Tackling Data Heterogeneity in Federated Learning via Loss Decomposition

Shuang Zeng[1*], Pengxin Guo[1*], Shuai Wang[2], Jianbo Wang[3], Yuyin Zhou[4], and Liangqiong Qu[1(✉)]

[1] Department of Statistics and Actuarial Science, The University of Hong Kong, Hong Kong SAR, China
liangqqu@hku.hk
[2] School of Cyberspace, Hangzhou Dianzi University, Hangzhou, China
[3] New H3C Technologies Co., Ltd.
[4] Computer Science and Engineering, University of California, Santa Cruz, USA

Abstract. Federated Learning (FL) is a rising approach towards collaborative and privacy-preserving machine learning where large-scale medical datasets remain localized to each client. However, the issue of data heterogeneity among clients often compels local models to diverge, leading to suboptimal global models. To mitigate the impact of data heterogeneity on FL performance, we start with analyzing how FL training influence FL performance by decomposing the global loss into three terms: local loss, distribution shift loss and aggregation loss. Remarkably, our loss decomposition reveals that existing local training-based FL methods attempt to reduce the distribution shift loss, while the global aggregation-based FL methods propose better aggregation strategies to reduce the aggregation loss. Nevertheless, a comprehensive joint effort to minimize all three terms is currently limited in the literature, leading to subpar performance when dealing with data heterogeneity challenges. To fill this gap, we propose a novel FL method based on global loss decomposition, called FedLD, to jointly reduce these three loss terms. Our FedLD involves a margin control regularization in local training to reduce the distribution shift loss, and a principal gradient-based server aggregation strategy to reduce the aggregation loss. Notably, under different levels of data heterogeneity, our strategies achieve better and more robust performance on retinal and chest X-ray classification compared to other FL algorithms. Our code is available at https://github.com/Zeng-Shuang/FedLD.

Keywords: Federated Learning · Data Heterogeneity · Principal Gradients.

1 Introduction

Federated Learning (FL), where computations are performed locally at each client without sharing data, presents a promising approach to accessing large, representative data for training robust deep learning models with enhanced generalizability [18]. In recent years, FL has witnessed some pivotal success on

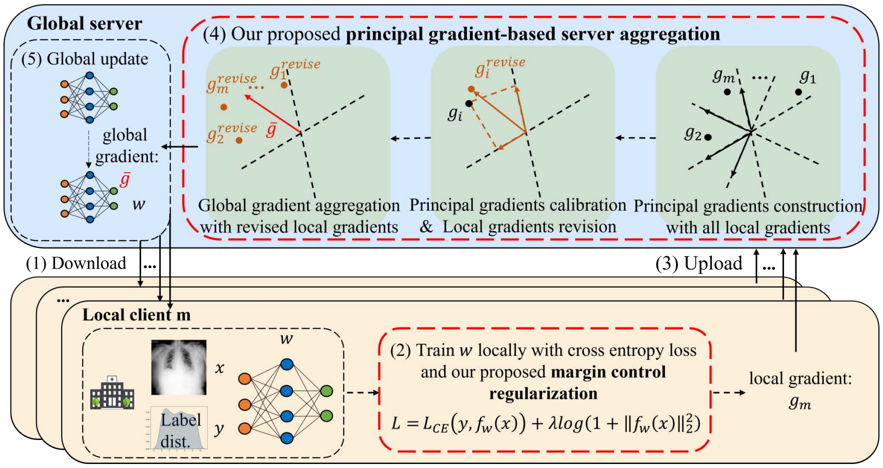
Global server
(4) Our proposed principal gradient-based server aggregation
(5) Global update
global gradient: $\bar{g}$
w
g_m^{revise}
g_1^{revise}
g_2^{revise}
$\bar{g}$
g_i^{revise}
g_i
g_m
g_1
g_2
Global gradient aggregation with revised local gradients
Principal gradients calibration & Local gradients revision
Principal gradients construction with all local gradients
(1) Download
(3) Upload
Local client m
x
y
Label dist.
w
(2) Train w locally with cross entropy loss and our proposed margin control regularization
$L = L_{CE}(y, f_w(x)) + \lambda log(1 + \|f_w(x)\|_2^2)$
local gradient: g_m

Fig. 1. Overview of the proposed FedLD. (1) Once each local client downloads the global model parameter $\boldsymbol{w}$, (2) it starts training locally with the cross-entropy loss and our proposed margin control regularization. (3) After that, each client uploads its local gradient to the global server. (4) Then, the global server aggregates these local gradients with our proposed principal gradient-based server aggregation, which includes three steps: First, use all local gradients to construct principal gradients; Second, calibrate principal gradients and use them to revise local gradients; Third, aggregate revised local gradients to generate the global gradient. (5) Finally, the server updates the global model parameter with the global gradient and sends it to local clients for the next round.

various medical applications, such as medical image segmentation [35], medical image classification [3], cancer boundary detection [20], among others [9,3,10]. Despite its widespread, FL suffers from data heterogeneity [31,32], as data can be non-independent and identically distributed (non-IID) across clients, which is particularly prevalent in medical scenarios [14,30,22,33,23].

