Balancing Model Performance and Rapid Personalization in Federated Learning with Learning Rate Scheduling

Anonymous authors
Paper under double-blind review

Abstract

Federated learning (FL) is a powerful technique for collaboratively training a single centralized model on distributed local data sources. By aggregating model information without disclosing the local training data, FL preserves data privacy. However, the inherent heterogeneity in local data sets challenges the performance of FL techniques, especially when data is very diverse across local sources. Personalized Federated Learning (PFL) can mitigate this challenge using multiple models but often requires additional memory and computation. This work does not propose a new PFL method but introduces how learning rate decay, within each training round, balances model performance across all local data sets and performance on local data after fine-tuning. We provide both theoretical insights and empirical evidence of efficacy across diverse domains, including vision, text, and graph data. Our extensive experiments demonstrate that learning rate scheduling alone outperforms other FL methods for generalization to new data for both new and existing users. Moreover, it performs comparably to PFL methods, particularly regarding new users, while maintaining similar computation and memory requirements as FL techniques.

1 Introduction

Federated Learning (FL) is a compelling solution for constructing a shared (global) model from several local data sources that inherently cannot be exchanged or aggregated (Yang et al., 2019; Kairouz et al., 2021; Li et al., 2020b; Mahlool & Abed, 2022; Zhang et al., 2021). Learning without sharing data becomes particularly crucial when data privacy or security is paramount, as exemplified by healthcare applications (Li et al., 2020a; Rieke et al., 2020). FL operates through an iterative procedure involving rounds of model improvement. These rounds start with distributing the current global model to local entities (users) and selecting participants to contribute to the model update. The chosen users participate by training their local copies of the model using their respective data and returning the updated model. Next, aggregation of the returned models produces a new global model. The described process represents the typical procedure for FL solutions, which learns only one shared model for all users. The most widely embraced FL method is FedAvg (McMahan et al., 2017), which computes the new global model as the parameter-wise average of returned local models.

When users have similar data, a single model can perform well. However, when local data are very diverse, ensuring good performance across users with a single shared global model can be challenging (Qu et al., 2022; Caldarola et al., 2022). Recent works (Chen et al., 2022; Tan et al., 2022) argue that alternatively, the focus should be on Personalized Federated Learning (PFL) methods that output models tailored to the local data. However, learning additional models can require extra memory and computation, and managing multiple models in PFL may lead to heightened communication requirements between users, a bottleneck in real-world applications. We desire a method that can perform well with heterogeneous user data but with a similar computational cost to FL techniques.

Building upon previous work by Nichol et al. (2018), which discovered that different FL solutions place different fixed emphases on **initial model success**, model performance on aggregated local datasets, and **rapid personalization**, model performance on local data after fine-tuning. This emphasis directly results from the distinct uses of gradients computed during local training for different methods. We introduce within-round learning rate decay, equivalent to scaling the magnitude of gradients computed during local training. Thus, decay will give us control over our solution's focus between initial model success and rapid personalization. Our modification is a generalization of FedAvg (McMahan et al., 2017) and does not require the computation or storage of additional gradients or models. In Section 3.2, we justify that a flexible balance should allow for better performance on various data heterogeneity and that our proposal can achieve such a balance without negatively impacting the convergence rate of FedAvg (Li et al., 2020c) with possibly diverse local data sets.

Finally, in Section 4, we provide extensive experimental results spanning diverse data sets across vision, text, and graph applications. We show that within-round learning rate decay enhances FL techniques and closes remaining performance gaps with PFL methods without the additional computation and communication overhead required by such methods. Importantly, we consistently observe a 1 to 4 percentage point improvement in average test set accuracy for new and existing users over FedAvg.

The organization of this work is as follows. First, Section 2 briefly summarizes related work from personalized federated learning and meta-learning. Second, Section 3 introduces our approach, mathematically justifies its improved flexibility, and establishes its convergence rate. Finally, Section 4 gives empirical evidence for performance improvements and computational cost compared with other benchmark methods.

2 Related Work

Several approaches have been developed to deal with heterogeneous client data in FL. A simple strategy to improve the performance of a single shared model for users with diverse data is to fine-tune the global model at the end of federated training, where **fine-tuning** consists of updating the shared model with the local data of each user to produce a personalized model for each user. It is important to clarify that fine-tuning is not a PFL solution, as the additional computation and models happen after the federated training is completed. To improve the fine-tuned models, instead of learning a shared model through traditional FL, meta-learning (Finn et al., 2017) methods have been applied during federated training (Jiang et al., 2019; Chen et al., 2018; Fallah et al., 2020; Charles & Konečný, 2021) to give a good starting model that accelerates the convergence of fine-tuning. Notably, the effectiveness of such an initialization heavily depends on the availability of sufficient data, as rapid convergence does not guarantee robust performance without subsequent fine-tuning. In addition to applying conventional meta-learning algorithms to the federated learning context, as seen in (Jiang et al., 2019), a variant named Reptile (Nichol et al., 2018) can be shown to be equivalent to FedAvg under the condition of equally sized local data sets. In this article, we exclusively consider aggregation with equally weighted users. Therefore, we will use the term FedAvg to encompass both FedAvg and Reptile and refer to FedAvg as a meta-learning method.

Next, PFL approaches the problem differently by introducing additional models learned during federated training to address data heterogeneity. These intricate solutions include learning a distinct model for each user throughout training (Li et al., 2021a), sharing only a subset of the entire model globally (Li et al., 2021b), or treating user data as a mixture of distributions (Marfoq et al., 2021). Marfoq et al. (2021) can be considered an example of a strategy that clusters or interpolates users (Briggs et al., 2020; Ghosh et al., 2020; Mansour et al., 2020; Sattler et al., 2019) to reduce the additional models to learn. In general, PFL methods often require substantial increases in memory and computation to produce and communicate additional models during federated training, which challenges scalability and resource efficiency. Furthermore, several prevailing methods may prove restrictive or inequitable for applications accommodating new users. For instance, Ditto's approach (Li et al., 2021a) of training personalized models during federated learning becomes infeasible for users who do not participate in training. Likewise, FedBN (Li et al., 2021b) and FedEM (Marfoq et al., 2021), which rely on insights from user data, encounter difficulties when new users lack sufficient data for tailoring a model to new users.

This work focuses on how including within-round learning rate decay can improve FL performance without significant extra computation or the need for additional models. We demonstrate that within-round learning rate decay is a generalization of FedAvg or Reptile. Moreover, unlike the aforementioned meta-learning applications to FL, the amount of learning rate decay can flexibly emphasize initial model performance and rapid personalization. Therefore, FedAvg/Reptile and FOMAML are particularly relevant baselines for comparison. While we do not propose a PFL method by definition, since fine-tuning occurs after federated training, we include PFL methods in our comparisons to assess the performance gap between our method and the state-of-the-art PFL techniques. Our experiments demonstrate consistent improvements compared to FedAvg and FOMAML and close the performance gap with PFL methods without their additional computational or memory overheads.

3 Methodology

This section begins with introducing our notation for the federated learning process. Before introducing the proposed method, we provide background information on the similarities and differences between the popular existing methods: FedSGD, FedAvg (McMahan et al., 2017), and FOMAML (Nichol et al., 2018). We believe this viewpoint of existing algorithms is essential for understanding the derivation and benefits of our proposal. Importantly, we justify that our proposed method can flexibly balance the goals of **initial model success**, model performance on the aggregated local datasets, and **rapid personalization**, model performance on local data after fine-tuning. Recall that **fine-tuning** consists of updating the shared model, after the completion of federated training, with the local data of each user to produce a personalized model for each user. Furthermore, we demonstrate how to maintain the same rate of convergence as FedAvg under certain assumptions while preserving the interpretability of the model performance trade-off mentioned above.

Notation. We consider Federated Learning constructed over $n = 1, \ldots, N$ communication rounds, each consisting of $k = 1, \ldots, K$ local update steps. The set $C = [1, \ldots, M]$ represents the users participating in federated learning, each with local objective function F_i. Federated learning aims to find the model that minimizes the average user objective. Note we distinguish F_i, which is evaluated on the entire local data set, from $F_i^{(n,k)}$, which is evaluated on data from the k-th local update step in round n. In each round, a subset of users, $S \subseteq C$, participates in the update of the global model, and the case $S = C$ is full user participation. We denote the global model at communication round n as $\theta_g^{(n)}$, and $\theta_i^{(n,k)}$ represents the i-th user's copy of the shared model after k of K planned local update steps with learning rate η.

