Feature Norm Regularized Federated Learning: Utilizing Data Disparities for Model Performance Gains

Ke Hu[1] , **Liyao Xiang**[2] , **Peng Tang**[1*] and **Weidong Qiu**[1†]
[1]School of Cyber Science and Engineering, Shanghai Jiao Tong University
[2]John Hopcroft Center for Computer Science, Shanghai Jiao Tong University
crestiny@sjtu.edu.cn, xiangliyao08@sjtu.edu.cn, tangpeng@sjtu.edu.cn, qiuwd@sjtu.edu.cn

Abstract

Federated learning (*FL*) is a machine learning paradigm that aggregates knowledge and utilizes computational power from multiple participants to train a global model. However, a commonplace challenge—non-independent and identically distributed (*non-i.i.d.*) data across participants—can lead to significant divergence in model updates, thus diminishing training efficacy. In this paper, we propose the Feature Norm Regularized Federated Learning (*FNR-FL*) algorithm to tackle the non-i.i.d. challenge. FNR-FL incorporates class average feature norms into the loss function by a straightforward yet effective regularization strategy. The core idea of FNR-FL is to penalize the deviations in the update directions of local models caused by the non-i.i.d. data. Theoretically, we provide convergence guarantees for FNR-FL when training under non-i.i.d. scenarios. Practically, our comprehensive experimental evaluations demonstrate that FNR-FL significantly outperforms existing FL algorithms in terms of test accuracy, and maintains a competitive convergence rate with lower communication overhead and shorter duration. Compared to FedAvg, FNR-FL exhibits a 66.24% improvement in accuracy and an 11.40% reduction in training time, underscoring its enhanced effectiveness and efficiency. The code is available on GitHub at: https://github.com/LonelyMoonDesert/FNR-FL.

1 Introduction

Federated learning, a cutting-edge machine learning paradigm, has found extensive applications across domains such as healthcare, finance, and smart devices, enabling collaborative model training while preserving data privacy and security [Banabilah *et al.*, 2022]. In federated learning, a crucial distinction from conducting deep learning on a single node is to aggregate locally updated models from individual participants to obtain a global model [Wang *et al.*, 2019]. Most commonly used aggregation algorithms in federated learning is FedAvg [McMahan *et al.*, 2017]. FedAvg performs weighted averaging on locally trained models to construct a global model, which is simple but notably effective under i.i.d. (independently and identically distributed) data [Clauset, 2011].

Despite the wide application of FedAvg, non-i.i.d. data presents challenges [Hsieh *et al.*, 2020]. These issues lead to variations in local data characteristics, hindering the convergence and generalization of global models. Recent studies [Li *et al.*, 2019] have shown that FedAvg performs poorly under non-i.i.d. data, as weighted averaging fails to integrate accurate knowledge from participants with skewed distributions.

In this work, we propose Feature Norm Regularized Federated Learning (*FNR-FL*), a federated learning framework leveraging the class average feature norm to enhance the performance of the global model under the non-i.i.d. data distribution. Compared to prior efforts dedicated to addressing non-i.i.d. issues, FNR-FL allows for significantly better test accuracy and faster convergence and excels in various non-i.i.d. scenarios, instead of excelling in only a limited number of settings.

The main achievements, including contributions to the field, can summarised as follows:

- We proposed a regularization algorithm based on class average feature norms, which serves as a plug-and-play module that seamlessly integrates with existing federated learning algorithms. This modular algorithm allows for enhanced model regularization without the need for substantial modifications to the underlying federated learning framework.
- Building upon this regularization algorithm, we have constructed a federated learning algorithm FNR-FL that achieves significantly superior test accuracy and faster convergence under the non-i.i.d. data distribution compared to other Federated Learning (*FL*) algorithms.
- Under mixed non-i.i.d. scenarios, FNR-FL has proven to outperform other algorithms, marking the first known federated learning algorithm tested under such mixed conditions. This strategic testing highlights the algorithm's robust performance in complex, real-world conditions.
- We have developed two innovative metrics which innovatively capture the trade-off between accuracy, communication, and computational costs, offering a clear benchmark for algorithm comparison.

*These authors are joint corresponding authors.
†These authors are joint corresponding authors.

2 Related Work

McMahan et al. [2017] developed FedAvg, a widely used federated learning algorithm that aggregates model parameters through weighted averaging. The essence of FedAvg involves clients uploading their local model parameters to a server, which computes the average of these parameters, weighting them according to the volume of data on each device. This approach has significantly shaped the development of efficient and scalable federated learning systems.

Li et al. [2020] introduced FedProx, a novel optimization framework tailored for federated networks, uniquely addressing both system and statistical heterogeneity. Its innovation lies in allowing flexible local computations while incorporating a proximal term to maintain overall algorithmic stability across diverse devices.

Karimireddy et al. [2020] developed SCAFFOLD, a pioneering stochastic algorithm that creatively tackles gradient dissimilarity in federated settings. Its key innovation is the incorporation of control variates to significantly reduce the gradient variance, leading to enhanced convergence rates and model performance.

Li et al. [2021a] proposed MOON (Model-Contrastive Learning), a groundbreaking approach in federated learning that introduces contrastive learning at the model level. This technique stands out for its simplicity and effectiveness, particularly in enhancing federated deep learning models' performance on non-i.i.d. datasets by encouraging model-level feature alignment.

Wang et al. [2020] introduced FedNova, a comprehensive theoretical framework designed for heterogeneous federated learning environments. Its main contribution is the natural integration of diverse local update steps and optimization techniques (such as gradient descent, stochastic gradient descent, and proximal updates), ensuring fair and efficient convergence across a wide range of network conditions and device capabilities.

The FedDF framework [2020] leverages a distillation technique to foster a more homogeneous knowledge transfer among decentralized datasets. This method is notable for its distillation from decentralized to centralized, aiding in overcoming the statistical heterogeneity inherent in FL.

FedGen [2021] by Dong et al. integrates a generative component into the federated learning process. This addition aims to synthesize pseudo-samples that represent the global data distribution, thus enhancing the robustness of the model against data heterogeneity.

3 Motivation

3.1 The Fundamental Reason of Non-i.i.d. FL Performance Decline

The primary cause of performance degradation in federated learning due to non-i.i.d. data lies in the varied data distributions across different participants. This variation leads to divergent update directions when training local models [Zhao *et al.*, 2018]. Simply put, the more diverse the data distribution, the greater the directional deviation in local model updates [Li *et al.*, 2021b], which hampers the effective consolidation of knowledge into the global model [Karimireddy *et al.*, 2020; Wang *et al.*, 2020; Li *et al.*, 2022; Gao *et al.*, 2022]. Consequently, this disparity in data distribution culminates in a decline in the global model's overall test accuracy.

