FedRDA: Representation Deviation Alignment in Heterogeneous Federated Learning

Wenjie Yao, Guanglu Sun, Suxia Zhu, Ruidong Wang, Xinzhong Zhu, *Member, IEEE*, HuiYing Xu, *Member, IEEE*, and Xiguang Wei

***Abstract*—Federatedlearning has garnered significant attention in the Internet of Things and healthcare applications due to its ability to train a shared global model across distributed clients. However, imbalanced data distribution leads to model discrepancies among clients. Most existing methods adopt implicit alignment strategies while overlooking explicit modeling of geometric and directional discrepancies in feature representations, which undermines local model optimization. To address this issue, we propose a method of representation deviation alignment in federated learning, which projects features onto the principal feature space to measure deviations between local and global feature representations explicitly. Specifically, Federated learning with Representation Deviation Alignment (FedRDA) employs a feature encoder to extract compact features and construct unbiased principal feature spaces for global and local models. Then, the residual projection in the feature space serves as a quantitative measure of the representation deviation, effectively capturing the latent direction differences between models. Besides, we introduce a representation consistency alignment strategy, which ensures that the distribution of local client features becomes more uniform within the global feature space. Extensive experiments on SVHN, CIFAR-10, CIFAR-100, Tiny-ImageNet, and GC10 demonstrate that FedRDA effectively reduces the classifier bias caused by representational differences.**

***Index Terms*—Data heterogeneity, feature deviation, federated learning, representation learning.**

Received 27 April 2025; revised 24 June 2025; accepted 30 June 2025. Date of publication 24 July 2025; date of current version 17 October 2025. This work was supported in part by the Key Research and development project of Heilongjiang Province under Grant JD2023SJ20, in part by the Heilongjiang Province “Double First-Class” Discipline Collaborative Innovation Project under Grant LJGXCG2024-P17, in part by the Key Project of Natural Science Foundation of Zhejiang Province under Grant LZ22F030003, in part by the Zhejiang Province Leading Geese Plan under Grant 2024C02G1123882, and in part by the National Natural Science Foundation of China under Grant 62376252. Paper no. TII-25-2721. *(Corresponding authors: Suxia Zhu; Xinzhong Zhu.)*

Wenjie Yao, Guanglu Sun, and Suxia Zhu are with the School of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China (e-mail: 2020410067@stu.hrbust.edu.cn; sunguanglu@hrbust.edu.cn; zhusuxia@hrbust.edu.cn).

Ruidong Wang, Xinzhong Zhu, and HuiYing Xu are with the School of Computer Science and Technology, Zhejiang Normal University, Jinhua 321004, China (e-mail: iswangrd@zjnu.edu.cn; zxz@zjnu.edu.cn; xhy@zjnu.edu.cn).

Xiguang Wei is with the Department of Data Science and Intelligent Development, AIMS SciTech, Hefei 230000, China (e-mail: weixiguang@aimstek.cn).

Digital Object Identifier 10.1109/TII.2025.3588630

I. Introduction

FEDERATED learning is a distributed training paradigm where models are trained locally on client devices, and their parameters are aggregated on a central server. It has been widely applied in fields, such as medical image analysis, the Internet of Things, and mobile services [1], [2]. However, data heterogeneity across clients often leads to nonindependent and identically distributed (Non-IID) data, which poses challenges for optimizing the global model. As shown in Fig. 1, low-dimensional representations of the same input may vary across different clients, a phenomenon commonly referred to as client drift or classifier bias [3].

Recently, various heterogeneous federated learning methods have been proposed. Most of these methods focus on feature information sharing among clients [4], [5] and adopt feature space alignment techniques to mitigate heterogeneity. However, as the number of participating clients increases, these methods often lead to significant resource overhead [6]. On the other hand, in scenarios with extreme heterogeneity among clients, substantial feature differences make it difficult for the global model to converge. Furthermore, anchor-based methods attempt to align feature spaces by sharing identical feature anchors across clients. Nevertheless, these methods suffer from slow global convergence rates [7], [8].

Although existing heterogeneous federated learning methods demonstrate some effectiveness, these methods lack fundamental investigation into the phenomenon of client drift. Classifier calibration with virtual representations (CCVR) [9] points out that the output of the final layer of feature extractors is more susceptible to classifier bias. Based on this observation, several approaches have been proposed. Knowledge transfer-based methods aim to guide clients to learn the global knowledge distribution rather than aligning the differences between heterogeneous knowledge [10], [11]. Some personalized federated learning (PFL) methods attempt to separate global and local information, followed by multitask optimization [12]. However, these methods overlook the consistency between global and local information. In addition, prototype-based methods optimize learning by transmitting abstract class prototypes [13], [14]. These methods primarily focus on minimizing the empirical error between global and local prototypes, rather than aligning the latent differences between prototypes.

To solve the problems mentioned above, we propose a method to address statistical heterogeneity, named heterogeneous Federated learning with Representation Deviation Alignment

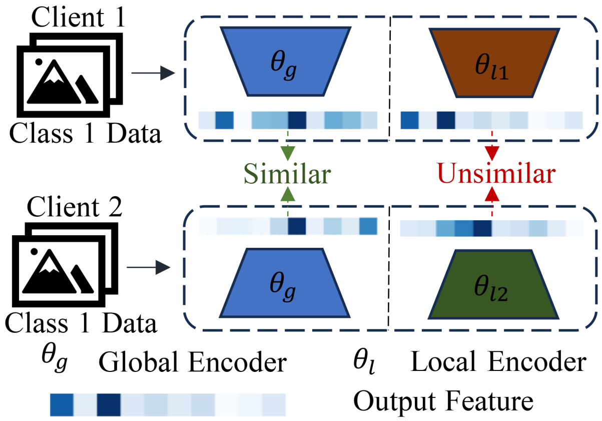


Fig. 1. Motivation of proposed method. For the same class of data across different clients, global model representations tend to be similar, while local model representations exhibit significant variability.

(FedRDA). FedRDA defines a principal feature subspace using the global singular vectors obtained singular value decomposition (SVD), and regularizes local client features through the residuals of orthogonal projection (OP), which guide the local features to minimize orthogonal deviation, thereby reducing representation shift during the federated optimization process. Specifically, FedRDA first encodes low-dimensional features to obtain model representations containing salient information while minimizing the empirical risk loss. Then, it establishes a global unbiased principal feature space based on the global model representations, which serves to quantify local representation deviation. Finally, during training, local model optimization is achieved by minimizing both subspace representation deviations and distribution discrepancies. FedRDA is comprehensively evaluated on multiple datasets, achieving favorable experimental results, which clearly validate the ability of FedRDA to enhance model generalization in heterogeneous environments. The main contributions of this article are as follows.

1) We propose a novel heterogeneous federated learning method that explicitly quantifies the spatial deviation of local models by projecting compact feature representations onto the residuals in the principal feature space.
2) We introduce a representation consistency alignment strategy that aligns the distributions of local and global model representations within the principal feature space while maintaining the global model features at the centroid position.
3) We conduct extensive experiments on five datasets, nine benchmark methods, and multiple experimental settings, with the highest accuracy improvement of 4.21% .

II. Related Works

A. Heterogeneous Federated Learning

Federated learning aims to enable model training at the edge of devices through collaboration among multiple clients, while safeguarding data privacy [2], [15]. To address the statistical Non-IID problem in federated learning, common solutions are categorized into *data-level* and *model-level*.

Data-level methods primarily focus on data cleaning or augmentation to directly modify datasets and mitigate the impact of statistical heterogeneity [16]. However, these methods fail to capture the optimization trends of the global model, leading to augmented data with local biases. Furthermore, GAN-based method leverages global information to generate new data but often incur significant computational overhead [1]. Knowledge distillation-based method reduces resource demands by transferring distributional information from the global model [17]. Nevertheless, directly aligning distributions in highly heterogeneous and large number of categories scenarios can lead to local information loss [18]. While data-level federated learning methods enhance local model generalization, they often overlook dependencies on the global model.

