FedSOL: Stabilized Orthogonal Learning with Proximal Restrictions in Federated Learning

Gihun Lee[1], Minchan Jeong[1], Sangmook Kim[2], Jaehoon Oh[3], Se-Young Yun[1]
[1]Graduate School of AI KAIST [3]Samsung Advanced Institute of Technology
[2]Department of Electrical and Computer Engineering, UBC

Abstract

Federated Learning (FL) aggregates locally trained models from individual clients to construct a global model. While FL enables learning a model with data privacy, it often suffers from significant performance degradation when clients have heterogeneous data distributions. This data heterogeneity causes the model to forget the global knowledge acquired from previously sampled clients after being trained on local datasets. Although the introduction of proximal objectives in local updates helps to preserve global knowledge, it can also hinder local learning by interfering with local objectives. To address this problem, we propose a novel method, Federated Stabilized Orthogonal Learning (FedSOL), which adopts an orthogonal learning strategy to balance the two conflicting objectives. FedSOL is designed to identify gradients of local objectives that are inherently orthogonal to directions affecting the proximal objective. Specifically, FedSOL targets parameter regions where learning on the local objective is minimally influenced by proximal weight perturbations. Our experiments demonstrate that FedSOL consistently achieves state-of-the-art performance across various scenarios.

1. Introduction

Federated Learning (FL) is an emerging distributed learning framework that preserves data privacy while leveraging client data for training [29, 30]. In this approach, individual clients train their local models using their private data, and a server aggregates these models into a global model. FL eliminates the need for direct access to clients' raw data, enabling the use of extensive data collected from various sources such as mobile phones, vehicles, and facilities [3, 71]. However, FL encounters a notorious challenge known as data heterogeneity [26, 36, 38]. As data from different clients often come from diverse underlying distributions, local datasets are not independently and identically distributed (Non-IID). This common issue in real-world scenarios leads to a misalignment between global and local objectives [27, 63]. Consequently, local learning deviates from the global objective, resulting in significantly degraded performance and slower convergence [26, 74].

Recent research suggests that such deviation in FL resembles *Catastrophic Forgetting* in Continual Learning (CL) [32, 56, 66]. In CL, fitting the model to a new task alters parameters critical for previous tasks, thus impairing their performance [44, 47, 52, 64]. Similarly in FL, local learning is prone to be overfitted on local datasets, causing the model to forget global knowledge not represented in the local distributions [32, 68]. Consequently, FL must navigate the balance between preserving previous global knowledge and acquiring new local knowledge during local learning. Inspired by CL, we aim to resolve the conflict between two objectives through *Orthogonal Learning* [18, 40]. In CL, an effective strategy for preserving previous knowledge is to minimizing the new task's loss using gradients that are orthogonal to the loss space of the old tasks [9, 10, 55, 75]. These orthogonal gradients enable the model to reduce the loss on the new task without negatively affecting performance on previous tasks. Our primary motivation is to adapt this strategy to FL by identifying gradients that are orthogonal to directions affecting the global knowledge while still enabling effective training on local datasets.

However, applying this strategy to FL presents unique challenges. First, implementing orthogonal gradients in CL often requires retaining past data or gradients for reference [9, 55]. This practice becomes problematic in FL as it can compromise data privacy and introduce additional communication overhead. Second, unlike in CL where task distributions are often disjoint [13, 72], FL clients may have overlapping data distributions, including instances from the same class. As a result, the global distribution, formed by combining local distributions, also presents overlap with individual local distributions. Such an overlap not only complicates the identification of an orthogonal update direction but also makes the process computationally demanding. Moreover, finding an orthogonal gradient that accommodates multiple local distributions is challenging and the obtained gradient may significantly undermine the effectiveness of learning on local datasets.

To address these problems, we initially focus on *proximal restrictions* in FL [27, 35, 37]. Many previous studies have integrated these restrictions into local objectives to tackle the data heterogeneity issue. By constraining the deviation of local learning from the global objective, the proximal restrictions maintain performance on the global distribution outside of local distributions. This leads us to interpret proximal objectives as losses that preserve global knowledge in FL, notably without requiring direct access to other clients' data or the need to communicate their information. Unfortunately, as the proximal objective is closely tied to the local objective, we find that directly negating the projected proximal gradient component in the local gradient substantially degrades the performance.

Instead, we hypothesize that identifying the parameter region where local gradients naturally align as orthogonal to the proximal gradients can effectively reduce interference between the two objectives. To this end, we propose a novel algorithm, Federated Stabilized Orthogonal Learning (FedSOL). At each local update, FedSOL adversarially perturbs weight parameters using the gradient of proximal objectives, and then captures the local gradient at these perturbed weights. FedSOL aims to find a parameter region where the local gradient is stable against proximal perturbation, ensuring that the resulting local gradient remains orthogonal to the proximal gradient.

To summarize, our main contributions are as follows:

- We suggest that orthogonal learning in CL could be an effective strategy in FL, by resolving conflicts between local and proximal objectives. (**Section 2**)
- We propose a novel FL method, FedSOL, which targets a parameter region where the local gradient is orthogonal to the proximal gradient. (**Section 3**)
- We validate the efficacy of FedSOL in various setups, demonstrating consistent state-of-the-art performance. We also highlight its robustness when integrated with different types of proximal objectives. (**Section 4**)
- We provide a comprehensive analysis of the benefits that FedSOL offers to FL, effectively preserving global knowledge during local learning and enhancing the smoothness of the global model. (**Section 5**)

2. Proximal Restriction in Local Learning

In this section, we first introduce the problem setup and the concept of proximal restriction in FL. We then discuss the trade-off between local and proximal objectives. We suggest that while orthogonal learning could be an effective solution, simple gradient projection cannot achieve it in FL.

2.1. Proximal Restriction in FL

Consider an FL system that consists of K clients and a central server. Each client k has a local dataset $\mathcal{D}^k$, where the entire dataset is a union of the local datasets as $\mathcal{D} = \bigcup_{k\in[K]}\mathcal{D}^k$. FL aims to train a global server model with weights $\boldsymbol{w}$ that minimize the loss across all clients:

$$\mathcal{L}_{\text{global}}(\boldsymbol{w}) = \sum_{k\in[K]} \frac{|\mathcal{D}_k|}{|\mathcal{D}|}\, \mathcal{L}^k_{\text{local}}(\boldsymbol{w})\,, \tag{1}$$

where $|\mathcal{D}^k|$ and $|\mathcal{D}|$ are the number of instances in each dataset. When using a proximal restriction objective, the loss function for each client k is a linear combination of its original local loss, $\mathcal{L}^k_{\text{local}}(\boldsymbol{w}_k)$, and a proximal loss, $\mathcal{L}^k_p(\boldsymbol{w}_k;\boldsymbol{w}_g)$, controlled by a hyperparameter β:

$$\mathcal{L}^k(\boldsymbol{w}_k) = \mathcal{L}^k_{\text{local}}(\boldsymbol{w}_k) + \beta\cdot\mathcal{L}^k_p(\boldsymbol{w}_k;\boldsymbol{w}_g). \tag{2}$$

Here, $\mathcal{L}^k_{\text{local}}(\boldsymbol{w}_k)$ is the loss on the client's local distribution (e.g., cross-entropy loss), and $\mathcal{L}^k_p(\boldsymbol{w}_k;\boldsymbol{w}_g)$ quantifies the discrepancy between the global model $\boldsymbol{w}_g$ and the local model $\boldsymbol{w}_k$. This discrepancy can be measured in various ways, such as the Euclidean distance between the parameters [37, 63] or the KL-divergence between probability vectors computed using the client's data [24, 32].

2.2. Forgetting in Local Learning

Recent studies suggest that data heterogeneity in FL leads to *Catastrophic Forgetting* during local learning [32, 56]. When the model is trained on skewed local datasets, local learning deviates from the global objectives—commonly referred to as client drift [27]. This drift causes trained local models to forget knowledge from the previous round of the global model, which the local datasets cannot fully represent. The FL performance is strongly tied to how well local learning preserves this global knowledge [66, 68]. Introducing proximal restrictions within the local objective effectively constrains such deviation, alleviating forgetting [32].