Two main approaches have been proposed to tackle data heterogeneity in FL: (**i**) regularizing *local training* to mitigate the deviation between local and global objectives and (**ii**) designing more efficient *global aggregation* strategies. For local training, methods such as FedProx [13], SCAFFOLD [11], FedCM [29], and FEDIIR [5] utilize the difference between local and global models as a regularization term to mitigate the deviation between local and global objectives. However, these methods may suffer from additional communication costs [11] or the inability of gradient differences to accurately capture model bias [6,29,5]. For global aggregation, several works have been proposed to develop efficient global aggregation strategies, such as FedMA [26], pFedLA [17], RobFedAvg [25], and GAMF [16]. However, these strategies may incur extra computing resources [16,25] or overlook the impact of local training on overall performance [2,34].

Most of the works address the issue of data heterogeneity by applying strategies either at the local or server side. In this paper, we ask: *How can the joint* *effect of local training and server aggregation be leveraged to improve FL performance in the presence of data heterogeneity?* To answer this, we decompose the global loss into three terms: local loss, distribution shift loss, and aggregation loss, which reveals that existing methods often focus on addressing only one or two of these terms, rather than all three. For example, in standard FL training, such as FedAvg [18], only the local loss is minimized, while the other two terms are ignored. Remarkably, existing local training-based methods attempt to further reduce the distribution shift loss through additional local training regularization [13,11,29,5], and global aggregation-based methods propose better aggregation strategies to minimize the aggregation loss [26,17,25,16].

However, a comprehensive joint effort to minimize all three terms - local loss, distribution shift loss, and aggregation loss - is currently limited in the literature. To fill this gap, we propose a novel FL method based on global loss decomposition, called FedLD, to jointly reduce these three loss terms, as shown in Fig. 1. Our FedLD involves a margin control regularization in local training to reduce the distribution shift loss, and a principal gradient-based server aggregation strategy to minimize the aggregation loss. Specifically, our margin control regularization encourages local models to learn stable features instead of shortcut features by adding the l_2-norm of the output logits to the standard cross entropy loss, thereby reducing the distribution shift loss. On the other hand, the local models trained on heterogeneous data distribution in FL may exhibit different or even conflicting judgments, leading to potential conflicts in clients' gradients. Naively aggregating these conflicting gradients in FL may lead to increased aggregation loss, thus resulting in poorly performing global model. Therefore, we propose a principal gradient-based server aggregation strategy to mitigate conflicting gradients by prioritizing principal directions that benefit all clients while discarding conflict-contributing directions, ultimately reducing the aggregation loss.

In summary, our key contributions are as follows: (i) We propose a novel global loss decomposition in FL, decomposing the global objective into local loss, distribution shift loss, and aggregation loss, which provides an analytical framework to assess the impact of these loss terms on FL performance. (ii) We provide a novel and practical algorithm, FedLD, which incorporates margin control regularization and a principal gradient-based server aggregation strategy to jointly reduce local loss, distribution shift loss, and aggregation loss. (iii) We conduct extensive numerical studies on retinal and chest X-ray classification datasets to verify the performance of our algorithm, which outperforms several classic baselines under different levels of data heterogeneity.

2 Methodology

2.1 Problem Setup

In this work, we address the problem of data heterogeneity in cross-device FL involving a central server and m clients, for a supervised image classification task. Each client i has its own local data distribution, denoted by $\mathcal{P}_i$. Let x

and y indicate the input features and labels extracted from client i's local data distribution $\mathcal{P}_i$, respectively, such that $(x, y) \sim \mathcal{P}_i(x, y)$. Then the objective is to minimize the aggregate loss function $\mathcal{L}(\boldsymbol{w})$, which is formulated as:

$$\mathcal{L}(\boldsymbol{w}) = \sum_{i=1}^{m} \frac{n_i}{n} \mathcal{L}_i(\boldsymbol{w}), \tag{1}$$

where $\mathcal{L}_i(\boldsymbol{w}) = \mathbb{E}_{(x,y)\sim\mathcal{P}_i(x,y)} \left[l\left(\boldsymbol{w}; x, y\right) \right]$ is the empirical loss of client i, n_i is the number of samples for client i and $n = \sum_{i=1}^{m} n_i$, and $\boldsymbol{w}$ denotes the global model parameter. In data heterogeneity setting, $\mathcal{P}_i \neq \mathcal{P}_j$ for different client i and client j. This disparity causes the local model to perform differently across clients, which degrades FL performance or even causes model divergence.

