

Federated Learning with Sample-level Client Drift Mitigation

Haoran Xu^{1*}, Jiaze Li^{1*}, Wanyi Wu², Hao Ren^{3†}

¹Zhejiang University, Hangzhou, China

²Shandong University, China

³School of Cyber Science and Engineering, Sichuan University, Chengdu 610065, China

xhaoran@zju.edu.cn, Jiaze_Li@zju.edu.cn,

wanyi_wu@mail.sdu.edu.cn, hao.ren@scu.edu.cn

Abstract

Federated Learning (FL) suffers from severe performance degradation due to the data heterogeneity among clients. Existing works reveal that the fundamental reason is that data heterogeneity can cause *client drift* where the local model update deviates from the global one, and thus they usually tackle this problem from the perspective of calibrating the obtained local update. Despite effectiveness, existing methods substantially lack a deep understanding of how heterogeneous data samples contribute to the formation of client drift. In this paper, we bridge this gap by identifying that the drift can be viewed as a cumulative manifestation of *biases* present in all local samples and the bias between samples is different. Besides, the bias dynamically changes as the FL training progresses. Motivated by this, we propose FedBSS that first mitigates the heterogeneity issue in a sample-level manner, orthogonal to existing methods. Specifically, the core idea of our method is to adopt a bias-aware sample selection scheme that dynamically selects the samples from small biases to large epoch by epoch to train progressively the local model in each round. In order to ensure the stability of training, we set the diversified knowledge acquisition stage as the warm-up stage to avoid the local optimality caused by knowledge deviation in the early stage of the model. Evaluation results show that FedBSS outperforms state-of-the-art baselines. In addition, we also achieved effective results on feature distribution skew and noise label dataset setting, which proves that FedBSS can not only reduce heterogeneity, but also has scalability and robustness.

Introduction

Federated Learning (FL), which enables collaborative training of models by sharing parameters among clients without exchanging raw data, has garnered significant attention for its ability to leverage vast amounts of data on clients while preserving privacy. The basic steps of FL are to iteratively execute local model training on multiple clients individually and subsequently aggregate all updated models at the server. Despite its implementation simplicity, the challenge arises from the inherently heterogeneous distribution of FL's training data across clients—characterized by non-identical and

independent (Non-IID) data—substantially deteriorating the performance of the obtained global model.

Many approaches have been devoted to addressing the data heterogeneity problem of FL. One of the representative categories is to consider that the essence of performance degradation is due to the *client drift* caused by the data heterogeneity where the local updates of each client greatly deviate from the aggregated global one (Karimireddy et al. 2020; Li et al. 2020; Zhang et al. 2022; Li and Zhan 2021; Gao et al. 2022; Louizos, Reisser, and Korzhnikov 2024). To mitigate this drift, many methods are proposed to calibrate the local update from the perspective of optimization (Gao et al. 2022; Karimireddy et al. 2020). For example, some methods leverages the difference between the local update and the global update in old rounds to compensate the local update in current round (Karimireddy et al. 2020; Gao et al. 2022) and other works add regularization on the local loss function to facilitate the local update to approach the global one (Zhang et al. 2022; Li and Zhan 2021; Louizos, Reisser, and Korzhnikov 2024). Besides a few works explore different aggregation strategies on the server (Chan et al. 2024).

Although these approaches have made great achievements, they lack a deep understanding of *how the heterogeneous data samples contribute to the formation of client drift*.

In this paper, we seek to tackle the Non-IID challenge of FL from the sample level by delving into the impact of each local sample on the drift of local update. Specifically, we identify that **there exists substantial bias in each sample and the client drift can be viewed as a cumulative manifestation of the biases of all samples**. Besides, the bias between samples is different from each other where the sample with large loss is considered to have a relatively large impact on the client drift. Also, we find that the bias of each sample dynamically changes during the FL training process.

Motivated by this finding, we propose a simple yet effective method, FedBSS, which includes two stages: Diversified knowledge acquisition stage and Progressive knowledge learning stage. The diversified knowledge acquisition phase acts as a model's cold start, warming it up by introducing varied knowledge through standard aggregation techniques. This step prevents early-stage deviations caused by limited knowledge scope, essentially initializing the global model with a diverse set of preliminary information guided by the

*These authors contributed equally.

†The corresponding author.

these related works are elaborated upon as follows.

