$\leq \left\|\frac{1}{N}\sum_{i=1}^{N}\nabla L_i(\boldsymbol{w}^r, \boldsymbol{\theta}_i^r)\right\| D.$$

Combining this with (10), we get

$$\left(\Delta^{(r+1)} + 2\delta^{(r+1)}\right) - \left(\Delta^{(r)} + \delta^{(r)}\right) \leq -\frac{\eta\sqrt{N}K}{8D^2}\left(\Delta^{(r)}\right)^2 + \mathcal{O}\left(\gamma\eta^2\sigma^2 K + \sqrt{N}\gamma^2\eta^3 G^2K^3 + \gamma^3\eta^4N^2G^2K^4\right),$$

which implies

$$\left(\Delta^{(r+1)} + 2\delta^{(r+1)}\right) - \left(\Delta^{(r)} + 2\delta^{(r)}\right) \leq -\frac{\eta\sqrt{N}K}{8D^2}\left(\Delta^{(r)}\right)^2 - \delta^{(r)} + \mathcal{O}\left(\gamma\eta^2\sigma^2 K + \sqrt{N}\gamma^2\eta^3 G^2K^3 + \gamma^3\eta^4N^2G^2K^4\right).$$

Notice that $\delta^{(r)} \geq \frac{\eta\sqrt{N}K}{2D^2}\left(\delta^{(r)}\right)^2$ since $\frac{\eta\sqrt{N}K}{2D^2}\delta^{(r)} \leq \frac{\eta\sqrt{N}K}{2D^2} \times 6\gamma\eta^2K^2G^2 \leq 1$ by our choice of η.
Thus,

$$\begin{aligned}
&\left(\Delta^{(r+1)} + 2\delta^{(r+1)}\right) - \left(\Delta^{(r)} + 2\delta^{(r)}\right) \\
&\leq -\frac{\eta\sqrt{N}K}{8D^2}\left(\left(\Delta^{(r)}\right)^2 + 4\left(\delta^{(r)}\right)^2\right) + \mathcal{O}\left(\gamma\eta^2\sigma^2 K + \sqrt{N}\gamma^2\eta^3 G^2K^3 + \gamma^3\eta^4N^2G^2K^4\right) \\
&\leq -\frac{\eta\sqrt{N}K}{16D^2}\left(\Delta^{(r)} + 2\delta^{(r)}\right)^2 + \mathcal{O}\left(\gamma\eta^2\sigma^2 K + \sqrt{N}\gamma^2\eta^3 G^2K^3 + \gamma^3\eta^4N^2G^2K^4\right).
\end{aligned}$$

Then using Lemma 3 we get

$$\Delta^{(R+1)} \leq \mathcal{O}\left(\frac{D^2}{\eta\sqrt{N}KR} + \frac{\Delta^{(1)} + \delta^{(1)}}{R} + \sqrt{\frac{D^2\gamma\eta\sigma^2}{\sqrt{N}}} + D\gamma\eta GK + DG\left(\gamma\eta\sqrt{N}K\right)^{\frac{3}{2}}\right).$$

Picking the best η, the right-hand side above is

$$\mathcal{O}\left(\left(\frac{D^4\gamma\sigma^2}{NKR}\right)^{\frac{1}{3}} + \sqrt{\frac{D^3\gamma G}{\sqrt{N}R}} + \left(\frac{D^8G^2\gamma^3}{R^3}\right)^{\frac{1}{5}} + \frac{\Delta^{(1)} + \delta^{(1)}}{R}\right).$$

For β-strongly convex losses, by Lemma 2, we have

$$\Delta^{(r)} \leq \frac{1}{2\beta}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla L_i(\boldsymbol{w}^r, \boldsymbol{\theta}_i^r)\right\|^2.$$