However, these proximal restrictions present a trade-off. While they serve to preserve global knowledge, they also inherently limit the model's ability to learn from local data [46, 54]. Striking the right balance between these two conflicting objectives during local learning is crucial for the success of FL. As the local model $\boldsymbol{w}_k$ begins with the same parameters as the distributed global model $\boldsymbol{w}_g$, it initially has a minimal proximal loss. Thereby, we posit that the main challenge is guiding the local learning to reduce the local loss $\mathcal{L}^k_{\text{local}}$ without inducing an increase in the proximal loss $\mathcal{L}^k_p$. Inspired by CL [9, 18, 55], we consider updating the local model using gradients that are orthogonal to the proximal gradient as an effective solution to this problem.

2.3. Proximal Gradient Projection

A straightforward approach to obtaining the update gradient, which is orthogonal to the proximal loss $\mathcal{L}^k_p$, is conducting a direct projection [9, 55] of the proximal gradient:

$$\boldsymbol{g}^{\text{Proj}}_u = \boldsymbol{g}_l - \frac{\boldsymbol{g}_l{}^T\,\boldsymbol{g}_p}{\boldsymbol{g}_p^T\,\boldsymbol{g}_p}\boldsymbol{g}_p \qquad \text{if} \quad \boldsymbol{g}_l^T\,\boldsymbol{g}_p < 0. \tag{3}$$

Here, we denote the local gradient as $\boldsymbol{g}_l = \nabla_{\boldsymbol{w}_k} \mathcal{L}^k_{\text{local}}(\boldsymbol{w}_k)$ and the proximal gradient as $\boldsymbol{g}_p = \nabla_{\boldsymbol{w}_k} \mathcal{L}^k_p(\boldsymbol{w}_k; \boldsymbol{w}_g)$. We omit the weights $\boldsymbol{w}$ for simplicity unless clarification is needed. By negating the conflicting component from the local gradient $\boldsymbol{g}_l$, the update gradient $\boldsymbol{g}_u^{\text{Proj}}$ becomes orthogonal to the proximal gradient $\boldsymbol{g}_p$. Note that we only project when the two gradients are in conflict (i.e., $\boldsymbol{g}_l^T \ \boldsymbol{g}_p < 0$), otherwise we use the original local gradient $\boldsymbol{g}_l$.

However, we find that this direct projection approach rather degrades the performance. In Table 1, we compare different usages of the proximal objective: as an auxiliary loss alongside the local objective, as in Equation 2 (Base), and as proximal gradient projection, as in Equation 3 (Projection). Note that the absence of a proximal loss (None) is equivalent to FedAvg [45]. The results show that performance significantly declines when the proximal gradient component is directly negated through gradient projection, even underperforming compared to FedAvg. This suggests that the two objectives are closely interconnected, and directly negating the proximal gradient might actually undermine the local learning. Therefore, we consider an approach that implicitly promotes the orthogonality of the update.

Table 1. Results of gradient projection on CIFAR-10 ($\alpha = 0.1$).

Proximal Loss	Usage: Base	Usage: Projection
None (FedAvg)	$56.13_{\pm 0.78}$	
L2 Distance	**59.80**$_{\pm 1.12}$	$56.35_{\pm 2.85}$ (- 3.45)
KL-Divergence	**60.31**$_{\pm 2.07}$	$50.88_{\pm 3.55}$ (- 9.43)

3. Proposed Method: FedSOL

In this section, we introduce Federated Stabilized Orthogonal Learning (FedSOL). Our primary motivation is to obtain a gradient for updating the local model that is orthogonal to the proximal gradient, yet still effectively reduces the local loss. The detailed algorithm is outlined in Algorithm 1.

3.1. Preliminary: Overview of SAM

We borrow the idea of recently proposed Sharpness-Aware Minimization (SAM) [19], which uses weight perturbations to achieve flatter minima. For the given loss $\mathcal{L}$, SAM optimizer solves the following min-max problem:

$$\min_{\boldsymbol{w}} \max_{\|\boldsymbol{\epsilon}\|_2 < \rho} \mathcal{L}(\boldsymbol{w} + \boldsymbol{\epsilon}). \tag{4}$$

In the above equation, the inner maximization identifies a parameter perturbation ϵ that maximizes loss change within the ρ-ball neighborhood. This is practically approximated by a single re-scaled gradient step $\boldsymbol{\epsilon}^* = \rho \nabla_{\boldsymbol{w}} \mathcal{L}(\boldsymbol{w}) / \|\nabla_{\boldsymbol{w}} \mathcal{L}(\boldsymbol{w})\|_2$. The outer minimization is

Algorithm 1 **Fed**erated **S**tabilized **O**rthogonal **L**earning

Input: local loss $\mathcal{L}^k_{\text{local}}$ and proximal loss $\mathcal{L}^k_p$ for each client $k \in [K]$, learning rate γ, and base perturbation strength ρ
Initialize global server weight $\boldsymbol{w}_g$
for each communication round t **do**
  Server samples clients $K^{(t)} \subset [K]$
  Server broadcasts $\boldsymbol{w}_g$ for all $k \in K^{(t)}$
  Client replaces $\boldsymbol{w}_k \leftarrow \boldsymbol{w}_g$
  for each client $k \in K^{(t)}$ **in parallel do**
    for each local step **do**
      # Set Adaptive Perturbation Radius (Sec 3.3)
      $\boldsymbol{\rho}_{\text{adaptive}} = \rho \cdot \boldsymbol{\Lambda}$ *(element-wise rescale)*
      # Perturb using Proximal Gradient (Sec 3.2)
      # (Optional) Use Partial Perturbation (Sec 3.3)
      $\boldsymbol{\epsilon}^*_p = \boldsymbol{\rho}_{\text{adaptive}} \odot \frac{\nabla_{\boldsymbol{w}_k} \mathcal{L}^k_p(\boldsymbol{w}_k; \boldsymbol{w}_g)}{\|\nabla_{\boldsymbol{w}_k} \mathcal{L}^k_p(\boldsymbol{w}_k; \boldsymbol{w}_g)\|}$
      # Update Local Model Parameters (Sec 3.2)
      $\boldsymbol{w}_k \leftarrow \boldsymbol{w}_k - \gamma \cdot \nabla_{\boldsymbol{w}_k} \mathcal{L}^k_{\text{local}}(\boldsymbol{w}_k + \boldsymbol{\epsilon}^*_p)$
    end for
  end for
  Upload $\boldsymbol{w}_k$ to server
  Server Aggregation : $\boldsymbol{w}_g \leftarrow \frac{1}{|K^{(t)}|} \sum_{k \in K^{(t)}} \boldsymbol{w}_k$
end for
Server output : $\boldsymbol{w}_g$

then conducted by a base optimizer, such as SGD [49], taking the gradient $\nabla_{\boldsymbol{w}} \mathcal{L}(\boldsymbol{w} + \boldsymbol{\epsilon}^*)$ at the perturbed weights. SAM demonstrates an exceptional ability to perform well across different model structures [11, 76] and tasks [1, 65] with high generalization performance. In FL, applying SAM improves the generalization of each client's local model [6, 54]. However, since this approach only addresses local objectives, its effectiveness in generalization is mostly confined to local data distributions [59, 60] and still encounter conflicts with the proximal objective.

3.2. Adversarial Proximal Perturbation

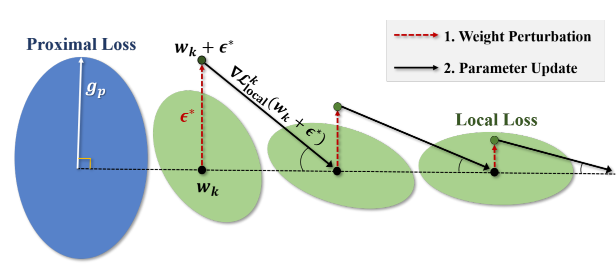

Figure 1. An overview of the FedSOL update. At each update, FedSOL computes its update gradient at the proximally perturbed weights. By withstanding the proximal perturbation, FedSOL obtains a local gradient that is orthogonal to the proximal gradient.

