Proceedings of Machine Learning Research 260:-, 2024 ACML 2024

FedLF: Adaptive Logit Adjustment and Feature Optimization in Federated Long-Tailed Learning

Xiuhua Lu[1] XHLU@QFNU.EDU.CN and **Peng Li**[1] LIPENG@QFNU.EDU.CN*†
[1]*School of Cyber Science and Engineering, Qufu Normal University*

Xuefeng Jiang[2] JIANGXUEFENG21B@ICT.AC.CN
[2]*Institute of Computing Technology, Chinese Academy of Sciences & University of Chinese Academy of Sciences*

Editors: Vu Nguyen and Hsuan-Tien Lin

Abstract

Federated learning offers a paradigm to the challenge of preserving privacy in distributed machine learning. However, datasets distributed across each client in the real world are inevitably heterogeneous, and if the datasets can be globally aggregated, they tend to be long-tailed distributed, which greatly affects the performance of the model. The traditional approach to federated learning primarily addresses the heterogeneity of data among clients, yet it fails to address the phenomenon of class-wise bias in global long-tailed data. This results in the trained model focusing on the head classes while neglecting the equally important tail classes. Consequently, it is essential to develop a methodology that considers classes holistically. To address the above problems, we propose a new method FedLF, which introduces three modifications in the local training phase: adaptive logit adjustment, continuous class centred optimization, and feature decorrelation. We compare seven state-of-the-art methods with varying degrees of data heterogeneity and long-tailed distribution. Extensive experiments on benchmark datasets CIFAR-10-LT and CIFAR-100-LT demonstrate that our approach effectively mitigates the problem of model performance degradation due to data heterogeneity and long-tailed distribution. our code is available at https://github.com/18sym/FedLF.

Keywords: Federated learning, long-tailed distribution, data heterogeneity

1. Introduction

Due to the availability of large-scale data Deng et al. (2009), Lin et al. (2014), Horn et al. (2018)and privacy-preserving policies Mohassel and Zhang (2017), there is no way for data distributed across clients to be sent to a central server for model training. To address this problem, federated learning (FL) enables multiple clients to collaboratively train a global model without uploading their local private data to the server. As deep learning continues to evolve, FL shows great potentials as a privacy-preserving and communication-efficient framework in various application domains Zeng et al. (2022), Nazir and Kaleem (2023).

However, in FL, the data sets of the participating clients come from different sources, and data heterogeneity is inevitable Li et al. (2023b,a). Existing federated learning methods mainly discuss the scenario of clients' data heterogeneity under a balanced global class distribution, ignoring the phenomenon of an unbalanced global class distribution in actual

* Xiuhua Lu and Peng Li contributed equally to this work
† Corresponding Authors: Peng Li and Xuefeng Jiang

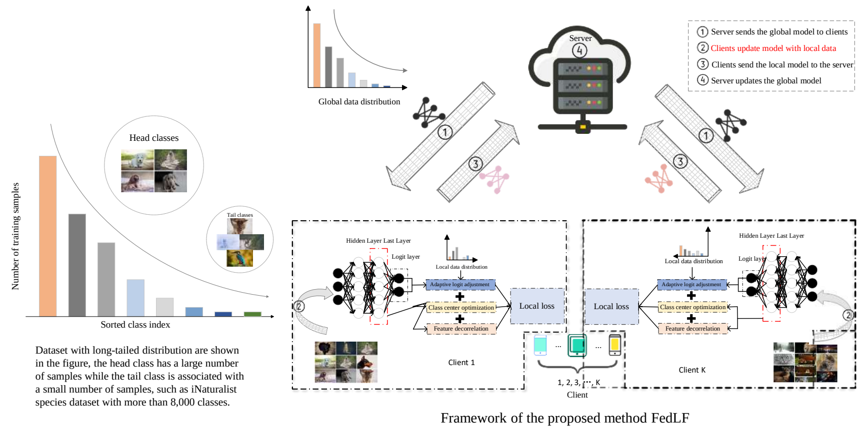


Figure 1: Left: global long-tailed distribution; Right: framework of FedLF

scenarios. In these scenarios, the global data often follows a long-tailed distribution, where a large number of samples are concentrated in a few classes, while other classes are represented by only a few samples. As shown in Fig. 1 (left), the classes with a large number of samples are called head class and the classes with a small number of samples are called tail class. Under the dual challenges of client data heterogeneity and the long-tailed distribution of global data, it gets harder for FL to train an effective global model, which we call federated long-tailed learning Zhang et al. (2023), Shang et al. (2022b). For instance, multiple companies come together for joint training to develop an autonomous driving model Nguyen et al. (2022). Normal driving behaviours are well represented in the dataset, while rare critical behaviours are underrepresented. This leads to local models that can handle typical situations but fail to correctly handle rare scenarios, making them perform poorly in emergencies, such as sudden obstacles, sharp turns and so on. Furthermore, insufficient recognition and prediction of different behaviours of pedestrians and non-motor vehicles will lead to the model not being able to respond correctly to sudden situations, increasing the risk of accidents. This indicates that tail class judgement is also a significant factor, and that a method to resolve the federated long-tail problem is urgently required.