3.1 Global and Local Updates for Federated Learning

Consider any given communication round. To simplify the notation, we omit the communication round n from the notation. We define the size K sequence of loss functions denoted as $\{F_i^{(j)}\}_{j=0}^{K-1}$, where K is the number of local update steps. Recall that $F_i^{(k)}$ is the local objective function of the i-th user evaluated on data from the k-th local update step. Let $g_i^{(k)} = \nabla F_i^{(k)}\left(\theta_i^{(k)}\right)$ denote the gradient of user i on the k-th local update step. Equation 1 defines the standard local update equation for the FedAvg ($K > 1$) or FedSGD ($K = 1$) procedure.

$$\theta_i^{(k)} = \theta_i^{(k-1)} - \eta g_i^{(k-1)} \text{ for all } k = 1, \ldots, K \tag{1}$$

After K-steps of local training, we aggregate the updated models $\theta_i^{(K)}$ or, equivalently, the changes made by local updates as in Equation 2.

$$\theta_g \leftarrow \frac{1}{|S|}\sum_{i \in S} \theta_i^{(K)} = \theta_g + \frac{1}{|S|}\sum_{i \in S} \underbrace{\left(\theta_i^{(K)} - \theta_g\right)}_{-\eta \times g_{Method}} \tag{2}$$

Our comparison of various methods will focus on understanding their expressions of the $\theta_i^{(K)} - \theta_g$ term given in Equation 2. For the FedAvg procedure, we can use Equation 1 recursively to write

$$\theta_i^{(K)} = \theta_i^{(0)} - \eta \sum_{j=0}^{K-1} g_i^{(j)} \tag{3}$$

Noting $\theta_i^{(0)} = \theta_g$, the change a user makes with the FedAvg procedure, $\theta_i^{(K)} - \theta_g$, is driven by $\sum_{j=0}^{K-1} g_i^j$. We refer to this component of a user's change as the **method gradient**, g_{Method}, due to its formulation and use in Equation 3. Similarly, other FL methods have a method gradient consisting of the gradients at local update steps. Equation 4 presents the specific gradients used from Equation 1 for FedSGD, FedAvg (McMahan et al., 2017), and FOMAML (Nichol et al., 2018).

$$g_{FedSGD} = g_i^{(0)}, \quad g_{FedAvg} = \sum_{j=0}^{K-1} g_i^{(j)}, \quad g_{FOMAML} = g_i^{(K-1)} \tag{4}$$

3.2 FedDecay: Generalizing Local Updates With Gradient Decay

Within-Round Learning Rate Decay. From Equation 4, FedSGD, FedAvg, and FOMAML fully include or omit certain gradient terms from the summation. We introduce within-round learning rate decay, **FedDecay**, to enable a more general use of the local gradients.

$$\theta_i^{(k)} = \theta_i^{(k-1)} - \eta\beta^{k-1} g_i^{(k-1)} \text{ for all } k = 1, \ldots, K \tag{5}$$

$$g_{FedDecay} = \sum_{j=0}^{K-1} \beta^j g_i^{(j)} \tag{6}$$

In this work, we primarily focus on using exponential decay as it simplifies the mathematical justification for the benefits of within-round learning rate decay. In Appendix A.1, the coming results for FedDecay are presented more generally for arbitrary positive sequences. Additionally, Appendix B.2 demonstrates that linear decay can improve performance. Equation 5 gives exponential within-round learning rate decay, which leads to Equation 6, the method gradient for FedDecay. Our proposed method generalizes previous work by recovering FedSGD when $\beta = 0$ and FedAvg when $\beta = 1$.

Balancing Initial Model Success and Rapid Personalization. The justification for FedDecay comes from a Talyor analysis of the method gradient from Equation 6. Following the analysis in Nichol et al. (2018), each method gradient is approximately the weighted sum of two other gradient terms: the first promotes initial model success, and the second contributes towards rapid personalization. Incorporating within-round learning rate decay allows for a flexible emphasis on each objective, which can be optimized by tuning the hyperparameters associated with the given decay scheme.

Consider the K-length sequence of loss functions $\{F_i^{(j)}\}_{j=0}^K$ for the i-th user's local update. Let $\tilde{g}_i^{(j-1)} = \nabla F_i^{(j-1)}(\theta_g)$ and $\tilde{H}_i^{(j-1)} = \nabla^2 F_i^{(j-1)}(\theta_g)$ denote the gradient and Hessian, respectively, of the j-th loss function evaluated instead at the most recent global model. We take expectation over the user i and mini-batches k and l from the local data. Equation 7 gives the expected method gradient of FedDecay for exponential decay and exists in greater generality in Section A.1.

$$\mathbb{E}\left[g_{FedDecay}\right] \approx \left(\frac{1-\beta^K}{1-\beta}\right)\mathbb{E}\left[\tilde{g}_i^{(k)}\right] - \left(\frac{(1-\beta^{K-1})(1-\beta^K)}{(1+\beta)(1-\beta)^2}\right)\mathbb{E}\left[\tilde{H}_i^{(k)}\tilde{g}_i^{(l)}\right] \tag{7}$$

We refer to the other expected gradient terms $\mathbb{E}[\tilde{g}_i^{(k)}]$ and $\mathbb{E}[\tilde{H}_i^{(k)}\tilde{g}_i^{(l)}]$ as AvgGrad and AvgGradInner, respectively. First, moving in the negative direction of AvgGrad attempts to minimize the expected loss across users and data, encouraging the model to perform well on aggregated local datasets, otherwise known as initial model success. Second, moving in the positive direction of AvgGradInner maximizes the inner product between gradients (as illustrated in Equation 8 by the chain rule) from distinct mini-batches within the same user. Large inner products indicate that the gradient will recommend moving in a similar direction regardless of the input data, which helps facilitate rapid personalization.

$$\mathbb{E}\left[\tilde{H}_i^k \tilde{g}_i^l\right] = \frac{1}{2}\mathbb{E}\left[\tilde{H}_i^k \tilde{g}_i^l + \tilde{H}_i^l \tilde{g}_i^k\right] = \frac{1}{2}\mathbb{E}\left[\frac{\partial}{\partial\theta_g}\left(\tilde{g}_i^k \tilde{g}_i^l\right)\right] \tag{8}$$

When users possess similar data, a method consisting of only AvgGrad terms can perform well across all users; however, in cases where users' data diverges significantly, an approach prioritizing AvgGradInner terms may be more suitable. Plausibly, most applications exist between the above two scenarios. In such cases, a method striking a flexible balance between AvgGrad and AvgGradInner terms could yield superior performance across a spectrum of statistical data heterogeneity. Please see Appendix B.1 for experimental evidence for the above claims. In Equation 7, large values of β place more and more relative emphasis on AvgGradInner, while all of the emphasis is on AvgGrad when $\beta = 0$. Thus, the new hyper-parameter β controls the emphasis of FedDecay between the objectives of initial model success and rapid personalization.

AvgGrad and AvgGradInner also appear in the expected method gradients of FedSGD, FedAvg, and FOMAML (Nichol et al., 2018). By comparing the ratio of the weights of AvgGradInner to AvgGrad for the considered methods, we can assess each method's relative prioritization of rapid personalization vs. initial model success. See Table 1 and contrast the flexible ratio of FedDecay with those of FedSGD, FedAvg, and FOMAML. FedSGD, FedAvg, and FOMAML have fixed ratios, given K. On the other hand, the ratio of FedDecay still depends on the choice of β.

Method	FedSGD	**FedDecay**	FedAvg	FOMAML
Ratio	$\frac{0}{1}$	$\frac{\left(\beta - \beta^K\right)\eta}{1-\beta^2}$	$\frac{(K-1)\eta}{2}$	$(K-1)\eta$
Limit ($K \to \infty$)	0	$\frac{\beta\eta}{1-\beta^2}$	∞	∞

Table 1: Ratios of coefficients for AvgGradInner (rapid personalization) to AvgGrad (initial model success) terms for several methods. Choice of β gives FedDecay much greater control over the ratio of emphasis on the ability to personalize vs. initial model success. Note that we are assuming that $\beta \in (0, 1)$ for FedDecay since FedSGD ($\beta = 0$) and FedAvg ($\beta = 1$) are already present.

Regardless of the similarity of local data sets, FedSGD, FedAvg, and FOMAML will always place the same fixed emphasis on initial model success and rapid personalization. Thus, FedDecay is the only method offering the flexibility to optimize this intricate trade-off, with hyper-parameter β, particularly as the number of local steps increases (communication becomes less frequent). This additional flexibility allows FedDecay to adapt to the problem-specific data heterogeneity for better performance.

Convergence Analysis. Within-round learning rate decay allows for an application-specific hyper-parameter tuning of the emphasis between initial model success and rapid personalization. Previous work by Li et al. (2020c) demonstrates the convergence of FedAvg on non-independent and non-identically distributed (i.i.d.) local data sets is achieved with some learning rate at iteration t of the form $\eta_t = \alpha_t = \frac{c}{t+d}$ where c and d are positive. Note that the local update step k of the communication round n is equivalent to iteration $t = n * K + k$. We argue that this strategy is insufficient for convergence and balancing initial model success with rapid personalization. Performing federated training for many communication rounds is needed to guarantee a close-to-optimal objective function value. However, for example, we should be able to do this for similar users while focusing on initial model success, which is impossible with the $\eta_t = \alpha_t$ learning rate scheduling. Such decay of the learning rate every iteration places more relative emphasis on rapid personalization with each communication round.

Instead, by composing our within-round learning rate decay, with previous decay across-round instead of each iteration, we can maintain the same convergence rate as FedAvg (Li et al., 2020c) while keeping the relative balance between initial model success and rapid personalization constant across communication rounds. The resulting learning rate is given by $\eta_t = \alpha_{\lfloor t/K \rfloor}\beta_{(t \bmod K)}$ where $\beta_0 \neq 0$ and $\beta_{k+1} \leq \beta_k$ for $k = 1, \ldots, K$. We extend the previous work Li et al. (2020c) to this modified decay scheme under the same assumptions. FedDecay converges with complexity of $\mathcal{O}(N^{-1})$ on non-i.i.d. data as total communication rounds $N \to \infty$ for both full and partial user participation. Please see Appendix A.2 for additional details and proofs.

Assumption 1 (Lipschitz Continuous Gradients)**.** *For all users $i \in \{1, \ldots, M\}$, the local objective function F_i is L-smooth. For all v and w in the domain of F_i,*

$$F_i(v) \leq F_i(w) + (v-w)^T \nabla F_i(w) + \frac{L}{2} \left\| v - w \right\|_2^2$$

Assumption 2 (Strong Convexity)**.** *For all users $i \in \{1, \ldots, M\}$, the local objective function F_i is μ-strongly convex. Similarly,*

$$F_i(v) \geq F_i(w) + (v-w)^T \nabla F_i(w) + \frac{\mu}{2} \left\| v - w \right\|_2^2$$

Assumption 3 (Bounded Variance)**.** *Let mini-batch $\xi^{(n,k)}$ be sampled uniformly from the i-th user's local data. For all n and k, an upper bound exists on the variance of stochastic gradients in each user i.*

$$\mathbb{E} \left\| \nabla F_i^{(n,k)} \left(\theta_i^{(n,k)}, \xi_i^{(n,k)} \right) - \nabla F_i \left(\theta_i^{(n,k)} \right) \right\|^2 \leq \sigma_i^2$$

where $F_i^{(n,k)}$ is the average for i-th user's local loss evaluated on data ξ_i.