The core idea of FedSOL is to identify the parameter region where the local gradient is minimally affected by adversarial weight perturbation using the proximal gradient. At each local update, FedSOL adversarially perturbs the weight parameters and obtains the local gradient as follows:

Step1: Weight Perturbation In FedSOL, we first identify a weight perturbation $\boldsymbol{\epsilon}_p^*$ that brings about the most significant change in the given proximal loss $\mathcal{L}_p^k$. Practically, this is conducted by using a single re-scaled step based on the proximal gradient $\boldsymbol{g}_p = \nabla_{\boldsymbol{w}_k}\mathcal{L}_p^k(\boldsymbol{w}_k; \boldsymbol{w}_g)$, controlled by the hyperparameter ρ for perturbation strength:

$$\boldsymbol{\epsilon}_p^* = \underset{\|\boldsymbol{\epsilon}\|_2 \le \rho}{\text{argmax}}\ \mathcal{L}_p^k(\boldsymbol{w}_k + \boldsymbol{\epsilon}; \boldsymbol{w}_g) \ \approx\ \rho\,\frac{\boldsymbol{g}_p}{\|\boldsymbol{g}_p\|_2}\,. \tag{5}$$

Step2: Parameter Update After the perturbation, update the parameters with the local gradient computed at the perturbed weights and a learning rate γ:

$$\boldsymbol{w}_k \leftarrow \boldsymbol{w}_k - \gamma \cdot \nabla_{\boldsymbol{w}_k}\mathcal{L}_{\text{local}}^k(\boldsymbol{w}_k + \boldsymbol{\epsilon}_p^*)\,, \tag{6}$$

In the above procedures, the model is updated using the gradient of the local loss $\mathcal{L}_{\text{local}}^k$, while the proximal loss $\mathcal{L}_p^k$ plays only an implicit role in perturbing weights. We emphasize that unlike SAM, which employs the same loss for both weight perturbations and parameter updates, FedSOL distinguishes between the roles of these two types of losses to address the knowledge trade-off in FL. A more detailed comparison of these two distinct approaches in the FL context is discussed in Appendix G.

3.3. Adaptive Perturbation Strength

In FedSOL, we introduce an adaptive perturbation strength reflecting the global and local parameter discrepancies. For each layer m, we construct a scaling vector $\boldsymbol{\lambda}^{(m)}$, where the i-th entry corresponds to each parameter in that layer:

$$\boldsymbol{\lambda}^{(m)}[i] = \frac{\left|\boldsymbol{w}_k^{(m)}[i] - \boldsymbol{w}_g^{(m)}[i]\right|}{\left\|\boldsymbol{w}_k^{(m)} - \boldsymbol{w}_g^{(m)}\right\|_2}. \tag{7}$$

Here, $\boldsymbol{w}_g^{(m)}$ and $\boldsymbol{w}_k^{(m)}$ represent the weights of the m-th layer in the global and local models, respectively. The denominator represents the normalization of the discrepancy within the layer, accounting for the layer-wise scale variance. This adaptive perturbation allows more perturbation for the parameter with large difference, and vice versa. It fits with the typical behavior of proximal loss, which increases as $\|\boldsymbol{w}_k - \boldsymbol{w}_g\|_2$ grows. By concatenating these layer-specific vectors, $\boldsymbol{\Lambda} = (\boldsymbol{\lambda}^{(1)}, \dots, \boldsymbol{\lambda}^{(m)}, \dots, \boldsymbol{\lambda}^{(\text{last})})$, and incorporating it into the Equation 5, the proximal perturbation $\boldsymbol{\epsilon}_p^*$ becomes:

$$\boldsymbol{\epsilon}_p^* = \rho \cdot \boldsymbol{\Lambda} \odot \frac{\boldsymbol{g}_p}{\|\boldsymbol{g}_p\|_2} \approx \underset{\|\boldsymbol{\Lambda}^{-1} \odot \boldsymbol{\epsilon}\|_2 \le \rho}{\text{argmax}}\ \mathcal{L}_p^k(\boldsymbol{w}_k + \boldsymbol{\epsilon}; \boldsymbol{w}_g)\,, \tag{8}$$

where $\odot$ denotes the element-wise product. Intuitively, the adaptive perturbation allows local learning to deviate certain parameters from the global model when they are significantly influential to justify larger weight perturbations. Note that using a fixed value for ρ corresponds to setting $\boldsymbol{\Lambda}$ in Equation 8 as a vector with all entries to one.

3.4. Partial Perturbation

As the data heterogeneity does not affect all layers equally [39, 42], we explore the use of *partial* perturbation in FedSOL by selectively perturbing specific layers instead of the entire model. We propose that perturbing only the last classifier layer is empirically sufficient for FedSOL. We provide a detailed discussion in Section 4.4. This approach yields performance nearly as high as full-model perturbation while significantly reducing computational requirements by avoiding multiple forward and backward computations across all layers. For optimal efficiency and performance, we apply FedSOL's weight perturbation exclusively to the last classifier head layer in the experiments unless otherwise specified.

3.5. Theoretical Analysis

To assess FedSOL's effect on local learning, we examine the changes in both $\mathcal{L}_{\text{p}}^k$ and the local loss $\mathcal{L}_{\text{local}}^k$ after a single FedSOL update. Note that the proximal loss $\mathcal{L}_{\text{p}}^k$ and local loss $\mathcal{L}_{\text{local}}^k$ each corresponds to the global knowledge and local knowledge. Specifically, we examine the impact of the FedSOL update gradient $\boldsymbol{g}_u^{\text{FedSOL}}$ on loss $\mathcal{L}^k$ at $\boldsymbol{w}_k$ with a learning rate γ, using a first-order Taylor approximation:

$$\begin{aligned}\Delta^{\text{FedSOL}}\mathcal{L}^k(\boldsymbol{w}_k) &= \mathcal{L}^k\big(\boldsymbol{w}_k - \gamma\,\boldsymbol{g}_u^{\text{FedSOL}}(\boldsymbol{w}_k)\big) - \mathcal{L}^k(\boldsymbol{w}_k)\\ &\approx -\gamma\langle \nabla_{\boldsymbol{w}_k}\mathcal{L}^k(\boldsymbol{w}_k),\ \boldsymbol{g}_u^{\text{FedSOL}}(\boldsymbol{w}_k)\rangle\,.\end{aligned} \tag{9}$$

In the above equation, $\mathcal{L}^k$ can be either local loss or proximal loss. We further scrutinize $\boldsymbol{g}_u^{\text{FedSOL}}$ itself as a first-order Taylor expansion of the local loss $\mathcal{L}_{\text{local}}^k$ at the perturbed weights $\boldsymbol{w}_k + \boldsymbol{\epsilon}_p^*$ as follows:

$$\begin{aligned}\boldsymbol{g}_u^{\text{FedSOL}}(\boldsymbol{w}_k) &= \nabla_{\boldsymbol{w}_k}\mathcal{L}_{\text{local}}^k(\boldsymbol{w}_k + \boldsymbol{\epsilon}_p^*)\\ &\approx \boldsymbol{g}_l(\boldsymbol{w}_k) + \rho\,\nabla^2_{\boldsymbol{w}_k}\mathcal{L}_{\text{local}}^k(\boldsymbol{w}_k)\hat{\boldsymbol{g}}_p(\boldsymbol{w}_k)\,.\end{aligned} \tag{10}$$

where $\hat{\boldsymbol{g}}_p = \boldsymbol{g}_p / \|\boldsymbol{g}_p\|_2$ represents the normalized proximal gradient. Note that $\boldsymbol{\epsilon}_p^*$ is solely used for weight perturbation, and hence its gradient is not computed. By integrating the approximation in Equation 10 into the loss difference defined in Equation 9, we derive the subsequent two key propositions. These propositions explain how FedSOL guides local learning to minimize the local loss, $\mathcal{L}_{\text{local}}^k$, without causing an increase in the proximal loss, $\mathcal{L}_{\text{p}}^k$.