Calibrating Local Update Previous research has tackled the challenge of Non-IID data in federated learning by calibrating the local model updates. Works such as SCAF-FOLD (Karimireddy et al. 2020) address client drift by employing control variates for gradients, albeit without directly considering inconsistencies between local and global objectives. FedDC (Gao et al. 2022), in a different approach, utilizes auxiliary drift variables to monitor and mitigate discrepancies between local and global model parameters. Regarding statistical heterogeneity, FedProx (Li et al. 2020) integrates an additional term into the local model’s objective function, suggesting that over-updating the local model could hinder global convergence. This proximal term serves as a penalty, discouraging significant deviations of the local model from the global one. Another strategy, presented by FedLC (Zhang et al. 2022), introduces a fine-tuned calibrated cross-entropy loss for local updates. It does so by incorporating a pairwise label margin, enhancing the sensitivity of the learning process to inter-class differences. Similarly, FedRS (Li and Zhan 2021) addresses the challenge of missing classes by proposing a ”Restricted Softmax” mechanism, which constrains the updating of weights associated with missing classes during local training. Fed-Dyn (Acar et al. 2021b), on the other hand, introduces a dynamic regularizer tailored to each client to align individual client models with the global model, thereby reducing communication overhead. More recently, integrating contrastive learning and mutual information into the loss function has emerged as another optimization strategy, as illustrated by Louizos et al. (Louizos, Reisser, and Korzhnikov 2024). Strategies such as incorporating additional regularizers into loss functions or leveraging gradient control mechanisms are all aimed at mitigating the drift of local models.

Aggregation Strategy The parameters of client models are inherently prone to drift, making server aggregation strategies another pivotal approach to mitigating the effects of Non-IID data. One such strategy, InCo (Chan et al. 2024) Aggregation, harnesses internal cross-layer gradients—combining gradients from both shallow and deep layers within a server model—to enhance similarity in the deeper layers without necessitating extra client-server communication. Recently, FedCDA (Wang et al. 2024a), on the other hand, employs a selective aggregation of cross-round local models, effectively minimizing disparities between the global and local models.

Problem Formulation

In this setting, we have N clients with their private datasets $\mathcal{D}_n = \{(\mathbf{x}_i, y_i)\}_{i=1}^{S_n}$ where \mathbf{x}_i is the data sample, y_i is its label and S_n is the number of training sample on n -th client. The objective of federated learning framework is to learn the global model parameter ϑ which minimizes the loss function $F(\vartheta)$ on training data of all clients without access to original data:

$$\min_{\vartheta \in \mathbb{R}^d} F(\vartheta) = \frac{1}{N} \sum_{n=1}^N F_n(\vartheta) \quad (1)$$

$$F_n(\vartheta) = \mathbb{E}_{\mathbf{x} \sim \mathcal{D}_n} f_n(\vartheta; \mathbf{x}) \quad (2)$$

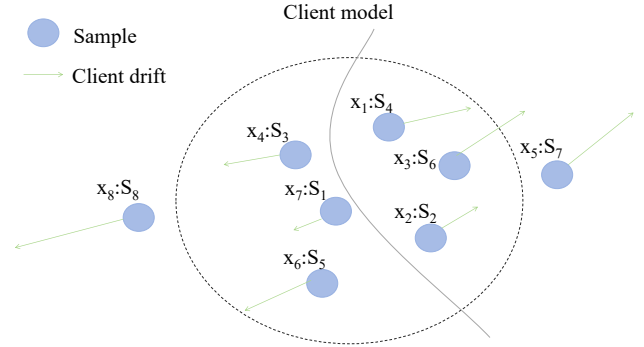


Figure 2: Different samples on client have a different degree of drift to the model.

$f_n(\vartheta; \mathbf{x})$ denotes the loss value with respect to model ϑ and random data sample \mathbf{x} .

Motivation

In this section, we revisit the principles of client drift and ask three questions:

Q1: What is the impact of different local samples on client drift?

This question stems from that previous studies did not study the impact of local samples on client drift. We investigate this by profiling the training process of clients over local client samples and identify that different client samples have different drift degrees to the training of the local model. Figure 2 shows the above motivation. In fact, this naturally holds because the loss of the global model is different for each client sample where loss is a measure of the gap between the knowledge of local data and the knowledge of the global data. Therefore, we take loss value as the basis to measure the degree of the sample’s influence on the client drift. To verify the above viewpoint, we track the training progress of different samples on a single client with ResNet-18 (He et al. 2016) model as an example. Based on loss, we sort client samples and extract the top 50% sample, the bottom 50% sample and all the samples respectively for training. Figure 3a is divided into four parts, which are the origin sample, 100% sample training, top 50% sample training and bottom 50% sample training according to the order of top left, top right, bottom left and bottom right. We can draw the following conclusion: the degree of the client model drift is also small when the loss of samples is small and vice versa.

Therefore, we can partition the client sample into two sets based on the loss value, namely, the set of biased samples and unbiased samples which indicates samples contribute large and small to the degree of model drift separately.

Q2: How to identify biased and unbiased samples from local samples?