2.2 Global Loss Objective Decomposition

To better understand the influence of data heterogeneity on FedAvg in each round, we decompose the loss function in Eq.(1) as follows:

$$\begin{aligned}\mathcal{L}(\boldsymbol{w}) =& \sum_{i=1}^{m} \frac{n_i}{n} \mathcal{L}_i(\boldsymbol{w}) \\ =& \underbrace{\sum_{i=1}^{m} \frac{n_i}{n} \mathcal{L}_i\left(\boldsymbol{w}_i\right)}_{\text{Local loss}} + \underbrace{\sum_{j=1}^{m} \sum_{i=1}^{m} \frac{n_j}{n} \frac{n_i}{n} \left(\mathcal{L}_j\left(\boldsymbol{w}_i\right) - \mathcal{L}_i\left(\boldsymbol{w}_i\right)\right)}_{\text{Distribution shift loss}} + \underbrace{\sum_{i=1}^{m} \frac{n_i}{n} \left(\mathcal{L}(\boldsymbol{w}) - \mathcal{L}\left(\boldsymbol{w}_i\right)\right)}_{\text{Aggregation loss}}.\end{aligned} \tag{2}$$

Here $\mathcal{L}_j\left(\boldsymbol{w}_i\right)$ denotes the empirical loss of client i's local model $\boldsymbol{w}_i$ when evaluated on the client j's local dataset. The formula in Eq.(2) holds because $\sum_{j=1}^{m} \frac{n_j}{n} \mathcal{L}_j(\cdot) = \mathcal{L}(\cdot)$. We can interpret different terms in Eq.(2) as follows: (i) $\mathcal{L}_i\left(\boldsymbol{w}_i\right)$ in the first term denotes the empirical loss of client i's local model $\boldsymbol{w}_i$ trained on its local dataset. We thus refer to the first term as the local loss. (ii) $\mathcal{L}_j\left(\boldsymbol{w}_i\right) - \mathcal{L}_i\left(\boldsymbol{w}_i\right)$ in the second term denotes the increase in the empirical loss of client i's local model $\boldsymbol{w}_i$ when evaluated on client j's local dataset compared to client i's local dataset. This increase arises from the data distribution shift between client i and client j. We thus call the absolute value of the second term the distribution shift loss. (iii) $\mathcal{L}(\boldsymbol{w}) - \mathcal{L}\left(\boldsymbol{w}_i\right)$ in the third term denotes the increase in the empirical loss on all samples of local datasets for the global model ($\boldsymbol{w}$, obtained after aggregating local models), compared with local model $\boldsymbol{w}_i$. As this increase comes from the server aggregation operation, we refer to the absolute value of this term as the aggregation loss.

The derived loss decomposition reveals that, in each round, FedAvg only minimizes the local loss (first term in Eq.(2)) through local training, while ignoring the distribution shift loss and aggregation loss. As shown in Fig. 2 (dashed lines), both the distribution shift loss and aggregation loss increase with data heterogeneity. This results in poorer performance and slower convergence of the FedAvg algorithm when facing increased data heterogeneity challenges. This observation motivates us to explore methods that jointly minimize the decomposed three terms in Eq.(2). To reduce the distribution shift loss, we need to improve the performance of each local model $\boldsymbol{w}_i$ on the data distributions of other clients during its local training step. To reduce the aggregation loss, we need to develop a more effective way to aggregate local models

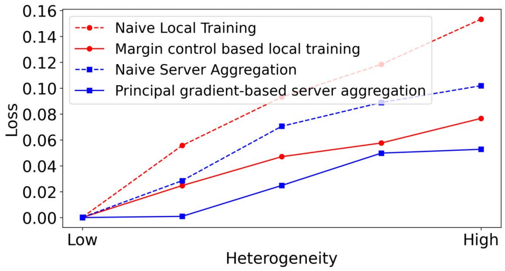

Fig. 2. Distribution shift loss of different local training methods (red lines) and aggregation loss of different server aggregation methods (blue lines) under different levels of heterogeneity in one FL round.

during server aggregation step. The overview of our method is shown in Fig. 1, and we will elaborate these two compenents in the following. Furthermore, the detailed description of our algorithm can be found in supplementary material.