Implication 1. *Our objective is to minimize discrepancies in model update directions among participants in federated learning, ultimately enhancing the overall performance of the global model.*

3.2 Quantify the Difference Among Local Model Updates

Acknowledging the critical role of minimizing update discrepancies in federated learning (Implication 1), our focus shifts to quantifying these differences effectively. Wei et al. [2023] suggest that by using a consistent feature extractor, it is feasible to approximate data distribution differences through the variance in feature norms.

Implication 2. *In a similar vein, we can postulate that, under the assumption of using the same dataset for evaluation, it is possible to gauge the difference among local model updates by leveraging the disparity among feature norms.*

3.3 Promoting Alignment of Model Update Directions Among Participants

To counteract the effects of non-i.i.d. data, FedProx integrates a proximal term in each local model's loss function. This term penalizes deviations from the global model, aligning local updates more closely with it.

Implication 3. *Informed by FedProx's methodology, we propose enhancing participant training by integrating a regularization term based on feature norm differences [Brownlee, 2018]. This term serves as a corrective measure for participants with diverse data distributions, encouraging more aligned contributions to the global model.*

4 Non-i.i.d. Scenarios

In our study, we have investigated three distinct categories of non-i.i.d. settings: feature distribution skew, label distribution skew, and quantity skew [Zhang *et al.*, 2022a]. Additionally, we have explored two mixed non-i.i.d. settings, which combine label distribution with quantity skew and label distribution with feature distribution skew.

4.1 Feature Distribution Skew

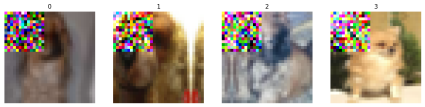
We simulated feature distribution skew by adding Gaussian noise to the data held by each participant. Using CIFAR-10 dataset samples, we show in Figure 1 how applying Gaussian noise with varying standard deviations affects the data. Original samples are in Figure 1a, while Figures 1b and 1c display the impact of different noise levels. As the standard deviation increases, the images become progressively obscured, exemplifying the feature distribution's deviation from its original form.

4.2 Label Distribution Skew

The Dirichlet distribution is utilized for allocating data with label distribution skew to different participants. Each class of samples is subject to a Dirichlet distribution sampling process

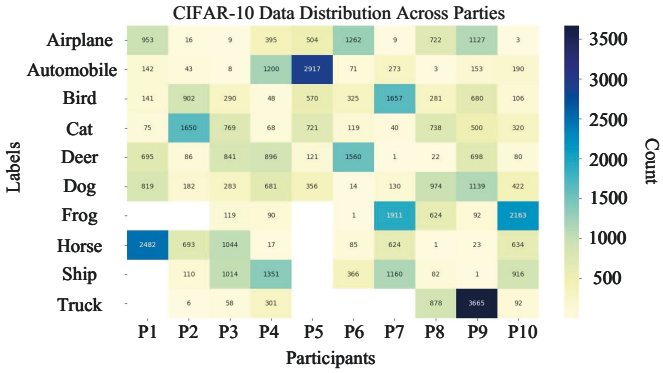
(a) original samples

(b) $noise \sim Gau(0.2)$

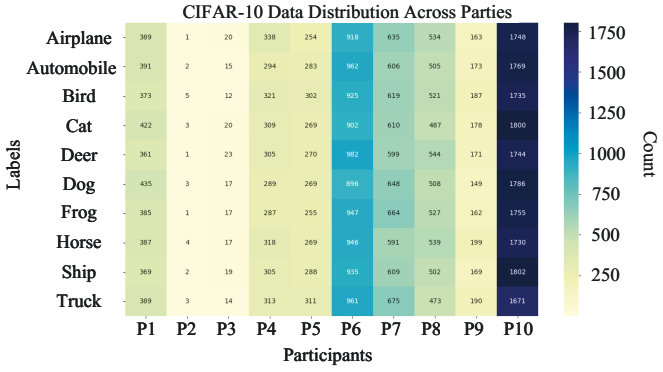


(c) $noise \sim Gau(0.5)$

Figure 1: Visualization of feature distribution skew on CIFAR-10



(a) Label distribution skew



(b) Quantity skew

Figure 2: Visualization of label distribution skew and quantity skew on CIFAR-10

individually. Taking the allocation of CIFAR-10 dataset to 10 participants as an example, when the parameter α is set to 0.5, the data allocation is depicted in Figure 2a. Darker colors indicate a higher quantity of samples, while white signifies the absence of samples in that class. From the illustration, it is evident that there are substantial variations in the number of samples from each class within the local datasets held by the participants.

4.3 Quantity Skew

Similar to label distribution skew, quantity skew arises when the sizes of local datasets, symbolized as $|D_i|$, differ among parties. We employ the Dirichlet distribution to distribute varying data sample quantities to each party. However, unlike label distribution skew, we maintain a roughly consistent data distribution across parties in this scenario [Archetti *et al.*, 2023]. This consistency allows us to isolate and analyze the specific impact of quantity skew on global model performance. Figure 2b demonstrates this, showing that within each participant's local datasets, the counts of different labels fall within a specific range.

5 Proposed Method

5.1 Class Average Feature Norm

In this section, we describe the computation of the class average feature norm. To facilitate our discussion, we enumerate key notations in Table 1 in the Supplementary Materials. In a federated learning scenario for image classification tasks involving n participants, each participant P_i possesses a local dataset denoted as D_i with N_i labeled samples:

$$D_i = \left\{(x_i^j, y_i^j)\right\}_{j=1}^{N_i} \tag{1}$$

Datasets D_i contain samples from different classes. Each participant's local model M_i consists of two main components: the feature extractor Mf_i and the classifier Mc_i. For a given sample x_j^i in D_i, we define its feature norm as the Euclidean norm (L_2 norm) of the extracted features as [Naik *et al.*, 2023]:

$$U_i^j = ||Mf_i(x_i^j, \vartheta_f^i)||_2 \tag{2}$$

where ϑ_f^i represents the parameters of the feature extractor Mf_i. After calculating the feature norms for each sample, we proceed to compute the average feature norm for each class of participant P_i. Specifically, for the class k in participant P_i, the computation of class average feature norm $\mathbf{F}_i^k$ is performed as follows:

$$\mathbf{F}_i^k = \frac{1}{\sum_{j=1}^{N_i} \delta_{[y_i^j=k]}} \sum_{j=1}^{N_i} \delta_{[y_i^j=k]} U_i^j \tag{3}$$

In this equation, $\delta_{[y_i^j=k]}$ is the indicator function, which is defined as:

$$\delta_{[y_i^j=k]} = \begin{cases} 1 & \text{if } y_i^j = k, \\ 0 & \text{if } y_i^j \neq k. \end{cases} \tag{4}$$

This process allows us to obtain class average feature norms for the different classes of participants P_i. The process of computation is summarized in Algorithm 1.