Assumption 4 (Bounded Expected Squared Norm)**.** *The expected squared norm of stochastic gradients is uniformly bounded. Similarly,*

$$\mathbb{E} \left\| \nabla F_i^{(n,k)} (\theta_i^{(n,k)}, \xi_i^{(n,k)}) \right\|^2 \leq G^2$$

Here, we consider the more general weighted objective function $F = \sum_{i \in C} p_i F_i$ and aggregation step given by Equation 9. Recall that F_i is the average local loss over all of the local data of the i-th user and $S \subseteq C$ of fixed size $|S|$ contains the users chosen to update the global model in a given round of Federated Learning. S is sampled uniformly without replacement for each round with probabilities p_i.

$$\theta_g \leftarrow \frac{|C|}{|S|} \sum_{i \in S} p_i \theta_i^{(K)} \tag{9}$$

Furthermore, let F^* and F_i^* denote the global minimums of the average objective function of users and the i-th user's average local loss over their data, respectively. We introduce $\Gamma = F^* - \sum_{i=1}^{M} p_i F_i^*$ to represent the degree to which data sets are non-independent and non-identically distributed. The magnitude of Γ reflects how heterogeneous the data distributions are and $\Gamma \to 0$ as the number of local data samples grows only when data are iid.

Theorem 1 (Convergence Of FedDecay)**.** *Let Assumptions 1 to 4 hold and L, μ, and G be defined therein and denote the condition number with $\kappa = L/\mu$. Choose a positive, locally decaying sequence $\beta_{k+1} \leq \beta_k$ for $k = 1, \ldots, K$. With a learning rate for iteration $t = nK + k$ (k local updates into round $n+1$) of the form $\eta_t = \alpha_{\lfloor t/K \rfloor} \beta_{t \bmod K}$ for positive $\beta_j \geq \beta_{j+1}$ and some $\alpha_j = \dfrac{c}{j+d}$. If $\alpha_j = \dfrac{3}{\mu \beta_{K-1} (j+d)}$ where $d = \max \left\{ \dfrac{12\kappa}{\beta_{K-1}}, 4 - \dfrac{2}{K} \right\}$ then*

$$\mathbb{E} \left[F(\theta_g^{(N)}) \right] - F^* \leq \frac{\kappa}{N+d-2} \Big(\frac{9(B+D)}{2\mu\beta_{K-1}^2} + \frac{(1/K)+d-2}{2} \times \mathbb{E} \left\| \theta_g^{(0)} - \theta_g^* \right\|^2 \Big) \tag{10}$$

$$\text{where } B = \sum_{i=1}^{M} p_i^2 \sigma_i^2 + 6L\Gamma + 2 \left(G(K-1)\beta_0 \beta_{K-1}^{-1} \right)^2 \tag{11}$$

$$\text{and } D = \frac{|C| - |S|}{|S| \left(|C| - 1 \right)} \tag{12}$$

The convergence analysis outlined in Theorem 1 emphasizes that the gap between the optimal solution F^* and the value $F(\theta_g^{(N)})$ evaluated on the global model, produced by FedDecay after N total communication rounds, converges to zero as $N \to \infty$. Akin to the outcomes seen with FedAvg, convergence happens at rate $\mathcal{O}(N^{-1})$. However, the advantage of FedDecay lies in its ability to achieve this convergence while accommodating a spectrum of diverse local updates from the sequence $\{\beta_k\}_{k=1}^{K}$. As illustrated in Section 3.2, incorporating within-round learning rate decay enables a versatile balance between initial model success and rapid personalization. Our research demonstrates that this enriched flexibility does not have to come at the expense of deteriorating algorithmic performance.

4 Experimental Results

In this section, we comprehensively evaluate our proposed method, FedDecay, alongside other prominent federated learning techniques, using the benchmark established by Chen et al. (2022). To ensure uniformity, we integrate our approach into their code base available at GitHub [1] [2]. We leverage this established infrastructure to experiment with identical data sets and models to facilitate a rigorous and fair comparison.

Data. As discussed in Yuan et al. (2022), generalization in federated learning refers both to new users (participation gap) and new data for existing users (out-of-sample gap). Our experiments involve holding out both users and data to study both types of generality in the following manner. For each data set, we maintain 20 percent of users as a holdout set, thereby simulating the challenge of generalization to new users. Also, we divide each user's data into distinct training, validation, and testing subsets. Our experiments encompass the various data sets FEMNIST (Caldas et al., 2018), SST2 (Wang et al., 2018; Socher et al., 2013), and PUBMED (Namata et al., 2012). This diversified selection of data sets empowers us to investigate performance across varied domains, including vision, text, and graph data. We use the default data settings chosen by Chen et al. (2022).

- FEMNIST: A 62-way handwritten character classification problem with images of size 28×28. This sub-sampled version of FEMNIST in Chen et al. (2022) contains 43,400 images from 3,550 authors as clients.
- SST2: A sentiment classification data set containing 68,200 movie review sentences labeled with human sentiment. Partitioned into 50 clients using Dirichlet allocation with $\alpha = 0.4$, it enables the assessment of text-based applications.
- PUBMED: A graph of 19,717 nodes and 44,338 edges, this data set classifies scientific publications into three classes. Louvain (Blondel et al., 2008) community partitioning parititons the graph into five users.

Model. In our experiments, we adhere to the model architectures prescribed by Chen et al. (2022) for consistency and comparative purposes.

- FEMNIST: We employ a Convolutional Neural Network (CNN) with a hidden size of 2,048. The model incorporates two convolutional layers with 5×5 kernels, followed by max pooling, batch normalization, ReLU activation, and two dense layers.
- SST2: Our model leverages a pre-trained BERT-Tiny model (Turc et al., 2019) with 2-layer Transformer encoders and a hidden size 128.
- PUBMED: We employ the Graph Isomorphism Neural Network (GIN) (Xu et al., 2019), featuring 2-layer convolutions with batch normalization, a hidden size 64, and a dropout rate of 0.5.

Methods. To establish the prowess of FedDecay, we subject it to a rigorous comparison with a range of federated learning techniques.

---

[1] Original benchmark at `https://github.com/alibaba/FederatedScope/tree/Feature/pfl_bench`

[2] Our modifications at `https://github.com/JoeLavond/FedDecay/tree/bench`

- Federated Learning: In this category, we include FedAvg/Reptile (McMahan et al., 2017; Nichol et al., 2018) and FOMAML (Nichol et al., 2018). These methods make up the most relevant baselines to our proposal as they all can be seen, like our proposal, as various within-round learning rate scheduling.
- Personalized Federated Learning: These encompass Ditto (Li et al., 2021a), FedBN (Li et al., 2021b), FedEM (Marfoq et al., 2021), and pFedMe (T Dinh et al., 2020). Interpret these PFL solutions as an upper bound on what performance could be for our method as they require additional computation cost and memory.

Hyperparameters. We extend the hyper-parameter optimization methodology outlined in the original benchmark paper. Leveraging a grid search technique, we conduct comprehensive explorations, leveraging early termination and hyperband stopping (Biewald, 2020). Our exhaustive search covers a spectrum of hyperparameters, as detailed in Table 2.

Hyper-parameter	Algorithm	Data Set	Tuning Grid
Local Epochs	-	-	$\{1, 3\}$
Batch Size	-	SST2	$\{16, 32, 64\}$
Learning Rate	-	-	$\{0.005, 0.01, \ldots, 1.0\}$
Regularization Rate	Ditto, pFedMe	-	$\{0.05, 0.1, 0.2, 0.5, 0.9\}$
Meta-learning Step	pFedMe	-	$\{1, 3\}$
Mixture Number	FedEM	-	$\{3\}$
Local Decay	FedDecay	-	$\{0, 0.2, \ldots, 1.0\}$

Table 2: Hyper-parameter grid search details. If no algorithm or data set is specified, the given hyper-parameter search will be applied for all.

The final hyperparameters for each run are selected based on the highest average validation accuracy across users. If runs do not terminate early, they will run for 1000, 500, and 500 epochs for FEMNIST, SST2, and PUBMED, respectively. Other fixed configurations worth noting are 20 percent partial user participation for FEMNIST and SST2, batch size 32 for FEMNIST, and full-user participation with full-batch training for PUBMED. To ensure a fair comparison and alignment with other methods, we maintain a fixed learning rate (α) for FedDecay, while β is the key hyperparameter explored. Fine-tuning happens for one epoch with the same learning rate α for all methods. We conduct our experiments on 4 NVIDIA GeForce RTX 3090 GPUs. See Appendix B.1 to analyze FedDecay's performance under misspecified β.

4.1 Generalization to New and Existing Users

Generalization Performance On New Users. To assess the robustness of the methods in accommodating new users, we introduce the following performance metrics: $\bar{Acc}$ (average accuracy), $\breve{Acc}$ (bottom ten percentile accuracy), and σ_{Acc} (standard deviation of accuracy). Average accuracy is the primary performance metric, and it is desirable to identify methods that often return large values for $\bar{Acc}$, indicating that the solution performs well for users, on average. In addition, we use the fairness metrics $\breve{Acc}$ and σ_{Acc} to understand if solutions are performing well for many users. A fair method consistently returns a large bottom ten percentile of accuracy and a low standard deviation, which indicates a lower bound on performance for the 90 percent majority of users and that the solution performs similarly across users. Please note that in the PUBMED data set comprising only five users, $\breve{Acc}$ and σ_{Acc} are not applicable due to the limited number of held-out users. Additionally, we repeat the single-model-based results across multiple seeds in Appendix B.3 to gain insights into result variability and partially assess the sensitivity of our findings to different initial conditions.