2.3 Margin Control in Clients' Local Training

Shortcut learning refers to a machine learning model's reliance on unstable correlations i.e. shortcut features [4]. It can lead to poor performance when the relationship between the label and the shortcut feature changes [12]. This shortcut learning could be a key factor causing a local model $\boldsymbol{w}_i$ trained on its local dataset to perform worse on other datasets with different distributions. In other words, shortcut learning increases the distribution shift loss in Eq.(2). This motivates us to find a method for mitigating shortcut learning in local training to reduce the distribution shift loss. [21] shows that training with cross entropy loss can lead to the preference for maximizing the margin i.e. range of the logits values, which in turn causes the model to rely more on shortcut features rather than stable features. Motivated by this, we penalize the margin in local training, aiming to reduce the reliance on shortcut features and encourage the model to learn more stable features, thereby reducing the distribution shift loss. Specifically, we use an approach which introduces a margin control regularization term by calculating the l_2-norm of the output logits and adding it to the cross entropy loss as follows:

$$\mathcal{L}_{\text{marg-log}} = \mathcal{L}_{CE}\left(y, f_{\boldsymbol{w}}(x)\right) + \lambda \log\left(1 + \|f_{\boldsymbol{w}}\left(x\right)\|_2^2\right), \tag{3}$$

where $\mathcal{L}_{CE}\left(y, f_{\boldsymbol{w}}(x)\right) = -\sum_{i=1}^{C} y_i \log\left(\operatorname{softmax}\left(f_{\boldsymbol{w},i}(x)\right)\right)$, is the standard cross-entropy loss function, and $y = [y_1, \cdots, y_C]^\top, f_{\boldsymbol{w}}(x) = [f_{\boldsymbol{w},1}(x), \cdots, f_{\boldsymbol{w},C}(x)]^\top$. Here C is the number of classes for our classification task, y is the one-hot vector for the label, and $f_{\boldsymbol{w}}(\cdot)$ is the output logits of the model $\boldsymbol{w}$. Margin control regularization helps to mitigate the reliance on shortcut features and encourages the model to focus on stable features. As shown in Fig. 2 (red lines), margin control regularization helps to reduce the distribution shift loss. While l_2-norm regularization on output logits was also used in FedSR [19] to align representations from different clients with a common reference, our margin control regularization is motivated by shortcut learning. This allows us to use other regularization techniques, such as evaluating log loss on a margin multiplied by a decreasing function or penalizing large margins by setting thresholds.

2.4 Principal Gradient-based Server Aggregation

One of the key factors contributing to the performance degradation of FL in data heterogeneity is the conflicting gradients of local models trained on diverse local datasets [2,34]. Consider a scenario of FL training on two clients with datasets A and B. These datasets are unevenly distributed subsets of the same population, where dataset A predominantly contains "class A" data, and dataset B mainly comprises "class B" data. The models trained on these two datasets may exhibit different or even conflicting judgments, leading to potential conflicts in the gradients of the clients trained on these datasets. Naively aggregating the conflicting gradients in FL may lead to a poorly performing global model, thus increasing the aggregation loss in Eq.(2). Therefore, we propose a principal gradient-based server aggregation approach to amend these conflicting gradients and force them to follow a direction that maximally benefits all participating clients. The specific steps are:

Step 1: Principal Gradients Construction. For client i, we flatten its local gradients into a vector, denoted as $\boldsymbol{g}_i \in \mathbb{R}^{d\times 1}$, where d is the flattened dimension of the local gradients and is usually fairly large for modern deep learning architectures. All the participating local gradient vectors make up a matrix $\boldsymbol{G} = [\boldsymbol{g}_1, \cdots, \boldsymbol{g}_m]$. Next, we perform singular value decomposition (SVD) on matrix $\boldsymbol{G}$, generating a series of eigenvalues and their corresponding eigenvectors, expressed as:

$$\lambda_z, \boldsymbol{v}_z = \mathrm{SVD}_z\left(\frac{1}{m}\boldsymbol{G}\boldsymbol{G}^\top\right), \tag{4}$$

where λ_z and $\boldsymbol{v}_z$ represent the z-th largest eigenvalue and its corresponding eigenvector, respectively. However, the high dimension of $\boldsymbol{G}\boldsymbol{G}^\top \in \mathbb{R}^{d\times d}$ makes naive SVD complex and prohibitive. Thus, we construct a bijection [27] to reduce computational complexity as follows:

$$\boldsymbol{G}^\top\boldsymbol{G}\boldsymbol{e}_z = \lambda_z\boldsymbol{e}_z \quad \Longrightarrow \quad \boldsymbol{G}\boldsymbol{G}^\top\boldsymbol{G}\boldsymbol{e}_z = \lambda_z\boldsymbol{G}\boldsymbol{e}_z \quad \Longrightarrow \quad \boldsymbol{v}_z = \boldsymbol{G}\boldsymbol{e}_z. \tag{5}$$