Notation	Semantics
n	number of participants
D_i	local dataset of participant P_i
(x_i^j, y_i^j)	j-th pair of sample and label of participant P_i
M_i	local model of participant P_i
Mf_i	feature extractor in local model of participant P_i
U_i^j	feature norm j-th sample of participant P_i
$\mathbf{F}_i^k$	class average feature norm of k-th class of participant P_i
$\Phi_i(w_i^t)$	objective function of participant P_i in t-th round
L	the parameter of L-smooth
μ	the parameter of μ-strongly convex
η_m	learning rate in m-th mini-batch SGD
$\bar{\mathbf{w}}_m$	parameter of global model in m-th mini-batch SGD
$\mathbf{w}_m^i$	parameter of local model of participant P_i in m-th mini-batch SGD
$\mathbf{w}^*$	parameter of optimal global model
b_m^i	a random batch selected from all batches of participant P_i

Table 1: Key notations

Algorithm 1 Calculate Class Average Feature Norms

Input: dataset D
Initialize: class feature norms $F \leftarrow \{\}$, class sample counts $C \leftarrow \{\}$, class average feature norms $avgF \leftarrow \{\}$
for each batch B *in* dataset D **do**
  for each sample (x_i, y_i) in batch B **do**
    compute the feature norm U_i for x_i using Eq. (2)
    append U_i to $F[y_i]$
    $C[y_i] += 1$
  end for
end for
for each (label k, feature norms F_k) *in* F **do**
  if C[k] > 0 **then**
    Calculate the class average feature norm $avgf$ using Eq. (3)
    $avgF$[k] $\leftarrow avgf$
  end if
end for
Output: class average feature norms $avgF$

Algorithm 2 Calculate Feature Norms Differences

Input: participant P_j, set of all participants $\mathcal{P}$, set of participants to be regularized $\mathcal{P}_{re}$, feature norm list of all participants F_n
$\Delta_j = \{\}$
for $m \in \mathcal{P} \setminus \mathcal{P}_{re}$ **do**
  for label $l \in F_n[P_j].keys()$ **do**
    $\Delta_{temp} = (F_n[m][l] - F_n[j][l])$
    $\Delta_j[l] \mathrel{+}= \Delta_{temp}$
  end for
end for
return Δ_j

5.2 FNR-FL: Feature Norm Regularized Federated Learning

Procedure of FNR-FL

In this subsection, we introduce the FNR-FL algorithm, a novel federated learning approach designed to tackle the challenges posed by non-i.i.d. data across different participants. This framework leverages feature norm regularization to align local model updates, thus enhancing global model performance. The detailed steps of this procedure are outlined in Algorithm 3.

In Algorithm 3, The server initializes and distributes the global model to all participant devices. Each participant, P_1 through P_n, trains this global model on their own local dataset. After training, they each evaluate their model against a global test dataset. They also calculate the class average feature norms using Algorithm 2. Participants with the relatively lower test accuracy are selected to refine their models further using the **FeatureNormRegularization**$(j, w_j^t, D_j, \Delta_j)$ in Algorithm 3. The server then collects these local models and performs an aggregation step.

Local training with computation of class average feature norm. In each communication round $t \in [T]$, all n participants are active. The server sends the global model w^t to each participant. Each participant P_i then updates its local model w_i^t on local dataset D_i for E_{train} epochs. Subsequently, each participant is required to compute the class average feature norms F_i^t (Algorithm 1) and the accuracy A_i of samples in the public dataset D_{public} using its updated local model w_i^t. The updated local model w_i^t, the class average feature norms F_i^t and the accuracy A_i are then sent to the server.

Computation of differences among feature norms. Upon receiving the test accuracies, denoted as $A_1, ..., A_n$, on the public dataset D_{public}, the server proceeds to select $n * p$ participants with the lowest accuracies to undergo feature norm regularization. Here, p serves as a threshold parameter determining the proportion of participants chosen for regularization.

Algorithm 3 FNR-FL

Input: local datasets D_i, number of participants n, number of communication rounds T, number of local training epochs E_{train}, number of regularization epochs E_{re}, public dataset D_{public}, learning rate η, percentage of regularized participants p,
Initialize: global model w^0
Server execues:
for $t = 1, 2, \cdots, T$ **do**
  initialize feature norm list F_n
  for $i = 1, 2, \cdots, n$ **in parallel do**
    send the global model w^t to participant P_i
    $w_i^t, F_i^t, accuracy \leftarrow$ **LocalTraining**(i, w^t, D_i)
    $F_n[i] = F_i^t$
  end for
  choose $n_{re} = \lfloor n * p \rfloor$ participants to refine local model updates
  for $j = 1, 2, \cdots, n_{re}$ **in parallel do**
    compute Δ_j using Algorithm 2
    $w_j^t \leftarrow$ **FeatureNormRefine**$(j, w_j^t, D_j, \Delta_j)$
  end for
  $w^{t+1} \leftarrow \sum_{i=1}^{n} \frac{|D^i|}{\sum_{i=1}^{n} |D^i|} w_i^t$
end for
return w^T

LocalTraining(i, w^t, D_i):
$w_i^t \leftarrow w^t$
for $k = 1, 2, \cdots, E_{train}$ **do**
  for each batch B *in* dataset D_i **do**
    $L(w_i^k; B) = \sum_{(x,y)\in B} \ell(w_i^k; x; y)$
    $w_i^k \leftarrow w_i^k - \eta \nabla L(w_i^k; B)$
  end for
end for
compute F_i^t on D_{public} using Algorithm 1
test w_i^k on D_{public} and record accuracy A_i
return w_i^k, F_i^t, A_i

FeatureNormRegularization$(j, w_j^t, D_j, \Delta_j)$:
for $k = 1, 2, \cdots, E_{re}$ **do**
  for each batch B *in* dataset D_{public} **do**
    $L(w_j^k; B) = \sum_{(x,y)\in B} \ell(w_j^k; x; y)$
    denote unique labels in B as L
    $J(j, F_n) = 0.0$
    for each label $l \in L$ **do**
      $\rho = \#(y == l)/|B|$
      $J(j, F_n) \mathrel{+}= \rho \cdot \Delta_j[l]$
    end for
    $\Phi(w_j^k) = L(w_j^k; B) + \lambda \cdot J(j, F_n)$
    $w_j^k \leftarrow w_j^k - \eta \nabla \Phi(w_j^k)$
  end for
end for
return w_j^k to the server

The set of these selected participants is denoted as $\mathcal{P}_{re}$ and the set of all participants is denoted as $\mathcal{P}$. For each participant P_j in $\mathcal{P}_{re}$, the server computes the differences of the class average feature norms between P_j and other participants in $\mathcal{P} \setminus \mathcal{P}_{re}$, denoted as Δ_j, using Algorithm 2. For each label l, we use calculate the differences of class average feature norm between participant P_j and other participants in $\mathcal{P} \setminus \mathcal{P}_{re}$, and record it with its label l as key in Δ_j.