A natural observation is that the loss of each sample in different rounds varies, causing a fixed loss threshold or ratio-based biased sample identification scheme to not apply to the scenario, which can also be observed from Figure 3c. To solve this, inspired by previous work (Fuchsguber et al. 2024; Zhu et al. 2008) **we identify that the uncertainty, i.e.,**

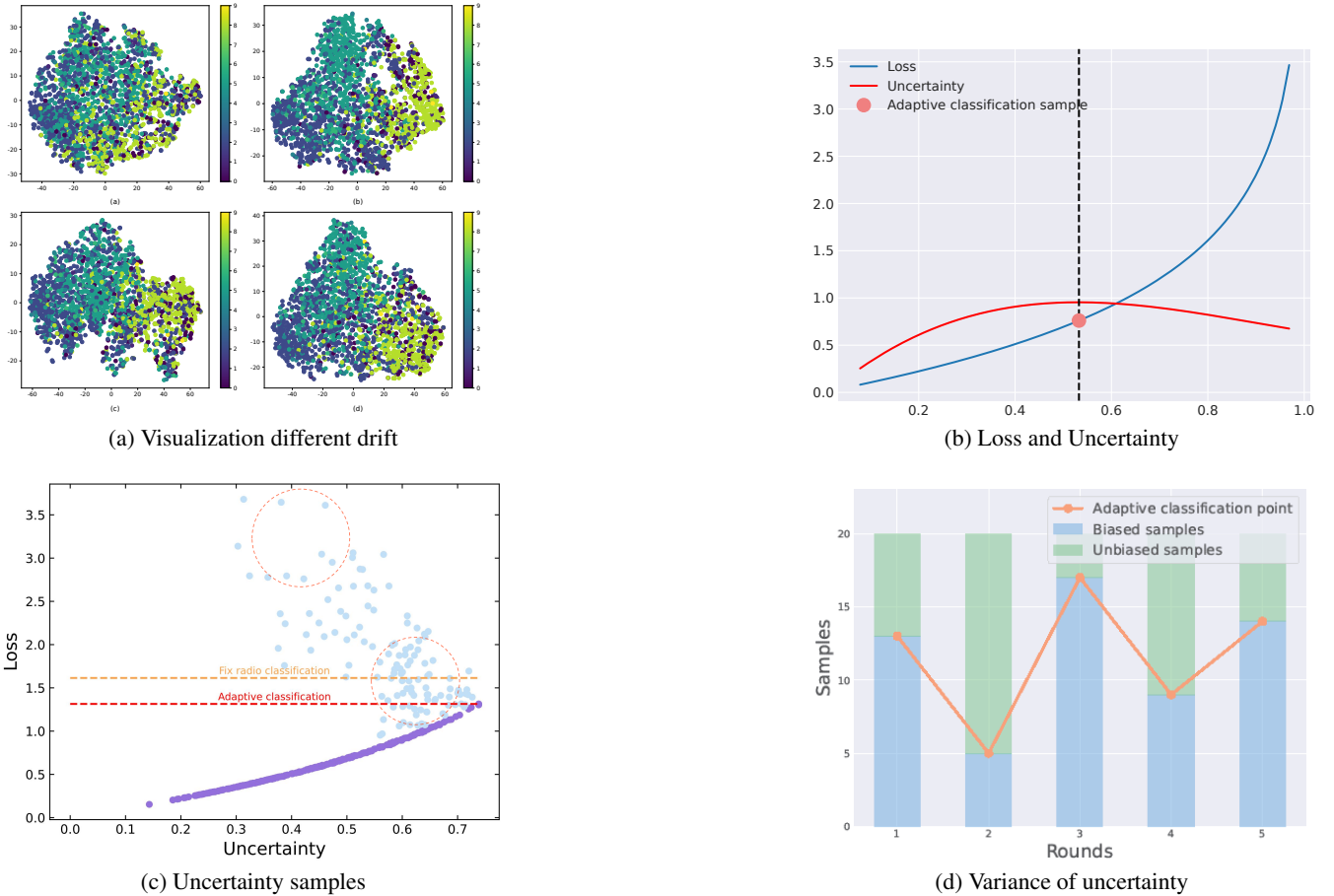


Figure 3: (a) Different impacts of various local samples on model drift. (b) The relationship between loss and uncertainty changes. (c) The relationship between uncertainty and loss as local samples vary. (d) The abrupt changes in the adaptive classification points in each round when a diversified knowledge acquisition stage is absent.

the confidence of the model on the probability of sample classification, can adaptively give the partition boundary of unbiased samples and biased samples.

Q3: How to mitigate the client drift with identified sample bias?

The core idea is that unbiased samples can reduce the drift degree of the model and increase the stability of training compared with biased samples. The intuitive idea is to filter out the biased sample part during client model training and only adopt the unbiased samples. However, the limit is that biased samples, as a part of all training samples, often contain information necessary for model training. Directly filtering out biased samples for training will affect the model’s knowledge acquisition of this part of the data, and ultimately affect the model’s performance. *To this end, the method should satisfy two requirements: 1. minimized bias to reduce client drift; 2. minimized knowledge sacrifice.* Built upon the above analysis, we introduce a progressive learning concept, which gradually adds biased samples to the local training process. The intuition behind this is that when we gradually add biased samples, we can gradually guide the model from local to global stability to learn the

knowledge in biased samples.

Methodology

Motivated by the above findings, we propose a novel and fine-grained framework named FedBSS which is illustrated in Figure 1. Our method FedBSS has the following innovations.