To address the impact of long-tailed distribution, a straightforward solution is to apply existing solutions to data heterogeneity and long-tailed distribution to the federated long-tailed problem. However, the extensive experimental results in Table 1 and Table 2 show that these solutions do not lead to effective improvement. We propose a novel privacy-preserving federated learning method, FedLF, inspired by FedRS Li and Zhan (2021) and Logit adjustment Menon et al. (2021). It counteracts the federated long-tailed problem by adaptive logit adjustment, continuous class centred optimization, and feature decorrelation. Specifically, we make three modifications to the local training. Firstly, we adaptively adjust the logits of the local model according to the different local data distribution information of each client. The purpose is to ensure that the model does not overfit the head classes and can pay more attention to the tail classes, ensuring that the model treats each class fairly. Secondly, we maintain a set of continuously optimizable class centers locally, which greatly improves intra-class compactness and inter-class separability by adjusting the distance between features and their corresponding class centers. Finally, we continuously reduce the similarity among features by introducing the relationship matrix of features. Unlike complex algorithms that need to deal with large amounts of data or expose private data Shang et al. (2022b), Luo et al. (2021), our method utilizes local information to operate on local training, ensuring simplicity, efficiency, and data privacy. The contributions of this paper can be summarised as follows:

- We study federated learning with client-side data heterogeneity and global long-tailed distribution, where the server does not have access to data and distribution sensitive information.
- We propose a novel privacy-preserving federated learning method FedLF to address the problem of poor tail classes performance of the model. In particular, only the local training process of the model needs to be modified to achieve satisfactory results without the risk of privacy disclosure.
- Our method achieves superior performance through extensive comparative experiments on the benchmark datasets CIFAR-10-LT and CIFAR-100-LT with seven state-of-the-art methods.

2. Related Work

2.1. Federated Learning with Data Heterogeneity

One of the most common challenges in federated learning is data heterogeneity. During the training process of federated learning, data heterogeneity hinders the convergence speed of the model and leads to degradation of model performance. Currently, many schemes are proposed to solve the data heterogeneity problem, which are mainly classified into two strategies: client side and server side. On the client side Li et al. (2020), Jin et al. (2022), strategies with local regulation limit the training process. For instance, FedProx Li et al. (2020) introduces regularization terms to prevent local updates from significantly deviating from the global model, thus minimizing the impact of data heterogeneity. An alternative approach is FedDyn Jin et al. (2022), which incorporates server-distributed penalty terms into each client's learning objective in each round of training, thereby guiding the local model towards global optimisation. On the server side Zhang et al. (2024), global knowledge is usually used to mitigate the negative impact of data heterogeneity among clients. An excellent example is FedTGP Zhang et al. (2024), which maintains a set of trainable global prototypes on the server to help clients train prototypes with better intra-class compactness and inter-class separation. Although these methods effectively solve the challenge of data heterogeneity among clients, they generally ignore the long-tailed distribution of global data, which often leads to poor performance of the model in judging the tail classes.

2.2. Long-tailed Learning

Long-tailed data distributions widely exist in the real world, which put forward new requirements for the development of deep learning, have received extensive attention in research Zhang et al. (2023). The existing solutions to the long-tailed problem are mainly divided into two perspectives: data side and model side. On the data side, it mainly contains two kinds of approaches: reweighting and resampling. Reweighting methods Lin et al. (2017), Cui et al. (2019) modify the weights of the loss values assigned to different classes of samples. For instance, Focal Loss Lin et al. (2017) assigns greater weight to challenging samples based on predicted probabilities, which enables the model to prioritize the training of difficult samples. Resampling methods Zhang and Pfister (2021), Zang et al. (2021) mitigate the detrimental effect of the limited number of tail classes on the performance of the model through under-sampling the head classes or over-sampling the tail classes. On the model side, it mainly includes model decoupling and logit adjustment mechanism. Model decoupling Kang et al. (2020), Wang et al. (2020) focuses on reducing the model's preferences by recalibrating the classifiers, which prompts the model to look at each class more fairly. Logit adjustment Menon et al. (2021), Hong et al. (2021) is a more fine-grained solutionand aims to equalize the impact of each class, improving the model's overall accuracy. Such as LADE Hong et al. (2021) matches the target label distribution by post-processing the model prediction trained by the cross-entropy loss and the Softmax function. However, all the above schemes assume a centralised training scenario. Faced with the challenges posed by distributed training, most of these methods are ineffective. Moreover, in the case of data heterogeneity, the complexity of training increases further, making it lose its effectiveness. Therefore, it is important to develop an approach that deals with the federated long-tailed problem.

2.3. Federated Learning with Long-tailed Data

To address the federated long-tailed learning problem, recent research uses mechanisms such as distillation Shang et al. (2022a), model decoupling Shang et al. (2022b), and so on. In FEDIC Shang et al. (2022a), A new distillation method with logit adjustment and a calibration gating network is proposed to alleviate the problems associated with long-tailed data. CReFF Shang et al. (2022b) develops a set of constantly updated federated features to retrain classifiers on the server-side, achieving comparable performance to training models on real data. While the above methods relieve the problems posed by long tailed data to some extent, they usually require complex operations using auxiliary data or may pose significant risks to data privacy. Therefore, we propose FedLF, which requires only a modification of the local training part to obtain superior performance. In addition, it also provides maximum performance of tail classes, with details in Table 1 and Table 2.

3. Proposed Method

Our work is inspired by FedRS and Logit adjustment. In this section, we first introduce some basic notations and then present three modifications for local training. Finally, we describe the overall optimization objective. The overall framework of FedLF is shown in Fig. 1 (right).

3.1. Preliminaries

Settings and Notations. We consider a classical federated learning setup where K clients with heterogeneous datasets $\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3, \ldots, \mathcal{D}_K$. They collaborate on a task of C classification. Our goal is to learn a global model without uploading the data to the server. In this paper, we set x_i be the i-th input sample and y_i corresponds to the label. The global data $\mathcal{D} \triangleq \cup_k \mathcal{D}^k$ is a long-tailed distribution $X = \{(x_i, y_i)|_{i=1}^{N_{all}}, x_i \in \mathcal{D}, y_i \in \{1, \ldots, C\}\}$, where N_{all} expresses the number of all samples in $\mathcal{D}$.

We define the total number of samples for each client dataset as N, which is not necessarily the same for each client. ϕ is the distance function, B is the batchsize of local training. For i-th sample x_i, we denote $h_i = F_\theta(x_i) \in \mathbb{R}^d$ as the feature vector, where d is the feature dimension. In order to simplify, the bias is omitted, and the weight matrix of the last classification layer is denoted as $W = [w_1, w_2, \ldots, w_C]^\top \in \mathbb{R}^{C\times d}$.