Here $\boldsymbol{e}_z$ represents the z-th largest eigenvector of the matrix $\boldsymbol{G}^\top\boldsymbol{G}$. In this way, we can transform the computation of SVD for $\boldsymbol{G}\boldsymbol{G}^\top \in \mathbb{R}^{d\times d}$ in Eq.(4) to the computation of SVD for $\boldsymbol{G}^\top\boldsymbol{G} \in \mathbb{R}^{m\times m}$, which is much cheaper, as $m \ll d$. After performing SVD in Eq.(5), we obtain a set of eigenvectors $\boldsymbol{V} = \{\boldsymbol{v}_1, \cdots, \boldsymbol{v}_m\}$ and consider them as the principal gradients.

Step 2: Local Gradients Revision based on Calibrated Principal Gradients. These eigenvectors are ordered based on the magnitude of the eigenvalues and are unoriented, since a negative multiple of an eigenvector is also a valid eigenvector. However, we need to determine the directions of these eigenvectors so that they point to the directions that can reduce the loss. For simplicity, we use the mean of local gradients $\hat{\boldsymbol{g}} = \frac{1}{m}\sum_i \boldsymbol{g}_i$ as a reference for calibrating the principal gradient direction. Specifically, we adjust the z-th largest oriented eigenvector to be positively related to the reference by following the calibration process:

$$\bar{\boldsymbol{v}}_z = \begin{cases} \boldsymbol{v}_z, & \text{if } \langle \boldsymbol{v}_z, \hat{\boldsymbol{g}}\rangle \geq 0 \\ -\boldsymbol{v}_z, & \text{otherwise} \end{cases}. \tag{6}$$

Next, we select the eigenvectors with the top L largest eigenvalues, i.e. $\{\bar{\boldsymbol{v}}_1, \cdots, \bar{\boldsymbol{v}}_L\}$, to form the principal coordinate system for local gradient projection. For each client i, we project its local gradient $\boldsymbol{g}_i$ onto this principal coordinate system, with the projection of $\boldsymbol{g}_i$ on the l-th eigenvector (axis) is calculated as $\boldsymbol{g}'_{i,l} = \frac{\boldsymbol{g}_i\bar{\boldsymbol{v}}_l}{\|\bar{\boldsymbol{v}}_l\|\|\bar{\boldsymbol{v}}_l\|}\bar{\boldsymbol{v}}_l$. We then aggregate all the projections together into a weighted sum, using the l-th eigenvalue λ_l as the weight for the l-th axis. Additionally, we apply a length correction to the weighted sum by multiplying it by $\frac{\|\boldsymbol{g}_i\|}{\|\boldsymbol{g}'_{i,l}\|}$:

$$\boldsymbol{g}_i^{revise} = \sum_{l=1}^{L} \frac{\|\boldsymbol{g}_i\|}{\|\boldsymbol{g}'_{i,l}\|} \frac{\lambda_l}{\|\lambda_l\|} \boldsymbol{g}'_{i,l}, \tag{7}$$

where $\boldsymbol{g}_i^{revise}$ is the revised gradient for the local client i. The length correction factor $\frac{\|\boldsymbol{g}_i\|}{\|\boldsymbol{g}'_{i,l}\|}$ in Eq.(7) aims to ensure that the magnitude of the revised local gradient for client i remains the same as the original local gradient $\boldsymbol{g}_i$. This is crucial because a reduced magnitude of the revised local gradient can hinder FL convergence [1].

Step 3: Global Gradient Aggregation. Last, we aggregate all the revised local gradients $\boldsymbol{g}_i^{revise}$ to construct the global gradients in the server as: $\bar{\boldsymbol{g}} = \sum_{i=1}^{m} \frac{n_i}{n} \boldsymbol{g}_i^{revise}$.