First, considering that different samples on client have a different degree of drift to the local updates, we evaluate and sort the loss set S_n of all the samples on n th client by global model from the server as follows:

$$S_n = \{(x_i^n, S_k^n)\}_{i=1}^{N_n}, \quad \text{with } S_i^n < S_j^n \text{ if } i < j \quad (3)$$

where S_n is the evaluation loss set of n -th client and S_k^n denotes the bottom k th loss in all sample from n th client. Second, in order to divide all sample into biased samples and unbiased sample from n th client, we need to set a classification sample S_{mid}^n which when the loss value $S_i^n \leq S_{mid}^n$, x_k^n sample corresponding to the loss value S_i^n is unbiased sample while the rest is biased samples. However, as mentioned

above, different rounds, different clients, and different samples will lead to different thresholds of classification points. Thus, we introduce the uncertainty to adaptively set the classification point for all clients on each round:

$$\alpha(x_i^n; \vartheta) = 1 - (\max(p(x_i^n; \vartheta)) - \min(p(x_i^n; \vartheta))) \quad (4)$$

where $\alpha(x_i^n; \vartheta)$ denotes the uncertainty of the model ϑ for the sample x_i^n on n th client.

We plot the relationship between loss and uncertainty in the model’s assessment of the sample as Figure 3b. With the increase of sample loss value, the model’s perception of the sample is essentially divided into three stages from figure. Specifically, the model’s judgment of sample is certain and accurate, the model’s judgment of sample is uncertain and inaccurate, and the model’s judgment of sample is certain but inaccurate. Based on the above properties, we can adaptively divide the sample into biased and unbiased sample by the highest uncertainty. Therefore, we take the sample with the highest uncertainty as the classification point at the beginning of client model training:

$$\alpha(S_{\text{mid}}^n) = \max(\alpha(x_i^n; \vartheta)) \quad (5)$$

Then, we propose progressively adding biased samples along during local training process in a epoch-by-epoch manner as follows:

$$\mathbf{X}_{t,e}^n = \mathbf{X}_{no}^n + \alpha \mathbf{X}_{bias}^n, \quad \alpha = \frac{1 - \cos\left(\frac{e}{E_{\text{total}}}\pi\right)}{2} \quad (6)$$

where $\mathbf{X}_{t,e}^n$ denotes the n th client training sample on client epoch e and global communication round t , \mathbf{X}_{no}^n is the unbiased sample on n th client, \mathbf{X}_{bias}^n is biased samples on n th client, E_e is current client training epoch and E_{total} is total epoch in the client training. Note that these samples are in the global round communication on the n th client.

One detail is that we do not adopt the scheme of linearly increasing biased samples epoch by epoch, such as $\mathbf{X}_{t,e}^n = \mathbf{X}_{no}^n + \frac{e}{E_{\text{total}}}\mathbf{X}_{bias}^n$. The specific comparison results are in section . Concretely, we can discover that the samples near the classification point are relatively dense, while the samples at the edge of loss are relatively scattered, and the span of loss is large from figure 3c. This may result in the linear increment method not fitting the sample curve well. More discussion is demonstrated in section .

In addition to the above, we also need an extra first stage as the warmup. This is because the model’s initial judgment of the point of uncertainty fluctuates greatly. Figure 3d has shown the point. The intuition behind this is that the initial knowledge of the model is not enough, and it cannot make a relatively stable judgment on the client sample. Thus in order to ensure the stability of the initial training stage, warmup is used as the cold start stage of the model, which is to enable the model to acquire more diversified knowledge at the initial stage. Note that Figure 1 only describes the second stage.

In summary, our algorithm is divided into two stages: the first stage is Diversified knowledge acquisition stage, at which the client normally selects all the samples. And

the second stage is Progressive knowledge learning stage. At this time, the algorithm first divides adaptive samples according to our above description, and then progressively adds unbiased samples during local client training.

Experiments

Experiment Setting

Datasets and models. We evaluate the performance of the proposed FedBSS over two models and four mainstream datasets. In Non-IID setting with label distribution skew, we consider Fashion-MNIST (Xiao, Rasul, and Vollgraf 2017), CIFAR-10 (Krizhevsky, Hinton et al. 2009) and CIFAR-100 (Krizhevsky, Hinton et al. 2009), which contains 10, 10, 100 classes respectively. For CIFAR-10 and CIFAR-100 datasets, we use ResNet-50 (He et al. 2016) as the backbone to train and test the performance while for Fashion-MNIST we use a simple CNN instead. The simple CNN has two 5x5 convolution layers (the first with 32 channels, the second with 64, each followed with 3x3 max pooling), a fully connected layer with 512 units and ReLU activation. In Non-IID setting with feature distribution skew, we utilize DomainNet dataset.

Data Partition. Follow the setting (Kairouz et al. 2021), we adopt two classic data partitioning strategies, namely label distribution skew and feature distribution skew.