Feature norm regularized model update. Consequently, the loss function $\Phi_j(w_j^t)$ of participant P_j incorporates its cross-entropy loss $L(w_i^k, B)$ and the weighted average sum $J(j, F_n)$ of each element in Δ_j with the proportion of the number of label l samples to the batch size $|B|$ as the weight:

$$\Phi_j(w_j^t) = \underbrace{L(w_j^k, B)}_{Cross-Entropy\ Loss} + \underbrace{\lambda \cdot J(j, F_n)}_{Regularization\ Term} \tag{5}$$

The hyperparameter λ controls the trade-off between fitting the local datasets well and encouraging convergence towards a global model with better performance.

Convergence Analysis of FNR-FL

In this section, we delineate the convergence characteristics of the FNR-FL algorithm we proposed. Initially, we establish certain assumptions, commonly adopted in preceding studies [Zhou, 2018; Bao *et al.*, 2022; T Dinh *et al.*, 2020; Haddadpour and Mahdavi, 2019].

Assumption 1 (L-smooth)**.** *The functions $\Phi_1, \ldots, \Phi_N$ are all L-smooth. This implies that for all x, y, the following inequality holds:*

$$\Phi_i(y) \leq \Phi_i(x) + \langle \nabla \Phi_i(x), y - x \rangle + \frac{L}{2} \|y - x\|^2. \tag{6}$$

Assumption 2 (μ-strongly convex)**.** *The functions are all μ-strongly convex, which means for all x, y:*

$$\Phi_i(y) \geq \Phi_i(x) + \langle \nabla \Phi_i(x), y - x \rangle + \frac{\mu}{2} \|y - x\|^2. \tag{7}$$

Assumption 3 (Bounded gradients)**.** *The stochastic gradients are unbiased and have bounded variance. Specifically, in m-th mini-batch gradient descent step of participant P_i:*

$$\mathbb{E}\left[\|\nabla \Phi_i(w_m^i, b_m^i) - \nabla \Phi_i(w_m^i)\|^2\right] \leq (\Delta G_i)^2, \tag{8}$$

and

$$\mathbb{E}\left[\|\nabla \Phi_i(w_m^i, b_m^i)\|^2\right] \leq G^2. \tag{9}$$

where b_m^i denotes a random batch in all batches of participant P_i in m-th mini-batch gradient descent step.

Consider the m-th mini-batch gradient descent step: $\bar{\mathbf{w}}_{m+1} = \bar{\mathbf{w}}_m - \eta_m \mathbf{g}_m$. We analyze the distance to the optimal point $\mathbf{w}^*$ in terms of the squared norm:

$$\begin{aligned} \|\bar{\mathbf{w}}_{m+1} - \mathbf{w}^*\|^2 = &\|\bar{\mathbf{w}}_m - \mathbf{w}^* - \eta_m \bar{\mathbf{g}}_m\|^2 \\ &+ \eta_m^2 \|\bar{\mathbf{g}}_m - \mathbf{g}_m\|^2 \\ &+ 2\eta_m \langle \bar{\mathbf{w}}_m - \mathbf{w}^* - \eta_m \bar{\mathbf{g}}_m, \bar{\mathbf{g}}_m - \mathbf{g}_m \rangle. \end{aligned} \tag{10}$$

where $\mathbf{g}_m = \nabla \Phi_i(\mathbf{w}_m^i, b_m^i)$, which denotes the gradient of the loss function $\nabla \Phi_i$ with respect to the current mini-batch b_m^i; and its average $\bar{\mathbf{g}}_m = \nabla \Phi_i(\mathbf{w}_m^i)$.

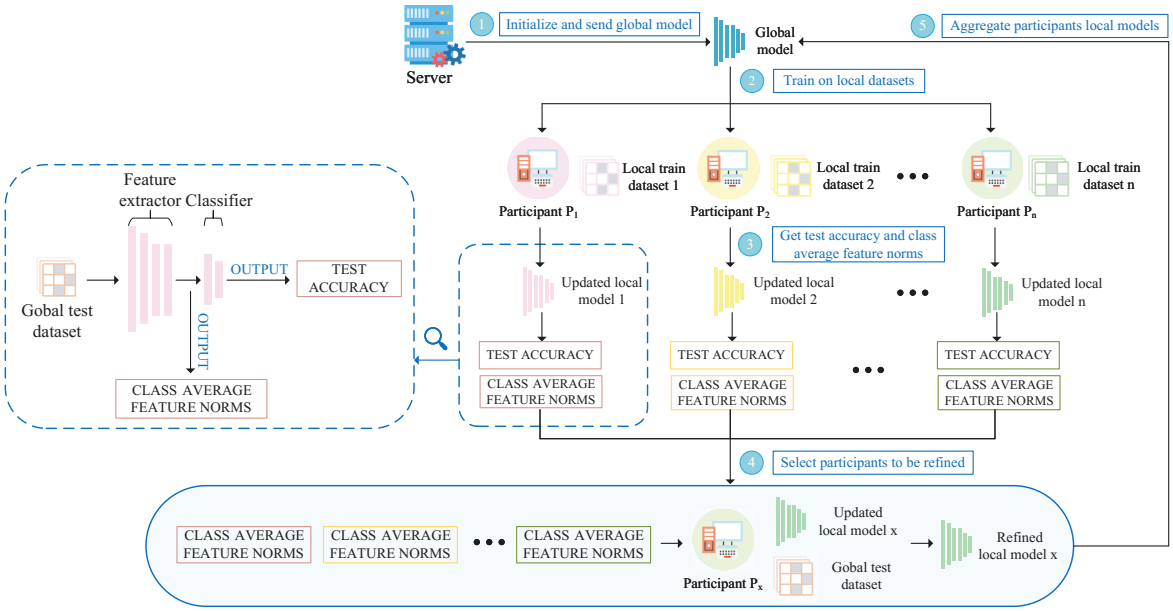


Figure 3: The framework of FNR-FL.