Remark 1. To better comprehend the gradient projection based on SVD, let's examine $\frac{1}{m}\boldsymbol{G}\boldsymbol{G}^\top$ in (4) as:

$$\frac{1}{m}\boldsymbol{G}\boldsymbol{G}^\top = \frac{1}{m}[\boldsymbol{g}_1, \cdots, \boldsymbol{g}_m] \otimes [\boldsymbol{g}_1, \cdots, \boldsymbol{g}_m] = \frac{1}{m}\sum_i \boldsymbol{g}_i \otimes \boldsymbol{g}_i = \frac{1}{m}\sum_i \mathcal{I}_i = -\frac{1}{m}\sum_i \mathcal{H}_i. \tag{8}$$

The $\mathcal{I}_i$ and $\mathcal{H}_i$ in the above formulation represent the Fisher Information matrix and Hessian matrix, respectively, and $\otimes$ represents the tensor product in mathematics. The approximation is a positive semi-definite covariance matrix. The eigenvalue λ_z and eigenvector v_z is one-to-one correspondence and arranged based on the size of eigenvalue. For the eigenvalue λ_z, it is the curvature of the loss in the direction of $\boldsymbol{v}_z$. Since the distribution of the eigenvalues could affect the training behavior, e.g the first-order optimization methods slow down significantly when $\{\lambda_z\}$ are highly spread out, we can get rid of those insignificant directions and only use the directions with the large curvature. Thus, the proposed principal gradient-based server aggregation can mitigate conflicting gradients by prioritizing principal directions that benefit all clients while discarding conflict-contributing directions. The results in Fig. 2 (blue lines) also demonstrate the effectiveness of the proposed principal gradient-based server aggregation, as it significantly reduces aggregation loss.

3 Experiments

3.1 Experimental Setup

We evaluate our method on two medical image classification datasets: Retina [30] with 5 clients and COVID-FL [30], a real-world federated dataset with 12 clients that exhibits both shifts in label and feature distributions. Following [30], we construct different levels of data heterogeneity for Retina by constructing label shifts using Dirichlet distribution with α 100, 0.1, 0.5 for split1, 2, 3 respectively, i.e., Split-1 (IID), Split-2 (moderate non-IID), and Split-3 (severe non-IID). The ResNet-50 [7] is adopted in all experiments. We compare our method with FedAvg [18], FedProx [13], FedBN [15], FedPAC [28], and FedGH [34]. For optimization, we adopt the SGD optimizer [24] with a learning rate of 0.01 for Retina and 0.005 for COVID-FL. The batch size is set to 50. The number of global communication rounds is set to 200, and the number of local training epochs is set to 1. We choose λ values of 0.1, 0.03, and 0.03 for Split-1, Split-2, and Split-3 of Retina, respectively, and 0.01 for COVID-FL. We set $L = 0.8m$, where m is the number of selected participating clients in each round. We set the client sampling rate to 1, unless otherwise stated.

3.2 Evaluation Results

Table 1. The comparison of final test accuracy (%) of different methods.

Method	Retina			COVID-FL
	Split-1	Split-2	Split-3	
FedAvg	83.63	82.26	81.13	79.86
FedProx	84.17	83.53	81.20	81.88
FedBN	83.91	75.91	65.25	56.34
FedPAC	78.01	71.64	52.81	81.63
FedGH	83.90	83.33	81.56	82.26
Ours	**85.20**	**83.83**	**82.30**	**83.87**

Table 2. Ablation study. **Margin** denotes margin control regularization. **Principal** denotes principal gradient-based server aggregation.

Margin	Principal	Retina		
		Split-1	Split-2	Split-3
×	×	83.63	82.26	81.13
×	✓	84.76	82.67	81.66
✓	×	84.63	83.5	81.96
✓	✓	85.20	83.83	82.30

As demonstrated in Table 1, our method outperforms all compared methods in all datasets with all levels of data heterogeneity. Importantly, as the data heterogeneity increases (i.e., from Split-1 to Split-3 in Retina), the performance improvement compared to the second-best method (i.e., FedGH) also increases (i.e., from 0.04% to 2.20%), which demonstrates the ability of our method to effectively alleviate the negative impact of data heterogeneity.

Table 3. The comparison of final test accuracy (%) of different methods on Retina with 50 clients. We apply client sampling with rate 0.1 for FL training.

Method	FedAvg	FedProx	FedBN	FedPAC	FedGH	Ours
Accuracy (%)	67.43	69.87	68.40	66.59	70.33	**71.30**

3.3 Analysis

Ablation Study. As demonstrated in Table 2, both of them can help improve the average test accuracy and the combination of them is able to achieve the most satisfactory model performance, which demonstrates the effectiveness of each component.