- **Label Distribution Skew:** To evaluate the performance of our work in a heterogeneous scenario with label distribution skew, we specify two Non-IID data partition methods called Shards (McMahan et al. 2017) and Dirichlet (Lin et al. 2020). In the Shards setting, the sorted samples are shuffled into $N * S$ shards, and assigned to N clients randomly. Each client owns an equal number of pieces. In the second setting, data distribution over clients satisfies the Dirichlet distribution by using α to characterize the degree of heterogeneity. We set α of Dirichlet: $\{0.1, 0.3, 0.5\}$ and shards for each client: $\{2, 4, 8\}$.
- **Feature Distribution Skew:** In this setting, clients share the same label space while different feature distribution, which has been extensively studied in previous works (Li et al. 2021; Peng et al. 2019; Yang et al. 2023). We conduct the classification task on natural images sourced from DomainNet (Peng et al. 2019), which consists of diverse distributions of images from six distinct data sources. we utilize all six distinct data domains. We selected ten categories—airplane, clock, axe, ball, bicycle, bird, strawberry, flower, pizza, and bracelet—for the classification task. Each client operates exclusively with the data from one domain, and in this setting, all six clients participated in the aggregation of the model.

Baselines. Beside of FedAvg, we also compare against various types of Non-IID federated learning approaches with the proposed method in our experiments. The first main type includes typical non-aggregation methods that calibrate local update, including Scaffold (Karimireddy et al. 2020), FedProx (Li et al. 2020), FedExP (Jhunjhunwala, Wang, and Joshi 2023), FedLC (Zhang et al. 2022) and FedRS (Li and Zhan 2021). Besides, another representative strategy is to se-

Method	Fashion-MNIST(%)			CIFAR-10(%)			CIFAR-100(%)		
Shards (S)	2	4	8	2	4	8	2	4	8
FedAvg	62.68±6.30	67.31±4.10	73.64±2.16	23.56±3.54	32.17±2.52	36.18±2.25	2.99±0.26	4.95±0.37	7.95±0.76
FedProx	63.42±5.23	68.71±3.55	72.58±3.54	20.65±1.84	27.18±3.13	31.89±1.17	2.34±0.21	4.47±0.48	6.30±0.62
FedExP	54.84±4.80	65.58±3.54	70.19±4.22	21.74±2.74	29.90±1.83	35.86±2.39	2.71±0.21	5.08±0.68	8.40±0.42
FedLC	64.67±0.25	74.20±0.39	76.64±0.44	21.28±0.23	25.19±0.35	32.07±0.27	4.33±0.09	7.06±0.11	9.18±0.12
FedRS	62.17±0.25	66.23±0.69	68.64±0.27	19.28±0.13	23.47±0.17	28.66±0.32	4.01±0.07	7.32±0.10	8.80±0.08
Scaffold	65.72±1.11	75.16±1.33	82.51±0.99	23.37±0.35	27.08±5.67	31.52±2.67	5.01±0.24	6.20±0.12	8.18±0.32
FedCDA	66.47±0.12	74.81±0.17	80.54±0.09	22.33±0.12	31.38±0.21	36.04±2.34	5.52±0.13	5.06±0.32	7.51±0.57
InCo	65.66±4.85	73.32±2.09	75.68±1.53	23.21±0.54	29.94±0.70	31.12±0.65	6.02±0.07	9.11±0.24	10.11±0.15
Ours	67.10±0.07	76.01±0.33	79.31±0.87	25.97±0.33	32.81±0.28	34.04±0.23	6.21±0.13	10.45±0.26	10.96±0.31
Dirichlet (α)	0.1	0.3	0.5	0.1	0.3	0.5	0.1	0.3	0.5
FedAvg	69.48±2.59	74.50±2.07	76.41±0.94	27.72±1.73	35.07±1.80	37.88±1.58	11.79±0.34	14.00±0.35	14.50±0.18
FedProx	61.39±6.17	70.65±3.38	73.87±2.37	23.36±2.57	26.86±2.92	30.16±2.47	7.99±0.55	10.70±0.33	11.03±0.18
FedExP	68.08±4.56	74.29±2.11	77.14±1.17	27.78±3.44	35.00±1.93	39.06±1.24	11.61±0.52	13.53±0.29	14.54±0.25
FedLC	67.11±0.31	69.27±0.16	74.19±1.09	27.99±0.28	32.42±1.25	40.76±1.02	11.88±0.26	13.00±0.12	14.17±0.58
FedRS	62.67±0.25	64.20±0.09	68.64±0.86	26.34±0.23	30.47±0.17	38.07±0.67	14.02±0.12	15.06±0.11	16.77±0.37
Scaffold	73.43±1.76	74.95±0.76	76.96±0.69	14.54±2.22	18.38±1.43	19.26±2.63	12.96±0.18	13.20±0.26	14.20±0.40
FedCDA	70.66±4.85	71.11±1.34	73.68±1.24	29.21±0.15	38.94±0.34	39.68±0.04	12.34±0.42	14.81±0.14	15.44±0.16
InCo	69.66±1.13	73.32±2.09	75.68±1.53	28.21±4.55	36.94±2.71	38.69±1.65	11.52±0.67	13.81±0.24	14.22±0.15
Ours	76.24±0.16	80.34±0.15	82.11±0.17	38.49±0.22	43.11±0.25	45.27±0.19	18.37±0.27	19.11±0.12	19.08±0.11

Table 1: The comparison of test accuracy of different methods. The best results are **bolded**.