We initiate our analysis by investigating the proficiency of each method in catering to new users. See Table 3 Our approach, FedDecay, notably outperforms all FL methods in terms of average test accuracy for all data sets. Furthermore, even in the FEMNIST data sets, it is worth highlighting that FedDecay significantly narrows the performance gap between FL and PFL methods. While personalized methods such as FedEM and pFedMe exhibit competitive results on FEMNIST, their performance drops noticeably in the SST2 data set. These findings underscore FedDecay's exceptional ability to generalize to new users across diverse applications effectively.

Type	Method	FEMNIST (image)			SST2 (text)			PUBMED (graph)		
		$\bar{Acc}$	$\breve{Acc}$	σ_{Acc}	$\bar{Acc}$	$\breve{Acc}$	σ_{Acc}	$\bar{Acc}$	$\breve{Acc}$	σ_{Acc}
Personalized Method	Ditto	0.5672	0.4444	0.0921	0.4746	0.0000	0.4010	0.2442	-	-
	FedBN	0.9059	0.8302	0.0544	**0.8030**	**0.6667**	0.1275	**0.8004**	-	-
	FedEM	**0.9175**	0.8333	0.0526	0.7404	0.6379	0.2141	0.7879	-	-
	pFedMe	0.9036	**0.8438**	0.0751	0.7785	0.6406	0.1179	0.7932	-	-
Single-model Based	FOMAML	0.8989	0.8113	0.0604	0.7680	0.6667	0.1057	0.7950	-	-
	FedAvg	0.9055	**0.8491**	0.0530	0.7680	0.6667	0.1057	0.7914	-	-
	FedDecay	**0.9152**	0.8421	0.0485	**0.8101**	**0.6724**	0.1087	**0.8039**	-	-

Table 3: Generalization to new users who did not participate in federated training after fine-tuning. FedDecay has the most considerable average test set accuracy ($\bar{Acc}$) of any single-model-based technique on all data sets, even outperforming personalized methods on SST2 and PUBMED. Note that these PFL methods are upper bounds, not relevant baselines, due to their higher computational cost and memory requirements. With five total users for PUBMED, only a single user is held out to evaluate generalization. Hence, there are no values for the bottom ten percentile ($\breve{Acc}$) or standard deviation (σ_{Acc}) for new users.

Generalization Performance On New Data For Existing Users Turning our attention to the performance of users who participate in the federated learning process, we anticipate that personalized federated learning methods, characterized by additional memory and computation for learning personalized models, should outperform FL. The results, presented in Table 4, affirm this expectation. However, among the FL methods, FedDecay emerges as the front-runner, achieving the highest average and bottom ten percentile test accuracy across all data sets. Furthermore, FedDecay even secures the highest average accuracy among all methods on the SST2 data set.

Delving into personalized federated learning techniques, we observe mixed results. Ditto and FedBN struggle on the SST2 data set, pFedMe faces challenges on PUBMED, and FedEM encounters difficulties on FEMNIST. Additionally, it is worth noting that FedDecay consistently outperforms other methods on at least one data set despite their additional computation and memory requirements. Please refer to Section 4.2 for a detailed exploration of the cost implications.

Type	Method	FEMNIST (image)			SST2 (text)			PUBMED (graph)		
		$\bar{Acc}$	$\breve{Acc}$	σ_{Acc}	$\bar{Acc}$	$\breve{Acc}$	σ_{Acc}	$\bar{Acc}$	$\breve{Acc}$	σ_{Acc}
Personalized Method	Ditto	0.9031	0.8333	0.0563	0.5949	0.0417	0.3449	0.8754	0.8465	0.0236
	FedBN	0.9182	0.8571	0.0548	0.7360	0.4000	0.2292	0.8788	**0.8540**	0.0248
	FedEM	0.8952	0.8200	0.0806	**0.7808**	0.5000	0.1845	**0.8822**	0.8478	0.0322
	pFedMe	**0.9280**	**0.8750**	0.0694	0.7699	**0.6087**	0.1672	0.8666	0.8205	0.0334
Single-model Based	FOMAML	0.8951	0.8140	0.0636	0.7654	**0.5556**	0.1634	0.8614	0.8292	0.0284
	FedAvg	0.8851	0.8039	0.0750	0.7654	**0.5556**	0.1634	0.8671	0.8342	0.0267
	FedDecay	**0.8986**	**0.8214**	0.0711	**0.7815**	**0.5556**	0.1727	**0.8722**	**0.8490**	0.0233

Table 4: Generalization to new data for existing users who participated in federated training after fine-tuning. FedDecay produces the best test set average and bottom ten percentile accuracy of any single-model-based method. Note that these PFL methods are upper bounds, not relevant baselines, due to their higher computational cost and memory requirements.

4.2 Computational and Memory Costs

The computational and communication costs incurred during training are integral to evaluating federated learning methods. Personalized federated learning methods, in particular, often demand increased computation and memory resources. In this context, we draw attention to the cost implications of various methods.

As depicted in Figure 1, FedDecay exhibits comparable computational costs to other FL techniques. The cost of learning rate scheduling is negligible per iteration, and the potential increase in communication rounds due to decaying updates does not meaningfully affect computation. In our experiments, FedDecay returns an identical cost to FedAvg for FEMNIST and SST2, which terminate in the same number of epochs. FedDecay achieves similar or superior results without considerably higher communication and computation costs.

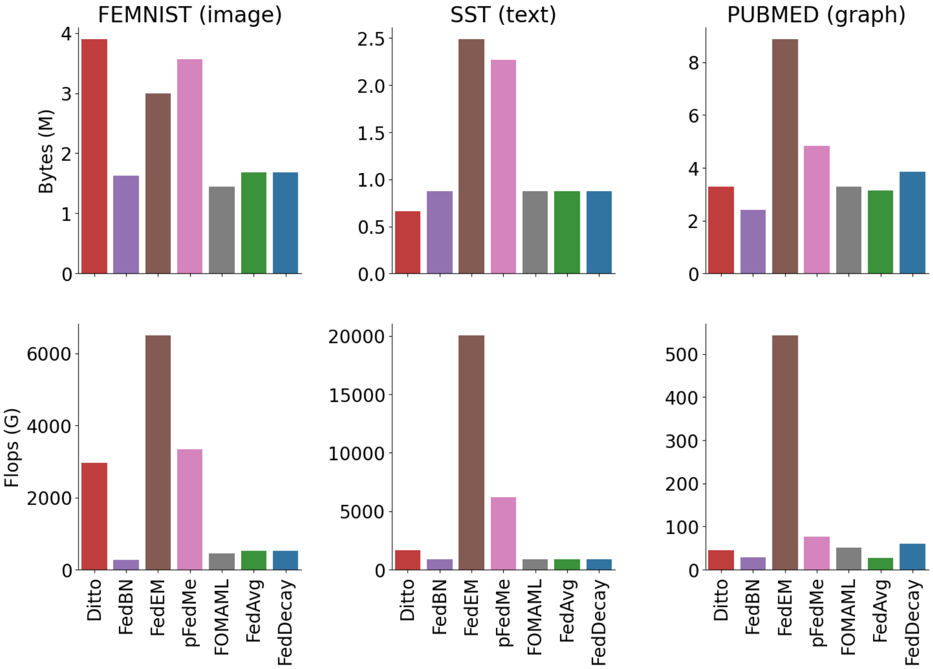

Figure 1: Computational And Memory Costs. Total bytes communication (*top*) and total floating point operations (*bottom*) performed by the best-performing run of each method. FedDecay requires similar costs to other FL techniques, like FedAvg. Furthermore, FedDecay incurs considerably lower costs than personalized methods, especially FedEM and pFedMe.

FedDecay requires substantially less memory and computation than FedEM and pFedMe, the two personalized methods that occasionally outperformed FedDecay in Section 4.1. Although FedDecay does not consistently outperform personalized federated learning methods, it is much more computationally efficient. It often dramatically closes the gap between other FL methods and personalized federated learning solutions. This realization emphasizes FedDecay's efficiency in striking a favorable balance between performance and resource utilization, making it an attractive solution for practical federated learning scenarios.

5 Concluding Remarks

Limitations. While our study offers valuable insights into the performance of our proposed method across various datasets, we acknowledge certain limitations that open avenues for future exploration and optimization. First, our analysis concentrated on a selection of data heterogeneity scenarios, encompassing natural and partitioned heterogeneity. We acknowledge that our findings may not fully encapsulate the dynamics present in more extreme non-iid scenarios.

Next, we focus mainly on exponential decay to show that scaling local gradient updates allows for balancing the objectives of initial model success and the ability to personalize rapidly. However, many alternative decay schemes for learning rates exist, but we have only briefly explored linear decay in Appendix B.2. Future research may explore these alternative decay schemes to enrich our understanding and potentially enhance our method's performance.

Finally, we evaluated our proposed method in isolation without extensively exploring its compatibility with other state-of-the-art techniques. While this approach enabled us to assess our method's intrinsic merits, practical deployment often requires combining multiple strategies for optimal results. Much of our theoretical work relies on stochastic gradient descent as the optimizer, and we intend to expand our theory to alternative optimizers in future work.