Lemma 1. *The bound of* $\|\bar{\mathbf{w}}_m - \mathbf{w}^* - \eta_m \bar{\mathbf{g}}_m\|^2$*:*

$$\begin{aligned}\|\bar{\mathbf{w}}_m - \mathbf{w}^* - \eta_m \bar{\mathbf{g}}_m\|^2 &\leq (1-\mu\eta_m)\|\bar{\mathbf{w}}_m - \mathbf{w}^*\|^2 \\ &+ 2\sum_{i=1}^{n}\left(\frac{|D^i|}{\sum_{i=1}^{n}|D^i|}\right)\|\bar{\mathbf{w}}_m - \mathbf{w}_m^i\|_2^2 + \frac{3\Gamma}{8L}.\end{aligned} \tag{11}$$

where $\Gamma = \Phi(\mathbf{w}^*) - \sum_{i=1}^{n}\left(\frac{|D^i|}{\sum_{i=1}^{n}|D^i|}\right)\Phi_i(\mathbf{w}^*)$.

Lemma 1 provides an upper bound of $\|\bar{\mathbf{w}}_m - \mathbf{w}^* - \eta_m \bar{\mathbf{g}}_m\|^2$ characterized by the learning rate η_m, μ and L.

Lemma 2. *In order to associate with our assumptions, we consider the expectation of* $\|\bar{\mathbf{g}}_m - \mathbf{g}_m\|^2$*:*

$$\mathbb{E}\|\bar{\mathbf{g}}_m - \mathbf{g}_m\|^2 \leq \sum_{i=1}^{n}\left(\frac{|D^i|}{\sum_{i=1}^{n}|D^i|}\right)^2 (\Delta G_i)^2. \tag{12}$$

Lemma 2 bounds the expectation of the gradient difference, which is influenced by the variability in each mini-batch's gradient, as indicated by Assumption 3.

Lemma 3. *Based on Lemma 1 and Lemma 2, we get:*

$$\begin{aligned}\mathbb{E}\|\bar{\mathbf{w}}_{m+1} - \mathbf{w}^*\|^2 &\leq (1-\mu\eta_m)\mathbb{E}\|\bar{\mathbf{w}}_m - \mathbf{w}^*\|^2 \\ &+ 8\eta_m^2(E_{re}-1)^2G^2 \\ &+ \eta_m^2\sum_{i=1}^{n}\left(\frac{|D^i|}{\sum_{i=1}^{n}|D^i|}\right)^2 (\Delta G_i)^2 + \frac{3\Gamma}{8L}.\end{aligned} \tag{13}$$

Lemma 3 introduces a recursive relationship derived from the previous lemmas.

Letting $\delta_m = \mathbb{E}\|\bar{\mathbf{w}}_m - \mathbf{w}^*\|^2$, which denotes the expected squared norm of the deviation of the global model $\bar{\mathbf{w}}_m$ from the optimal $\mathbf{w}^*$ at the m-th mini-batch SGD; and thus $\delta_{m+1} = \mathbb{E}\|\bar{\mathbf{w}}_{m+1} - \mathbf{w}^*\|^2$, we can get the following recurrence relation:

$$\begin{aligned}\delta_{m+1} &\leq (1-\mu\eta_m)\delta_m \\ &+ 8\eta_m^2(E_{re}-1)^2G^2 \\ &+ \eta_m^2\sum_{i=1}^{n}\left(\frac{|D^i|}{\sum_{i=1}^{n}|D^i|}\right)^2 (\Delta G_i)^2 + \frac{3\Gamma}{8L}.\end{aligned} \tag{14}$$

This inequality demonstrates the decrement of δ_m over iterations, thus assuring convergence to the optimal global model w^* as m increases. The requisite conditions for assuring the convergence of the sequence $\{\delta_m\}$ are enumerated as follows:

1. The learning rate η_m decays.
2. $0 < (1-\mu\eta_m) < 1$.
3. $(E_{re}-1)^2G^2$, $(\Delta G_i)^2$ and $\frac{3\Gamma}{8L}$ are bounded.

Given that all the prescribed conditions are satisfied, we ascertain that δ_m convergences to 0 as m increases to infinity, signifying the assured convergence of the global model $\bar{\mathbf{w}}_m$ to the optimal global model w^*. While in practice the number of iterations m is finite, the convergence behavior of δ_m as outlined still assures that we can approximate the desired precision within our computational budget. For the complete proof, see the Supplementary Materials.

6 Evaluations

6.1 Experimental Setup

In our experimental setup, all participants are actively involved in every round of the training process. We use the SGD optimizer [Robbins and Monro, 1951] with a learning rate of 0.1. The batch size of the training data is set to 64 and the test data is set to 32 by default. The number of local training epochs is set to 10 by default and the regularization epochs is set to 5 by default. We conduct each training process for 10 rounds.

6.2 Performance Under Single Types of Non-i.i.d. Scenarios

To investigate the effectiveness of the proposed FNR-FL, we report the performance of FNR-FL, FedAvg, FedProx, SCAFFOLD, MOON, FedNova, FedDyn, FedFTG, FedDF, FedDC, and FedGen when training ResNet-18 [He *et al.*, 2016] or VGG-11 [Krizhevsky, 2009] model on CIFAR-10 [Krizhevsky, 2009]. Experiments are conducted under three single types of non-i.i.d. data distribution.

Test Accuracy

Table 2 shows FNR-FL excelling in non-i.i.d. scenarios on CIFAR-10, notably outperforming other FL methods in feature distribution skew with accuracies of 0.9976 on ResNet-18 and 0.9931 on VGG-11. In label distribution and quantity skew, FNR-FL achieves 0.9970 and 0.9992 for ResNet-18 respectively, demonstrating strong results that underscore its robustness in federated learning.

Further evaluations on benchmark datasets MNIST and FEMNIST are presented in Table 3. FNR-FL achieves accuracies of 0.9953 in feature distribution skew and 0.9944 in quantity skew on MNIST, and 0.9889 and 0.9944 respectively on FEMNIST. These results are comparable to those on CIFAR-10, confirming FNR-FL's consistent effectiveness across diverse data scenarios, further demonstrating its practical utility in federated learning systems.