Basic Algorithm of Federated Learning. In this paper, we use FedAvg McMahan et al. (2017) as the foundational algorithm, upon which we propose our improvements. The typical federated learning process unfolds as follows: In round t, the server initially distributes the global model $\mathbf{w}^t$ to all participating clients. Each client k, using their unique local dataset $\mathcal{D}_k$ for $k = 1, \ldots, K$, updates their local model $\mathbf{w}_k^t$ according to the following update rule:

$$\mathbf{w}_k^{t+1} \leftarrow \mathbf{w}_k^t - \eta \nabla_{\mathbf{w}} \ell\left(\mathbf{w}^t; \mathcal{D}_k\right), \tag{1}$$

where η denotes the learning rate, and ℓ denotes the loss function, typically a cross-entropy loss in classification tasks. Following the local updates, a subset of clients, denoted by K^t, is selected to upload their updated models to the server. The server then aggregates these models using a weighted averaging scheme based on the volume of data each client contributes, thereby producing a new global model for the subsequent round $t+1$:

$$\mathbf{w}^{t+1} = \sum_{k\in K^t} \frac{|\mathcal{D}_k|}{\sum_{k\in K^t} |\mathcal{D}_k|} \mathbf{w}_k^{t+1}. \tag{2}$$

3.2. Adaptive Logit Adjustment

To address the imbalance in the distribution of classes, we locally adjust the classifier's influence weights on each class on the client side. This adjustment makes the training process more fair and improves sensitivity to all classes. Specifically, the score matrix for each class is multiplied by the adjusted local label distribution matrix. The details are as follows:

Firstly, the local label distribution is $dist = [n_1, n_2, ..., n_C]$, where n_i denote the number of samples in class i. All elements within $dist$ are divided by the total number of local samples. The normalised $ndist$ is used to calculate the adjustment matrix $adist$:

$$adist = \frac{ndist}{\max(ndist)} \cdot (1.0 - \alpha) + \alpha \cdot 1. \tag{3}$$

Here, the smoothing factor α is a critical hyperparameter, 1 is the unit vector. When α is close to 0, the $adist$ are primarily determined by the original normalized distribution. As α approaches 1, the weights in $adist$ for each class become nearly equal. This smoothing step introduces a certain degree of uniformity while preserving the original distributional characteristics of the data. It ensures that the model does not become overly sensitive to classes with extreme distribution during training. By retaining the essential information of each class and reducing the influence of outliers, this approach enhances the model's ability to learn from less frequent classes. Ultimately, this helps improve the model's generalization ability and fairness across uneven datasets.

Secondly, the calculated *adist* is element-wise multiplied with the original score matrix $h_i \cdot W^\top$to obtain the adjusted logits z_i:

$$z_i = adist \odot h_i \cdot W^\top. \tag{4}$$

Finally, the loss function L_A is calculated from the modified logits $z_{i,j}$:

$$L_A = -\frac{1}{N} \sum_{i=1}^{N} y_i \log \left(\frac{\exp\left(z_{i,y_i}\right)}{\sum_{j=1}^{C} \exp\left(z_{i,j}\right)} \right), \tag{5}$$

where $z_{i,j}$ denotes the adjusted logit of sample x_i on class j, N denoted total number of sample.

3.3. Class Center Optimization

Inspired by contrast loss, we adjusting the Euclidean distance of features and the corresponding class centers, which improve the model's ability to discriminate among samples of different classes. Particularly, we maintain constantly updated set of class centers locally $\hat{P} = [\hat{p_c} \mid c = 1, 2, ..., C]$. We assume that h_i^c represents the features whose sample x_i is class c, N_c is the total number of local samples belonging to class c, and $\hat{p_c}$ is computed as follows:

$$\hat{p_c} = \frac{1}{N_c} \sum_{i=1}^{N_c} h_i^c, \tag{6}$$

and then, we keep optimizing the class centers $\hat{P}$ during the training process by:

$$L_C = \sum_{c=1}^{C} \sum_{i=1}^{N_c} -\log\left(\frac{e^{-\phi(h_i^c, \hat{p_c})}}{e^{-\phi(h_i^c, \hat{p_c})} + \sum_{c', c' \neq c} e^{-\phi(h_i^c, \hat{p_{c'}})}}\right). \tag{7}$$

Here, $c^{'}$ represents all classes not equal to class c. Although the above formula is the standard contrast loss, it does not significantly reduce the intra-class distance, and the learned inter-class boundaries are not clear enough.

To further clarify the boundaries, we force the learning of inter-class margins Q during training by introducing a class spacing threshold τ, which is calculated as follows:

$$Q = \min\left(\left(\max_{c \subseteq [C], c^{'} \subseteq [C], c \neq c^{'}} \phi(\hat{p}_c, \hat{p}_{c'})\right), \tau\right), \tag{8}$$

where τ is a hyperparameter. The loss function is rewritten as follows:

$$L_C = \sum_{c=1}^{C} \sum_{i=1}^{N_C} -(\log(\frac{e^{-\phi(h_i^c, \hat{p_c})+Q)}}{e^{-(\phi(h_i^c, \hat{p_c})+Q)} + \sum_{c', c' \neq c} e^{-\phi(h_i^c, \hat{p_{c'}})}}). \tag{9}$$

The L_C partially takes into account the distribution of sample features, which helps the model to better distinguish among samples of different classes and improves the classification performance of the model. The introduction of the class spacing threshold controls the spacing among classes to avoid the influence of abnormal samples and enhances the generalization ability of the model.