Conclusion. In summary, this study introduces an unexplored approach to federated learning by incorporating gradient decay into local updates within each round of training. Within-round learning rate decay is a tool for balancing the shared model's emphasis on initial model success with the ability to personalize rapidly. This additional flexibility allows FedDecay to adapt, more decay for more considerable user differences, to the problem-specific data heterogeneity for better performance without changing the order of convergence. Importantly, using within-round learning rate decay does not require additional memory and computationally requires only a simple grid search to find improved solutions in our experiments, Section 4. Empirically, exponential within-round learning rate decay consistently improves performance, especially for new users, over the popular FedAvg procedure and returns similar solutions to personalized methods that require additional memory and computation.

References

Lukas Biewald. Experiment tracking with weights and biases, 2020. URL `https://www.wandb.com/`. Software available from wandb.com.

Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

Christopher Briggs, Zhong Fan, and Peter Andras. Federated learning with hierarchical clustering of local updates to improve training on non-iid data. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–9, 2020. doi: 10.1109/IJCNN48605.2020.9207469.

Debora Caldarola, Barbara Caputo, and Marco Ciccone. Improving generalization in federated learning by seeking flat minima. In *European Conference on Computer Vision*, pp. 654–672. Springer, 2022.

Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakub Konečnỳ, H Brendan McMahan, Virginia Smith, and Ameet Talwalkar. Leaf: A benchmark for federated settings. *arXiv preprint arXiv:1812.01097*, 2018.

Zachary Charles and Jakub Konečnỳ. Convergence and accuracy trade-offs in federated learning and meta-learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 2575–2583. PMLR, 2021.

Daoyuan Chen, Dawei Gao, Weirui Kuang, Yaliang Li, and Bolin Ding. pfl-bench: A comprehensive benchmark for personalized federated learning. *Advances in Neural Information Processing Systems*, 35:9344–9360, 2022.

Fei Chen, Mi Luo, Zhenhua Dong, Zhenguo Li, and Xiuqiang He. Federated meta-learning with fast convergence and efficient communication. *arXiv preprint arXiv:1802.07876*, 2018.

Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. *Advances in Neural Information Processing Systems*, 33:3557–3568, 2020.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pp. 1126–1135. PMLR, 2017.

Avishek Ghosh, Jichan Chung, Dong Yin, and Kannan Ramchandran. An efficient framework for clustered federated learning. *IEEE Transactions on Information Theory*, 68:8076–8091, 2020. URL `https://api.semanticscholar.org/CorpusID:219530697`.

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

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

Li Li, Yuxi Fan, Mike Tse, and Kuo-Yi Lin. A review of applications in federated learning. *Computers & Industrial Engineering*, 149:106854, 2020a. ISSN 0360-8352. doi: https://doi.org/10.1016/j.cie.2020.106854. URL `https://www.sciencedirect.com/science/article/pii/S0360835220305532`.

Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020b. doi: 10.1109/MSP.2020.2975749.

Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *International Conference on Machine Learning*, pp. 6357–6368. PMLR, 2021a.

Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. In *International Conference on Learning Representations*, 2020c. URL `https://openreview.net/forum?id=HJxNAnVtDS`.

Xiaoxiao Li, Meirui JIANG, Xiaofei Zhang, Michael Kamp, and Qi Dou. FedBN: Federated learning on non-IID features via local batch normalization. In *International Conference on Learning Representations*, 2021b. URL `https://openreview.net/forum?id=6YEQUn0QICG`.

Dhurgham Hassan Mahlool and Mohammed Hamzah Abed. A comprehensive survey on federated learning: Concept and applications. *Mobile Computing and Sustainable Informatics: Proceedings of ICMCSI 2022*, pp. 539–553, 2022.

Yishay Mansour, Mehryar Mohri, Jae Ro, and Ananda Theertha Suresh. Three approaches for personalization with applications to federated learning, 2020.

Othmane Marfoq, Giovanni Neglia, Aurélien Bellet, Laetitia Kameni, and Richard Vidal. Federated multi-task learning under a mixture of distributions. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 15434–15447. Curran Associates, Inc., 2021. URL `https://proceedings.neurips.cc/paper_files/paper/2021/file/82599a4ec94aca066873c99b4c741ed8-Paper.pdf`.

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*, pp. 1273–1282. PMLR, 2017.

Galileo Namata, Ben London, Lise Getoor, Bert Huang, and U Edu. Query-driven active surveying for collective classification. In *10th international workshop on mining and learning with graphs*, volume 8, pp. 1, 2012.

Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018.

Zhe Qu, Xingyu Li, Rui Duan, Yao Liu, Bo Tang, and Zhuo Lu. Generalized federated learning via sharpness aware minimization. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 18250–18280. PMLR, 17–23 Jul 2022. URL `https://proceedings.mlr.press/v162/qu22a.html`.

Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N Galtier, Bennett A Landman, Klaus Maier-Hein, et al. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):119, 2020.

Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 32:3710–3722, 2019. URL `https://api.semanticscholar.org/CorpusID:203736521`.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

Canh T Dinh, Nguyen Tran, and Josh Nguyen. Personalized federated learning with moreau envelopes. *Advances in Neural Information Processing Systems*, 33:21394–21405, 2020.

Alysa Ziying Tan, Han Yu, Lizhen Cui, and Qiang Yang. Towards personalized federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962*, 2019.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5446. URL `https://aclanthology.org/W18-5446`.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=ryGs6iA5Km`.

Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019.

Honglin Yuan, Warren Richard Morningstar, Lin Ning, and Karan Singhal. What do we mean by generalization in federated learning? In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=VimqQq-i_Q`.

Chen Zhang, Yu Xie, Hang Bai, Bin Yu, Weihong Li, and Yuan Gao. A survey on federated learning. *Knowledge-Based Systems*, 216:106775, 2021. ISSN 0950-7051. doi: https://doi.org/10.1016/j.knosys.2021.106775. URL `https://www.sciencedirect.com/science/article/pii/S0950705121000381`.

A Supplementary Proofs

A.1 Balancing Initial Model Success and Rapid Personalization

We extend the work of Nichol et al. (2018) to understand how the updates of FedDecay impact initial model success, rapid personalization, and generalization. Consider the gradient from a sequence of loss functions $\{F_i^{(j)}\}_{j=0}^{K-1}$ where $K \in \{z \in \mathbb{Z} \mid z \geq 2\}$. For example, the above sequence could be a local objective function evaluated on different mini-batches. Consider the following update for any sequence of scaling coefficients $\{\beta_j \mid \beta_0 \neq 0\}_{j=0}^{K-1}$.

$$\theta_i^{(k)} = \theta_i^{(k-1)} - \eta\beta_{k-1} \times \nabla F_i^{(k-1)}(\theta_i^{(k-1)}) \text{ for } k = 1, \ldots, K$$

Let $g_i^{(j)} = \nabla F_i^{(j)}(\theta_i^{(j)})$ and define $\tilde{g}_i^{(j)} = \nabla F_i^{(j)}(\theta_g)$ and $\tilde{H}_i^{(j)} = \nabla^2 F_i^{(j)}(\theta_g)$ as the gradient and Hessian of the j-th loss function evaluated at the initial point.

$$\begin{aligned}
g_i^{(j)} = \nabla F_i^{(j)}(\theta_i^{(j)}) &\approx \nabla F_i^{(j)}(\theta_g) + \nabla^2 F_i^{(j)}(\theta_g)\left(\theta_i^{(j)} - \theta_g\right) \\
&= \tilde{g}_i^{(j)} + \tilde{H}_i^{(j)}\left(\theta_i^{(j)} - \theta_g\right) \\
&= \tilde{g}_i^{(j)} - \eta\tilde{H}_i^{(j)} \sum_{h=0}^{j-1} \beta_h \nabla F_i^{(h)}(\theta_i^{(h)}) \\
&\approx \tilde{g}_i^{(j)} - \eta\tilde{H}_i^{(j)} \sum_{h=0}^{j-1} \beta_h \, \tilde{g}_i^{(h)}
\end{aligned}$$

Apply the previous expansion to the update of FedDecay.

$$\begin{aligned}
g_{FedDecay} = \frac{\theta_i^{(K)} - \theta_g}{-\eta} &= \sum_{j=0}^{K-1} \left(\frac{\theta_i^{(j+1)} - \theta_i^{(j)}}{-\eta}\right) \\
&= \sum_{j=0}^{K-1} \beta_j \nabla F_i^{(j)}(\theta_i^{(j)}) \\
&\approx \sum_{j=0}^{K-1} \beta_j \left(\tilde{g}_i^{(j)} - \eta\tilde{H}_i^{(j)} \sum_{h=0}^{j-1} \beta_h \tilde{g}_i^{(h)}\right) \\
&= \sum_{j=0}^{K-1} \beta_j \tilde{g}_i^{(j)} - \eta \sum_{j=0}^{K-1} \left(\beta_j \tilde{H}_i^{(j)} \sum_{h=0}^{(j-1)} \beta_h \tilde{g}_i^{(h)}\right)
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}\left[g_{FedDecay}\right] &\approx \sum_{j=0}^{K-1} \beta_j \tilde{g}_i^{(j)} - \eta \sum_{j=0}^{K-1} \left(\beta_j \tilde{H}_i^{(j)} \sum_{h=0}^{j-1} \beta_h \tilde{g}_i^{(h)}\right) \\
&= \mathbb{E}\left[\tilde{g}_i^{(j)}\right] \left(\sum_{j=0}^{K-1} \beta_j\right) - \mathbb{E}\left[\tilde{H}_i^{(j)} \tilde{g}_i^{(h)}\right] \left(\eta \sum_{j=0}^{K-1} \left(\beta_j \sum_{h=0}^{j-1} \beta_h\right)\right)
\end{aligned}$$

Hence the following ratio of $\mathbb{E}\left[\tilde{H}_i^{(j)} \tilde{g}_i^{(h)}\right]$ to $\mathbb{E}\left[\tilde{g}_i^{(j)}\right]$ for FedDecay. Let $\mathcal{B}(k) = \sum_{j=0}^{k-1} \beta_{j-1}$.

$$R_{FedDecay} = \frac{\eta \sum_{j=0}^{K-1} \beta_j \mathcal{B}(j)}{\mathcal{B}(K)}$$

Note that these equations do not need to be within-round decay and that some expressions hold more generally. However, our interest is in addressing our concern that existing meta-learning techniques, which facilitate rapid personalization, do not require a model to have good initial success.