Convergence

The convergence curves of training ResNet-18 on CIFAR-10 under three single types of non-i.i.d. scenarios are illustrated in Figure 4. We can observe that the FNR-FL algorithm consistently exhibits the same level of convergence speed as other FL algorithms and achieves exceptionally better test accuracies.

Communication and Computational Cost

Metrics for performance evaluation are presented in Table 4, where the $\mathit{Traffic}$ represents the cumulative data exchanged, including both uploads and downloads and Time computes the duration to complete 10 rounds of training. The metrics $\kappa = \mathit{Accuracy} * 10^4 / \mathit{Time}$ and $\rho = \mathit{Accuracy} * 10^4 / \mathit{Traffic}$ were computed to measure test accuracy per unit of time and traffic, respectively, providing a quantifiable tradeoff between effectiveness and efficiency. FNR-FL stands out for its exceptional efficiency in non-i.i.d. scenarios.

6.3 Performance Under Mixed Types of Non-i.i.d. Scenarios

We evaluated the performance of federated learning algorithms on the CIFAR-10 dataset under mixed scenarios of label and feature distribution skew as well as label and quantity distribution skew. In Table 5 (Supplementary Materials), FNR-FL outshines all FL algorithms in label+feature distribution skew at noise levels of 0.1 and 0.5, with top accuracies of 0.9755 and 0.9111, respectively. Other algorithms show significant performance drops, especially at higher noise. For label+quantity skew, FNR-FL leads with 0.8826 accuracy, while others hover around 0.3000.

As noted by recent studies [Li *et al.*, 2022], mixed types of skew present more complex challenges than single types. FNR-FL's ability to maintain high accuracy levels addresses the critical need for effective algorithms capable of operating under mixed types of non-i.i.d. scenarios.

6.4 Orthogonality of FNR-FL with Existing FL Algorithms

The experimental results in Table 6 in the Supplementary Materials demonstrate the effectiveness of integrating the FNR-FL algorithm with other FL algorithms. Across all scenarios, combinations involving FNR-FL consistently outperform their standalone FL algorithms, indicating a substantial enhancement in handling non-i.i.d. data distributions. These results suggest that FNR-FL's regularization mechanism effectively augments existing FL algorithms [Acar *et al.*, 2021].

6.5 Effect of Feature Norm Regularization

The ablation study assesses the impact of feature norm regularization (FNR) within the FNR-FL algorithm on a participant's local model. Notably, classes with lower local sample sizes, such as 'frog' (471) and 'horse' (485), show more significant accuracy improvements. This observation is aligned with the hypothesis that FNR enables participants to benefit from the shared knowledge across the federation, particularly when their local data is insufficient to capture the complexity of the class. In contrast, classes with more substantial local representations, like 'automobile' (512) and 'ship' (528), exhibit smaller gains. This can be attributed to the participant's local model already achieving a higher accuracy due to the ample class-specific data, hence the marginal utility of the shared knowledge from FNR is less impactful [Du *et al.*, 2022; Shao *et al.*, 2023; Qi *et al.*, 2023].

6.6 Effect of Noise

The experiment evaluates the performance and robustness of FNR-FL against other FL algorithms across different noise levels within feature and label distribution skews. The results are presented in Table 7. FNR-FL demonstrates remarkable noise immunity, preserving high accuracy despite increasing noise. Its robustness to noise underscores its suitability for real-world federated learning applications.

6.7 Effect of Training Hyperparameters

Table 6 shows the impact of various hyperparameters on global model accuracy. Our findings reveal that changes in batch size, learning rate, and the number of training epochs do not significantly affect the test accuracy of the global model. This suggests that our proposed FNR-FL maintains stable performance across a variety of hyperparameter settings.

Non-i.i.d. scenarios	Feature distribution skew		Label distribution skew		Quantity skew	
Model	ResNet-18	VGG-11	ResNet-18	VGG-11	ResNet-18	VGG-11
FNR-FL	**0.9976**	**0.9931**	**0.9970**	**0.9992**	**0.9982**	**0.9939**
FedAvg (AISTATS, 2017 [McMahan *et al.*, 2017])	0.6001	0.7485	0.8393	0.8311	0.9107	0.9194
FedProx (MLsys, 2020 [Li *et al.*, 2020])	0.5956	0.7285	0.8773	0.8050	0.8874	0.8916
SCAFFOLD (ICML, 2020, [Karimireddy *et al.*, 2020])	0.7425	0.7733	0.9077	0.8657	0.7017	0.9064
MOON (CVPR, 2021, [Li *et al.*, 2021a])	0.5530	0.7193	0.6515	0.8432	0.7336	0.8424
FedNova (NeurIPS, 2020, [Wang *et al.*, 2020])	0.6223	0.7475	0.9043	0.8522	0.7293	0.7622
FedDyn (ICPADS, 2023, [Jin *et al.*, 2023])	0.6071	0.7555	0.8463	0.8381	0.9177	0.9264
FedFTG (CVPR, 2022, [Zhang *et al.*, 2022b])	0.6451	0.7935	0.8843	0.8761	0.9557	0.9644
FedDF (NeurIPS, 2020, [Lin *et al.*, 2020])	0.6101	0.7585	0.8493	0.8411	0.9207	0.9294
FedDC (CVPR, 2022, [Gao *et al.*, 2022])	0.6401	0.7885	0.8793	0.8711	0.9507	0.9594
FedGen (ICML, 2021, [Zhu *et al.*, 2021])	0.6021	0.7505	0.8413	0.8331	0.9127	0.9214

Table 2: Test accuracy on CIFAR-10 under single types of non-i.i.d. scenarios

Non-i.i.d scenarios	Dataset	FNR-FL	FedAvg	FedProx	SCAFFOLD	MOON	FedNova
Feature distribution skew	MNIST	0.9953	0.9685	0.9772	0.9805	0.9787	0.9786
	FEMNIST	0.9889	0.7591	0.7571	0.7591	0.7629	0.7666
Label distribution skew	MNIST	0.9943	0.9841	0.9829	0.9828	0.9838	0.9854
	FEMNIST	0.9941	0.9841	0.9801	0.9880	0.9903	0.9839
Quantity skew	MNIST	0.9951	0.9903	0.9894	0.9880	0.9903	0.9893
	FEMNIST	0.9944	0.9916	0.9868	0.7598	0.9903	0.7598