3.4. Feature Decorrelation

To further enhance the robustness of the model against correlated feature distributions, we introduce the loss function for decorrelation. Through evaluating the covariance matrix of the features and penalising non-diagonal elements , this loss function aims to minimise the correlation among features. The method is as follows:

The feature matrix $\mathbf{X}$ is first normalised:

$$\mathbf{X}_{ij}^{\text{norm}} = \frac{\mathbf{X}_{ij} - \mu_j}{\sigma_j}, \tag{10}$$

Here $\mathbf{X}_{ij}^{\text{norm}}$ is the element in row i and column j. μ_j and σ_j represent the mean and standard deviation of column j. Then estimate the correlation matrix $\mathbf{Cor}$:

$$\mathbf{Cor} = \frac{1}{B} \mathbf{X}^{\mathbf{norm}T} \mathbf{X}^{\mathbf{norm}}, \tag{11}$$

where X^{norm} is the normalised representation matrix, B is the batch size. The loss function L_D is computed as follows:

$$L_D = \sum_{i=1}^{N} \sum_{j=1}^{N} (\mathbf{Cor}_{ij})^2 . \tag{12}$$

Here, Cor_{ij} is the element of row i and column j of the correlation matrix Cor. The non-diagonal ($Cor_{ij}, i \neq j$) elements within the matrix are the values corresponding to the correlation among the features. We effectively reduce the correlation among the features by continuously optimizing the loss function, enhance the robustness and generalization of the model.

3.5. Overall Optimization Objective

We combine the above three losses, and this combined loss function optimizes intra-class compactness and inter-class separation while adapting to logits adjustment, and improves feature independence. It significantly improves the generalization ability and effectiveness of the overall model. The combine loss are as follows:

$$L = L_A + \lambda L_C + \gamma L_D, \tag{13}$$

Here, λ and γ are hyperparameters, and the clients trains using L. Algorithm 1 summarizes the recommendation framework FedLF.

Algorithm 1: FedLF

Input: Initialized global model $\mathbf{w}^0$, smoothing factor α, class spacing threshold τ, weights of the two loss functions λ and γ.

Output: global model $\mathbf{w}^{t+1}$.

for $t = 1$ **to** T **do**
  Randomly select a set of active clients K^t.
  `// Clients execute:`
  for $k \in K^t$ **do**
    Calculate L_A by Equation 5;
    Calculate L_C by Equation 7;
    Calculate L_D by Equation 12;
    Update local model $\mathbf{w}_k^{t+1}$ by Equation 1;
    Send $\mathbf{w}_k^{t+1}$ to the server.
  end
  `// Server executes:`
  Aggregate local models to $\mathbf{w}^{t+1}$ by Equation 2.
end

4. Experimental Results

In this section, we compare FedLF with seven state-of-the-art methods to demonstrate that FedLF is effective in relieving the federated long-tailed problem. In order to evaluate the effectiveness of all methods in more detail, we conduct extensive experiments on two benchmark datasets and evaluated four accuracies, namely head classes accuracy, middle classes accuracy, tail classes accuracy and overall accuracy.

4.1. Experimental Setup

The basic experimental setup is as follows: the total number of clients K is set to 20, the number of local iterations E for each client is 5, and the local batchsize B is 32.

Implementation details. All experiments use ResNet-8 He et al. (2016) as the base model and run under PyTorch with an Nvidia GeForce RTX 3060 Laptop GPU. We conduct experiments on the CIFAR-10/100-LT Krizhevsky (2009) dataset and set IF Cao et al. (2019) to 100, 50,and 10. IF is long-tailed factor to describe the degree of imbalance in the long-tailed case. The heterogeneous dataset is partitioned into each client using Dirichlet coefficients based on previous research Xiao et al. (2024), where Dirichlet coefficients are set to 0.5 and 1.0. The learning rate is mildly set to 0.1. We set the online rate of clients to 40%, and the clients online each time are randomly selected.

In addition, we have four different metrics when evaluating model performance: header classes accuracy, middle classes accuracy, tail classes accuracy, and all classes accuracy. To define the head and tail classes, we introduce thresholds. We set the classes with more samples above the thresholds as head classes, and the classes with fewer samples than the thresholds as tail classes. We have different thresholds for different degrees of long-tailed. Specifically, for CIFAR-10-LT, the thresholds are set to (1500, 200) when the long-tail factor IF is 100 and 50, and (1500, 600) when the IF is 10. For CIFAR-100-LT, the threshold is set to (200,20) when the long-tail factor IF is 100 and 50, and (300, 60) when the IF is 10.

Baselines. We compare with the following baseline: FedAvg McMahan et al. (2017), FedBN Li et al. (2021), FedRS Li and Zhan (2021), FEDIC Shang et al. (2022a), Focal Loss Lin et al. (2017), FedProx Li et al. (2020), CReFF Shang et al. (2022b).

Hyperparameters. Unless stated otherwise, most hyperparameters of these baseline are configured according to the original literature. We utilize the official open-source codes of these methods. There are four hyperparameters in FedLF, namely smoothing coefficient α, class spacing threshold τ, and the weights of the loss function λ, γ. The smoothing coefficient α is set to 0.25 with reference to Li and Zhan (2021). It enhances the generalization ability of the model, reduces the sensitivity to outliers, and effectively reduces the interference of extreme sample distributions on model training. We set the threshold τ to 100 with reference to Zhang et al. (2024). In our loss function formulation: $L = L_A + \lambda L_C + \gamma L_D$, the hyperparameters λ and γ are crucial for balancing the contributions of the losses L_C and L_D to the aggregate loss L. We set both λ and γ to 0.01, a decision driven by the need to ensure that the additional terms enhance the model's performance without overwhelming the primary loss, L_A. The values of λ and γ are determined through a systematic exploration of various settings, where 0.01 emerged as the optimal value that subtly integrates the corrective effects of L_C and L_D, improving overall model robustness and accuracy. For details, please see Fig. 3.