Method	Fashion-MNIST(%)			CIFAR-10(%)			CIFAR-100(%)		
Noisy Label Ratio	0.1	0.3	0.5	0.1	0.3	0.5	0.1	0.3	0.5
FedAvg	80.19±0.13	78.44±0.11	75.28±0.31	42.15±0.27	36.78±0.42	28.23±0.35	13.70±0.16	11.13±0.23	8.08±0.16
FedProx	79.94±0.10	78.17±0.15	75.60±0.27	37.96±0.24	32.67±0.20	26.30±0.53	10.76±0.27	8.75±0.15	6.39±0.14
FedExP	80.27±0.11	78.28±0.13	75.74±0.25	41.82±0.18	36.60±0.24	29.32±0.50	13.44±0.23	11.11±0.18	7.51±0.24
FedLC	80.11±0.11	78.18±0.09	76.98±0.12	39.23±0.26	35.14±0.30	30.41±0.18	15.08±0.39	11.34±0.21	10.39±0.17
FedRS	79.86±0.22	77.59±0.11	74.88±0.13	37.88±0.24	34.28±0.25	27.68±0.38	12.09±0.18	10.09±0.17	9.21±0.28
Scaffold	78.65±0.09	78.88±0.09	76.40±0.17	40.72±0.36	37.14±0.19	29.36±0.41	12.37±0.24	9.94±0.15	7.39±0.06
FedCDA	79.63±0.15	78.20±0.12	75.74±0.30	39.83±0.15	35.27±0.21	30.49±0.28	14.18±0.22	10.33±0.21	9.10±0.21
InCo	79.89±0.56	77.27±0.60	75.16±0.46	38.12±1.13	34.55±0.99	24.78±1.75	11.02±0.34	9.54±0.29	8.28±0.54
Ours	80.63±0.14	79.91±0.13	78.88±0.08	44.46±0.22	40.27±0.29	38.96±0.19	16.01±0.17	15.47±0.28	12.13±0.34

Table 2: The comparison of test accuracy on noisy label datasets. The best results are **bolded**.

lect aggregation on the sever, such as InCo (Chan et al. 2024) and FedCDA (Wang et al. 2024a).

Implementation Details. We implement the whole experiment in a simulation environment based on RTX 3090 GPUs. We use 100 clients in total and randomly choose 10% each round for local training. We set the local epoch to 10, batch size to 64, and learning rate to $1e - 3$. We employ SGD optimizer with momentum of $1e - 4$ and weight decay of $1e - 5$ for all methods and datasets. At the same time, we set the number of global communication rounds to 200. For evaluation, we compute the average accuracy and standard deviation over the final 10 rounds of each run. For our method, we set the number of warmup rounds to 50.

Result with Label Distribution Skew

We report the comparison results with other baselines in Table 1. In order to demonstrate the generalization of our method, we compare them on two different Non-IID set-

tings, Shards and Dirichlet distribution. We apply different data distributions on different datasets. We can see that our proposed FedBSS achieves the best performance on almost all settings. It demonstrates the effectiveness and benefit of FedBSS. Specifically, on relatively larger datasets such as CIFAR-10 Dirichlet 0.5, FedBSS with ResNet-50 achieves 45.27% accuracy whereas the best baseline method FedLC achieves 40.76% accuracy. In addition, FedBSS with simple CNN also makes improvements on relatively smaller datasets, and the improvement is not less than in large models. At the same time, we can also see that the results of our method on relatively small datasets and simple CNN are not the best, which may be because the client biased samples of different rounds is less drift to the global model on small datasets and simple models, and can not provide better performance of the global model by progressive learning. In conclusion, we can notice our FedBSS makes more improvements on the large model, complex datasets than small

model, simple datasets.

Note that the baseline approaches we compare can actually be divided into two categories. One is calibrating local update, and the other is setting aggregating strategy. The performance of InCo in the table is not good enough. The main reason is that its perception of the client is relatively limited, so the improvement is small. Furtherly, we can see that scaffold outperforms our method FedBSS on some datasets, while it has not the good performance on some other datasets. This indicates that Scaffold is not stable and always sensitive to datasets and hyperparameters. The performance of our method FedBSS is usually more prominent when there is high heterogeneity setting. This may be because the biased samples drifts the model to a large extent when there is high heterogeneity. Therefore, our method can more effectively improve its performance in such scenarios. Besides, our experimental results show that the server aggregating strategy-based methods also improve less on small models and simple datasets. We consider that the main reason is that small models have little parameter, which the effect of directly improving the model parameters is not obvious in this scenario.

Result with Feature Distribution Skew

In the above experiments, we define data heterogeneity with label distribution skew. In this section, we test our approach on feature Non-IID setting with feature distribution skew. We utilize domainNet dataset which contains six domains: clipart, infograph, painting, quickdraw, real, and sketch. We selected ten categories for the classification task and ResNet-18 as backbone. Each client operates exclusively with the data from one domain, and in this setting, all six clients participated in the aggregation of the model. Figure 4 is the result. We can find that our approach FedBSS outperforms FedAvg’s performance not only on six very different domains, but also on the final global model.