Proposition 1 (Proximal Objective Orthogonality). *Given a local loss* $\mathcal{L}^k_{\text{local}}$ *and its Hessian matrix* $\nabla^2\mathcal{L}^k_{\text{local}} \succcurlyeq 0$ *evaluated at* $\boldsymbol{w}_k$*, the change of proximal loss by FedSOL update reduces the conflicts* $\langle \boldsymbol{g}_l, \boldsymbol{g}_p \rangle \leq 0$ *in FedAvg update* $\Delta^{\text{FedAvg}}\mathcal{L}^k_p = -\gamma \langle \boldsymbol{g}_l, \boldsymbol{g}_p \rangle$ *as* ρ *increases:*

$$\Delta^{\text{FedSOL}}\mathcal{L}^k_p \approx -\gamma\Big(\langle \boldsymbol{g}_l, \boldsymbol{g}_p \rangle + \rho \cdot \underbrace{\hat{\boldsymbol{g}}_p^\top \nabla^2\mathcal{L}^k_{\text{local}}\, \boldsymbol{g}_p}_{\geq 0}\Big). \quad (11)$$

Proposition 2 (Local Objective Equivalence). *The change of local loss* $\mathcal{L}^k_{\text{local}}$ *by FedSOL update is equivalent to the FedAvg update conducted at* $\nabla\mathcal{L}^k_{\text{local}}(\boldsymbol{w}_k + \frac{\rho}{2}\boldsymbol{\epsilon}^*_p)$ *as:*

$$\Delta^{\text{FedSOL}}\mathcal{L}^k_{\text{local}}(\boldsymbol{w}_k) \approx \Delta^{\text{FedAvg}}\mathcal{L}^k_{\text{local}}\Big(\boldsymbol{w}_k + \frac{\rho}{2}\boldsymbol{\epsilon}^*_p\Big). \quad (12)$$

Firstly, **Proposition 1** examines FedSOL's impact on the proximal loss. It suggests that FedSOL's update gradient, $\boldsymbol{g}^{\text{FedSOL}}_u$, directs the local updates to be orthogonal to the proximal gradient. This helps maintain a low proximal loss, $\mathcal{L}^k_{\text{p}}$, during local learning, which initially has a very low value as the learning starts from the distributed global model. This indicates that FedSOL implicitly regularizes the negative impact of the local gradient on proximal loss. This regularization effect grows as the curvature of local loss $\nabla^2\mathcal{L}^k_{\text{local}}$ local becomes steeper.

Meanwhile, **Proposition 2** compares the change of local loss $\mathcal{L}^k_{\text{local}}$ under FedSOL to its counterpart in FedAvg. This proposition suggests that, although FedSOL calculates the local gradient at perturbed weights, its impact on the local loss is almost identical to that of FedAvg. This implies that FedSOL effectively reduces local loss without significantly slowing down the learning process. As a result, FedSOL successfully mitigates the conflict between the proximal objective and the local objective. The detailed proofs of the propositions are provided in Appendix K.

4. Experiment

4.1. Experimental Setups

Data Setups We use 6 datasets: MNIST [15], CIFAR-10 [31], SVHN [50], CINIC-10 [12], PathMNIST [70], and TissueMNIST [70]. We distribute data to clients using two strategies: Sharding [45, 51] and Latent Dirichlet Allocation (LDA) [32, 62]. Sharding sorts data by label and assigns equal-size shards to clients. The heterogeneity level increases as the shard per user, s, becomes smaller. On the other hand, LDA assigns class c data samples to each client k with probability $p_c(\approx \text{Dir}(\alpha))$, where the heterogeneity increases as α becomes smaller. Although only the statistical distributions varies across the clients in Sharding strategy, both the distribution and dataset size differ in LDA.

Learning Setups We distribute MNIST, CIFAR-10, and SVHN datasets across 100 clients with a sampling ratio of 0.1, while CINIC-10, PathMNIST, and TissueMNIST across 200 clients with a ratio of 0.05. We use a model architecture as described in [45], which consists of two convolutional layers, max-pooling layers, and two fully connected layers. Each client optimizes its local datasets for 5 local epochs using momentum SGD with a learning rate of 0.01, momentum 0.9, and weight decay 1e-5. The learning rate is multiplied by a factor of 0.99 after each communication round. We conducted a total of 300 communication rounds, except for MNIST, PathMNIST, and TissueMNIST, for which we conducted 200 rounds, sufficient for the server model to reach performance saturation. For the proximal loss, we employed the KL-divergence loss function. We provide more detailed experimental setups in Appendix B.

4.2. Proximal Orthogonality of FedSOL

In Figure 2, we examine the interaction of FedSOL's update gradient, g^{FedSOL}_u, with the proximal loss $\mathcal{L}_p$. As the perturbation strength ρ increases, the direction of g^{FedSOL}_u becomes increasingly orthogonal to the proximal gradient g_p (Figure 2(a)). This enhanced orthogonality helps maintain a low proximal loss during local learning (Figure 2(b)).

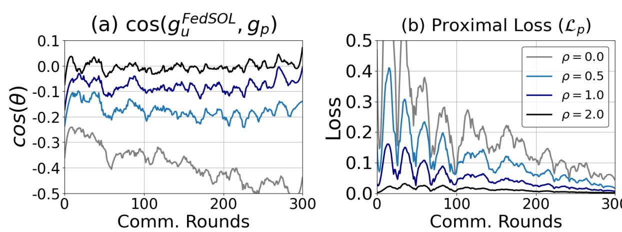

Figure 2. Effect of FedSOL on local learning in CIFAR-10 (α=0.1) by varying ρ values. (a) Average proximal loss across local models. (b) Cosine similarity between FedSOL gradient (g^{FedSOL}_u) and proximal gradient (g_p) during local learning.

4.3. Performance on Data Heterogeneity

Heterogeneity Level Table 2 presents a comparison between FedSOL and other baselines. The results show that many recently proposed FL methods tend to underperform even when compared to the standard FedAvg baseline. A similar observation is also reported in [32, 74], which points out that many FL methods are sensitive to the learning scenarios. In contrast, FedSOL achieves state-of-the-art results in most cases, consistently outperforming FedAvg across all evaluated scenarios. Particularly, FedSOL shows remarkable improvement at high heterogeneity levels, such as in Sharding (s=2) and LDA (α=0.05) scenarios. We further provide the learning curves in Appendix C, perturbation strategies in Appendix H, personalized performance in Appendix D, and larger dataset experiments in Appendix E.

Table 2. Test accuracy@1(%) comparison among baselines and FedSoL on different datasets. The values in the parenthesis are the standard deviation. The arrow (↓, ↑) shows the comparison to the FedAvg. We set $s \in \{2, 3, 5, 10\}$ and $\alpha \in \{0.05, 0.1, 0.3, 0.5\}$ for CIFAR-10 datasets, whereas $s = 2$ and $\alpha = 0.1$ for the others.