4.2. Results and Analysis

The results are shown in Table 1 ($\alpha = 0.5$) and Table 2 ($\alpha = 1$). In the table, results in bold and underlined represent the best and second best results for that column. Our method achieves superior results in all experiments. Compared to the baseline method FedAvg, our method achieves the highest performance improvement of 13.27% when IF=100, $\alpha = 0.5$ at CIFAR-10-LT. This demonstrates that our method has good generalization ability in the case of severe long-tailed global class distributions.

FedProx, FedBN and FedRS mainly address data heterogeneity. The results of FedProx and FedBN perform similarly to FedAvg because they primarily focus on addressing data heterogeneity without considering long tailed distribution. In addition, FedRS, although not specifically designed for long-tailed distribution, still performs well in long-tailed cases, suggesting that logit adjustment effectively addresses the long-tailed problem.

CReFF and FEDIC consider long-tailed distribution and show good results in solving this problem. In few cases, the CReFF method slightly outperforms our method, but when identifying tail classes, our method significantly outperforms CReFF.

The performance gap between the above methods and ours is due to the fact that FedProx, FedBN and FedRS mainly solve the problem of data heterogeneity among clients

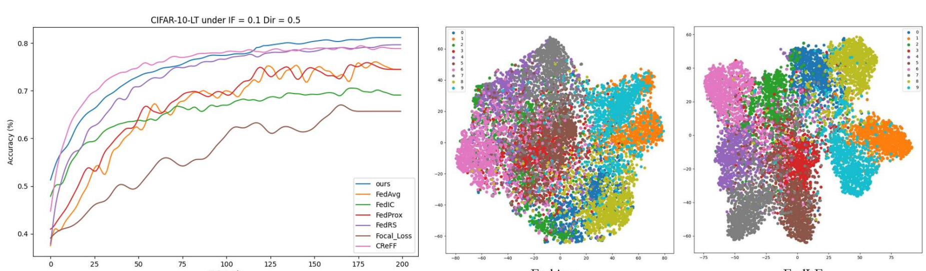
Table 1: test accuracy (%) by compared FL methods on CIFAR-10/100-LT at α=0.5.

Datsset	Non-IID / Imbalance Factor / Method/Model	IF=100 Head	IF=100 Middle	IF=100 Tail	IF=100 All	IF =50 Head	IF =50 Middle	IF =50 Tail	IF =50 All	IF=10 Head	IF=10 Middle	IF=10 Tail	IF=10 All
		$\alpha = 0.5$											
CIFAR-10-LT	FedAvg	86.17	53.67	28.07	55.74	90.87	55.84	31.70	61.52	**85.34**	63.30	64.90	74.48
	FedProx	83.33	59.60	28.27	57.32	**91.53**	58.52	31.80	63.08	84.86	70.70	53.60	76.07
	CReFF	82.30	50.48	18.70	**69.21**	89.13	59.30	31.50	71.90	84.38	73.22	57.20	78.51
	FedBN	**86.47**	50.35	29.20	54.84	91.50	54.30	36.30	61.86	84.44	62.48	65.60	73.77
	FEDIC	62.40	60.90	**78.03**	66.49	68.97	63.88	71.03	67.55	73.50	61.40	73.76	71.21
	FedRS	65.17	**69.05**	67.87	67.53	70.73	**70.30**	79.85	72.34	72.54	**83.40**	**86.00**	78.51
	Focal Loss	80.97	50.95	13.23	48.64	88.73	47.36	17.60	53.82	76.76	65.42	64.50	71.00
	FedLF	72.73	65.67	69.67	69.01	74.43	68.86	**82.15**	**73.19**	80.38	78.45	85.90	**80.16**
CIFAR-100-LT	FedAvg	**67.15**	32.10	3.67	30.58	65.25	31.53	7.50	35.30	**66.77**	41.51	12.25	44.73
	FedProx	66.90	33.24	6.10	31.83	64.88	31.84	9.11	35.68	65.09	42.96	17.13	45.46
	CReFF	63.12	47.83	9.00	**34.60**	**65.34**	**48.06**	10.32	37.64	60.35	**53.74**	15.00	47.08
	FedBN	62.65	30.06	5.03	29.07	60.79	28.91	4.78	32.22	55.68	42.04	13.00	42.72
	FEDIC	50.30	41.26	10.49	33.67	48.13	39.08	13.00	36.74	57.84	40.16	17.65	41.93
	FedRS	50.45	41.80	8.53	33.23	48.79	40.45	15.44	37.12	51.27	48.69	15.88	46.70
	Focal Loss	58.70	28.86	2.23	26.84	59.17	27.10	5.44	30.94	60.14	41.39	12.63	43.21
	FedLF	52.10	**41.86**	**11.33**	34.48	49.67	42.41	**16.06**	**39.52**	53.18	50.63	**19.38**	**48.69**

Table 2: Test accuracy (%) by compared FL methods on CIFAR-10/100-LT at α=1.0.