Ability of Loss Reduction. We simulate different levels of data heterogeneity using the Dirichlet distribution [8] with the concentration parameter $\alpha \in \{100, 10, 1, 0.1, 0.01\}$ on the Retina dataset. First, we train local models with or without margin control regularization. Second, we aggregate the local models trained in the naive way with principal gradient-based aggregation or naive server aggregation. As shown in Fig. 2, margin control regularization significantly reduces distribution shift loss compared with

naive local training (red lines), and principal gradient-based aggregation significantly reduces aggregation loss compared with naive server aggregation (blue lines).
Generalization to Multiple Clients. We further simulate 50 clients on the Retina dataset using the Dirichlet distribution with the concentration parameter $\alpha = 0.5$. As shown in Table 3, our method also achieves the best result with multiple clients, which demonstrates the generalizability of our method.

4 Conclusion

In this paper, we propose a global loss decomposition to understand the impact of data heterogeneity on FL performance, which decomposes the global loss into three terms: local loss, distribution shift loss and aggregation loss. We then propose two strategies, margin control regularization and principal gradient-based server aggregation, to reduce them jointly, thus tackling data heterogeneity in FL. Our loss decomposition provides an analytical tool for analysing the impact of different operations on FL performance, and our proposed margin control regularization and principal gradient-based server aggregation can seamlessly integrate into any FL frameworks. Extensive experiments demonstrate that our algorithm effectively reduces the impact of data heterogeneity on FL performance.

Acknowledgments. This work was supported by National Natural Science Foundation of China (62306253) Early career fund (27204623), Guangdong Natural Science Fund-General Programme (2024A1515010233), and UCSC hellman fellowship.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. An, X., Shen, L., Hu, H., Luo, Y.: Federated learning with manifold regularization and normalized update reaggregation. Advances in Neural Information Processing Systems **36** (2024)
2. Charles, Z., Garrett, Z., Huo, Z., Shmulyian, S., Smith, V.: On large-cohort training for federated learning. Advances in neural information processing systems **34**, 20461–20475 (2021)
3. Dayan, I., Roth, H.R., Zhong, A., Harouni, A., Gentili, A., Abidin, A.Z., Liu, A., Costa, A.B., Wood, B.J., Tsai, C.S., et al.: Federated learning for predicting clinical outcomes in patients with covid-19. Nature medicine **27**(10), 1735–1743 (2021)
4. Geirhos, R., Jacobsen, J.H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., Wichmann, F.A.: Shortcut learning in deep neural networks. Nature Machine Intelligence **2**(11), 665–673 (2020)
5. Guo, Y., Guo, K., Cao, X., Wu, T., Chang, Y.: Out-of-distribution generalization of federated learning via implicit invariant relationships (2023)
6. Hamilton, W.L., Ying, R., Leskovec, J.: Representation learning on graphs: Methods and applications. arXiv preprint arXiv:1709.05584 (2017)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)