Result on Noise Label Datasets

In real scenarios, data labels are often noisy or absent. In this section, we employ FedBSS on noise label dataset to test the robustness of our method. Table 2 shows results of our method FedBSS. We set different noise label ratio which is 0.1, 0.3 and 0.5 on three datasets. We can find that our method achieve better performance compared with other Non-IID methods. Specifically, our approach offers a significant improvement over other approaches for both relatively smaller dataset Fashion-MNIST and relatively larger datasets CIFAR-100. With the increase of noise label ratio, our method improves more and more significantly, while FedBSS can still maintain good performance even under the most severe noise label ratio condition which is 0.5. It proves that our approach is robust and scalable.

More discussion and experiments

In this section, we discuss two parts which are mentioned on section in our method FedBSS. The first one is about first warmup stage. The second one is about sample selection strategy.

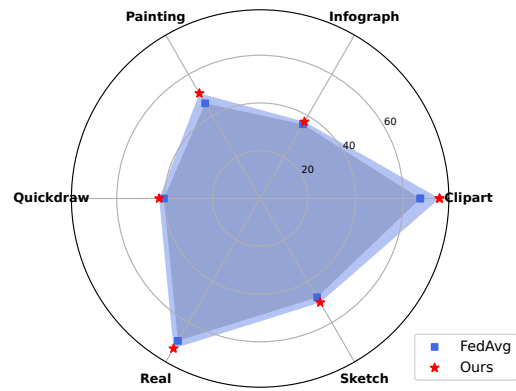


Figure 4: Result on DomainNet dataset

Warmup Stage For the first one, we compare with three baselines. Our method sets the warmup rounds as 50. We compare with the baseline which varies the warmup rounds as $\{0, 25, 100\}$. We can see that our method which sets the warmup rounds as 50. Note that this does not mean that 50 rounds is always optimal. However, we can discover that when the warmup round is zero which means the first warmup stage doesn’t exist, the performance of the model is always worse. This may be due to the lack of diversity knowledge in the first stage. Certainly, when warmup rounds are too high, the performance is worse too. This is because the first stage is too long, resulting in too many global model drifts, which makes the progressive learning effect of the second stage not good. Therefore, the warmup rounds are a relatively sensitive parameter. We think the first warmup phase is a good trade-off between minimizing client drift and minimizing knowledge sacrifice. Specifically, the first phase helps the model acquire diversified knowledge, while the second phase mitigated the phenomenon of client draft.

Sample Selection Strategy We have discussed the biased sample selection strategy in section . To summarize, there are three strategies: filter, linearly addition and our method. We compared these three strategies. Our method proves the superiority and it outperforms the other two strategies. It proves that filtering biased samples may cause the model to lose part of its knowledge and fall into local optimal solutions or slow convergence. On the other hand, the effect of linear inclusion of biased sample is not good, which may be due to the sample distribution, which is more concentrated in the middle part and dispersed in the edge part. This may result in a linear function that does not fit well.

Conclusion

This paper focuses on Non-IID setting in federated learning. We propose a novel two-stage method for FL called FedBSS that exploits the drift properties of different samples. Our method is to adopt a bias-aware sample selection scheme that dynamically selects the samples from small biases to large epoch by epoch to train progressively the local model in each round. We demonstrate its effectiveness and robustness.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (NSFC Grant No. 62402331), the Fundamental Research Funds for the Central Universities (Grant No. YJ202429), the Fundamental Research Funds for the Central Universities (No. SCU2024D012), and the Science and Engineering Connotation Development Project of Sichuan University (No. 2020SCUNG129).