Datsset	Non-IID / Imbalance Factor / Method/Model	IF=100 Head	IF=100 Middle	IF=100 Tail	IF=100 All	IF =50 Head	IF =50 Middle	IF =50 Tail	IF =50 All	IF=10 Head	IF=10 Middle	IF=10 Tail	IF=10 All
		$\alpha = 1$											
CIFAR-10-LT	FedAvg	83.93	61.28	35.17	60.24	91.93	60.84	22.60	62.52	**87.76**	69.07	47.20	76.23
	FedProx	83.03	65.87	35.40	61.88	90.13	61.78	28.45	63.62	85.02	73.02	65.50	78.27
	CReFF	**89.17**	52.08	23.50	**69.94**	**93.60**	55.34	25.65	72.68	86.46	71.95	60.50	80.56
	FedBN	80.00	60.65	36.70	59.27	79.90	58.60	35.65	61.68	73.48	82.94	62.20	76.14
	FEDIC	73.37	59.35	68.50	66.30	71.40	59.15	71.60	69.05	73.43	68.00	75.94	73.16
	FedRS	62.27	69.45	69.50	67.31	71.43	70.80	77.30	72.29	79.08	80.95	**90.90**	80.93
	Focal Loss	74.53	56.63	21.10	51.34	85.40	51.64	9.30	53.30	82.54	66.90	66.60	73.69
	FedLF	65.27	**69.87**	**70.77**	68.76	74.17	**71.92**	**80.25**	**74.05**	81.60	**82.95**	86.60	**82.64**
CIFAR-100-LT	FedAvg	66.95	33.88	3.87	31.18	**67.33**	32.03	6.78	35.96	66.86	45.33	14.25	46.33
	FedProx	66.85	33.64	3.83	31.35	66.92	31.88	9.83	36.32	66.50	46.93	**16.00**	47.15
	CReFF	**67.23**	**55.87**	3.20	**35.23**	65.89	53.62	14.32	38.65	**70.83**	**58.01**	15.06	48.03
	FedBN	57.10	32.94	3.10	29.28	60.83	28.81	9.72	33.06	60.18	41.40	10.13	43.03
	FEDIC	49.85	41.36	8.60	33.98	50.63	40.19	11.68	37.26	65.39	42.60	13.03	45.36
	FedRS	51.15	43.10	8.37	34.29	46.54	41.48	17.00	37.90	48.73	50.89	12.50	47.34
	Focal Loss	60.55	28.78	3.30	27.49	60.21	29.83	2.33	32.17	64.55	41.64	12.12	44.32
	FedLF	49.55	44.54	**9.93**	35.16	52.17	**42.19**	**13.83**	**40.12**	54.27	52.31	15.12	**49.77**

and do not take into account the long-tailed factor, which is the most common factor in real life. CReFF and FEDIC is designed for the long-tailed problem have good overall performance but do not effectively improve the accuracy of determining the tail classes. In contrast, our approach takes into account the long-tailed problem. It obtains good overall performance without losing the ability to judge tail classes. Fig. 2 represents the convergence curves of algorithms for the IF = 10 and $\alpha = 0.5$ and feature TSNE plot after 200 rounds of training.



(1) FedAvg (2) FedLF

Figure 2: (1) denotes the convergence curves on CIFAR-10-LT with IF=10 and $\alpha = 0.5$, (2) denotes feature TSNE plot after 200 rounds of training.

4.3. Ablation Study

Our loss function consists of three components, in order to assess the impact of each component of FedLF on model performance in the same long-tailed environment, we design a

Table 3: Ablation Experiments result(%) for IF=50 and IF=10 at $\alpha = 0.5$

Long-tailed factor	Components		Multi-precision			
	L_C	L_D	Head	Middle	Tail	All
IF = 50	✗	✗	72.70	68.34	72.10	70.40
	✓	✗	73.63	68.34	79.20	72.10
	✗	✓	73.13	69.70	77.80	72.35
	✓	✓	74.43	68.86	82.15	73.19
IF = 10	✗	✗	71.64	83.53	89.80	78.21
	✓	✗	73.96	82.75	89.70	79.05
	✗	✓	74.92	81.13	91.10	79.02
	✓	✓	77.48	81.80	88.20	80.28

Table 4: Ablation Experiments result(%) on CIFAR-10-LT at α=0.1 and IF=0.1

Non-IID and IF factor	α = 0.1 and IF = 0.1			
Method/Model	Head	Middle	Tail	All
FedAvg	75.80	45.30	45.00	60.52
FedProx	**79.32**	62.52	43.60	69.03
CReFF	<u>75.98</u>	44.55	42.30	<u>73.56</u>
FedBN	57.80	50.08	**88.90**	57.82
FEDIC	74.60	54.23	46.90	72.16
FedRS	72.12	<u>63.30</u>	65.50	67.93
Focal Loss	65.36	61.180	43.60	61.51
FedLF	75.70	**72.20**	<u>69.90</u>	**73.72**

series of ablation experiments. By progressively removing or modifying weights of components, we aim to clarify the contribution of each component to the overall performance. In particular, we hope to reveal the role of each component in improving model accuracy under data distributions characterized by varying degrees of long-tailed.

Comparisons under the same long-tailed. We evaluate the impact of each component in the same long-tailed environment to demonstrate its effectiveness.

- Effectiveness of L_C: The effect of L_C is evident in Table 3. The experimental results are significantly improved with the addition of this component, which increases the ability of intra-class compactness and inter-class separability of features.

- Effectiveness of L_D: As can be seen from Table 3, L_D plays an vital role. After adding this component, the experimental results are significantly improved, indicating that this loss function effectively reduces the correlation of features.

Comparisons under the different long-tailed and heterogeneous. As shown in Table 1 and Table 2, varying degrees of long-tailed distribution significantly impacts the algorithm's accuracy. As the extent of the long-tailed distribution increases, the accuracy of the algorithm correspondingly decreases. To ensure the rigor of our conclusions, we conduct ablation experiments under different levels of long-tail distribution to verify the effectiveness of each component. The experimental results are shown in Table 3, indicating that each component is effective under different degrees of long-tailed environmental conditions. In addition, we conduct an experiment in a serious heterogeneous scenario, such as α=0.1 in Table 4. The results show that FedLF demonstrates effectiveness in such severe heterogeneous scenarios.

Impact of different weights. We investigate the effect of the weights λ of L_C and γ of L_D on model performance. The 3 show that different parts of the loss function play different roles in training and optimization. Therefore, to better exploit the contribution of each loss term, we adjust the weights of L_C and L_D separately and observe their effects on model performance. The results are shown in Fig. 3. The model performance is optimal when λ and γ are set to 0.01.