Again, using this in (10), we get

$$\left(\Delta^{(r+1)} + 2\delta^{(r+1)}\right) - \left(\Delta^{(r)} + \delta^{(r)}\right) \leq -\frac{\beta\eta\sqrt{N}K}{4}\Delta^{(r)} + \mathcal{O}\left(\gamma\eta^2\sigma^2 K + \sqrt{N}\gamma^2\eta^3 G^2K^3 + \gamma^3\eta^4N^2G^2K^4\right),$$

which implies

$$\left(\Delta^{(r+1)} + 2\delta^{(r+1)}\right) - \left(\Delta^{(r)} + 2\delta^{(r)}\right) \leq -\frac{\beta\eta\sqrt{N}K}{4}\left(\Delta^{(r)} + 2\delta^{(r)}\right) + \mathcal{O}\left(\gamma\eta^2\sigma^2 K + \sqrt{N}\gamma^2\eta^3 G^2K^3 + \gamma^3\eta^4N^2G^2K^4\right)$$

by picking $\frac{\beta\eta\sqrt{N}K}{2} \leq 1$. Using Lemma 4, we get

$$\Delta^{(R+1)} \leq \left(1 - \frac{\beta\eta\sqrt{N}K}{4}\right)^R\left(\Delta^{(1)} + 2\delta^{(1)}\right) + \mathcal{O}\left(\frac{\gamma\eta\sigma^2}{\beta\sqrt{N}} + \frac{(\gamma\eta GK)^2}{\beta} + \frac{\gamma^2\eta^3N^{\frac{3}{2}}G^2K^3}{\beta}\right)$$

To make the first term above vanish, we pick η such that $\frac{\beta\eta\sqrt{NK}}{4} = \frac{2\log R}{R}$. Then $\left(1 - \frac{\beta\eta\sqrt{NK}}{4}\right)^R = \left(1 - \frac{2\log R}{R}\right)^R \leq \frac{1}{R^2}$, and we get

$$\Delta^{(R+1)} = \widetilde{\mathcal{O}}\left(\frac{\Delta^{(1)} + \delta^{(1)}}{R^2} + \frac{\gamma}{\beta}\frac{\sigma^2}{\beta NKR} + \frac{\gamma^2 G^2}{\beta^3 NR^2} + \frac{\gamma^2 G^2}{\beta^4 R^3}\right).$$

□

E Experiments

We test our algorithms on two synthetic datasets. The evaluation on real data will be provided in future versions.

E.1 Synthetic Data (I)

In this subsection, we run experiments on synthetic loss functions, and compare the robustness of FEDRESNAIVE, FEDRESSGD, FEDRESAVG with Option I and II against client drift. The purpose is to verify that FEDRESSGD and FEDRESAVG (Option II), which we develop theorems for, are indeed robust to client drift, while the other two algorithms do not. The synthetic loss function we use is inspired by the Theorem II of [7]. Specially, we consider the case with number of clients $N = 2$, and with one-dimensional global/local parameters $w \in \mathbb{R}$ and $\theta \in \mathbb{R}$. The loss functions of the two clients are

$$L_1(w, \theta_1) = 0.1(w + \theta_1)^2 + 10w \tag{11}$$

$$L_2(w, \theta_2) = 0.1\theta_2^2 - 10w. \tag{12}$$

We assume that the loss functions are deterministic (i.e., every time client i samples a loss function, it will be exactly L_i). Clearly, the global optimal solution satisfies $w^* + \theta_1^* = 0$ and $\theta_2^* = 0$ and gives the optimal average loss of 0. We fix $\eta = \lambda = 0.01$, and set $\alpha = 1$ in FEDRESAVG. All entries of all parameters are initialized with $\mathcal{N}(0, 0.1)$.

The results are plotted in Figure 1.

Observations. From Figure 1, we see that FEDRESNAIVE and FEDRESAVG (Option I) suffer from the issue of client drift: under a fixed learning rate, when the number of local updates per communication (K) is large, their parameters fail to converge to the optimal ones. The problem exacerbates as K grows larger.

E.2 Synthetic Data (II)

In this subsection, we construct a synthetic dataset similar to that considered in [8]. It also resembles the Example 1 we give in Section 2. We consider a M-class classification problem with features $\boldsymbol{x}_g = \boldsymbol{x}_l = \boldsymbol{x} \in \mathbb{R}^d$ and label y that is generated by