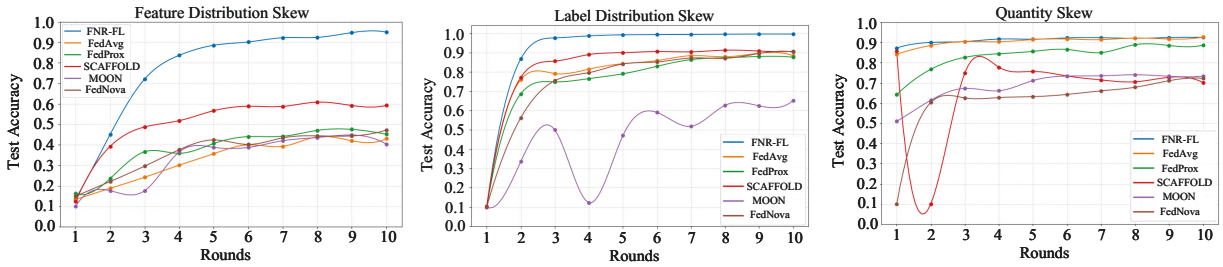
Table 3: Test accuracy on MNIST and FEMNIST under single types of non-i.i.d. scenarios

		Federated Learning Algorithms					
Non-i.i.d. scenarios	Metric	FNR-FL	FedAvg	FedProx	SCAFFOLD	MOON	FedNova
Feature Distribution Skew	*Traffic* (MB)	8920	8920	8920	13380	8920	8920
	Time (s)	6060	6840	8160	6840	10692	6525
	Accuracy	0.9976	0.6001	0.5956	0.7425	0.5530	0.6223
	κ	**1.6462**	0.8774	0.7299	1.0855	0.5172	0.9536
	ρ	**1.1184**	0.6728	0.6677	0.5549	0.6199	0.6976
Label Distribution Skew	*Traffic* (MB)	8920	8920	8920	13380	8920	8920
	Time (s)	7133	6006	11310	7176	8814	7098
	Accuracy	0.9970	0.8393	0.8773	0.9077	0.6515	0.9043
	κ	**1.3977**	1.3974	0.7757	1.2649	0.7392	1.2740
	ρ	**1.1177**	0.9409	0.9835	0.6784	0.7304	1.0138
Quantity Skew	*Traffic* (MB)	8920	8920	8920	13380	8920	8920
	Time (s)	5400	6786	9360	7332	11544	6006
	Accuracy	0.9982	0.9107	0.8874	0.7017	0.7336	0.7293
	κ	**1.8485**	1.3420	0.9481	0.9570	0.6355	1.2143
	ρ	**1.1191**	1.0210	0.9948	0.5244	0.8224	0.8176

Table 4: Performance evaluation of federated learning algorithms (10 participants, 10 rounds)

Non-i.i.d. scenarios		FNR-FL	FedAvg	FedProx	SCAFFOLD	MOON	FedNova
label+feature skew	noise=0.1	0.9755	0.8067	0.7837	0.8371	0.6360	0.8029
	noise=0.5	0.9111	0.4504	0.4587	0.5763	0.3728	0.3942
label+quantity skew		0.8826	0.3094	0.3048	0.311153	0.2993	0.309439

Table 5: Test accuracy on CIFAR-10 under mixed types of non-i.i.d. scenarios (ResNet-18)



(a) Feature distribution skew (b) Label distribution skew (c) Quantity skew

Figure 4: Convergence curves of training ResNet-18 on CIFAR-10

Non-i.i.d. scenarios	FedAvg+ FNR-FL	FedProx+ FNR-FL	SCAFFOLD+ FNR-FL	MOON+ FNR-FL	FedNova+ FNR-FL
Feature distribution skew	0.9976	0.9784	0.9783	0.9678	0.9664
Label distribution skew	0.9970	0.9838	0.9892	0.9774	0.9766
Quantity skew	0.9982	0.9999	0.9637	0.9532	0.9427

Table 6: Orthogonality of FNR-FL with existing FL algorithms (ResNet-18, CIFAR-10)

Non-i.i.d. scenarios	Noise	FNR-FL	FedAvg	FedProx	SCAFFOLD	MOON	FedNova
Feature distribution skew	0.0	**0.9995**	0.9238	0.9039	0.9290	0.7371	0.9257
	0.1	**0.9984**	0.8549	0.8219	0.8581	0.5677	0.8469
	0.5	**0.9505**	0.4303	0.4517	0.5925	0.4030	0.4723
Label distribution skew	0.0	**0.9970**	0.8850	0.8773	0.9077	0.6515	0.9043
	0.1	**0.9755**	0.8067	0.7837	0.8371	0.6360	0.8029
	0.5	**0.9111**	0.4504	0.4587	0.5763	0.3728	0.3942

Table 7: Effect of noise (ResNet-18, CIFAR-10)

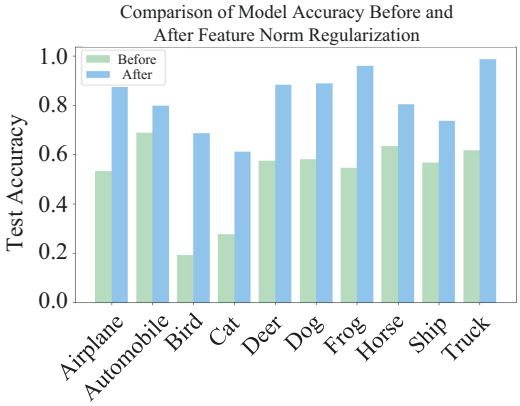


Figure 5: Comparison of model accuracy before and after feature norm regularization

	Batch size		Learning rate		Training epochs	
Value	16	64	0.01	0.05	1	5
Epoch 1	0.9380	0.9679	0.9679	0.7009	0.9297	0.9679
Epoch 2	0.9782	0.9836	0.9836	0.9794	0.9802	0.9836
Epoch 3	0.9843	0.9866	0.9866	0.9819	0.9847	0.9866
Epoch 4	0.9856	0.9882	0.9882	0.9858	0.9864	0.9882
Epoch 5	0.9862	0.9889	0.9889	0.9867	0.9882	0.9889

Figure 6: Effect of different hyperparameters on global model accuracy

7 Conclusions and Future Work

In this paper, we introduce the FNR-FL algorithm as a new approach to enhance performance in non-i.i.d. data scenarios. The core philosophy of FNR-FL revolves around leveraging feature norms as a metric to quantify and mitigate the divergence in local model updates. Our evaluations demonstrate that FNR-FL achieves superior test accuracy without excessive overhead and maintains convergence. An orthogonal experiment highlights its modularity, seamlessly enhancing FL algorithms. The necessity of feature norm regularization is confirmed through an ablation experiment.

For future work, we aim to further optimize FNR-FL for communication cost reduction by integrating model compression techniques. We also plan to explore applying FNR-FL to privacy-preserving federated learning and to develop a decision-making algorithm that analyzes data distribution characteristics in federated settings to recommend the most appropriate FL algorithm.