Model-level methods are the dominant solutions to address heterogeneity in federated learning. Among these, PFL ensures global convergence while preserving client-specific information. Knowledge transfer-based methods aim to guide clients to learn the global knowledge distribution rather than aligning the differences between heterogeneous knowledge [10], [11]. However, these methods require targeted parameter tuning, making them less adaptable to new environments. Regularization-based methods are comparatively simpler. For example, FedProx [19] constrains local models with a proximal term, and pFedMe [20] employs the Moreau envelope as a regularization term to enable PFL. Nonetheless, as client personalization improves, the performance of global model can deteriorate. Prototype-based methods address heterogeneity by utilizing class feature representations from individual clients [21]. By enhancing the uniformity of class descriptions, these methods effectively improve model generalization [13]. However, in highly heterogeneous scenarios, prototype-based approaches may introduce additional noise, resulting in greater deviation.

B. Feature Representation in Federated Learning

In federated learning, the Non-IID nature of client data distributions is most prominently reflected in the differences in features. CCVR [9] have shown that the feature layer at the input of classifier input most directly manifests these feature discrepancy. Recent research has proposed that by fixing a common classifier for all clients, it is possible to drive the learned features across different clients toward greater consistency, thereby mitigating training bias caused by data heterogeneity [22].

In addition, some approaches leverage global semantic knowledge to align global and local features. The uniformity and variance for heterogeneous federated learning (FedUV) [23] method introduces two regularization terms to counteract local model biases. FedCP [24] uses a federated conditional strategy to separate global feature information from personalized data on the client side, alleviating model bias. Federated stabilized orthogonal learning (FedSOL) [25] incorporates implicit neighbor constraints to promote global alignment and reduce the impact of bias on the model. Federated bias-eliminating augmentation learning (FedBEAL) [26] proposes a feature deviation eliminator to mitigate discrepancies among client features, while FedAug [26] embeds a shared generator to capture consensus features across clients, expanding the training space for each client. These methods, however, rely on model-specific design details and do not provide precise measurements of feature deviation. Although previous works are effective in alleviating client bias in statistically heterogeneous scenarios, they do not measure and optimize the degree of deviation at the feature level, making it difficult to find the optimal model space. This limits the ability to align features and express global information features in heterogeneous scenarios.

III. Notations and Preliminary

A. Federated Learning

The federated learning update process is represented as $\theta^{t+1} = \sum_{i=1}^{K} \frac{m}{n}\theta_i^t$, where K represents the number of client models participating in the update from the local set of clients $C = \{c_1, c_2, \ldots, c_N\}$, with N representing the total number of clients. Here, n is the total data size across all clients, partitioned into datasets $\{D_1, D_2, D_3, \ldots, D_k\}$, where each client dataset $D_k = \{(\mathbf{x}_1, y_1^1), \ldots, (\mathbf{x}_m, y_m^p))\}$ as a local data size m specific to that client. The term θ_i^t represents the model weights of the ith client after the completion of training in the tth round. Φ_k and $\mathcal{H}_k$ represent the feature extractor and classification head of the kth client model, respectively.

B. Representation Discrepancy

To quantify the discrepancy between local and global representations, we employ SVD to decouple the global shared features and local-specific features in a geometrically interpretable manner. The key properties of SVD include the orthogonality and uniqueness of singular vectors, the nonnegativity of singular values, and their descending order, which can be interpreted as a translation and rotation of features that maps them from the original space to a new representation space. For any feature matrix $\mathbf{X}$, the SVD is expressed as follows:

$$\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T \tag{1}$$

where $\mathbf{U}$ and $\mathbf{V}$ represent the principal directions of data distribution in the feature space. By retaining the largest r singular values and their corresponding $\mathbf{U}$ and $\mathbf{V}$, a low-dimensional representation of the feature space can be obtained. Specifically

$$\mathbf{X}_r = \mathbf{U}_r\Sigma\mathbf{V}_r^T \tag{2}$$

where $\mathbf{X}_r$ is the best approximation of the matrix in the r-dimensional feature space, referred to as the **principal feature space**. Dimensions $r+1$ through m represent the **residual feature space**. If projections in the residual space are excessively large, then they indicate anomalies or deviations in the features.

The key insight is that the principal components encode globally shared discriminative patterns across clients, while the remaining components represent residual variations that may capture client-specific features. For any local feature vector $\boldsymbol{h}_i^L \in \mathbb{R}^d$, its projection onto the residual subspace is

$$\boldsymbol{h}_i^{\text{proj}} = \boldsymbol{U}_{\text{res}}\boldsymbol{U}_{\text{res}}^T\boldsymbol{h}_i^L \tag{3}$$

which isolates the component of $\boldsymbol{h}_i^L$ orthogonal to the global principal subspace. The magnitude $\|\boldsymbol{h}_i^{\text{proj}}\|_2$ quantifies the representation deviation between local features and the global consensus.

IV. Method

In this section, we introduce FedRDA, as shown in Fig. 2, FedRDA consists of two key modules: **feature space construction module** and **representation consistency alignment module**. First, we employ a feature encoder to extract compact features, which capture low-error representations, including global features extracted by the global model and features extracted by the local model during the previous training epoch. Next, we construct an unbiased principal feature space biased toward local data to measure the representation deviation between local and global models. Finally, based on the measured representation deviation, we perform spatial and distributional alignment to guide the local model toward optimizing within the global feature space. The ultimate goal of FedRDA is to minimize the deviation of client features from the global feature space and ensure that each client achieves uniform distribution within local feature space.

A. Low-Dimensional Compact Feature Extraction

1) Compact Encoder: To mitigate the impact of overfitting dominant category relative to minority category, we design a feature compacting process through linear projection and classification. Given high-dimensional features $\mathbf{X} \in \mathbb{R}^d$ from Φ_k, we first obtain compact representations through $\mathbf{Z} = \mathbf{X}\mathbf{W}_p + \mathbf{b}_p$ with $\mathbf{W}_p \in \mathbb{R}^{d\times r_1}$ and $\mathbf{b}_p \in \mathbb{R}^{r_1}$. These projected features are then processed through the classification layer

$$\mathcal{L}_e = \frac{1}{m}\sum_{i=1}^{m} \ell(\mathbf{Z}^i\mathbf{W}_c + \boldsymbol{b}_c, \mathbf{y}^i) \tag{4}$$

where $\mathbf{W}_c \in \mathbb{R}^{r_1\times p}$ and $\mathbf{b}_c \in \mathbb{R}^p$ represent the learning weight and bias vector of last layer, respectively, $\ell(\cdot)$ represents the cross-entropy loss function, and $\mathbf{y}^i$ represents the true label of ith sample.

To preserve feature fidelity while enhancing compactness, we simultaneously maintain direct classification on original features

$$\mathcal{L}_o = \frac{1}{m}\sum_{i=1}^{m} \ell(\mathbf{X}^i\mathbf{W}_o + \boldsymbol{b}_o, \mathbf{y}^i) \tag{5}$$

where $\mathbf{W}_o \in \mathbb{R}^{d\times p}$ and $\mathbf{b}_c \in \mathbb{R}^p$ represent the learning weight and bias vector of classifier, respectively. The unified objective combines both components

$$\mathcal{L} = \mathcal{L}_o + \alpha\mathcal{L}_e \tag{6}$$

where $\alpha \in [0, 1]$ is controlling fidelity-compactness tradeoff.