Exponential decay $\beta_j = \beta^j$ allows for the simplification of the above ratio after using the finite sum formula for the geometric series, $\mathcal{B}(k) = \frac{1-\beta^k}{1-\beta}$.

$$\begin{aligned}
R_{FedDecay} &= \eta \left(\sum_{j=0}^{K-1} \beta^j \left(\frac{1-\beta^j}{1-\beta} \right) \right) \times \left(\frac{1-\beta}{1-\beta^K} \right) \\
&= \eta\beta \left(\frac{(1-\beta^K)(1-\beta^{K-1})}{(1-\beta)^2(1+\beta)} \right) \times \left(\frac{1-\beta}{1-\beta^K} \right) \\
&= \eta\beta \left(\frac{1-\beta^{K-1}}{(1-\beta)(1+\beta)} \right) \\
&= \eta\beta \left(\frac{1-\beta^{K-1}}{1-\beta^2} \right)
\end{aligned}$$

A.2 Convergence of FedDecay on Heterogenous Data

Full User Participation. We expand the theoretical work of Li et al. (2020c). Furthermore, we adopt the following additional notations and use their Lemma 1 and Lemma 2. Let iteration $t = nK + 1$ denote the k-th local update step of communication round $n + 1$ of federated training. The i-th users model at iteration t is provided by w_t^i. Hence, all previous notations can be converted similarly to $w_t^i = \theta_i^{\lfloor t/K \rfloor,\, t \bmod K}$.

Recall that $S \subseteq C$ denotes the set of users participating in the update of the global model for a given round n of federated training. Here, $S = C$ and let p_i denote the aggregation weight of the i-th user as introduced in Section 3.2. We are most interested in assigning equal probability or aggregation weight to all users since our objective is an improved initialization (for all users).

$$\begin{aligned}
v_{t+1}^i &= w_t^i - \eta_t \nabla \hat{F}_i(w_t^i, \xi_t^i) \\
w_{t+1}^i &= \begin{cases} v_{t+1}^i & \text{if } (t+1) \bmod K \neq 0 \\ \sum_{i=1}^M p_i v_{t+1}^i & \text{else} \end{cases}
\end{aligned}$$

Also, we need to define the following aggregated sequences, which are always equivalent under full user participation. Furthermore, shorthand notation for various gradients is used for simplicity. Note $\bar{w}_t$ is only accessible when $t \bmod K = 0$, $\bar{g}_t = \mathbb{E} g_t$, and $\bar{v}_{t+1} = \bar{w}_t - \eta_t g_t$.

$$\bar{v}_t = \sum_{i=1}^M p_i v_t^i \qquad g_t = \sum_{i=1}^M p_i \nabla F_i(w_t^i, \xi_t^i)$$

$$\bar{w}_t = \sum_{i=1}^M p_i w_t^i \qquad \bar{g}_t = \sum_{i=1}^M p_i \nabla F_i(w_t^i)$$

We take the following two lemmas from Li et al. (2020c) and adopt the third to our learning rate scheduling.

Lemma 1. *Assume Assumption 1 and 2. If $\eta_t \leq \frac{1}{4L}$, we have*

$$\begin{aligned}
\mathbb{E} \left\| \bar{v}_{(t+1)} - w^* \right\|^2 \leq (1-\eta_t)\, \mathbb{E} \left\| \bar{w}_t - w^* \right\|^2 + \eta_t^2 \mathbb{E} \left\| g_t - \bar{g}_t \right\|^2 \\
+ 2\mathbb{E} \left[\sum_{i=1}^M p_i \left\| \bar{w}_t - w_i^t \right\|^2 \right] + 6L\eta_t^2 \Gamma
\end{aligned}$$

Lemma 2. *Under Assumption 3, it follows that*

$$\mathbb{E} \left\| g_t - \bar{g}_t \right\|^2 \leq \sum_{i=1}^M p_i^2 \sigma_i^2$$

Lemma 3. *Under Assumption 4, a locally decaying, cyclic learning rate of the form* $\eta_t = \alpha_{\lfloor t/K \rfloor} \beta_{t \bmod K}$ *such that* $\beta_{t+1} \le \beta_t$ *for* $t = 0, \dots, K-1$ *satisfies*

$$E\left[\sum_{i=1}^{M} p_i \left\|\bar{w}_t - w_t^i\right\|^2\right] \le \eta_t^2 \left(G(K-1)\beta_{K-1}^{-1}\right)^2$$

Proof. Note that for all $t \ge 0$ there exists $t_0 \le t$ such that $t - t_0 \le K-1$ and $w_{t_0}^i = \bar{w}_{t_0}$ for all $i \in [1, \dots, M]$

$$\begin{aligned}
&\mathbb{E}\left[\sum_{i=1}^{M} p_i \left\|\bar{w}_t - w_t^i\right\|^2\right] \\
&= \mathbb{E}\left[\sum_{i=1}^{M} p_i \left\|\left(w_t^i - \bar{w}_{t_0}\right) - \left(\bar{w}_t - \bar{w}_{t_0}\right)\right\|^2\right] \\
&\le \mathbb{E}\left[\sum_{i=1}^{M} p_i \left\|\left(w_t^i - \bar{w}_{t_0}\right)\right\|^2\right] \\
&\text{as } \mathbb{E}\left\|X - \mathbb{E}X\right\| \le \mathbb{E}\left\|X\right\|^2 \text{ where } X = w_t^i - \bar{w}_{t_0} \\
&\le \sum_{i=1}^{M} p_i \mathbb{E}\left[(K-1)\sum_{i=t_0}^{t-1} \eta_t^2 \left\|\nabla \hat{F}_i\left(w_t^i, \xi_t^i\right)\right\|^2\right] \\
&\text{as } \left\|\sum_{i=t_0}^{t-1} \eta_t \nabla \hat{F}_i\left(w_t^i, \xi_t^i\right)\right\|^2 \quad \le (t - t_0)\sum_{i=t_0}^{t-1} \eta_t^2 \left\|\nabla \hat{F}_i\left(w_t^i, \xi_t^i\right)\right\|^2 \\
&\le (K-1)\sum_{i=1}^{M} p_i \sum_{i=t_0}^{t-1} \eta_{t_0}^2 G^2 \\
&\text{as } \mathbb{E}\left\|\nabla \hat{F}_i\left(w_t^i, \xi_t^i\right)\right\|^2 \le G^2 \text{ and } \eta_t \le \eta_{t_0} \text{ for } t_0 \le t \le t_0 + K \\
&\le (K-1)\sum_{i=1}^{M} p_i \sum_{i=t_0}^{t-1} \left(\frac{G\beta_0}{\beta_{K-1}}\right)^2 \\
&\text{as } \eta_{t_0} \le \eta_t \left(\frac{\beta_0}{\beta_{K-1}}\right) \text{ for } t_0 \le t \le t_0 + K \\
&\le \eta_t^2 \left(G(K-1)\beta_0\beta_{K-1}^{-1}\right)^2 \text{ as } \sum_{i=1}^{M} p_i = 1
\end{aligned}$$

□

The following proves convergence under full user participation $S_{\cdot} = C$.

Proof. Recall that for all t, under full user participation $\bar{w}_t = \bar{v}_t$. Let $\Delta_t = \mathbb{E}\left\|\bar{w}_t - w^*\right\|^2$. From Lemma 1, 2, and 3, it follows that

$$\begin{aligned}
&\Delta_{t+1} \le \left(1 - \eta_t \mu\right)\Delta_t + \eta_t^2 B \\
&\text{where } B = \sum_{i=1}^{M} p_i^2 \sigma_i^2 + 6L\Gamma + 2\left(G(K-1)\beta_0\beta_{K-1}^{-1}\right)^2
\end{aligned}$$

We can assume without loss of generality that $\beta_{t \bmod K} \le 1$ and $\beta_0 = 1$. Otherwise

$$\eta_t = \alpha_{\lfloor t/K \rfloor}\beta_{t \bmod K} = \tilde{\alpha}_{\lfloor t/K \rfloor}\tilde{\beta}_{t \bmod K}$$

where $\tilde{\alpha}_t = \beta_0\left(\frac{c}{t+d}\right)$ and $\tilde{\beta}_t = \frac{\beta_t}{\beta_0} \leq 1$. Additionally, we can assume that all β's are positive-valued as we could reduce K until this is satisfied.