Non-IID Partition Strategy : Sharding

Method	MNIST	CIFAR-10 $s = 2$	CIFAR-10 $s = 3$	CIFAR-10 $s = 5$	CIFAR-10 $s = 10$	SVHN	CINIC-10	PathMNIST	TissueMNIST
FedAvg [45]	96.16$_{(0.19)}$	51.48$_{(3.41)}$	62.94$_{(0.00)}$	70.96$_{(0.91)}$	74.60$_{(0.88)}$	73.63$_{(3.16)}$	42.40$_{(2.70)}$	57.40$_{(1.48)}$	49.36$_{(1.64)}$
FedProx [37]	95.86$_{(0.12)}$ ↓	52.80$_{(2.66)}$ ↑	58.19$_{(0.55)}$ ↓	64.71$_{(0.74)}$ ↓	69.37$_{(1.21)}$ ↓	71.09$_{(3.13)}$ ↓	40.00$_{(3.01)}$ ↓	60.77$_{(3.64)}$ ↑	48.20$_{(1.95)}$ ↓
FedNova [63]	94.13$_{(0.36)}$ ↓	46.89$_{(2.57)}$ ↓	61.12$_{(0.88)}$ ↓	67.11$_{(0.25)}$ ↓	70.59$_{(0.52)}$ ↓	67.35$_{(2.84)}$ ↓	40.94$_{(2.29)}$ ↓	58.85$_{(4.10)}$ ↑	36.44$_{(0.95)}$ ↓
Scaffold [27]	95.91$_{(0.18)}$ ↓	62.60$_{(0.70)}$ ↑	68.53$_{(0.99)}$ ↑	74.28$_{(0.39)}$ ↑	76.71$_{(0.16)}$ ↑	77.84$_{(2.28)}$ ↑	47.76$_{(0.45)}$ ↑	71.12$_{(1.04)}$ ↑	30.99$_{(6.09)}$ ↓
FedNTD [32]	96.62$_{(0.06)}$ ↑	**67.25**$_{(1.08)}$ ↑	**70.47**$_{(0.33)}$ ↑	75.21$_{(0.39)}$ ↑	76.46$_{(0.07)}$ ↑	**85.30**$_{(0.78)}$ ↑	52.72$_{(1.12)}$ ↑	65.00$_{(1.26)}$ ↑	52.63$_{(0.59)}$ ↑
FedSAM [54]	96.12$_{(0.19)}$ ↓	51.85$_{(3.14)}$ ↑	60.90$_{(0.93)}$ ↓	69.29$_{(0.39)}$ ↓	72.98$_{(0.34)}$ ↓	65.85$_{(3.77)}$ ↓	45.91$_{(2.02)}$ ↑	67.32$_{(3.15)}$ ↑	49.62$_{(1.61)}$ ↑
FedASAM [6]	**97.08**$_{(0.15)}$ ↑	52.08$_{(2.19)}$ ↑	63.24$_{(1.16)}$ ↑	70.95$_{(0.76)}$ ↓	74.74$_{(0.88)}$ ↑	79.48$_{(2.17)}$ ↓	43.15$_{(2.73)}$ ↑	59.47$_{(2.91)}$ ↑	49.46$_{(1.91)}$ ↑
FedSOL (Ours)	**97.15**$_{(0.08)}$ ↑	**66.72**$_{(0.61)}$ ↑	**69.88**$_{(0.15)}$ ↑	**75.82**$_{(0.34)}$ ↑	**77.79**$_{(0.19)}$ ↑	**85.18**$_{(0.37)}$ ↑	**55.17**$_{(0.32)}$ ↑	**73.85**$_{(1.55)}$ ↑	**53.42**$_{(0.46)}$ ↑

Non-IID Partition Strategy : LDA

Method	MNIST	CIFAR-10 $\alpha = 0.05$	CIFAR-10 $\alpha = 0.1$	CIFAR-10 $\alpha = 0.3$	CIFAR-10 $\alpha = 0.5$	SVHN	CINIC-10	PathMNIST	TissueMNIST
FedAvg [45]	96.11$_{(0.19)}$	42.27$_{(1.34)}$	56.13$_{(0.78)}$	67.32$_{(0.94)}$	73.90$_{(0.66)}$	55.36$_{(4.85)}$	36.49$_{(4.37)}$	65.98$_{(4.76)}$	42.78$_{(2.03)}$
FedProx [37]	96.05$_{(0.13)}$ ↓	50.58$_{(0.57)}$ ↑	59.80$_{(1.12)}$ ↑	68.39$_{(0.81)}$ ↑	72.87$_{(0.55)}$ ↑	72.40$_{(3.15)}$ ↑	40.09$_{(3.97)}$ ↑	70.44$_{(1.92)}$ ↑	52.25$_{(1.40)}$ ↑
FedNova [63]	88.24$_{(1.37)}$ ↓	10.00$_{(\textit{Failed})}$ ↓	10.00$_{(\textit{Failed})}$ ↓	64.67$_{(0.77)}$ ↓	70.04$_{(0.45)}$ ↓	53.07$_{(3.30)}$ ↓	21.89$_{(1.71)}$ ↓	38.94$_{(2.34)}$ ↓	15.03$_{(3.74)}$ ↓
Scaffold [27]	94.18$_{(0.32)}$ ↓	10.00$_{(\textit{Failed})}$ ↓	10.00$_{(\textit{Failed})}$ ↓	**71.92**$_{(0.17)}$ ↑	**75.49**$_{(0.21)}$ ↑	21.46$_{(1.75)}$ ↓	16.89$_{(2.25)}$ ↓	18.07$_{(0.04)}$ ↓	32.04$_{(0.07)}$ ↓
FedNTD [32]	96.97$_{(0.27)}$ ↑	58.08$_{(0.48)}$ ↑	**63.16**$_{(1.02)}$ ↑	71.56$_{(0.26)}$ ↑	74.91$_{(0.33)}$ ↑	79.25$_{(0.61)}$ ↑	50.22$_{(3.71)}$ ↑	74.26$_{(1.25)}$ ↑	44.55$_{(1.95)}$ ↑
FedSAM [54]	95.72$_{(0.43)}$ ↓	36.14$_{(1.21)}$ ↓	52.14$_{(0.94)}$ ↓	64.83$_{(0.56)}$ ↓	70.74$_{(0.40)}$ ↓	13.27$_{(2.78)}$ ↓	36.70$_{(4.28)}$ ↑	66.64$_{(3.76)}$ ↑	44.07$_{(3.02)}$ ↑
FedASAM [6]	96.60$_{(0.10)}$ ↑	43.12$_{(1.25)}$ ↑	57.00$_{(0.30)}$ ↑	67.45$_{(0.92)}$ ↑	73.91$_{(0.51)}$ ↑	60.25$_{(4.56)}$ ↑	36.93$_{(4.60)}$ ↑	69.45$_{(3.19)}$ ↑	42.73$_{(2.35)}$ ↑
FedSOL (Ours)	**97.44**$_{(0.11)}$ ↑	**60.01**$_{(0.30)}$ ↑	**64.13**$_{(0.46)}$ ↑	**71.94**$_{(0.57)}$ ↑	**75.60**$_{(0.32)}$ ↑	**83.92**$_{(0.29)}$ ↑	**55.07**$_{(1.48)}$ ↑	**78.88**$_{(0.46)}$ ↑	**53.40**$_{(0.85)}$ ↑

Learning Factors We examine the learning factors that influence FedSOL's performance in Figure 3. In our experiments, FedSOL consistently surpasses FedAvg across various factors, achieving its best performance within the ρ range between 0.5 and 2.0. Most notably, FedSOL's gains increase as a smaller portion of clients participate in each round. For example, FedAvg's performance significantly declines at a sampling ratio of 0.02, falling to near-random accuracy. However, FedSOL remains robust under such conditions. Further comparisons are in Appendix J.

Model Architecture We conduct further experiments using different model architectures: VggNet-11 [57], ResNet-18 [21], and SL-ViT [33]. The results in Table 3 validate the efficacy of FedSOL across various model architectures.

Table 3. Comparison of methods on different model architectures. The heterogeneity is set as LDA ($\alpha = 0.1$).