2) Global State Feature Collection: Unlike statistical heterogeneity mitigation approaches that rely on sharing feature information between clients, we treat the global model as a trusted model. The feature representations extracted by the global model are used as a global reference. After each communication round, when the global model is sent to the clients, the local dataset $\mathbf{D}_k$

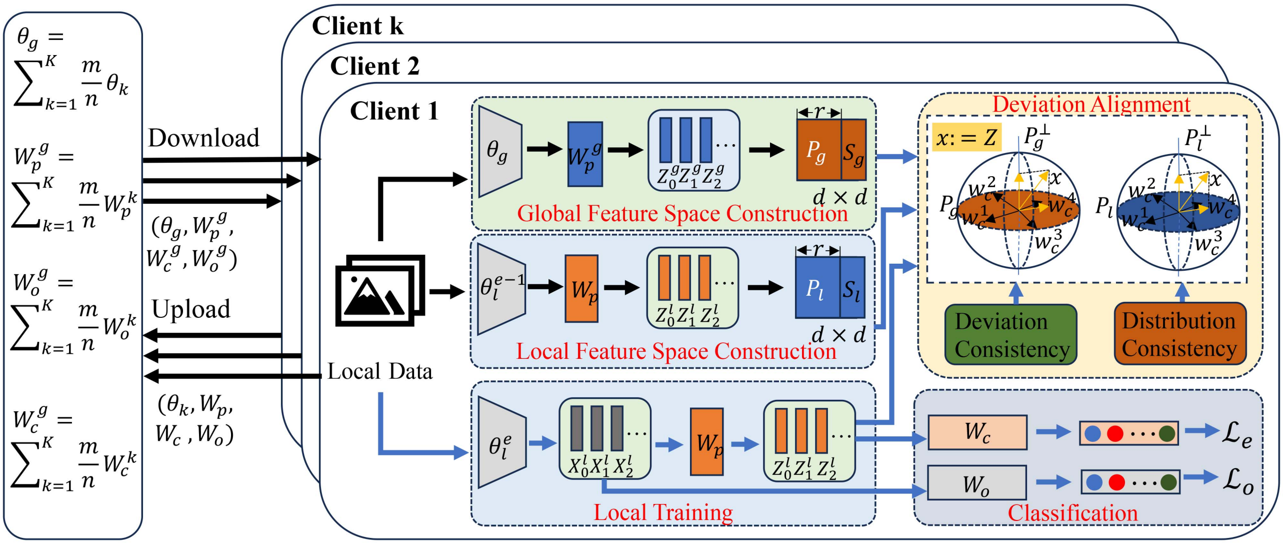


Fig. 2. Proposed framework of FedRDA, where the global representation is the output features of the global model in the local dataset, and the local representation is the features output by the local model in the previous epoch. The black lines indicate the feature extraction process prior to each communication round, and the blue lines represent the subsequent optimization and learning process.

is processed to extract features and perform compact encoding by (4), resulting in global feature representation $\mathbf{Z}^g$.

3) Local State Feature Collection: The local feature representation $\mathbf{Z}^l$ extracted by the local model is used as a local reference. The local dataset $\mathbf{D}_k$ is processed to extract features and perform compact encoding by (4).

B. Representation Deviation Alignment

To address the Non-IID problem, we aim to align the representation direction of local models with the global model. Unlike existing explicit parameter alignment methods, we utilize SVD to decompose the global feature space into orthogonal basis vectors and construct a global feature direction guidance matrix. This enables local client features to achieve implicit alignment with the global representation space through OP. The representation deviation alignment process of FedRDA is divided into two steps: **feature space construction** and **representation consistency alignment**.

1) Feature Space Construction: To calculate directional deviation in the feature space, we first construct the principal feature space by feature representation $\mathbf{Z}$. To simplify computation, according to (4), eliminate the impact of the bias term **b**to define the origin of the feature space as follows:

$$\mathbf{o} = -(\mathbf{Z}\mathbf{W})^{+}\mathbf{b} \tag{7}$$

where $(\cdot)^{+}$ represents the pseudoinverse. The term $\mathbf{o}$ signifies the offset distance from the original feature space, thereby establishing the origin of the unbiased feature space. This allows for the formation of new representation $\mathbf{Z}_o = \mathbf{Z} + \mathbf{o}$.

Then, we construct the unbiased principal feature space using the global features

$$\mathbf{Z}_o^T \mathbf{Z}_o = \mathbf{Q}\Lambda\mathbf{Q}^{-1} \tag{8}$$

where $\Lambda \in \mathbb{R}^d$ is the covariance matrix of eigenvalues, sorted in descending order. We select the top r eigenvalues to form a r-dimensional principal feature vector space, represented as $\mathbf{P} \in \mathbb{R}^{(r\times d)}$.

Based on (7) and (8), we can obtain the global unbiased principal feature space $\mathbf{P}_g$ and the local unbiased principal feature space $\mathbf{P}_l$ by global unbiased representation $\mathbf{Z}_o^g$ and local unbiased representation $\mathbf{Z}_o^l$. The specific calculation formulas are as follows:

$$\mathbf{P}_g = \mathbf{Q}_g[:, :r], \mathbf{P}_l = \mathbf{Q}_l[:, :r] \tag{9}$$

where $\mathbf{Q}_g$ is the eigenvector matrix obtained from the SVD of the global unbiased representation $\mathbf{Z}_o^g$ and $\mathbf{Q}_l$ is the eigenvector matrix obtained from the SVD of the local unbiased representation $\mathbf{Z}_o^l$. Here, the top r eigenvectors corresponding to the largest eigenvalues form the principal feature spaces $\mathbf{P}_g$ and $\mathbf{P}_l$, which are used for representation deviation alignment.

2) Representation Consistency Alignment: To achieve feature alignment within the principal feature space $\mathbf{P}$, we decompose a feature $\mathbf{x}$ into components within and orthogonal to $\mathbf{P}$ as follows: let the projection of $\mathbf{x}$ onto $\mathbf{P}$ be denoted as $\mathbf{x}^{P\perp}$, where $\mathbf{x} = \mathbf{x}^{\mathbf{P}} + \mathbf{x}^{\mathbf{P}\perp}$. The orthogonal component $\mathbf{x}^{\mathbf{P}\perp}$ is determined by the feature vectors spanning dimensions $(r+1)$ to d, represented by $\mathbf{S}$. The deviation of feature $\mathbf{x}$ from space $\mathbf{P}$ can thus be expressed as $\mathbf{x}^{\mathbf{P}\perp} = \mathbf{S}\mathbf{S}^T\mathbf{x}$.

a) Deviation consistency: Due to statistical heterogeneity, feature outputs $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_b]$ derived from dataset samples may have low representation for some feature samples. To address this, we normalize the deviation scores uniformly, defining the intrinsic deviation score as follows:

$$d(\mathbf{X}, \mathbf{S}) = \frac{\sum_{i=1}^{b} \max(l_{j=1}^{i}, \ldots, l_{p}^{i})}{\sum_{i=1}^{b} \sqrt{\mathbf{x}_i^T \mathbf{S}\mathbf{S}^T \mathbf{x}_i}} \sqrt{\mathbf{x}_i^T \mathbf{S}\mathbf{S}^T \mathbf{x}_i} \tag{10}$$

where l_p^i represents the score of class p for the ith feature $\mathbf{x}_i$ in the sampled dataset, obtained as the output of the final classification head after applying the softmax function, ensuring that the deviation score does not exceed the actual classification output.

To measure the deviation of the local features $\mathbf{Z}_o^b$, we first project the local feature representations onto the local feature subspace. The final deviation value E is computed as follows:

$$E = \frac{e^{u_g} + e^{u_l}}{\sum_{i=1}^{p} e^{l_i} + e^{u_g} + e^{u_l}} \tag{11}$$

where $u_g = d(\mathbf{Z}_o^b, \mathbf{S}_g) - d(\mathbf{Z}_o^g, \mathbf{S}_g)$ represents the deviation of local features relative to the global feature space and $u_l = d(\mathbf{Z}_o^b, \mathbf{S}_l) - d(\mathbf{Z}_o^g, \mathbf{S}_l)$ represents the deviation in the local feature space. The result is influenced by three components: the local prediction score, the deviation of local features from the global feature space, and the deviation in the local feature space. A large relative deviation value suggests significant differences between the local and global feature representations, necessitating feature optimization. A small deviation value indicates that the learned features closely approximate the global feature representation, reflecting a smooth feature learning process.

b) Distribution consistency: To ensure that local and global feature spaces are aligned and follow a consistent distribution while reducing deviations, we define the distribution consistency value

$$U = D_{KL}(u'_l \| u'_g) = \sum_{i=1}^{b} u'_l(i) log \frac{u'_l(i)}{u'_g(i)} \tag{12}$$

where $u'(\cdot)$ represents the probability distribution output of u after applying softmax function.