For $\eta_t = \alpha_{\lfloor t/K \rfloor}\beta_{t \bmod K}$ where $\alpha_t = \frac{c}{t+d}$ for some $c > \frac{2}{\mu\beta_{K-1}}$ and $d > 2 - \frac{1}{K}$ such that $\eta_1 \leq \frac{1}{4L}$.

$$\begin{aligned}
&\frac{c}{(t/K)+d} \leq \alpha_{\lfloor t/K \rfloor} \leq \frac{c}{(t/K)+d-1} \\
&\Rightarrow \frac{c\beta_{K-1}}{(t/K)+d} \leq \eta_t \leq \frac{c}{(t/K)+d-1} \\
&\text{as } \beta_{K-1} \leq \beta_{t \bmod K} \leq 1
\end{aligned}$$

By induction, we prove $\Delta_t \leq \dfrac{v}{(t/K)+d-2}$ where

$$v = \max\left\{\frac{c^2 B}{\beta_{K-1}c\mu - 2}, \left[(1/K)+d-2\right]\Delta_1\right\}$$

Note that $t = 1$ holds trivially by the definition of v. Assuming the conclusion holds for some t, then

$$\begin{aligned}
&\Delta_{t+1} \leq (1-\eta_t\mu)\,\Delta_t + \eta_t^2 B \\
&\leq \left(1 - \frac{\beta_{K-1}c\mu}{(t/K)+d}\right)\left(\frac{v}{(t/K)+d-2}\right) + \left(\frac{c}{(t/K)+d-1}\right)^2 B \\
&\quad \text{by the induction hypothesis, } 1-\eta_t\mu \leq 1 - \frac{\beta_{K-1}c\mu}{(t/K)+d}, \\
&\qquad \text{and } \eta_t \leq \left(\frac{c}{(t/K)+d-1}\right) \\
&\leq \left(1 - \frac{\beta_{K-1}c\mu}{(t/K)+d}\right)\left(\frac{v}{(t/K)+d-2}\right) + \frac{c^2 B}{\left[(t/K)+d\right]\left[(t/K)+d-2\right]} \\
&\qquad \text{as } (t/K)+d-2 > (1/K)+2-(1/K)-2 = 0 \\
&\qquad \Rightarrow \left[(t/K)+d\right]\left[(t/K)+d-2\right] \leq \left[(t/K)+d-1\right]
\end{aligned}$$

Letting $a = \dfrac{1}{\left[(t/K)+d\right]\left[(t/K)+d-2\right]}$, we continue with the previous quantity.

$$\begin{aligned}
&= a\left(\left[(t/K)+d-2\right]v + c^2B - \left[c\mu\beta_{K-1}-2\right]v\right) \\
&\leq a\Big(\left[(t/K)+d-2\right]v + c^2B - \left[c\mu\beta_{K-1}-2\right]\left[\frac{c^2B}{\beta_{K-1}c\mu}\right]\Big) \\
&\qquad \text{as } \beta_{K-1}c\mu - 2 > 0 \text{ and } v \leq \frac{c^2B}{\beta_{K-1}c\mu} \\
&= \frac{\left[(t/K)+d-2\right]v}{\left[(t/K)+d\right]\left[(t/K)+d-2\right]} \\
&= \frac{v}{(t/K)+d} \\
&\leq \frac{v}{(t/K)+d}\left(\frac{(t/K)+d}{(t/K)+d-\left(2+\frac{1}{K}\right)}\right) \\
&= \frac{v}{(t/K)+d-\left(2+\frac{1}{K}\right)} \\
&= \frac{v}{\left(\frac{t+1}{K}\right)+d-2}
\end{aligned}$$

Then by L-smoothness of $F(\cdot)$

$$\mathbb{E}\left[F(\bar{w}_t) - F^*\right] \leq \frac{L}{2}\Delta_t \leq \frac{L}{2}\left(\frac{v}{(t/K) + d - 2}\right)$$

Specifically, if we choose the constants to be set as:
$c = \dfrac{3}{\mu\beta_{K-1} - 2}$ and $d = \max\left\{\dfrac{12L}{\mu\beta_{K-1}}, 4 - \dfrac{2}{K}\right\}$ then,

$$\begin{aligned}
\alpha_1 &= \frac{c}{d+1} < \frac{c}{d} = \frac{3}{\mu\beta_{K-1}d} \leq \frac{3}{12L + \mu\beta_{K-1}} \leq \frac{1}{4L} \\
&\quad \text{as } d \geq \frac{12L}{\mu\beta_{K-1}} \\
v &= \max\left\{\frac{c^2B}{\beta_{K-1}c\mu - 2}, \left[(1/K) + d - 2\right]\Delta_1\right\} \\
&\leq \frac{c^2B}{\beta_{K-1}c\mu - 2} + \left[(1/K) + d - 2\right]\Delta_1 \\
&= c^2B + \left[(1/K) + d - 2\right]\Delta_1 \\
&\quad \text{as } c = \frac{3}{\mu\beta_{K-1}} \\
&= \left(\frac{3}{\mu\beta_{K-1}}\right)^2 B + \left[(1/K) + d - 2\right]\Delta_1 \\
&= \frac{2}{\mu}\left(\frac{9B}{2\mu\beta_{K-1}^2} + \frac{(1/K) + d - 2}{2}\Delta_1\right) \\
\mathbb{E}\left[F(\bar{w}_t) - F^*\right] &\leq \frac{L}{2}\left(\frac{v}{(t/K) + d - 2}\right) \\
&\leq \frac{\kappa}{(t/K) + d - 2}\left(\frac{9B}{2\mu\beta_{K-1}^2} + \frac{(1/K) + d - 2}{2}\Delta_1\right)
\end{aligned}$$

□

Partial User Participation. Here, we focus on the case where the random set of users at iteration t (S_t) of size $|S|$ is selected to update the global model. Consider when the central server forms $S.$ by sampling uniformly without replacement. We modify the definition of w_t to incorporate the new averaging scheme.

$$w_{t+1}^i = \begin{cases} v_{t+1}^i & \text{if } (t+1) \bmod K \neq 0 \\ \sum_{i \in S_{t+1}} p_i \frac{M}{|S|} v_{t+1}^i & \text{else} \end{cases}$$

We rely on the following previous work by Li et al. (2020c).

Lemma 4. *If* $t + 1 \bmod K = 0$ *and* $S.$ *is sampled uniformly without replacement, then*

$$\mathbb{E}_{S_t}\left(\bar{v}_{t+1}\right) = \bar{v}_{t+1}$$

Lemma 5. *If* $t + 1 \bmod K = 0$, $S.$ *is sampled uniformly without replacement,* $p_i = \frac{1}{n}$ *for all* $i \in [1, \ldots, M]$, *and* $\eta_t = \alpha_{\lfloor t/K \rfloor}\beta_{t \bmod K}$ *where both* α_j *and* β_j *are non-increasing, then*

$$\mathbb{E}_{S_t}\left\|\bar{v}_{t+1} - \bar{w}_{t+1}\right\|^2 \leq \eta_t^2\left(\frac{M - |S_t|}{K(M-1)}\right)\left(GK\beta_0\beta_{K-1}\right)^2$$

Proof. Replace $\eta_{t_0} \leq 2\eta_t$ with $\eta_{t_0} \leq \eta_t\left(\frac{\beta_0}{\beta_{K-1}}\right)$ in the original proof. □

The following proves convergence under partial user participation in updating the global model. This proof completes Theorem 1.

Proof.

$$\begin{aligned}
&\mathbb{E}_{S_{t+1}} \left\| \bar{w}_{t+1} - w^* \right\|^2 \\
&\quad = \mathbb{E}_{S_{t+1}} \left\| \bar{w}_{t+1} - \bar{v}_{t+1} + \bar{v}_{t+1} - w^* \right\|^2 \\
&\quad = \mathbb{E}_{S_{t+1}} \left\| \bar{w}_{t+1} - \bar{v}_{t+1} \right\|^2 \\
&\qquad + \mathbb{E}_{S_{t+1}} \left\| \bar{v}_{t+1} - w^* \right\|^2 + 2\mathbb{E}_{S_{t+1}} \langle \bar{w}_{t+1} - \bar{v}_{t+1}, \bar{v}_{t+1} - w^* \rangle \\
&\quad = \mathbb{E}_{S_{t+1}} \left\| \bar{w}_{t+1} - \bar{v}_{t+1} \right\|^2 + \mathbb{E}_{S_{t+1}} \left\| \bar{v}_{t+1} - w^* \right\|^2
\end{aligned}$$

The last equality follows from Lemma 4. Next, by Lemma 5, we have

$$\mathbb{E} \left\| \bar{w}_{t+1} - w^* \right\|^2 \leq \begin{cases} (1 - \eta_t \mu)\mathbb{E} \left\| \bar{w}_t - w^* \right\|^2 + \eta_t B \\ \quad \text{if } t + 1 \bmod K \neq 0 \\ (1 - \eta_t \mu)\mathbb{E} \left\| \bar{w}_t - w^* \right\|^2 + \eta_t (B + D) \\ \quad \text{if } t + 1 \bmod K = 0 \end{cases}$$

Furthermore, the second bound holds for all t since $D > 0$. Next, similar to our proof under full user participation:

- Assume without loss of generality that $\beta_{t \bmod K} \leq 1$ and $\beta_0 = 1$.
- $\eta_t = \alpha_{\lfloor t/K \rfloor} \beta_{t \bmod K}$ where $\alpha_j = \frac{c}{t+d}$ for some $c > \frac{2}{\mu \beta_{K-1}}$ and $d > 2 - \frac{1}{K}$.

Then $\Delta_t \leq \dfrac{v}{(t/K) + d - 2}$ where $v = \max \left\{ \dfrac{c^2(B + D)}{\beta_{K-1} c \mu - 2}, \left[(1/K) + d - 2 \right] \Delta_1 \right\}$.
Hence $\mathbb{E}\left[F(\bar{w}_t) \right] - F^* \leq \dfrac{L}{2} \Delta_t \leq \dfrac{L}{2} \left(\dfrac{v}{(t/K) + d - 2} \right)$.
With $c = \dfrac{3}{\mu \beta_{K-1}}$ and $d = \max \left\{ \dfrac{12L}{\mu \beta_{K-1}}, 4 - \dfrac{2}{K} \right\}$,

$$\mathbb{E}\left[F(\bar{w}_t) - F^* \right] \leq \frac{\kappa}{(t/K) + d - 2} \left(\frac{9(B + D)}{2\mu \beta_{K-1}^2} + \left(\frac{(1/K) + d - 2}{2} \right) \Delta_1 \right)$$

□

B Supplementary Results

B.1 Sensitivity Analysis of FedDecay

FedDecay demonstrates sensitivity to the hyper-parameter β choice; see Table 5. Specifically, in the case of the FEMNIST data set, runs with $\beta = 0.8$ were prematurely halted due to hyperband stopping, indicating inferior performance compared to smaller β choices. Conversely, the remaining β values improved accuracy for new and existing users. Notably, the FEMNIST data set, comprised of handwritten characters contributed by different authors, showcases inherent data heterogeneity. This real-world example underscores the presence of substantial user similarities, owing to the shared language in which all characters are written.