References

[Acar *et al.*, 2021] Durmus Alp Emre Acar, Yue Zhao, Ramon Matas Navarro, Matthew Mattina, Paul N. Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization, 2021.

[Archetti *et al.*, 2023] Alberto Archetti, Eugenio Lomurno, Francesco Lattari, André Martin, and Matteo Matteucci. Heterogeneous datasets for federated survival analysis simulation. In *Companion of the 2023 ACM/SPEC International Conference on Performance Engineering*, pages 173–180, 2023.

[Banabilah *et al.*, 2022] Syreen Banabilah, Moayad Aloqaily, Eitaa Alsayed, Nida Malik, and Yaser Jararweh. Federated learning review: Fundamentals, enabling technologies, and future applications. *Information processing & management*, 59(6):103061, 2022.

[Bao *et al.*, 2022] Yajie Bao, Michael Crawshaw, Shan Luo, and Mingrui Liu. Fast composite optimization and statistical recovery in federated learning. In *International Conference on Machine Learning*, pages 1508–1536. PMLR, 2022.

[Brownlee, 2018] Jason Brownlee. Gentle introduction to vector norms in machine learning. *Machine Learning Mastery*, 2018.

[Clauset, 2011] Aaron Clauset. A brief primer on probability distributions. In *Santa Fe Institute*, 2011.

[Du *et al.*, 2022] Zhixu Du, Jingwei Sun, Ang Li, Pin-Yu Chen, Jianyi Zhang, Hai" Helen" Li, and Yiran Chen. Rethinking normalization methods in federated learning. In *Proceedings of the 3rd International Workshop on Distributed Machine Learning*, pages 16–22, 2022.

[Gao *et al.*, 2022] Liang Gao, Huazhu Fu, Li Li, Yingwen Chen, Ming Xu, and Cheng-Zhong Xu. Feddc: Federated learning with non-iid data via local drift decoupling and correction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10112–10121, 2022.

[Haddadpour and Mahdavi, 2019] Farzin Haddadpour and Mehrdad Mahdavi. On the convergence of local descent methods in federated learning. *arXiv preprint arXiv:1910.14425*, 2019.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[Hsieh *et al.*, 2020] Kevin Hsieh, Amar Phanishayee, Onur Mutlu, and Phillip Gibbons. The non-iid data quagmire of decentralized machine learning. In *International Conference on Machine Learning*, pages 4387–4398. PMLR, 2020.

[Jin *et al.*, 2023] Cheng Jin, Xuandong Chen, Yi Gu, and Qun Li. Feddyn: A dynamic and efficient federated distillation approach on recommender system. In *2022 IEEE 28th International Conference on Parallel and Distributed Systems (ICPADS)*, pages 786–793. IEEE, 2023.

[Karimireddy *et al.*, 2020] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pages 5132–5143. PMLR, 2020.

[Krizhevsky, 2009] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

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

[Li *et al.*, 2020] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020.

[Li *et al.*, 2021a] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10713–10722, 2021.

[Li *et al.*, 2021b] Xiaoxiao Li, Meirui Jiang, Xiaofei Zhang, Michael Kamp, and Qi Dou. Fedbn: Federated learning on non-iid features via local batch normalization. *arXiv preprint arXiv:2102.07623*, 2021.

[Li *et al.*, 2022] Qinbin Li, Yiqun Diao, Quan Chen, and Bingsheng He. Federated learning on non-iid data silos: An experimental study. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pages 965–978. IEEE, 2022.

[Lin *et al.*, 2020] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. *Advances in Neural Information Processing Systems*, 33:2351–2363, 2020.

[McMahan *et al.*, 2017] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[Naik *et al.*, 2023] Shraddha M Naik, Chinnamuthu Subramani, Ravi Prasad K Jagannath, and Anand Paul. Exponential filtering technique for euclidean norm-regularized extreme learning machines. *Pattern Analysis and Applications*, pages 1–10, 2023.

[Qi *et al.*, 2023] Tao Qi, Fangzhao Wu, Chuhan Wu, Liang He, Yongfeng Huang, and Xing Xie. Differentially private knowledge transfer for federated learning. *Nature Communications*, 14(1):3785, 2023.

[Robbins and Monro, 1951] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

[Shao *et al.*, 2023] Jiawei Shao, Zijian Li, Wenqiang Sun, Tailin Zhou, Yuchang Sun, Lumin Liu, Zehong Lin, and Jun Zhang. A survey of what to share in federated learning:

Perspectives on model utility, privacy leakage, and communication efficiency. *arXiv preprint arXiv:2307.10655*, 2023.

[T Dinh *et al.*, 2020] Canh T Dinh, Nguyen Tran, and Josh Nguyen. Personalized federated learning with moreau envelopes. *Advances in Neural Information Processing Systems*, 33:21394–21405, 2020.

[Wang *et al.*, 2019] Shiqiang Wang, Tiffany Tuor, Theodoros Salonidis, Kin K Leung, Christian Makaya, Ting He, and Kevin Chan. Adaptive federated learning in resource constrained edge computing systems. *IEEE journal on selected areas in communications*, 37(6):1205–1221, 2019.

[Wang *et al.*, 2020] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in neural information processing systems*, 33:7611–7623, 2020.

[Wei and Huang, 2023] Xiao-Xiang Wei and Hua Huang. Edge devices clustering for federated visual classification: A feature norm based framework. *IEEE Transactions on Image Processing*, 32:995–1010, 2023.

[Zhang *et al.*, 2022a] Jie Zhang, Zhiqi Li, Bo Li, Jianghe Xu, Shuang Wu, Shouhong Ding, and Chao Wu. Federated learning with label distribution skew via logits calibration. In *International Conference on Machine Learning*, pages 26311–26329. PMLR, 2022.

[Zhang *et al.*, 2022b] Lin Zhang, Li Shen, Liang Ding, Dacheng Tao, and Ling-Yu Duan. Fine-tuning global model via data-free knowledge distillation for non-iid federated learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10174–10183, 2022.

[Zhao *et al.*, 2018] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*, 2018.

[Zhou, 2018] Xingyu Zhou. On the fenchel duality between strong convexity and lipschitz continuous gradient. *arXiv preprint arXiv:1803.06573*, 2018.

[Zhu *et al.*, 2021] Zhuangdi Zhu, Junyuan Hong, and Jiayu Zhou. Data-free knowledge distillation for heterogeneous federated learning. In *International conference on machine learning*, pages 12878–12889. PMLR, 2021.