Rethinking Data Heterogeneity in Federated Learning: Introducing a New Notion and Standard Benchmarks

Saeed Vahidian, Mahdi Morafah, Chen Chen, Mubarak Shah, Life Fellow, IEEE, and Bill Lin, Member, IEEE

Abstract—Though successful, federated learning (FL) presents new challenges for machine learning, especially when the issue of data heterogeneity, also known as Non-IID data, arises. To cope with the statistical heterogeneity, previous works incorporated a proximal term in local optimization or modified the model aggregation scheme at the server side or advocated clustered federated learning approaches, where the central server groups agent population into clusters with jointly trainable data distributions to take advantage of a certain level of personalization. While effective, they lack a deep elaboration on what kind of data heterogeneity and how the data heterogeneity impacts the accuracy performance of the participating clients. In contrast to many of the prior FL approaches, we demonstrate not only the issue of data heterogeneity in current setups is not necessarily a problem but also in fact it can be beneficial for the FL participants. Our observations are intuitive: 1) Dissimilar labels of clients (label skew) are not necessarily considered data heterogeneity, and 2) the principal angle between the clients’ data subspaces spanned by their corresponding principal vectors of data is a better estimate of the data heterogeneity.

Impact Statement—FL is becoming a compelling learning paradigm in the artificial intelligence (AI) area. However, FL suffers from a notorious issue which is the existence of statistical Non-IID data across different distributed clients. Due to diverse participants, severe data heterogeneity can be present in different clients data, which has been demonstrated to result in unstable and slow convergence. For instance, training a global model across hospitals to identify brain/cancer tumors where every hospitals’ images come from a different domain/distribution. To simulate this statistical data heterogeneity, data heterogeneity has been simply modeled as Non-IID label skew which tends to be a rigid data partitioning and is hardly representational and thorough. In this paper, we proposed a new notion and framework for Non-IID partitioning in FL settings by analyzing the principal angles between subspaces of different classes of each dataset. We perform an extensive empirical analysis and show that our approach generates a more severe and realistic data heterogeneity partitioning compared to the one used in prior arts.

Index Terms—Federated learning, new notion of Non-IIDness, statistical data heterogeneity, subspace similarity.

I. INTRODUCTION

D


eep learning has emerged as a fast-development technique in computer vision, natural language processing [1], and conversational AI. Though successful, the efficacy of machine learning and deep learning algorithms relies on large quantities of data. However, in areas such as health care, the data may be distributed across numerous hospitals or data centers and cannot be accessed by a central server or cloud due to privacy constraints. For instance, hospitals may have only a few images of particular cancer that must be kept private. A lot of works on privacy-preserving data management and data mining [2] in a centralized setting have been proposed so far, however, they cannot tackle the cases of distributed databases.

Driven by such realistic requirements, in order to conduct data mining/machine learning, it is necessary to exploit data from such distributed databases while preserving data privacy [1]. Federated learning (FL) [3] rises to this challenge due to its ability to collectively train neural networks while preserving data privacy. One of the standard FL algorithms is FedAvg [3]. In each round of FedAvg, clients train models with their local datasets independently, and then the server aggregates the locally trained models, and finally, the parameters of local models are averaged elementwise by the server to obtain a shared Global model [4].

FL however introduces distinct issues not present in classical distributed learning [5]. One of the challenges that currently confine the applicability of existing FL methods to real-world datasets is the data heterogeneity in the data distribution between participating clients [6], [7].

There have been a plethora of works proposing solutions to FL under Non-IID data in recent years. They can be categorized into five groups:

1) alleviating nonguaranteed and weight divergence [8], [9], [10], [11], [12], [13], where the local objectives of the clients are modified such that the local model is consistent with the global model to some extent;
2) modifying the aggregation scheme at the server side [12], [14], [15], [16], [17];
3) data sharing [18], [19], [20], [21], where the server shares a small subset of an auxiliary dataset with the clients to
help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
4) personalized federated learning [6, 22, 23, 24, 25, 26, 27], take the advantage of a certain level of personalization in training the individual clients models rather than training a single global model;
5) using curriculum learning (CL) in FL to mitigate the effect of data heterogeneity across clients.