This trend aligns with our assertion in Section 3.2, where we indicated that prioritizing AvgGrad terms (via smaller β values) would be beneficial in cases where users share similarities. Consequently, we observe FedSGD outperforming FedAvg in this context. However, FedSGD, which does not emphasize AvgGradInner

Method	β	FEMNIST (image) Existing $\overline{Acc}$	 New $\overline{Acc}$	SST2 (text) Existing $\overline{Acc}$	 New $\overline{Acc}$	PUBMED (graph) Existing $\overline{Acc}$	 New $\overline{Acc}$
FedSGD	0.0	88.51	90.55	73.60	80.30	86.70	79.14
FedDecay	0.2	**89.86**	**91.52**	67.01	59.02	85.67	79.68
FedDecay	0.4	89.74	90.77	69.42	67.26	**87.22**	**80.39**
FedDecay	0.6	89.76	90.93	**78.15**	**81.01**	86.65	79.86
FedDecay	0.8	Hyperband Stopped		74.64	76.14	86.59	79.32
FedAvg	1.0	87.91	89.78	76.54	76.80	86.29	79.50

Table 5: Performance of FedDecay under misspecified values for decay coefficient, β.

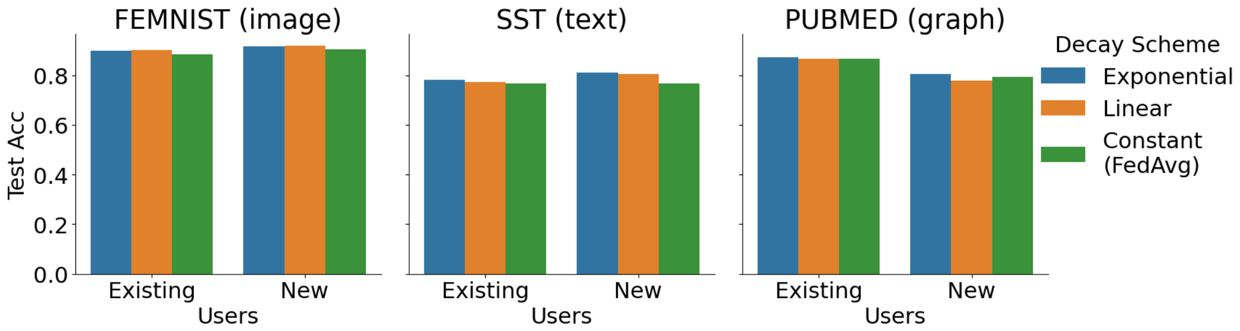

Figure 2: Alternative Without-Round Decay Scheme Performance. Linear decay scheme can also provide improved performance over FedAvg. Within-round decay in general is an option for improving performance without additional computation or memory cost.

(generalization, rapid personalization), is beaten by FedDecay. Here, we observe that having a more flexible balance on initial model success, generalization, and fast personalization results in better performance for various choices of β.

Unlike FEMNIST, heterogeneous users are created from SST2 by partitioning data into 50 clients using Dirichlet allocation with $\alpha = 0.4$. Many extreme non-iid data sets exist, but $\alpha = 0.4$ results in reasonably non-iid users. Revisiting the claim discussed in our Section 3.2, we anticipate that an increased emphasis (achieved via larger β values) on AvgGradInner is required as non-iid user diversity grows. In general, larger values of β perform better on SST2. While FedSGD demonstrates superior test-set performance compared to FedAvg, its validation-set accuracy lags behind FedAvg's, preventing it from securing the best run on SST2 for FedAvg in Section 4.1. More importantly, FedDecay balances AvgGrad and AvgGradInner terms, yielding significantly improved metrics over FedAvg. Notably, FedDecay still has the best overall performance among the three methods. Lastly, in the case of the PUBMED data set, our method's performance seems resilient to the choice of β. However, a straightforward grid search enhances performance for new and existing users over FedAvg and FedSGD.

B.2 Linear Within-Round Learning Rate Decay

In this section, we explore the potential of alternative decay schemes within the FedDecay framework. Results in Section 3.2 allow for more general sequences than exponential decay. While exponential decay has been a focus, we believe other strategies for scaling gradient emphasis during federated training could enhance performance by their balance between initial model success, generalization, and rapid personalization. We expand our analysis to include the use of linear decay for FedDecay. We define linear decay as $\beta_j = \max\{1 - j(1-\beta), 0\}$, where $1 - \beta$ is employed for consistency aligning with FedSGD ($\beta = 0$) and FedAvg ($\beta = 1$). Figure 2 illustrates the average test set accuracy for new and existing users across different decay schemes.

It is evident from the results that both exponential and linear decay schemes yield performance improvements over FedAvg. While linear decay performs slightly worse than FedAvg on the new PUBMED user, it demonstrates robust performance on FEMNIST and SST2. These findings underscore the potential for alternative decay schemes to enhance the overall model performance within the federated learning paradigm. We emphasize that decay can be an effective tool for improving performance without additional cost.

B.3 Robustness of FedDecay to Choice of Seed

We address conducting a series of experiments to ensure the robustness of our findings against the selection of random seeds. Given the constraints of our computational resources, we focus on replicating the less computational FL methods using three distinct fixed seeds. We chose FEMNIST and PUBMED to have one example of full and partial user participation. We present the averaged top-run metrics for new and existing users across the FEMNIST and PUBMED data sets in Table 6 and Table 7. Notably, the FedDecay approach consistently achieves the highest average accuracy among the FL algorithms, catering to new and existing users. Moreover, it demonstrates exceptional performance regarding the best ten-percentile accuracy for existing users. This reinforces the reliability and generalizability of our proposed FedDecay method.

	Existing Users			New Users		
Method	$\overline{Acc}$	$\breve{Acc}$	σ_{Acc}	$\overline{Acc}$	$\breve{Acc}$	σ_{Acc}
FOMAML	0.8971	0.8223	0.0687	0.9044	**0.8371**	0.0578
FedAvg	0.8948	0.8227	0.0701	0.9095	0.8318	0.0563
FedDecay	**0.8993**	**0.8285**	0.0688	**0.9127**	0.8294	0.0545

Table 6: Generalization metrics for new and existing users on the FEMNIST (image) data set. Values reported are the average metric for each single-model-based method's top run across several seeds. FedDecay produces the maximum average test set accuracy on new and existing users.

	Existing Users			New Users		
Method	$\overline{Acc}$	$\breve{Acc}$	σ_{Acc}	$\overline{Acc}$	$\breve{Acc}$	σ_{Acc}
FOMAML	0.8597	0.8234	0.0293	0.7855	-	-
FedAvg	0.8666	0.8313	0.0328	0.7914	-	-
FedDecay	**0.8704**	**0.8420**	0.0244	**0.7950**	-	-

Table 7: Generalization metrics for new and existing users on the PUBMED (graph) data set. Values reported are the average metric for each single-model-based method's top run across several seeds. FedDecay produces the maximum average test set accuracy on new and existing users. With five total users, only a single user is held out to evaluate generalization. Hence, there are no values for the bottom ten percentile ($\breve{Acc}$) or standard deviation (σ_{Acc}) for new users.

B.4 Method Hyper-parameters After Tuning

We provide the hyperparameter configurations for all methods that produce the top average validation set accuracy. These hyperparameter configurations are then used to compute test set metrics for our main experiments.

Method	Local Epochs (K)	Batch Size	Learning Rate (α)	Regularization Rate	Meta-learning Steps	Local Decay (β)
Ditto	3	-	0.10	0.50	-	-
FedBN	3	-	0.01	-	-	-
FedEM	3	-	0.10	-	-	-
pFedMe	3	-	0.50	0.05	3.0	-
FOMAML	3	-	0.50	-	-	-
FedAvg	1	-	0.10	-	-	-
FedDecay - Exponential	3	-	0.05	-	-	0.2
FedDecay - Linear	3	-	0.10	-	-	0.6

Table 8: Main experiment hyper-parameters for all methods after tuning on FEMNIST

Method	Local Epochs (K)	Batch Size	Learning Rate (α)	Regularization Rate	Meta-learning Steps	Local Decay (β)
Ditto	3	16.0	0.05	0.8	-	-
FedBN	3	64.0	0.10	-	-	-
FedEM	3	16.0	0.05	-	-	-
pFedMe	3	64.0	0.05	0.8	3.0	-
FOMAML	3	16.0	0.05	-	-	-
FedAvg	3	16.0	0.05	-	-	-
FedDecay - Exponential	3	16.0	0.05	-	-	0.6
FedDecay - Linear	3	16.0	0.05	-	-	0.4

Table 9: Main experiment hyper-parameters for all methods after tuning on SST

Method	Local Epochs (K)	Batch Size	Learning Rate (α)	Regularization Rate	Meta-learning Steps	Local Decay (β)
Ditto	1	-	0.005	0.1	-	-
FedBN	3	-	0.005	-	-	-
FedEM	3	-	0.500	-	-	-
pFedMe	3	-	0.500	0.5	1.0	-
FOMAML	3	-	0.500	-	-	-
FedAvg	1	-	0.500	-	-	-
FedDecay - Exponential	3	-	0.500	-	-	0.4
FedDecay - Linear	3	-	0.500	-	-	0.4

Table 10: Main experiment hyper-parameters for all methods after tuning on PUBMED