Model	Method	CIFAR-10	SVNH	PathMNIST
Vgg11	FedAvg	41.30$_{\pm 1.07}$	50.02$_{\pm 4.25}$	61.79$_{\pm 9.88}$
	FedProx	40.45$_{\pm 1.41}$	31.07$_{\pm 6.72}$	63.47$_{\pm 2.68}$
	FedNTD	**60.55**$_{\pm 2.14}$	56.62$_{\pm 2.64}$	69.82$_{\pm 2.27}$
	FedSOL	56.39$_{\pm 1.40}$	**74.74**$_{\pm 0.04}$	**78.38**$_{\pm 1.12}$
Res18	FedAvg	49.92$_{\pm 0.62}$	76.98$_{\pm 2.90}$	57.91$_{\pm 1.27}$
	FedProx	59.00$_{\pm 2.58}$	82.09$_{\pm 2.35}$	75.84$_{\pm 1.58}$
	FedNTD	57.79$_{\pm 3.42}$	78.50$_{\pm 0.18}$	76.87$_{\pm 0.57}$
	FedSOL	**66.32**$_{\pm 0.48}$	**85.97**$_{\pm 0.04}$	**80.59**$_{\pm 0.11}$
SL-ViT	FedAvg	35.48$_{\pm 2.09}$	53.94$_{\pm 5.17}$	72.44$_{\pm 1.91}$
	FedProx	38.73$_{\pm 1.23}$	58.25$_{\pm 4.23}$	74.10$_{\pm 1.23}$
	FedNTD	47.59$_{\pm 2.84}$	61.46$_{\pm 1.76}$	71.65$_{\pm 1.71}$
	FedSOL	**47.95**$_{\pm 1.51}$	**67.19**$_{\pm 0.33}$	**77.96**$_{\pm 0.47}$

Proximal Losses We utilize KL-divergence as the proximal loss in our primary experiments. However, FedSOL is compatible with various other different proximal objectives. Table 4 demonstrates the impact of integrating FedSOL with other proximal objectives: FedProx [37], FedNova [63], Scaffold [27], FedDyn [2], and Moon [35]. These methods are compared in two distinct scenarios: as an auxiliary objective alongside the original local objective following Equation 2 (Base), and as proximal perturbation within FedSOL (Combined). The perturbation is applied to the entire model, not just the head, to assess the overall effect of each proximal objective within FedSOL. The results

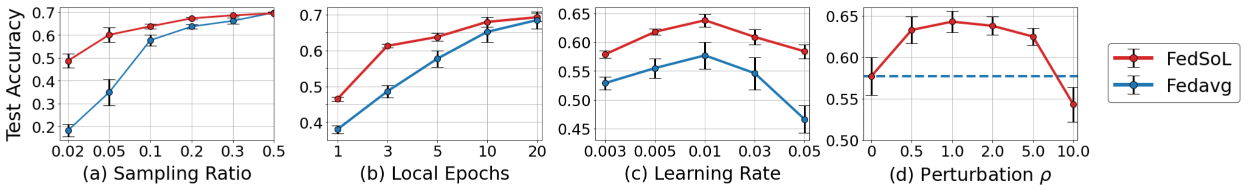

Figure 3. Performance of *FedAvg* and *FedSOL* on CIFAR-10 (α=0.1) with various setups: (a) sampling ratio, (b) the number of local epochs, (c) initial learning rate, and (d) perturbation strength. The error bars stand for the standard deviations.

show enhanced performance when combined with FedSOL.

Table 4. Comparison of proximal methods when combined with FedSOL (ρ=2.0). The heterogeneity is set as LDA (α = 0.1).

Method	**CIFAR-10**		**SVHN**		**CINIC-10**	
	Base	Combined	Base	Combined	Base	Combined
FedProx	59.80	**63.93** ↑	72.40	**84.32** ↑	40.09	**55.25** ↑
FedNova	10.00	**31.77** ↑	53.07	**79.95** ↑	21.89	**42.37** ↑
Scaffold	10.00	**62.70** ↑	21.46	**77.52** ↑	16.89	**49.96** ↑
FedDyn	60.80	**62.85** ↑	78.15	**79.43** ↑	48.25	**52.17** ↑
MOON	55.72	**60.91** ↑	29.67	**76.82** ↑	38.15	**49.14** ↑

4.4. Ablation Study

Adaptive Perturbation Strength The effect of the adaptive perturbation is depicted in Figure 4. As shown in Figure 4(a), adaptive perturbation not only improve performances but also reduces sensitivity to the selection of ρ. Meanwhile, Figure 4(b) displays the average values for the layer-wise scaling factor λ across the local models. The result highlights the increased deviation in the later layers, as a consequence of the data heterogeneity.

Partial Perturbation The results in Table 5 reveal that perturbing only the last classifier layer (*Head* in Table 5) is sufficient for FedSOL. The performance reaches as high as the full-model perturbation, yet the required computation is considerably lower. Interestingly, perturbing all layers except the head (*Body* in Table 5) incurs nearly the same computational cost yet results in diminished performance, highlighting the importance of the later layers. We conduct further experiments on larger models in Appendix F and discuss the local computational cost in Appendix I.

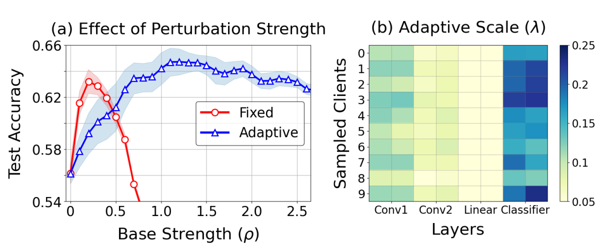

Figure 4. Effect of adaptive perturbation strength in CIFAR-10 (α=0.1). (a) Server test accuracy after 300 rounds. (b) Layer-wisely averaged λ values of FedSOL ($\rho = 1.0$) at round 200.

Table 5. Effect of partial perturbation in FedSOL on CIFAR10 (α=0.1). The FLOPs shows relative computation w.r.t. FedAvg. δ stands for the computation for the proximal loss.

Target Position	**Perturbation (ρ)**					**FLOPs**
	0.0	0.5	1.0	1.5	2.0	
All (*full*)		61.17	**64.16**	**64.38**	**63.94**	$2\times + \delta$
Body (*partial*)	56.13	60.98	62.95	63.94	63.80	$1.96\times + \delta$
Head (*partial*)		**62.65**	63.62	64.13	63.25	$1.33\times + \delta$

5. Analysis

Weight Divergence To assess the deviation of local learning from the global model, we measure the L2 distance between models: $\|w_g - w_k\|$, where w_g is the global model and w_k is the client k's trained local model. Figure 5 shows the results averaged across sampled clients. In Figure 5(a), FedSOL effectively reduces the divergence, ensuring that local models stay closely aligned with the global model. In Figure 5(b), FedSOL also promotes increased consistency among local models, reducing their mutual divergence.

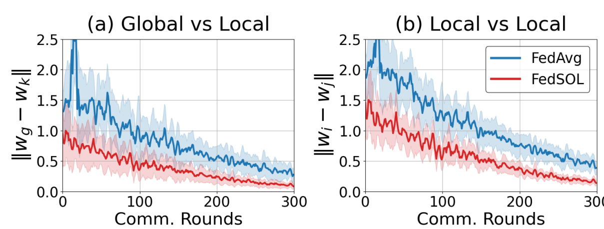

Figure 5. Analysis on weight divergence in FedAvg and FedSOL (ρ=2.0) on CIFAR-10 LDA (α=0.1). (a) shows global-local model divergence, while (b) presents the divergence across local models.

Knowledge Preservation We examine the effect of FedSOL on knowledge preservation during local learning. The results in Figure 6 show performance on the global distribution. As depicted in Figure 6, local models using FedAvg experience a significant drop in performance on the global distribution after local learning. In contrast, FedSOL maintains high performance on the global distribution, indicating that its orthogonal learning approach effectively preserves global knowledge. Further analysis of class-wise accuracy for FedAvg and FedSOL server models is presented in Figure 6(b). The results demonstrate that while FedAvg exhibits significant fluctuations and inconsistent class-wise accuracy, FedSOL consistently maintains its class-wise accuracy as communication proceeds.

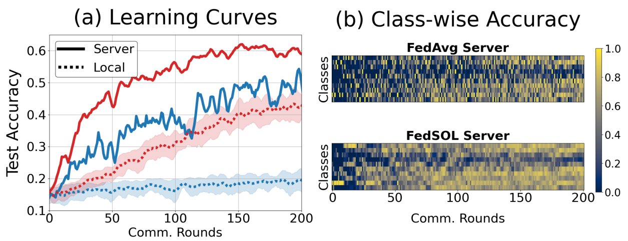

Figure 6. Comparison of FedAvg (*blue* lines) and FedSOL (ρ=2.0) (*red* lines) on CIFAR-10 (s=2). (a) learning curves for global and local models. The shaded areas reflect standard deviation across clients. (b) class-wise accuracy of the global model.