C. Loss Function

The objective of FedRDA has been to ensure that the original model and compact features accurately classify data while aligning local features with the global feature space. Therefore, the final optimization objective has been defined as follows:

$$\mathcal{L} = \mathcal{L}_o + \alpha \mathcal{L}_e + \gamma(E + U) \tag{13}$$

where the loss function consists of three components: first term, $\mathcal{L}_o$, represents the main classification loss. The second term, $\mathcal{L}_e$, represents the classification loss of compact feature, encourages the model to output high-quality, compact features. The third term collectively form a consistency alignment strategy. γ is a hyperparameter that controls the level of deviation. The E mitigates the deviation between local and global feature representations, the U ensures that local and global features follow a consistent distribution trend.

V. Experiments

A. Datasets and Hyperparameters Settings

FedRDA is primarily designed for federated learning classification tasks. To evaluate its effectiveness, we followed prior works [7], [25] and selected four datasets, SVHN, CIFAR10, CIFAR100, and Tiny-ImgaeNet, for experiments. SVHN, CIFAR-10, and CIFAR-100 contain 50 000 training images and 1000 test images, following the official dataset settings. Tiny-ImageNet consists of 200 classes, with each class containing 500 training images, 50 validation images, and 50 test images. GC10-DET is a steel defect detection dataset consisting of 1832 training images and 461 testing images, covering ten defect categories. The image resolution is resized to 224 × 224 for model input. We use different backbone models tailored to the complexity and resolution of each dataset: SimpleCNN (2 Conv + 2 FC layers) for SVHN and CIFAR-10, visual geometry group (VGG)-11 for CIFAR-100, residual neural network (ResNet)-18 for Tiny-ImageNet, and GC10-DET, which involve larger input resolutions or more complex visual patterns. The Non-IID data partition follows a dirichlet distribution, with ρ controlling the level of heterogeneity. To ensure rigorous comparisons with baseline methods, the experiments in this work are configured with ten clients, the default heterogeneity level $\rho = 0.1$, a learning rate of 0.01, a batch size of 64, three local training epochs, and the stochastic gradient descent (SGD) optimizer with a momentum of 0.9. All experiments were conducted on a workstation equipped with a single RTX 3090 GPU and a 2.90 GHz Intel(R) Xeon(R) Gold 6226R CPU.

Algorithm 1: Federated Learning With Representation Deviation Alignment.

```
Input: Initial global model $\theta_g^r$, which includes $W_c$, $W_p$
  and $W_o$. Key hyperparameters are as follows: learning
  rate $\eta$, participation ratio $\lambda$, number of local training
  epochs $E$, communication rounds $R$, heterogeneity
  level $\rho$.
Output: Global model $\theta_g$.
produce model aggregation
  for$r = 0, 1, \ldots, R - 1$do :
    Randomly sample $K$ clients based on $\lambda$ and $N$
    for $k \in N$ parralle do:
      send global model $\theta_g^r$ to client k
      $\theta_k^r, |D_k| \leftarrow \mathbf{LocalUpdate}(k, \theta_g^r)$
    end for
    $\theta_g^{r+1} = \sum_{k \in N} \frac{|\bar{D_k}|}{\sum_{k \in N} |D_k|} \cdot \theta_k^r$
  end for
end produce
function LocalUpdate$(k, \theta_g^{r+1})$
$\boldsymbol{Z}_g \leftarrow$ get global feature with $\theta_g^r$ and $D_k$
build global feature space by (9)
  for $e = 0, 1, 2, \ldots, E - 1$ do:
    for $(\boldsymbol{x}_i, y_i) \in D_k$ :
      $Z_l \leftarrow$ get global feature with $\theta_e$ and $(\boldsymbol{x}_i, y_i)$
      build local feature space by (9)
      $E_g \leftarrow$ deviation compute
      $U \leftarrow$ compute consistent distribution
      $\mathcal{L} = \mathcal{L}_o + \alpha \mathcal{L}_e + \gamma(E + U)$
    end for
  end for
  return $\theta_k^r, |D_k|$
```

TABLE I
RESULTS OF THE GLOBAL MODEL ACCURACY (%) ACROSS FOUR DATASETS UNDER DIFFERENT LEVELS OF HETEROGENEITY ARE PRESENTED

Methods	SVHN		CIFAR-10		CIFAR-100		Tiny-ImageNet		GC10	
	$\rho = 0.1$	$\rho = 0.05$	$\rho = 0.1$	$\rho = 0.05$	$\rho = 0.1$	$\rho = 0.05$	$\rho = 0.1$	$\rho = 0.05$	$\rho = 0.1$	$\rho = 0.05$
FedAvg [2]	81.45±0.01	79.67±0.01	50.43±0.03	**51.85±0.02**	46.79±0.05	40.69±0.06	42.16±0.02	38.40±0.04	64.77±0.71	45.08±0.41
FedProx [19]	41.13±0.13	32.52±0.10	50.72±0.02	51.37±0.02	46.73±0.04	40.72±0.06	41.70±0.02	38.32±0.04	53.39±0.24	36.76±0.07
Moon [28]	74.81±0.01	71.11±0.01	50.58±0.02	50.11±0.03	46.27±0.05	40.51±0.06	38.64±0.04	38.51±0.04	62.58±0.02	54.92±0.04
FedNova [4]	82.31±0.01	78.95±0.03	47.47±0.04	40.93±0.20	46.77±0.05	40.63±0.06	41.89±0.02	38.50±0.03	50.89±2.22	–
FedFM [7]	70.81±0.01	70.03±0.15	42.17±0.12	38.69±0.62	37.06±0.15	30.02±0.08	24.76±0.05	26.70±0.05	33.26±0.63	37.24±0.03
FedUV [23]	81.16±0.03	79.24±0.01	48.57±0.01	47.16±0.02	42.29±0.03	36.77±0.05	34.45±0.01	31.25±0.02	64.99±0.64	46.17±0.11
FedSOL [25]	64.40±0.05	54.09±0.06	46.76±0.08	42.16±0.11	36.46±0.04	32.16±0.05	41.82±0.01	37.75±0.01	30.63±0.01	28.45±0.01
FedPAC [29]	69.50±0.22	64.53±0.49	42.02±0.17	39.74±0.24	46.17±0.26	38.51±0.55	19.51±0.17	17.44±0.61	31.15±0.01	30.85±0.11
FedFA [30]	68.23±0.01	59.81±0.01	51.36±0.75	49.36±0.39	46.91±0.04	40.75±0.12	41.07±0.03	38.16±0.05	51.78±0.75	46.61±0.55
FedRDA	**82.42±0.74**	**81.28±2.56**	**54.64±0.01**	51.18±0.01	**48.19±0.05**	**42.08±0.06**	**45.10±0.01**	**40.30±0.03**	**65.21±2.23**	**57.99±1.36**

1 When $\rho = 0.05$, on the GC10 dataset, we observe through repeated trials that the accuracy of FedNova remains constant at 2.1882%.
The best-performing method in each setting is highlighted in **boldface**, while the second-best method is indicated with an underline.

B. Baselines

To validate the effectiveness of FedRDA, we compare with four categories of baseline methods. First, classical federated learning algorithms, including FedAvg [2], FedProx [19], and FedNova [27], optimize models by introducing regularization terms. Second, contrastive learning-based methods, such as model-contrastive federated learning (Moon) [4], leverage the similarity between model representations to perform model-level contrastive learning. Third, anchor-based method, FedFM [7], guides clients toward a shared optimization target by introducing common anchors across clients. In addition, generalization-enhancing algorithms, such as FedUV [23] and FedSOL [25], indirectly improve global model performance by enhancing the generalization capabilities of local models. Finally, methods similar to ours, such as FedPAC [28] and FedFA [29], address federated heterogeneity problems through feature-based alignment mechanisms.