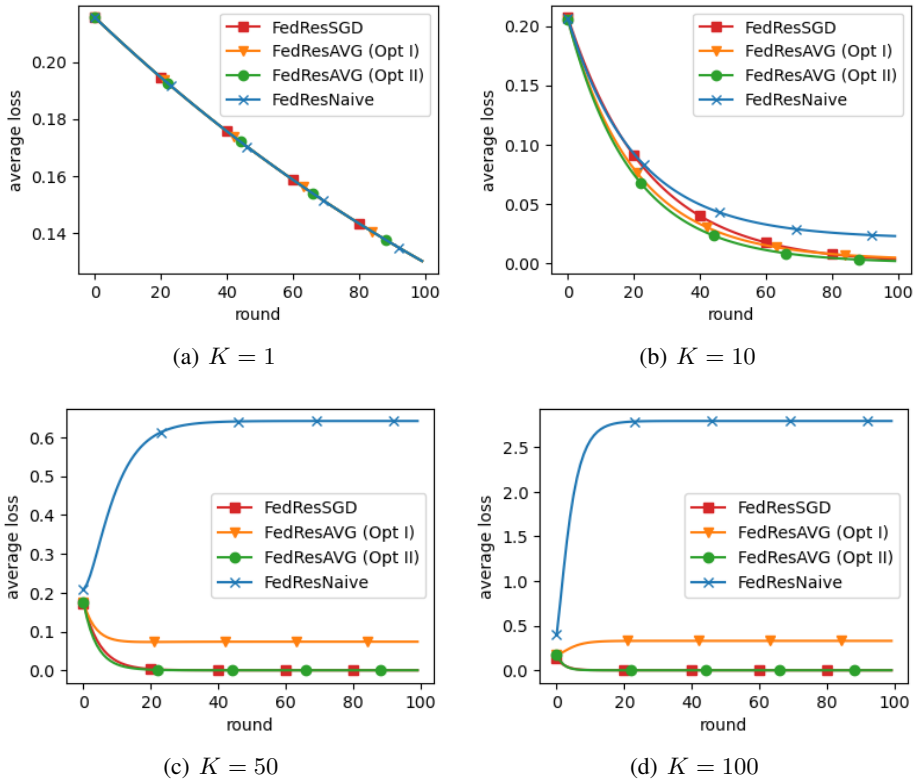
$$y = \operatorname{argmax}\left\{(1 - \alpha)\boldsymbol{W}^*\boldsymbol{x} + \alpha\boldsymbol{\Theta}_i^*\boldsymbol{x} + n\right\} \tag{13}$$

for some $\alpha \in [0, 1]$, zero-mean random noise $n \in \mathbb{R}^M$, and some ground truth matrices $\boldsymbol{W}^*, \boldsymbol{\Theta}_i^* \in \mathbb{R}^{M\times d}$ that are unknown. The value of α controls the heterogeneity of the clients: when $\alpha = 0$, the clients' data have the same distribution; when α becomes larger, they become more and more different. We use logistic loss to evaluate the performance.

In the experiment, we set the number of clients $N = 10$, number of class $M = 2$, and feature dimension $d = 10$. We let all entries of $\boldsymbol{W}^*$ and $\boldsymbol{\Theta}_i^*$ be independently drawn from $\mathcal{N}(0, 1)$. For each sample, we independently draw $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{0}, I)$ and $n \sim \mathcal{N}(\boldsymbol{0}, I)$.

In the experiment, we use logistic regression with linear models to model the class probability:

$$p(y) \propto \exp\left((\boldsymbol{W}\boldsymbol{x} + \boldsymbol{\Theta}_i\boldsymbol{x})_y\right),$$



(a) $K = 1$ (b) $K = 10$

(c) $K = 50$ (d) $K = 100$

Figure 1: Average loss versus communication round with $N = 2$ and synthetic losses (Eq. (11) and (12)), under different K (K is the number of local updates within one communication round)

where ($\boldsymbol{v}_y$ denotes the y-th entry of the vector $\boldsymbol{v}$. We compare our algorithms with FEDAVG and SCAFFOLD, which only maintain a global model, and a close variant of FEDPROX that gives local models. In FEDPROX, in the local update phase, each client minimizes the following function when receiving loss ℓ:

$$\ell(\boldsymbol{w}_i) + \frac{\mu}{2} \left\| \boldsymbol{w}_i - \overline{\boldsymbol{w}} \right\|^2, \tag{14}$$