Smoothness of Loss Landscape It has been suggested that models at flatter minima more easily preserve previous knowledge after adapting to new distributions [5, 7, 13, 69]. In Figure 7, we visualize the loss landscapes [34] of global models obtained from FedAvg, FedASAM, and FedSOL. In these plots, each axis corresponds to one of the two dominant eigenvectors (top-1 and top-2) of the Hessian matrix, representing the directions of the most significant shifts in the loss landscape. Along with each landscape, we include the value of the dominant eigenvalue (λ_1) and its ratio to the fifth-largest eigenvalue (λ_1/λ_5), following the criteria used in [19, 48]. The smaller ratio observed in FedSOL indicates that variations in the loss are more evenly distributed across different directions. These results demonstrate FedSOL's effectiveness in smoothing the loss landscape.

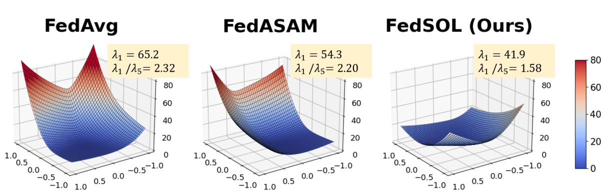

Figure 7. Loss landscape visualization of global model on CIFAR-10 LDA (α=0.1). The λ_1 and λ_5 in each figure stand for the top-1 and top-5 eigenvalues of the Hessian matrix.

6. Related Work

Federated Learning (FL) Federated learning is a distributed learning paradigm to train models without directly accessing private client data [29, 30]. The standard algorithm, FedAvg [45], aggregates trained local models by averaging their parameters. FedAvg ideally performs well when all client devices are active and IID distributed [58, 67]. While various FL algorithms have been introduced, they commonly conduct parameter averaging in a certain manner [27, 32, 37, 73]. However, its performance significantly degrades under heterogeneous data distributions among clients [26, 38, 74]. Our work aims to address this data heterogeneity issue by modifying the local learning.

Proximal Restriction in FL A prevalent strategy to address data heterogeneity in FL involves introducing a proximal objective into local learning as an auxiliary loss [27, 32, 35, 37]. This approach aims to restrict the deviation of local learning induced by the biased local distributions. For example, FedProx [37] employs an L_2 distance between models, while Scaffold [27] employs the estimated global direction as a control variate to adjust local gradients. However, such proximal objectives may hinder the acquisition of new knowledge during local learning due to conflicts with the local objective [46, 54]. In our work, we leverage the proximal objective for weight perturbation, thereby enabling local learning to be orthogonal to the proximal objective.

Orthogonal Learning in CL In CL, many approaches have adopted orthogonal learning, which align new task gradients orthogonal to old task loss spaces [13, 18, 40, 55]. This typically involves retaining data or gradients as memory [9, 10]. In FL, few recent attempts have applied similar strategies [4, 41]. For example, FOT [4] uses a random Gaussian matrix with SVD, and GradMA [41] employs gradient memory and quadratic programming, both requiring substantial computational resources. In our work, we use the proximal objective to preserve knowledge and update using a local gradient orthogonal to it. However, we observe that directly opposing the proximal gradient can adversely affect performance, a finding that aligns with recent CL research about the drawbacks of direct gradient projection [69, 75]. To address this, we propose FedSOL, which implicitly finds the orthogonal local gradient.

7. Conclusion

In this study, we propose a novel FL algorithm, FedSOL. Inspired by CL, FedSOL identifies the local gradient that is orthogonal to the proximal gradient during local learning. This orthogonal learning strategy helps to maintain previous global knowledge throughout the local learning process. We conduct extensive experiments to validate the efficacy of FedSOL and demonstrate its benefits in FL.

Acknowledgement

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by Korea government (MSIT) [No. 2021-0-00907, Development of Adaptive and Lightweight Edge-Collaborative Analysis Technology for Enabling Proactively Immediate Response and Rapid Learning, 90%] and [No. 2019-0-00075, Artificial Intelligence Graduate School Program (KAIST), 10%].

References

[1] Momin Abbas, Quan Xiao, Lisha Chen, Pin-Yu Chen, and Tianyi Chen. Sharp-maml: Sharpness-aware model-agnostic meta learning. In *International conference on machine learning*, pages 10–32. PMLR, 2022. 3

[2] Durmus Alp Emre Acar, Yue Zhao, Ramon Matas Navarro, Matthew Mattina, Paul N Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. *arXiv preprint arXiv:2111.04263*, 2021. 6

[3] Mohammed Aledhari, Rehma Razzak, Reza M Parizi, and Fahad Saeed. Federated learning: A survey on enabling technologies, protocols, and applications. *IEEE Access*, 8: 140699–140725, 2020. 1

[4] Yavuz Faruk Bakman, Duygu Nur Yaldiz, Yahya H Ezzeldin, and Salman Avestimehr. Federated orthogonal training: Mitigating global catastrophic forgetting in continual federated learning. *arXiv preprint arXiv:2309.01289*, 2023. 8

[5] Lorenzo Bonicelli, Matteo Boschini, Angelo Porrello, Concetto Spampinato, and Simone Calderara. On the effectiveness of lipschitz-driven rehearsal in continual learning. *Advances in Neural Information Processing Systems*, 35: 31886–31901, 2022. 8

[6] Debora Caldarola, Barbara Caputo, and Marco Ciccone. Improving generalization in federated learning by seeking flat minima. *arXiv preprint arXiv:2203.11834*, 2022. 3, 6, 4, 5

[7] Sungmin Cha, Hsiang Hsu, Taebaek Hwang, Flavio P Calmon, and Taesup Moon. Cpr: classifier-projection regularization for continual learning. *arXiv preprint arXiv:2006.07326*, 2020. 8

[8] Pratik Chaudhari, Anna Choromanska, Stefano Soatto, Yann LeCun, Carlo Baldassi, Christian Borgs, Jennifer Chayes, Levent Sagun, and Riccardo Zecchina. Entropy-sgd: Biasing gradient descent into wide valleys. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124018, 2019. 3

[9] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. *arXiv preprint arXiv:1812.00420*, 2018. 1, 2, 8

[10] Arslan Chaudhry, Naeemullah Khan, Puneet Dokania, and Philip Torr. Continual learning in low-rank orthogonal subspaces. *Advances in Neural Information Processing Systems*, 33:9900–9911, 2020. 1, 8

[11] Xiangning Chen, Cho-Jui Hsieh, and Boqing Gong. When vision transformers outperform resnets without pretraining or strong data augmentations. *arXiv preprint arXiv:2106.01548*, 2021. 3

[12] Luke N Darlow, Elliot J Crowley, Antreas Antoniou, and Amos J Storkey. Cinic-10 is not imagenet or cifar-10. *arXiv preprint arXiv:1810.03505*, 2018. 5, 1

[13] Danruo Deng, Guangyong Chen, Jianye Hao, Qiong Wang, and Pheng-Ann Heng. Flattening sharpness for dynamic gradient projection memory benefits continual learning. *Advances in Neural Information Processing Systems*, 34: 18710–18721, 2021. 1, 8

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1

[15] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012. 5, 1

[16] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 1

[17] 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. 2

[18] Mehrdad Farajtabar, Navid Azizan, Alex Mott, and Ang Li. Orthogonal gradient descent for continual learning. In *International Conference on Artificial Intelligence and Statistics*, pages 3762–3773. PMLR, 2020. 1, 2, 8

[19] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020. 3, 8, 4

[20] Chaoyang He, Songze Li, Jinhyun So, Xiao Zeng, Mi Zhang, Hongyi Wang, Xiaoyang Wang, Praneeth Vepakomma, Abhishek Singh, Hang Qiu, et al. Fedml: A research library and benchmark for federated machine learning. *arXiv preprint arXiv:2007.13518*, 2020. 1

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016. 6, 1

[22] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 2

[23] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018. 3

[24] Donglin Jiang, Chen Shan, and Zhihui Zhang. Federated learning algorithm based on knowledge distillation. In *2020 International Conference on Artificial Intelligence and Computer Engineering (ICAICE)*, pages 163–167. IEEE, 2020. 2

[25] Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. *arXiv preprint arXiv:1912.02178*, 2019. 3

[26] 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. 1, 8

[27] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pages 5132–5143. PMLR, 2020. 1, 2, 6, 8