C. Performance Across Different Datasets and Heterogeneity Levels

Table I presents the experimental results of all baseline methods on three datasets under heterogeneous settings with $\rho \in \{0.1, 0.05\}$. The proposed FedRDA method achieved state-of-the-art performance in nine out of ten experimental configurations. Notably, on the CIFAR-10 dataset, it outperformed the baseline FedAvg by 3.92% . On the CIFAR-100 dataset, FedRDA surpassed the second-best algorithm by 1.4% and 1.36%, respectively. Similarly, on the Tiny-ImageNet dataset, FedRDA demonstrated strong performance, achieving improvements of at least 2.94% and 1.9% . This success is attributed to the ability of FedRDA to effectively utilize the preference information of global and local model features during federated training. By measuring and mitigating the discrepancies between these features, FedRDA reduces bias and enhances performance.

In scenarios with extreme heterogeneity ($\rho = 0.1$ and $\rho = 0.05$), the proposed algorithm demonstrates superior effectiveness compared to classic federated algorithms, such as FedAvg. For general tasks, compact feature representations contain abundant latent deviation information, enabling FedRDA to effectively align features and mitigate local model representation deviations. Most baseline methods also maintain competitive performance under these conditions. However, for complex tasks, the experimental results of baseline methods show significant variability, whereas FedRDA consistently outperforms all baselines, because FedRDA focuses on learning latent differences between model representations rather than directly aligning distributions. In Non-IID environments with complex tasks, models are more sensitive to feature information, and direct feature space alignment can impair the learning capacity of local models.

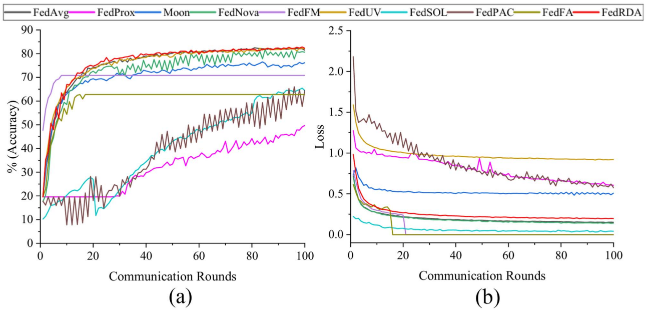


Fig. 3. Accuracy and loss curves on the SVHN dataset with $\alpha = 0.1$.

As shown in Fig. 3, following the approach of FedUV [23] and FedFA [29], we provide an analysis of convergence. From Fig. 3(a), in terms of training accuracy, FedRDA demonstrates relatively faster convergence, achieving higher accuracy within fewer training rounds. From Fig. 3(b), regarding training loss, FedRDA maintains a consistently faster convergence rate. Based on the motivation and methodology proposed in this work, when the model is fully trained and both global and local feature representations become stable, the feature discrepancy also stabilizes. FedRDA implicitly measures and optimizes this discrepancy, facilitating more efficient global model learning.

D. Impact of Participating Clients

Fig. 4 presents experiments to evaluate the impact of the number of participating clients on FedRDA. The parameters are set as $N \in \{50, 100\}$ and the sampling rate as $\lambda \in \{0.4, 0.8\}$. The experimental analysis is detailed as follows.

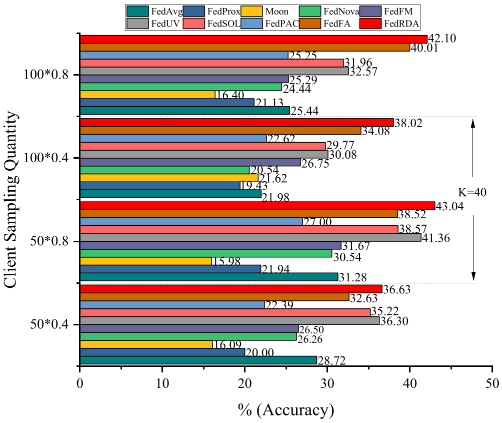


Fig. 4. Accuracy with different numbers of clients and sampling rates.

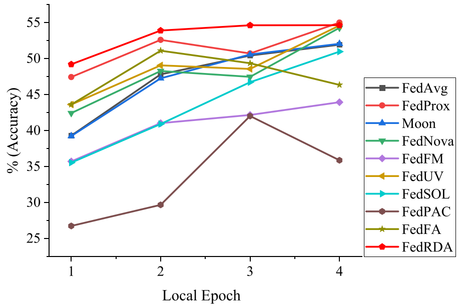


Fig. 5. Change curves of different local epochs.

FedRDA demonstrates strong stability under low client participation, indicating that the proposed deviation-aware alignment method can effectively reduce representation discrepancies across clients, even when the participating clients vary. While higher client participation generally leads to better performance, the proposed method still achieves superior generalization in low-participation scenarios, benefiting from the alignment of principal semantic subspaces. Overall, the results suggest that FedRDA possesses participation-aware capabilities, making it well-suited for deployment in federated settings with dynamic or sparse client involvement.

E. Impact of Different Local Training Epochs

Generally, as the number of local training rounds increases, the feature space of model becomes more aligned with client-specific features, drifting further from the global feature space. As shown in Fig. 5, FedRDA outperforms most baseline algorithms under varying local training epochs and achieves the best generalization performance with fewer training rounds, because, with fewer epochs, representation deviations among models are less influenced by Non-IID conditions, allowing FedRDA to effectively align representations by capturing latent differences.

F. Effectiveness Analysis of FedRDA

In this section, we investigate the impact of two key parameters in FedRDA, projection dimension r_1 and deviation computation dimension (SVD principal feature selection dimension) r, on its generalization capability. The parameters are set as $d/r_1 \in \{0.25, 0.5, 0.75, 1\}$ and $r = \beta \cdot \{1, 2, \ldots, 9\}$, where $\beta = 5$ for convolutional neural network (CNN) and $\beta = 20$ for other models. For the VGG11 and ResNet18 classifier, the feature dimension d is 512, while for the CNN classifier, the feature dimension d is 84. The experimental analysis is as follows.

TABLE II
ACCURACY(%) UNDER DIFFERENT COMPACT FEATURE DIMENSIONS

Dim	ResNet		VGG		CNN	
	Acc	Deviation	Acc	Deviation	Deviation	Deviation
0/4	63.03	–	66.98	–	50.43	–
1/4	65.12	2.84	69.64	2.27	56.32	2.59
2/4	65.36	3.33	70.28	2.28	55.53	2.61
3/4	64.87	3.56	69.14	2.28	54.59	2.61
4/4	66.40	3.77	70.81	2.27	56.49	2.59

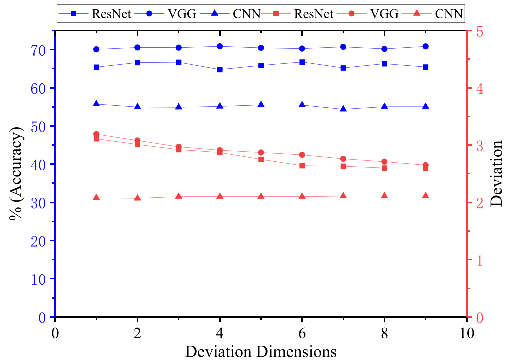


Fig. 6. Experimental results of different deviation calculation dimensions.

1) Impact of Feature Dimensions: As shown in Table II, reveals that appropriate dimension reduction enhances the ability of model to extract generalizable features, thereby improving generalization performance. Reducing the dimension to half of the input feature dimension achieves optimal generalization. However, further dimension reduction leads to the loss of essential features, resulting in decreased performance. Consequently, we recommend setting the feature dimension to one-half in FedRDA. Conversely, increasing the feature dimension raises computational complexity and exacerbates deviation, which can adversely impact performance.