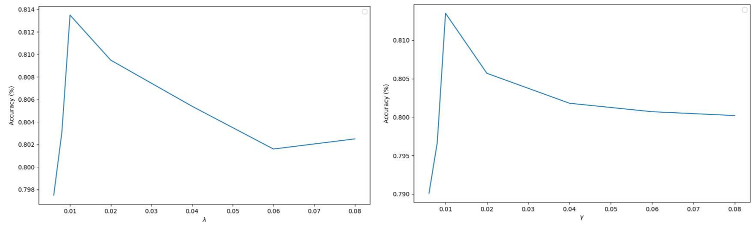


Figure 3: Impact of λ and γ weights.

5. Conclusion

In this paper, we present FedLF to enhance federated learning under data heterogeneity and global long-tailed data. FedLF is a client-side approach, which introduces three modifications in the local training phase: adaptive logits adjustment, continuous class centred optimization, and feature decorrelation. Furthermore, the effectiveness of each component of FedLF is verified. Experiments show that FedLF achieves superior results on datasets with heterogeneous and long-tailed settings compares to seven other state-of-the-art methods. For future works, we aim to study federated learning robustness against noisy labels Wu et al. (2023); Jiang et al. (2022, 2024a,b), especially in long-tailed data environments.

References

Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Aréchiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 1565–1576, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/621461af90cadfdaf0e8d4cc25129f91-Abstract.html.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge J. Belongie. Class-balanced loss based on effective number of samples. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 9268–9277. Computer Vision Foundation / IEEE, 2019. doi: 10.1109/CVPR.2019.00949. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Cui_Class-Balanced_Loss_Based_on_Effective_Number_of_Samples_CVPR_2019_paper.html.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, pages 248–255. IEEE Computer Society, 2009. doi: 10.1109/CVPR.2009.5206848. URL https://doi.org/10.1109/CVPR.2009.5206848.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer

Society, 2016. doi: 10.1109/CVPR.2016.90. URL https://doi.org/10.1109/CVPR.2016.90.

Youngkyu Hong, Seungju Han, Kwanghee Choi, Seokjun Seo, Beomsu Kim, and Buru Chang. Disentangling label distribution for long-tailed visual recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 6626–6636. Computer Vision Foundation / IEEE, 2021. doi: 10.1109/CVPR46437.2021.00656. URL https://openaccess.thecvf.com/content/CVPR2021/html/Hong_Disentangling_Label_Distribution_for_Long-Tailed_Visual_Recognition_CVPR_2021_paper.html.

Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alexander Shepard, Hartwig Adam, Pietro Perona, and Serge J. Belongie. The inaturalist species classification and detection dataset. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 8769–8778. Computer Vision Foundation / IEEE Computer Society, 2018. doi: 10.1109/CVPR.2018.00914. URL http://openaccess.thecvf.com/content_cvpr_2018/html/Van_Horn_The_INaturalist_Species_CVPR_2018_paper.html.

Xuefeng Jiang, Sheng Sun, Yuwei Wang, and Min Liu. Towards federated learning against noisy labels via local self-regularization. In Mohammad Al Hasan and Li Xiong, editors, *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022*, pages 862–873. ACM, 2022. doi: 10.1145/3511808.3557475. URL https://doi.org/10.1145/3511808.3557475.

Xuefeng Jiang, Jia Li, Nannan Wu, Zhiyuan Wu, Xujing Li, Sheng Sun, Gang Xu, Yuwei Wang, Qi Li, and Min Liu. Fnbench: Benchmarking robust federated learning against noisy labels. 2024a. doi: 10.36227/techrxiv.172503083.36644691/v1. URL http://dx.doi.org/10.36227/techrxiv.172503083.36644691/v1.

Xuefeng Jiang, Sheng Sun, Jia Li, Jingjing Xue, Runhan Li, Zhiyuan Wu, Gang Xu, Yuwei Wang, and Min Liu. Tackling noisy clients in federated learning with end-to-end label correction. *arXiv preprint arXiv:2408.04301*, 2024b.

Cheng Jin, Xuandong Chen, Yi Gu, and Qun Li. Feddyn: A dynamic and efficient federated distillation approach on recommender system. In *28th IEEE International Conference on Parallel and Distributed Systems, ICPADS 2022, Nanjing, China, January 10-12, 2023*, pages 786–793. IEEE, 2022. doi: 10.1109/ICPADS56603.2022.00107. URL https://doi.org/10.1109/ICPADS56603.2022.00107.

Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=r1gRTCVFvB.

Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009. URL https://api.semanticscholar.org/CorpusID:18268744.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In Inderjit S. Dhillon, Dimitris S. Papailiopoulos, and Vivienne Sze, editors, *Proceedings of Machine Learning and Systems 2020, MLSys 2020, Austin, TX, USA, March 2-4, 2020*. mlsys.org, 2020. URL https://proceedings.mlsys.org/paper_files/paper/2020/hash/1f5fe83998a09396ebe6477d9475ba0c-Abstract.html.

Xiaoxiao Li, Meirui Jiang, Xiaofei Zhang, Michael Kamp, and Qi Dou. Fedbn: Federated learning on non-iid features via local batch normalization. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=6YEQUn0QICG.

Xin-Chun Li and De-Chuan Zhan. Fedrs: Federated learning with restricted softmax for label distribution non-iid data. In Feida Zhu, Beng Chin Ooi, and Chunyan Miao, editors, *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, pages 995–1005. ACM, 2021. doi: 10.1145/3447548.3467254. URL https://doi.org/10.1145/3447548.3467254.

Xujing Li, Min Liu, Sheng Sun, Yuwei Wang, Hui Jiang, and Xuefeng Jiang. Fedtrip: A resource-efficient federated learning method with triplet regularization. In *2023 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 809–819. IEEE, 2023a.