[28] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2016. 3

[29] Jakub Konečnỳ, H Brendan McMahan, Daniel Ramage, and Peter Richtárik. Federated optimization: Distributed machine learning for on-device intelligence. *arXiv preprint arXiv:1610.02527*, 2016. 1, 8

[30] Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016. 1, 8

[31] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 and cifar-100 datasets. *URl: https://www. cs. toronto. edu/kriz/cifar. html*, 6, 2009. 5, 1

[32] Gihun Lee, Yongjin Shin, Minchan Jeong, and Se-Young Yun. Preservation of the global knowledge by not-true self knowledge distillation in federated learning. *arXiv preprint arXiv:2106.03097*, 2021. 1, 2, 5, 6, 8

[33] Seung Hoon Lee, Seunghyun Lee, and Byung Cheol Song. Vision transformer for small-size datasets. *arXiv preprint arXiv:2112.13492*, 2021. 6, 1

[34] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. *Advances in neural information processing systems*, 31, 2018. 8

[35] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10713–10722, 2021. 2, 6, 8, 1

[36] 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, 2020. 1

[37] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2:429–450, 2020. 2, 6, 8

[38] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. *arXiv preprint arXiv:1907.02189*, 2019. 1, 8

[39] Zexi Li, Xinyi Shang, Rui He, Tao Lin, and Chao Wu. No fear of classifier biases: Neural collapse inspired federated learning with synthetic and fixed classifier. *arXiv preprint arXiv:2303.10058*, 2023. 4

[40] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017. 1, 8

[41] Kangyang Luo, Xiang Li, Yunshi Lan, and Ming Gao. Gradma: A gradient-memory-based accelerated federated learning with alleviated catastrophic forgetting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3708–3717, 2023. 8

[42] Mi Luo, Fei Chen, Dapeng Hu, Yifan Zhang, Jian Liang, and Jiashi Feng. No fear of heterogeneity: Classifier calibration for federated learning with non-iid data. *arXiv preprint arXiv:2106.05001*, 2021. 4, 1, 2

[43] Othmane Marfoq, Giovanni Neglia, Richard Vidal, and Laetitia Kameni. Personalized federated learning through local memorization. In *International Conference on Machine Learning*, pages 15070–15092. PMLR, 2022. 2

[44] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, pages 109–165. Elsevier, 1989. 1

[45] 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. PMLR, 2017. 3, 5, 6, 8, 1, 2

[46] Matias Mendieta, Taojiannan Yang, Pu Wang, Minwoo Lee, Zhengming Ding, and Chen Chen. Local learning matters: Rethinking data heterogeneity in federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8397–8406, 2022. 2, 8, 3

[47] Martial Mermillod, Aurélia Bugaiska, and Patrick Bonin. The stability-plasticity dilemma: Investigating the continuum from catastrophic forgetting to age-limited learning effects. *Frontiers in psychology*, 4:504, 2013. 1

[48] Peng Mi, Li Shen, Tianhe Ren, Yiyi Zhou, Xiaoshuai Sun, Rongrong Ji, and Dacheng Tao. Make sharpness-aware minimization stronger: A sparsified perturbation approach. *arXiv preprint arXiv:2210.05177*, 2022. 8

[49] Yurii E Nesterov. A method for solving the convex programming problem with convergence rate o (1/kˆ 2). In *Dokl. akad. nauk Sssr*, pages 543–547, 1983. 3

[50] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011.*, 2011. 5, 1

[51] Jaehoon Oh, Sangmook Kim, and Se-Young Yun. Fedbabu: Towards enhanced representation for federated image classification. *arXiv preprint arXiv:2106.06042*, 2021. 5, 2

[52] German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019. 1

[53] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 1

[54] Zhe Qu, Xingyu Li, Rui Duan, Yao Liu, Bo Tang, and Zhuo Lu. Generalized federated learning via sharpness aware minimization. *arXiv preprint arXiv:2206.02618*, 2022. 2, 3, 6, 8, 4, 5

[55] Gobinda Saha, Isha Garg, and Kaushik Roy. Gradient projection memory for continual learning. *arXiv preprint arXiv:2103.09762*, 2021. 1, 2, 8

[56] Neta Shoham, Tomer Avidor, Aviv Keren, Nadav Israel, Daniel Benditkis, Liron Mor-Yosef, and Itai Zeitak. Overcoming forgetting in federated learning on non-iid data. *arXiv preprint arXiv:1910.07796*, 2019. 1, 2

[57] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 6, 1

[58] Sebastian U Stich. Local sgd converges fast and communicates little. *arXiv preprint arXiv:1805.09767*, 2018. 8

[59] Yan Sun, Li Shen, Shixiang Chen, Liang Ding, and Dacheng Tao. Dynamic regularized sharpness aware minimization in federated learning: Approaching global consistency and smooth landscape. *arXiv preprint arXiv:2305.11584*, 2023. 3, 4

[60] Yan Sun, Li Shen, Tiansheng Huang, Liang Ding, and Dacheng Tao. Fedspeed: Larger local interval, less communication round, and higher generalization accuracy. *arXiv preprint arXiv:2302.10429*, 2023. 3, 4

[61] Antonio Torralba, Rob Fergus, and William T Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 30(11):1958–1970, 2008. 1

[62] Hongyi Wang, Mikhail Yurochkin, Yuekai Sun, Dimitris Papailiopoulos, and Yasaman Khazaeni. Federated learning with matched averaging. *arXiv preprint arXiv:2002.06440*, 2020. 5, 2

[63] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. *arXiv preprint arXiv:2007.07481*, 2020. 1, 2, 6

[64] Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *arXiv preprint arXiv:2302.00487*, 2023. 1

[65] Pengfei Wang, Zhaoxiang Zhang, Zhen Lei, and Lei Zhang. Sharpness-aware gradient matching for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3769–3778, 2023. 3

[66] Zhenyi Wang, Enneng Yang, Li Shen, and Heng Huang. A comprehensive survey of forgetting in deep learning beyond continual learning. *arXiv preprint arXiv:2307.09218*, 2023. 1, 2

[67] Blake Woodworth, Kumar Kshitij Patel, Sebastian Stich, Zhen Dai, Brian Bullins, Brendan Mcmahan, Ohad Shamir, and Nathan Srebro. Is local sgd better than minibatch sgd? In *International Conference on Machine Learning*, pages 10334–10343. PMLR, 2020. 8

[68] Chencheng Xu, Zhiwei Hong, Minlie Huang, and Tao Jiang. Acceleration of federated learning with alleviated forgetting in local training. *arXiv preprint arXiv:2203.02645*, 2022. 1, 2

[69] Enneng Yang, Li Shen, Zhenyi Wang, Shiwei Liu, Guibing Guo, and Xingwei Wang. Data augmented flatness-aware gradient projection for continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5630–5639, 2023. 8

[70] Jiancheng Yang, Rui Shi, Donglai Wei, Zequan Liu, Lin Zhao, Bilian Ke, Hanspeter Pfister, and Bingbing Ni. Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific Data*, 10(1):41, 2023. 5, 1

[71] 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. 1

[72] Guanxiong Zeng, Yang Chen, Bo Cui, and Shan Yu. Continual learning of context-dependent processing in neural networks. *Nature Machine Intelligence*, 1(8):364–372, 2019. 1

[73] Lin Zhang, Li Shen, Liang Ding, Dacheng Tao, and Ling-Yu Duan. Fine-tuning global model via data-free knowledge distillation for non-iid federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10174–10183, 2022. 8

[74] 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. 1, 5, 8

[75] Zhen Zhao, Zhizhong Zhang, Xin Tan, Jun Liu, Yanyun Qu, Yuan Xie, and Lizhuang Ma. Rethinking gradient projection continual learning: Stability/plasticity feature space decoupling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3718–3727, 2023. 1, 8

[76] Juntang Zhuang, Boqing Gong, Liangzhe Yuan, Yin Cui, Hartwig Adam, Nicha Dvornek, Sekhar Tatikonda, James Duncan, and Ting Liu. Surrogate gap minimization improves sharpness-aware training. *arXiv preprint arXiv:2203.08065*, 2022. 3