2) Impact of Different Deviation Dimensions: As shown in Fig. 6, increasing the dimensionality of the feature vectors implies that more dominant features are extracted for deviation calculation, which results in an overall reduction in bias. However, using fewer feature vectors for computation does not necessarily lead to stronger generalization capabilities, because in the context of federated learning, the dominant features vary across different clients. The deviation values gradually decrease as the feature dimensions increase, and the deviation magnitude is positively correlated with accuracy, because, FedRDA aligns model representations by optimizing deviations, effectively uncovering deeper latent differences. This facilitates fine-grained

TABLE III
ACCURACY(%) OF DIFFERENT MODELS ON THE CIFAR-10 DATASET

Models	FedAvg		FedRDA	
	$\rho = 0.1$	$\rho = 0.05$	$\rho = 0.1$	$\rho = 0.05$
CNN	50.43±0.03	51.85±0.02	54.64±0.01	51.18±0.01
VGG	62.41±0.26	48.03±0.09	68.43±0.10	47.68±0.29
ResNet	60.58±0.12	49.56±0.01	67.23±0.04	54.42±0.10
MobileNet	43.67±0.07	30.95±0.07	45.13±0.09	31.28±0.10
ShuffleNet	28.27±0.10	28.99±0.14	18.82±0.28	29.95±0.09
EfficientNet	23.16±0.10	20.11±0.13	20.90±0.13	23.76±0.18
DenseNet	29.72±0.31	43.78±0.04	41.77±0.73	46.98±0.01

TABLE IV
PERFORMANCE UNDER DIFFERENT DECOMPOSITION METHODS

Methods	Acc(%)	Deviation	Time(s)
SVD	66.70	3.56	259.44
QR	65.04	19399.46	755.64
PCA	65.86	2.22	185.43

alignment, thereby enhancing the generalization capability of the global model.

3) Impact of Different Network Models: In this section, we study the impact and effectiveness of seven different model structures on FedRDA. The experimental analysis is as follows.

As shown in Table III, the results reveal that FedRDA demonstrates high adaptability across most network designs, significantly enhancing the generalization of the global model in heterogeneous settings. In particular, VGG and ResNet models show improvements of 6.02% and 6.65% under heterogeneity level $\rho = 0.1$, respectively. Lightweight models, such as MobileNet and ShuffleNet, also exhibit improved performance. However, EfficientNet, which leverages neural architecture search, shows only moderate gains, because the heterogeneous data distribution causes inconsistent layer representations, affecting performance. Overall, FedRDA outperforms FedAvg across diverse model architectures, demonstrating the ability to effectively align model representations.

4) Comparative Analysis of Decomposition Methods: As shown in Table IV, we introduce orthogonal-triangular (QR) decomposition and principal component analysis (PCA) to construct the feature space via feature decomposition. QR decomposition provides an orthogonal basis but lacks the ability to rank feature directions by semantic importance, and thus fails to capture the dominant representation patterns. PCA constructs principal directions based on the eigenstructure of the covariance matrix to capture major feature trends. The superior performance of SVD indicates that effective alignment relies not only on feature orthogonality but also on the semantic prioritization of feature directions, making it well-suited for addressing heterogeneity in federated learning.

5) Comparative Analysis of Compact Encoder Methods: As shown in Table V, we employ auto-encoder (AE) and random OP for compact feature encoding, where the original feature dimension is 512 and the encoded dimension is 384. It can be observed that multi-layer perceptron (MLP)-based encoding achieves superior performance. AE, as a classical dimensionality reduction technique, suffers from degraded performance due to the introduction of reconstruction loss. OP, which reduces dimensionality using randomly generated orthogonal matrices, enhances feature discriminability to some extent but lacks the learning capacity required to capture representative feature embeddings. In contrast, MLP serves as a simple yet effective encoder that can learn to obtain compact representations, thereby facilitating subsequent deviation computation.

TABLE V
PERFORMANCE UNDER DIFFERENT COMPACT ENCODE METHODS

Methods	Acc(%)	Deviation	Time(s)
MLP	66.70±0.11	3.56	259.44
AE	65.05±0.14	3.49	262.50
OP	66.38±2.59	2.30	253.17

TABLE VI
ACCURACY(%) OF DIFFERENT MOUDLES ON THE THREE DATASET

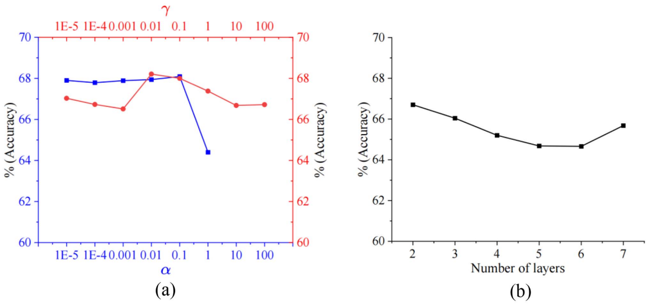
Methods	CIFAR10	CIFAR100	Tiny-ImageNet
wo-$(E+U)$	65.37±0.09	53.35±0.01	44.68±0.01
wo-$\mathcal{L}_e$	63.03±0.01	50.41±0.01	43.49±0.01
wo-$\mathcal{L}_o$	59.54±0.33	50.26±0.01	43.58±0.01
FedRDA	66.70±0.11	53.71±0.01	45.10±0.01

G. Ablation Study

In this section, we study the impact of different modules in FedRDA under the ResNet model across three datasets. We set $d/r_1 = 0.75, \beta = 4, \alpha = 1$, and $\gamma = 1$. The ablation settings are as follows.

1) **wo**$-\mathcal{L}_o$: Compared to FedRDA, this variant removes the primary classifier and relies solely on compact feature classification and representation deviation alignment, and performance is tested.
2) **wo**$-\mathcal{L}_e$: Compared to FedRDA, this variant removes the low-dimensional compact feature encoder and performance is tested.
3) **wo**$-(E+U)$: Compared to FedRDA, this variant removes the representation consistency alignment strategy and performance is tested.

As shown in Table VI. For **wo**$-\mathcal{L}_o$, we observe degraded performance after its removal, particularly on the CIFAR-10 and CIFAR-100 datasets. This is because the primary classifier is responsible for classifying the original features used for projection, which encode both local and global knowledge. As a result, these features are not suitable for representing feature deviation. This finding highlights the necessity of retaining both the primary classifier and the projection layer. Removing the $\mathcal{L}_e$ component (**wo**$-\mathcal{L}_e$) leads to a performance decline, as the compact feature encoder helps FedRDA obtain more representative model representations while avoiding deviations caused by noise. This component effectively extracts deviation information across varying



(a) (b)

Fig. 7. (a) Impact of different fidelity–compactness trade-offs and levels of feature deviation. (b) Impact of the number of compact encoding layers.

TABLE VII
Comparison of Computational Across Different Algorithms

Methods	Params/MB	FLOPs/G	Time/S
FedAvg [2]	11.1643	5.5542	232.25
FedProx [19]	11.1643	16.6626	286.59
Moon [28]	11.1694	166.26	348.81
FedNova [4]	11.1643	11.1084	288.78
FedFM [7]	11.1643	11.3179	242.61
FedUV [23]	11.1643	5.5542	288.24
FedSOL [25]	11.1643	11.1084	232.59
FedPAC [29]	11.1643	7.4056	279.39
FedFA [30]	12.6839	6.6866	241.59
FedRDA	12.4763	6.9305	259.44

degrees of heterogeneity, proves that extracting compact features can mitigate the influence of dominant category on clients and enhance the generalization capability of the global model. When the consistency alignment strategy is removed ($\mathbf{wo}-(E+U)$), performance also decreases. This indicates that FedRDA can effectively capture latent direction differences between model representations, enabling the consistency strategy to align these representations and improve overall performance.