8. Hsu, T.M.H., Qi, H., Brown, M.: Measuring the effects of non-identical data distribution for federated visual classification. arXiv preprint arXiv:1909.06335 (2019)
9. Kaissis, G.A., Makowski, M.R., Rückert, D., Braren, R.F.: Secure, privacy-preserving and federated machine learning in medical imaging. Nature Machine Intelligence **2**(6), 305–311 (2020)
10. Kalra, S., Wen, J., Cresswell, J.C., Volkovs, M., Tizhoosh, H.: Decentralized federated learning through proxy model sharing. Nature communications **14**(1), 2899 (2023)
11. Karimireddy, S.P., Kale, S., Mohri, M., Reddi, S., Stich, S., Suresh, A.T.: Scaffold: Stochastic controlled averaging for federated learning. In: International conference on machine learning. pp. 5132–5143. PMLR (2020)
12. Koh, P.W., Sagawa, S., Marklund, H., Xie, S.M., Zhang, M., Balsubramani, A., Hu, W., Yasunaga, M., Phillips, R.L., Gao, I., et al.: Wilds: A benchmark of in-the-wild distribution shifts. In: International Conference on Machine Learning. pp. 5637–5664. PMLR (2021)
13. Li, T., Sahu, A.K., Zaheer, M., Sanjabi, M., Talwalkar, A., Smith, V.: Federated optimization in heterogeneous networks. Proceedings of Machine learning and systems **2**, 429–450 (2020)
14. Li, X., Gu, Y., Dvornek, N., Staib, L.H., Ventola, P., Duncan, J.S.: Multi-site fmri analysis using privacy-preserving federated learning and domain adaptation: Abide results. Medical Image Analysis **65**, 101765 (2020)
15. Li, X., Jiang, M., Zhang, X., Kamp, M., Dou, Q.: Fedbn: Federated learning on non-iid features via local batch normalization. arXiv preprint arXiv:2102.07623 (2021)
16. Liu, C., Lou, C., Wang, R., Xi, A.Y., Shen, L., Yan, J.: Deep neural network fusion via graph matching with applications to model ensemble and federated learning. In: International Conference on Machine Learning. pp. 13857–13869. PMLR (2022)
17. Ma, X., Zhang, J., Guo, S., Xu, W.: Layer-wised model aggregation for personalized federated learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10092–10101 (2022)
18. McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: Artificial intelligence and statistics. pp. 1273–1282. PMLR (2017)
19. Nguyen, A.T., Torr, P., Lim, S.N.: Fedsr: A simple and effective domain generalization method for federated learning. Advances in Neural Information Processing Systems **35**, 38831–38843 (2022)
20. Pati, S., Baid, U., Edwards, B., Sheller, M., Wang, S.H., Reina, G.A., Foley, P., Gruzdev, A., Karkada, D., Davatzikos, C., et al.: Federated learning enables big data for rare cancer boundary detection. Nature communications **13**(1), 7346 (2022)
21. Puli, A.M., Zhang, L., Wald, Y., Ranganath, R.: Don't blame dataset shift! shortcut learning due to gradients and cross entropy. Advances in Neural Information Processing Systems **36** (2024)
22. Qu, L., Balachandar, N., Rubin, D.L.: An experimental study of data heterogeneity in federated learning methods for medical imaging. arXiv preprint arXiv:2107.08371 (2021)
23. Qu, L., Zhou, Y., Liang, P.P., Xia, Y., Wang, F., Adeli, E., Fei-Fei, L., Rubin, D.: Rethinking architecture design for tackling data heterogeneity in federated learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10061–10071 (2022)

24. Ruder, S.: An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747 (2016)
25. Uddin, M.P., Xiang, Y., Yearwood, J., Gao, L.: Robust federated averaging via outlier pruning. IEEE Signal Processing Letters **29**, 409–413 (2021)
26. Wang, H., Yurochkin, M., Sun, Y., Papailiopoulos, D., Khazaeni, Y.: Federated learning with matched averaging. arXiv preprint arXiv:2002.06440 (2020)
27. Wang, Z., Grigsby, J., Qi, Y.: Pgrad: Learning principal gradients for domain generalization. arXiv preprint arXiv:2305.01134 (2023)
28. Xu, J., Tong, X., Huang, S.L.: Personalized federated learning with feature alignment and classifier collaboration. arXiv preprint arXiv:2306.11867 (2023)
29. Xu, J., Wang, S., Wang, L., Yao, A.C.C.: Fedcm: Federated learning with client-level momentum. arXiv preprint arXiv:2106.10874 (2021)
30. Yan, R., Qu, L., Wei, Q., Huang, S.C., Shen, L., Rubin, D., Xing, L., Zhou, Y.: Label-efficient self-supervised federated learning for tackling data heterogeneity in medical imaging. IEEE Transactions on Medical Imaging (2023)
31. Yang, C., Wang, Q., Xu, M., Chen, Z., Bian, K., Liu, Y., Liu, X.: Characterizing impacts of heterogeneity in federated learning upon large-scale smartphone data. In: Proceedings of the Web Conference 2021. pp. 935–946 (2021)
32. Zhang, J., Zeng, S., Zhang, M., Wang, R., Wang, F., Zhou, Y., Liang, P.P., Qu, L.: Flhetbench: Benchmarking device and state heterogeneity in federated learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12098–12108 (2024)
33. Zhang, M., Qu, L., Singh, P., Kalpathy-Cramer, J., Rubin, D.L.: Splitavg: A heterogeneity-aware federated deep learning method for medical imaging. IEEE Journal of Biomedical and Health Informatics **26**(9), 4635–4644 (2022)
34. Zhang, X., Sun, W., Chen, Y.: Tackling the non-iid issue in heterogeneous federated learning by gradient harmonization. arXiv preprint arXiv:2309.06692 (2023)
35. Ziller, A., Usynin, D., Braren, R., Makowski, M., Rueckert, D., Kaissis, G.: Medical imaging deep learning with differential privacy. Scientific Reports **11**(1), 13524 (2021)