Xujing Li, Sheng Sun, Min Liu, Ju Ren, Xuefeng Jiang, and Tianliu He. Federated classification tasks in long-tailed data environments via classifier representation adjustment and calibration. *Authorea Preprints*, 2023b.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In David J. Fleet, Tomás Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, volume 8693 of *Lecture Notes in Computer Science*, pages 740–755. Springer, 2014. doi: 10.1007/978-3-319-10602-1_48. URL https://doi.org/10.1007/978-3-319-10602-1_48.

Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2999–3007. IEEE Computer Society, 2017. doi: 10.1109/ICCV.2017.324. URL https://doi.org/10.1109/ICCV.2017.324.

Mi Luo, Fei Chen, Dapeng Hu, Yifan Zhang, Jian Liang, and Jiashi Feng. No fear of heterogeneity: Classifier calibration for federated learning with non-iid data. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 5972–5984, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/2f2b265625d76a6704b08093c652fd79-Abstract.html.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In Aarti Singh and Xiaojin (Jerry) Zhu, editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, volume 54 of *Proceedings of Machine Learning Research*, pages 1273–1282. PMLR, 2017. URL http://proceedings.mlr.press/v54/mcmahan17a.html.

Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=37nvvqkCo5.

Payman Mohassel and Yupeng Zhang. Secureml: A system for scalable privacy-preserving machine learning. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, pages 19–38. IEEE Computer Society, 2017. doi: 10.1109/SP.2017.12. URL https://doi.org/10.1109/SP.2017.12.

Sajid Nazir and Mohammad Kaleem. Federated learning for medical image analysis with deep neural networks. *Diagnostics*, 13(9), 2023. ISSN 2075-4418. doi: 10.3390/diagnostics13091532. URL https://www.mdpi.com/2075-4418/13/9/1532.

Anh Nguyen, Tuong Do, Minh Tran, Binh X. Nguyen, Chien Duong, Tu Phan, Erman Tjiputra, and Quang D. Tran. Deep federated learning for autonomous driving. In *2022 IEEE Intelligent Vehicles Symposium, IV 2022, Aachen, Germany, June 4-9, 2022*, pages 1824–1830. IEEE, 2022. doi: 10.1109/IV51971.2022.9827020. URL https://doi.org/10.1109/IV51971.2022.9827020.

Xinyi Shang, Yang Lu, Yiu-Ming Cheung, and Hanzi Wang. FEDIC: federated learning on non-iid and long-tailed data via calibrated distillation. In *IEEE International Conference on Multimedia and Expo, ICME 2022, Taipei, Taiwan, July 18-22, 2022*, pages 1–6. IEEE, 2022a. doi: 10.1109/ICME52920.2022.9860009. URL https://doi.org/10.1109/ICME52920.2022.9860009.

Xinyi Shang, Yang Lu, Gang Huang, and Hanzi Wang. Federated learning on heterogeneous and long-tailed data via classifier re-training with federated features. In Luc De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, pages 2218–2224. ijcai.org, 2022b. doi: 10.24963/IJCAI.2022/308. URL https://doi.org/10.24963/ijcai.2022/308.

Tao Wang, Yu Li, Bingyi Kang, Junnan Li, Jun Hao Liew, Sheng Tang, Steven C. H. Hoi, and Jiashi Feng. The devil is in classification: A simple framework for long-tail instance segmentation. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XIV*, volume 12359 of *Lecture Notes in Computer Science*, pages 728–744. Springer, 2020. doi: 10.1007/978-3-030-58568-6_43. URL https://doi.org/10.1007/978-3-030-58568-6_43.

Nannan Wu, Li Yu, Xuefeng Jiang, Kwang-Ting Cheng, and Zengqiang Yan. Fednoro: Towards noise-robust federated learning by addressing class imbalance and label noise heterogeneity. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI 2023, 19th-25th August 2023, Macao, SAR, China*, pages 4424–4432. ijcai.org, 2023. doi: 10.24963/ijcai.2023/492. URL https://doi.org/10.24963/ijcai.2023/492.

Zikai Xiao, Zihan Chen, Liyinglan Liu, Yang Feng, Jian Wu, Wanlu Liu, Joey Tianyi Zhou, Howard Hao Yang, and Zuozhu Liu. Fedloge: Joint local and generic federated learning under long-tailed data. *CoRR*, abs/2401.08977, 2024. doi: 10.48550/ARXIV.2401.08977. URL https://doi.org/10.48550/arXiv.2401.08977.

Yuhang Zang, Chen Huang, and Chen Change Loy. FASA: feature augmentation and sampling adaptation for long-tailed instance segmentation. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 3437–3446. IEEE, 2021. doi: 10.1109/ICCV48922.2021.00344. URL https://doi.org/10.1109/ICCV48922.2021.00344.

Tengchan Zeng, Omid Semiari, Mingzhe Chen, Walid Saad, and Mehdi Bennis. Federated learning on the road autonomous controller design for connected and autonomous vehicles. *IEEE Trans. Wirel. Commun.*, 21(12):10407–10423, 2022. doi: 10.1109/TWC.2022.3183996. URL https://doi.org/10.1109/TWC.2022.3183996.

Jianqing Zhang, Yang Liu, Yang Hua, and Jian Cao. Fedtgp: Trainable global prototypes with adaptive-margin-enhanced contrastive learning for data and model heterogeneity in federated learning. In Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan, editors, *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 16768–16776. AAAI Press, 2024. doi: 10.1609/AAAI.V38I15.29617. URL https://doi.org/10.1609/aaai.v38i15.29617.

Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, and Jiashi Feng. Deep long-tailed learning: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(9):10795–10816, 2023. doi: 10.1109/TPAMI.2023.3268118. URL https://doi.org/10.1109/TPAMI.2023.3268118.

Zizhao Zhang and Tomas Pfister. Learning fast sample re-weighting without reward data. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 705–714. IEEE, 2021. doi: 10.1109/ICCV48922.2021.00076. URL https://doi.org/10.1109/ICCV48922.2021.00076.