References

- Acar, D. A. E.; Zhao, Y.; Navarro, R. M.; Mattina, M.; Whatmough, P. N.; and Saligrama, V. 2021a. Federated Learning Based on Dynamic Regularization. arXiv:2111.04263.
- Acar, D. A. E.; Zhao, Y.; Navarro, R. M.; Mattina, M.; Whatmough, P. N.; and Saligrama, V. 2021b. Federated Learning Based on Dynamic Regularization. arXiv:2111.04263.
- Chan, Y.-H.; Zhou, R.; Zhao, R.; Jiang, Z.; and Ngai, E. C. H. 2024. Internal Cross-layer Gradients for Extending Homogeneity to Heterogeneity in Federated Learning. arXiv:2308.11464.
- Fuchsgruber, D.; Wollschläger, T.; Charpentier, B.; Oroz, A.; and Günemann, S. 2024. Uncertainty for Active Learning on Graphs. arXiv:2405.01462.
- Gao, L.; Fu, H.; Li, L.; Chen, Y.; Xu, M.; and Xu, C.-Z. 2022. FedDC: Federated Learning with Non-IID Data via Local Drift Decoupling and Correction. In *IEEE Conference on Computer Vision and Pattern Recognition*.
- He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778. IEEE.
- Jhunjunwala, D.; Wang, S.; and Joshi, G. 2023. FedExp: Speeding Up Federated Averaging via Extrapolation. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5*.
- Kairouz, P.; McMahan, H. B.; Avent, B.; Bellet, A.; Bennis, M.; Bhagoji, A. N.; Bonawitz, K.; Charles, Z.; Cormode, G.; Cummings, R.; D’Oliveira, R. G. L.; Rouayheb, S. E.; Evans, D.; Gardner, J.; Garrett, Z.; Gascón, A.; Ghazi, B.; Gibbons, P. B.; Gruteser, M.; Harchaoui, Z.; He, C.; He, L.; Huo, Z.; Hutchinson, B.; Hsu, J.; Jaggi, M.; Javidi, T.; Joshi, G.; Khodak, M.; Konečný, J.; Korolova, A.; Koushanfar, F.; Koyejo, S.; Lepoint, T.; Liu, Y.; Mittal, P.; Mohri, M.; Nock, R.; Özgür, A.; Pagh, R.; Raykova, M.; Qi, H.; Ramage, D.; Raskar, R.; Song, D.; Song, W.; Stich, S. U.; Sun, Z.; Suresh, A. T.; Tramèr, F.; Vepakomma, P.; Wang, J.; Xiong, L.; Xu, Z.; Yang, Q.; Yu, F. X.; Yu, H.; and Zhao, S. 2021. Advances and Open Problems in Federated Learning. *Foundations and Trends® in Machine Learning*, 14(1).
- Karimireddy, S. P.; Kale, S.; Mohri, M.; Reddi, S.; Stich, S.; and Suresh, A. T. 2020. SCAFFOLD: Stochastic Controlled Averaging for Federated Learning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 5132–5143. PMLR.
- Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.
- Lee, G.; Jeong, M.; Shin, Y.; Bae, S.; and Yun, S.-Y. 2022. Preservation of the Global Knowledge by Not-True Distillation in Federated Learning. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 38461–38474. Curran Associates, Inc.
- Li, Q.; He, B.; and Song, D. 2021. Model-Contrastive Federated Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- Li, T.; Sahu, A. K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; and Smith, V. 2020. Federated Optimization in Heterogeneous Networks. In *Proceedings of Machine Learning and Systems 2020, MLSys 2020, Austin, TX, USA, March 2-4, 2020*.
- Li, X.; Jiang, M.; Zhang, X.; Kamp, M.; and Dou, Q. 2021. Fed{BN}: Federated Learning on Non-{IID} Features via Local Batch Normalization. In *International Conference on Learning Representations*.
- Li, X.-C.; and Zhan, D.-C. 2021. FedRS: Federated Learning with Restricted Softmax for Label Distribution Non-IID Data. KDD ’21, 995–1005. ISBN 9781450383325.
- Lin, T.; Kong, L.; Stich, S. U.; and Jaggi, M. 2020. Ensemble distillation for robust model fusion in federated learning. *Advances in Neural Information Processing Systems*, 33: 2351–2363.
- Louizos, C.; Reisser, M.; and Korzhenkov, D. 2024. A Mutual Information Perspective on Federated Contrastive Learning. arXiv:2405.02081.
- McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS*.
- Peng, X.; Bai, Q.; Xia, X.; Huang, Z.; Saenko, K.; and Wang, B. 2019. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, 1406–1415.
- Wang, H.; Xu, H.; Li, Y.; Xu, Y.; Li, R.; and Zhang, T. 2024a. FedCDA: Federated Learning with Cross-rounds Divergence-aware Aggregation. In *The Twelfth International Conference on Learning Representations*.
- Wang, H.; Zheng, P.; Han, X.; Xu, W.; Li, R.; and Zhang, T. 2024b. FedNLR: Federated Learning with Neuron-wise Learning Rates. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD ’24*, 3069–3080. Association for Computing Machinery. ISBN 9798400704901.
- Xiao, H.; Rasul, K.; and Vollgraf, R. 2017. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. *CoRR*, abs/1708.07747.
- Yang, Z.; Zhang, Y.; Zheng, Y.; Tian, X.; Peng, H.; Liu, T.; and Han, B. 2023. FedFed: Feature Distillation against Data Heterogeneity in Federated Learning. In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Zhang, J.; Li, Z.; Li, B.; Xu, J.; Wu, S.; Ding, S.; and Wu, C. 2022. Federated Learning with Label Distribution Skew via

Logits Calibration. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, 26311–26329. PMLR.

Zhu, J.; Wang, H.; Yao, T.; and Tsou, B. K. 2008. Active Learning with Sampling by Uncertainty and Density for Word Sense Disambiguation and Text Classification. In Scott, D.; and Uszkoreit, H., eds., *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, 1137–1144. Manchester, UK: Coling 2008 Organizing Committee.

Zhu, Z.; Hong, J.; and Zhou, J. 2021. Data-Free Knowledge Distillation for Heterogeneous Federated Learning. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 12878–12889. PMLR.