As shown in Fig. 7(a), we examine the sensitivity of FedRDA to the hyperparameters α and γ, which control the weights of $\mathcal{L}_e$ and $(E+U)$, respectively. The results (illustrated in Fig. 7) demonstrate that FedRDA maintains robust performance within a wide range of values. While extremely small or large values may affect optimization balance, the model exhibits stable accuracy when α and γ are set within a moderate range ($[0.01, 0.1]$). Furthermore, as shown in Fig. 7(b), we investigate the impact of the number of projection layers on performance. We observe that increasing the number of layers does not improve effectiveness. This is because the goal of projection is to obtain compact features for global deviation computation. When the number of projection layers is excessive, local feature information is disrupted, leading to the failure of $\mathcal{L}_e$.

H. Computational Analysis

As shown in Table VII, this section compares the model parameter size, computational cost, and average per-round client training time of FedRDA with other baseline methods. FedRDA maintains a lower parameter size and computational cost because it only introduces a projection layer for compact feature extraction and classification, while SVD decomposition and deviation computation are performed solely on compact features, resulting in lower computational overhead compared to FedProx and MOON. FedNova requires local model parameter normalization and FedSOL involves additional fine-tuning of local model parameters, both leading to increased computational cost. FedFM incurs additional computation due to anchor box calculation and matching operations. FedPAC introduces significant computational overhead by performing personalized model fine-tuning and feature alignment, while FedFA increases both parameter size and computational cost by enforcing feature alignment at each convolutional layer.

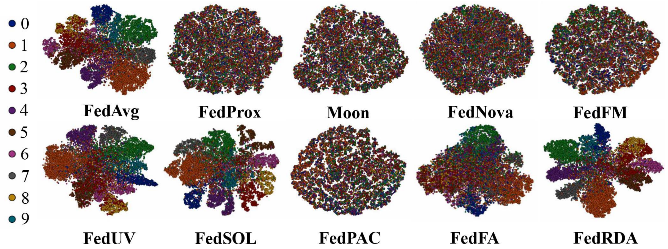


Fig. 8. Feature embedding visualization of baselines and FedRDA in the SVHN dataset.

I. Visualization Analysis

As shown in Fig. 8, we visualize the feature representations using t-distributed stochastic neighbor embedding (t-SNE) for dimensionality reduction. We observe that the classical method FedAvg is capable of distinguishing some class-specific features. In contrast, traditional methods, such as FedProx, MooN, and FedNova, which primarily focus on privacy preservation, tend to exhibit significant overlap in feature clusters under highly heterogeneous scenarios. The anchor-based method FedFM shows blurred cluster boundaries under extreme Non-IID settings. Regularization-based methods, such as FedUV and FedSOL, achieve better clustering for certain features, outperforming FedAvg. Feature alignment methods similar to FedRDA, such as FedPAC and FedFA, also display overlapping clusters under high heterogeneity. FedFA, which aligns features using anchor boxes, shows a tendency to better cluster major features. In comparison, FedRDA demonstrates well-separated clustering for most features. This superior performance is attributed to the proposed deviation alignment strategy, which implicitly aligns local features toward the shared global representation space, facilitating more effective feature learning.

J. Case Study

As shown in Fig. 9, we visualize clientwise feature distributions using t-SNE on the global test set and compute the average centroid distance (CD) across clients to assess feature alignment. A smaller CD indicates lower interclient bias and higher feature consistency. FedAvg achieves partial alignment across clients, but the feature distributions within each client

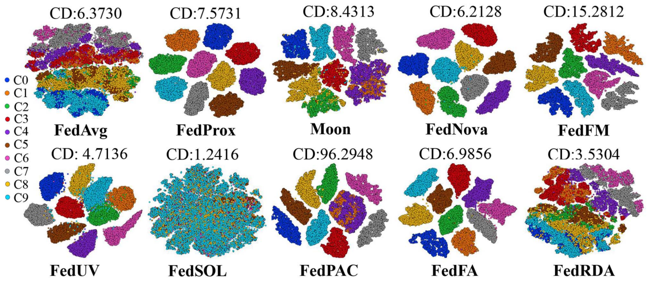


Fig. 9. Visualization of feature embeddings for the baselines and FedRDA client models on the global test set of SVHN (each color denotes a different client).

remain highly dispersed, which hinders global model aggregation. FedProx displays severe interclient representational divergence, highlighting substantial model drift and limited generalization. Contrastive learning and generalization-oriented methods fail to significantly alleviate such inconsistencies. FedPAC achieves moderate distribution alignment through explicit feature matching. FedSOL learns bias-invariant parameters and achieves the lowest CD, enhancing client generalization and global consistency. However, similar to FedAvg, it suffers from highly dispersed intraclient features, making class-level alignment difficult and limiting global generalization. In contrast, FedRDA performs alignment within a principal feature subspace, which facilitates structured consistency across clients. By explicitly modeling representational deviation, it yields more coherent feature distributions and enhances the global model's generalization ability.

K. Privacy Analysis

FedRDA provides a level of privacy protection comparable to that of FedUV [23] and FedSOL [25], while outperforming FedFA [29] and FedPAC [28] in terms of privacy preservation. First, as shown in Table I, during the construction of the feature space, FedRDA utilizes local client models for feature extraction without sharing features across clients. Second, under statistically heterogeneous settings, FedRDA performs feature selection during deviation-based alignment, thereby limiting the amount of information an adversary can infer even in the presence of gradient leakage during communication, which enhances user privacy. Finally, privacy requirements may vary across application scenarios, for cases requiring stronger protection, techniques, such as differential privacy or homomorphic encryption, can be integrated to achieve a better tradeoff between privacy and performance.

VI. Conclusion

This article proposed a novel heterogeneity bias mitigation algorithm based on feature representation deviation. Specifically, FedRDA used a feature encoder to learn model representations that capture deviation information under heterogeneous conditions and constructed a global unbiased principal feature space. Subspace projection is then used to quantify representation deviations between models, followed by the introduction of a consistency strategy to align model representations. Extensive experiments on three datasets with varying complexity. The results demonstrate that FedRDA outperforms both traditional algorithms and state-of-the-art methods designed to address statistical heterogeneity. Future work will focus on exploring how client model weights can be leveraged to further mitigate statistical heterogeneity.

Although FedRDA achieved notable improvements in generalization performance, future work may further enhance communication efficiency by leveraging deviation-aware alignment to reduce unnecessary parameter uploaded.

References

[1] R. Taheri, M. Shojafar, M. Alazab, and R. Tafazolli, "FED-IIoT: A robust federated malware detection architecture in industrial IoT," *IEEE Trans. Ind. Inform.*, vol. 17, no. 12, pp. 8442–8452, Dec. 2021.

[2] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Artif. Intell. Statist.*, 2017, pp. 1273–1282.

[3] H. Kang, M. Kim, B. Lee, and H. Kim, "FedAND: Federated learning exploiting consensus ADMM by nulling drift," *IEEE Trans. Ind. Inform.*, vol. 20, no. 7, pp. 9837–9849, Jul. 2024.

[4] Q. Li, B. He, and D. Song, "Model-contrastive federated learning," in *Proc. IEEE/CVF Conf. Comput. Vis. pattern Recognit.*, 2021, pp. 10708–10717.

[5] W. Huang, M. Ye, Z. Shi, and B. Du, "Generalizable heterogeneous federated cross-correlation and instance similarity learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 2, pp. 712–728, Feb. 2024.

[6] W. Huang, M. Ye, and B. Du, "Learn from others and be yourself in heterogeneous federated learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 10133–10143.

[7] R. Ye, Z. Ni, C. Xu, J. Wang, S. Chen, and Y. C. Eldar, "FedFM: Anchor-based feature matching for data heterogeneity in federated learning," *IEEE Trans. Signal Process.*, vol. 71, pp. 4224–4239, 2023.

[8] T. Zhou, J. Zhang, and D. H. Tsang, "FedFA: Federated learning with feature anchors to align features and classifiers for heterogeneous data," *IEEE Trans. Mobile Comput.*, vol. 23, no. 6, pp. 6731–6742, Jun. 2024.