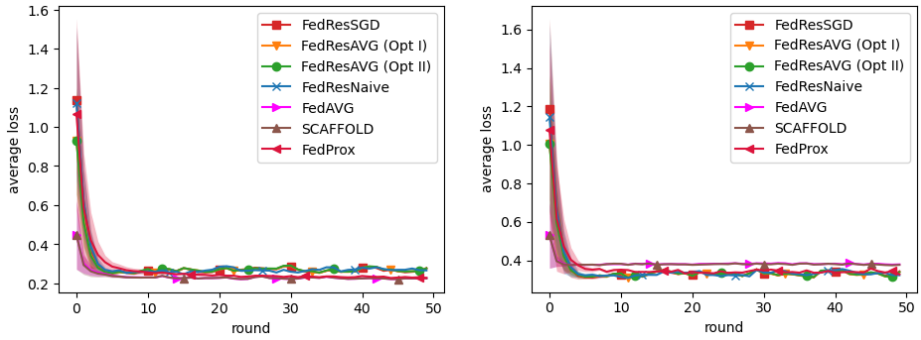
where $\boldsymbol{w}_i$ is the local copy of the global parameter, and $\overline{\boldsymbol{w}}$ is the most recent global parameter sent by the server. The idea is to restrict the drift of $\boldsymbol{w}_i$ from $\overline{\boldsymbol{w}}$. Our implementation is by performing gradient descent on (14). That is, we update $\boldsymbol{w}_i$ with

$$\boldsymbol{w}_i \leftarrow \boldsymbol{w}_i - \eta \left(\nabla \ell(\boldsymbol{w}_i) + \mu \boldsymbol{w}_i - \mu \overline{\boldsymbol{w}} \right).$$

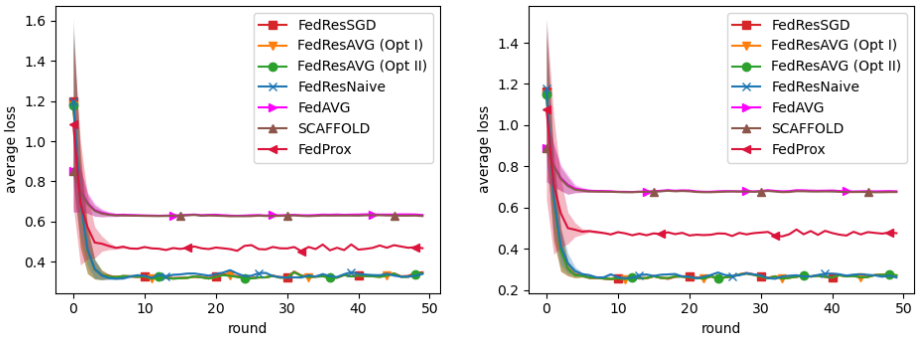
One can also view this as a close variant of SCAFFOLD, but with $\boldsymbol{c}_i = -\mu \boldsymbol{w}_i$ and $\boldsymbol{c} = -\mu \overline{\boldsymbol{w}}$. Although the goal of FEDPROX is still to learn a global model, in the experiment, we allow FEDPROX to use the local copy of the global model $\boldsymbol{w}_i$ (before synchronization with the server) to make predictions, and thus each client essentially has a personalized local model. We show that the regularization term $\frac{\mu}{2} \left\| \boldsymbol{w}_i - \overline{\boldsymbol{w}} \right\|^2$ that brings all clients' personalized model close can hurt the performance when their data distribution is actually not close.

When $\mu = 0$, each $\boldsymbol{w}_i$ is updated from the global model $\overline{\boldsymbol{w}}$ with local data, and thus becomes very similar to our Federated Residual Learning framework where $\boldsymbol{w}_i - \overline{\boldsymbol{w}}$ is essentially the local model (see our discussions in Section 2). We find that increasing μ only monotonically worsens the performance in the considered scenario.

In Figure 2, we plot the results. We fix $K = 50$, $N = 10$. For each algorithm, we choose learning rates through a grid search within $[0.1, 0.001]$ with consecutive values roughly 3 times with each other. The reported loss is the test loss after every communication round. Notice that for FEDPROX, we only plot the result for the case with $\mu = 0.05$.



(a) $\alpha = 0$ (b) $\alpha = 0.3$



(c) $\alpha = 0.7$ (d) $\alpha = 1$

Figure 2: Average loss versus communication round with $N = 10$ and synthetic losses. The loss is logistic loss on a binary classification problem whose label is generated according to Eq. (13). For FEDPROX, we only plot the result with $\mu = 0.05$

Observations. In this synthetic data, maybe because the loss functions are more benign (compared to the one used in Section E.1), the problem of client drift is not present (in terms of performance, FEDAVG $\approx$ SCAFFOLD, and all four algorithms under the FedRes framework behave very similarly). The four FedRes algorithms naturally outperform FEDAVG and SCAFFOLD that only learn global models when the clients' data distribution are different. The performance of FEDPROX is between FedRes and the global approaches.