[9] M. Luo, F. Chen, D. Hu, Y. Zhang, J. Liang, and J. Feng, "No fear of heterogeneity: Classifier calibration for federated learning with non-IID data," *Adv. Neural Inf. Process. Syst.*, vol. 34, pp. 5972–5984, 2021.

[10] T. Lin, L. Kong, S. U. Stich, and M. Jaggi, "Ensemble distillation for robust model fusion in federated learning," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst.*, 2020, pp. 2351–2363.

[11] G. Lee, M. Jeong, Y. Shin, S. Bae, and S.-Y. Yun, "Preservation of the global knowledge by not-true distillation in federated learning," in *Proc. 36th Int. Conf. Neural Inf. Process. Syst.*, 2022, pp. 38461–38474.

[12] L. Meng et al., "Improving global generalization and local personalization for federated learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 1, pp. 76–87, Jan. 2025.

[13] T. Liu, Z. Qi, Z. Chen, X. Meng, and L. Meng, "Cross-training with prototypical distillation for improving the generalization of federated learning," in *Proc. 2023 IEEE Int. Conf. Multimedia Expo*, 2023, pp. 648–653.

[14] W. Huang, M. Ye, Z. Shi, H. Li, and B. Du, "Rethinking federated learning with domain shift: A prototype view.," in *Proc. 2023 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 16312–16322.

[15] N. Bugshan, I. Khalil, M. S. Rahman, M. Atiquzzaman, X. Yi, and S. Badsha, "Toward trustworthy and privacy-preserving federated deep learning service framework for industrial Internet of Things," *IEEE Trans. Ind. Inform.*, vol. 19, no. 2, pp. 1535–1547, Feb. 2023.

[16] T. Yoon, S. Shin, S. J. Hwang, and E. Yang, "FedMix: Approximation of mixup under mean augmented federated learning," in *Proc. Int. Conf. Learn. Representations*, 2021.

[17] S. Guo, H. Chen, Y. Liu, C. Yang, Z. Li, and C. H. Jin, "Heterogeneous federated learning framework for IIoT based on selective knowledge distillation," *IEEE Trans. Ind. Inform.*, vol. 21, no. 2, pp. 1078–1089, Feb. 2025.

[18] L. Collins, H. Hassani, A. Mokhtari, and S. Shakkottai, "FedAvg with fine tuning: Local updates lead to representation learning," in *Proc. 36th Int. Conf. Neural Inf. Process. Syst.*, 2022, pp. 10572–10586.

[19] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proc. Mach. Learn. Syst.*, vol. 2, pp. 429–450, 2020.

[20] C. T Dinh, N. Tran, and J. Nguyen, “Personalized federated learning with moreau envelopes,” in *Proc. 34th Int. Conf. Neural Inf. Process. Syst.*, 2020, pp. 21394–21405.

[21] Z. Qi, L. Meng, Z. Chen, H. Hu, H. Lin, and X. Meng, “Cross-Silo prototypical calibration for federated learning with non-IID data,” in *Proc. 31st ACM Int. Conf. Multimedia*, 2023, pp. 3099–3107.

[22] H. Chen, A. Frikha, D. Krompass, J. Gu, and V. Tresp, “FRAug: Tackling federated learning with non-IID features via representation augmentation,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 4826–4836.

[23] H. M. Son, M.-H. Kim, T.-M. Chung, C. Huang, and X. Liu, “FedUV: Uniformity and variance for heterogeneous federated learning,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 5863–5872.

[24] J. Zhang et al., “FedCP: Separating feature information for personalized federated learning via conditional policy,” in *Proc. 29th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, 2023, pp. 3249–3261.

[25] G. Lee, M. Jeong, S. Kim, J. Oh, and S.-Y. Yun, “FedSOL: Stabilized orthogonal learning with proximal restrictions in federated learning,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 12512–12522.

[26] Y.-Y. Xu, C.-S. Lin, and Y.-C. F. Wang, “Bias-eliminating augmentation learning for debiased federated learning,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 20442–20452 .

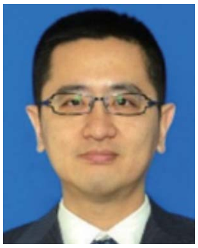
[27] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, “Tackling the objective inconsistency problem in heterogeneous federated optimization,” in *Proc. 34th Int. Conf. Neural Inf. Process. Syst.*, 2020, pp. 7611–7623.

[28] J. Xu, X. Tong, and S.-L. Huang, “Personalized federated learning with feature alignment and classifier collaboration,” in *Proc. 11th Int. Conf. Learn. Representations*, 2023. [Online]. Available: https://openreview.net/forum?id=SXZr8aDKia

[29] Z. Tailin, J. Zhang, and D. Tsang, “FedFA: Federated learning with feature alignment for heterogeneous data,” *IEEE Trans. Mobile Comput.*, vol. 23, no. 6, pp. 6731–6742, 2023.

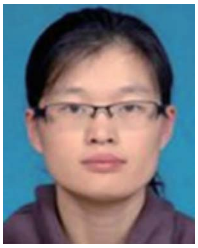
Wenjie Yao is currently working toward the Ph.D. degree in computer science and technology with the Harbin University of Science and Technology, Harbin, China.

His current research interests include graph data mining, time series analysis, and personal federated learning.



Guanglu Sun received the Ph.D. degree in computer application technology from the Harbin Institute of Technology, Harbin, China, in 2008.

He is currently a Professor and a Senior Member with China Computer Federation. His main research interests include artificial intelligence, network and information security, intelligent information processing, etc.



Suxia Zhu received the Ph.D. degree in computer system architecture from the Harbin Institute of Technology, Harbin, China, in 2009.

She is currently a Professor with the School of Computer Science and Technology, Harbin University of Science and Technology, Harbin. Her research interests include artificial intelligence, privacy and security, IoT, and parallel computing.

Ruidong Wang received the Ph.D. degree in computer applied technology from the Harbin University of Science and Technology, Harbin, China, in 2024.

He is currently a Lecturer with the School of Computer Science and Technology, Zhejiang Normal University, Jinhua, China. His current research interests include graph data mining, time series analysis, and anomaly detection.

Xinzhong Zhu (Member, IEEE) received the Ph.D. degree in information and communication engineering from Xidian University, Xi’an, China, in 2018.

He is currently a Professor with the School of Computer Science and Technology and Dean of Hangzhou Research Institute of Artificial Intelligence, Zhejiang Normal University, Jinhua, China, and also the Deputy Director of Zhejiang Provincial Key Laboratory of Intelligent Education Technology and Application, and President of Research Institute of Ningbo Cixing Company, Ltd., Ningbo, China. He has authored or coauthored more than 30 peer-reviewed papers, including those in highly regarded journals and conferences, such as IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Image Processing, IEEE Transactions on Multimedia, IEEE Transactions on Knowledge and Data Engineering, CVPR, NeurIPS, AAAI, IJCAI, *and Association for Computing Machinery Multimedia*. His research interests include machine learning, deep clustering, computer vision, and object detection, segmentation, recognition, and tracking.

Dr. Zhu is a Member of the ACM and certified as CCF Distinguished Member.

HuiYing Xu (Member, IEEE) received the M.S. degree in software engineering from the National University of Defense Technology, Changsha, China, in 2005.

She is currently an Associate Professor with the School of Computer Science and Technology, Zhejiang Normal University, Jinhua, China, and also the Researcher with Research Institute of Ningbo Cixing Company, Ltd., Ningbo, China. Her research interests include Kernel learning and feature selection, object detection, learning with incomplete data, and their applications.

Prof. Xu is a Member of the China Computer Federation.

Xiguang Wei received the Ph.D. degree in electronic engineering from the University of Hong Kong, Hong Kong, in 2017.

He is currently the Director with the Department of Data Science and Intelligent Development, AIMS SciTech, Hefei, China. He focusing on privacy-preserving federated learning across multiple terminals, medical AI, and interpretable machine learning, with an emphasis on practical deployment.

Dr. Wei is recognized as a high-level talent in Shenzhen.