It was shown that CL is particularly effective in the early stages of training as the larger initial steps of the optimization help construct a more balanced and IID data distribution on the client;
bring significant challenges in learning accuracy of FL algorithms. Refer to Sections II-C, II-D, III-C, and III-D.

3) Under the new notion of heterogeneity that will be proposed herein, data heterogeneity can have detrimental effects such that for some of the clients it is not justified to participate in federation. Refer to Table IV and Section V.

4) None of the existing state-of-the-art (SOTA) FL algorithms beats the others according to the new notion of data heterogeneity that will be presented in this article.

Here we formally introduce our proposed paradigm, where the goal is to define a new notion of data heterogeneity and suggest standard and real Non-IID setups. We hope that this notion, along with introduced setups, becomes standard and inspire the federated learning community.

II. OVERVIEW

A. Preliminaries

Principal angles between two subspaces: Let \( \mathcal{U} = \text{span}\{u_1, \ldots, u_p\} \) and \( \mathcal{W} = \text{span}\{w_1, \ldots, w_q\} \) be \( p \) and \( q \)-dimensional subspaces of \( \mathbb{R}^p \), where \( \{u_1, \ldots, u_p\} \) and \( \{w_1, \ldots, w_q\} \) are orthonormal, with \( 1 \leq p \leq q \). There exists a sequence of \( p \) angles \( 0 \leq \Theta_1 \leq \Theta_2 \leq \ldots \leq \Theta_p \leq \pi/2 \) called the principal angles. The sequence of \( p \) principal angle between them is defined as

\[
\Theta(\mathcal{U}, \mathcal{W}) = \min_{u \in \mathcal{U}, w \in \mathcal{W}} \left( \arccos \frac{|u^T w|}{|u||w|} \right)
\]

where \(|\cdot||\) is the induced norm. The smallest principal angle is \( \Theta_1(u_1, v_1) \), where the vectors \( u_1 \) and \( v_1 \) are the corresponding principal vectors.

Truncated Singular Value Decomposition (SVD): Truncated SVD of a real \( m \times n \) matrix \( M \) is a factorization of the form \( M = U_p \Sigma_p V_p^T \), where \( U_p = [u_1, u_2, \ldots, u_p] \) is an \( m \times p \) orthonormal matrix, \( \Sigma_p \) is a \( p \times p \) rectangular diagonal matrix with nonnegative real numbers on the main diagonal, and \( V_p = [v_1, v_2, \ldots, v_p] \) is a \( p \times n \) orthonormal matrix, where \( u_1 \in U_p \) and \( v_i \in V_p \) are the left and right singular vectors, respectively.

Hierarchical clustering: When forming disjoint clusters where the number of clusters is not known in advance, hierarchical clustering (HC) [7] is of interest. Agglomerative HC is one of the well-known clustering techniques in machine learning that takes a proximity (adjacency) matrix as input and groups similar objects into clusters. HC starts by treating each client as a separate cluster. At each step of the clustering, the pairwise \( L_2 \) (Euclidean) distance between all clusters is computed to identify their similarity. The two clusters that are most closest ones are merged. This iterative process continues till all the clusters are merged into one. Finally, a distance threshold is defined to determine when to stop merging clusters. In this article, the distance threshold is called clustering threshold.

B. Methodology

In our approach the data heterogeneity/homogeneity should be identified by analyzing the principal angles between the client data subspaces. More particularly, each client in FL applies a truncated SVD step on its own local data in a single-shot manner to derive a small set of principal vectors, which form the principal bases of the underlying data. These principal bases provide a signature that succinctly captures the main characteristics of the underlying distribution. These principal bases efficiently identifies distribution heterogeneity/homogeneity among clients by comparing the principal angles between the clients data subspaces spanned by the provided principal vectors. The greater the difference in data heterogeneity between two clients is, the more orthogonal their subspaces will be.

To measure the statistical heterogeneity among different clients’ domains, in this article, we leverage the angle between clients data subspaces spanned by the most significant left singular vectors of clients data. To begin, we introduce the data heterogeneity via the new notion presented in this article and then we will generate Non-IID data partitioning across the clients using the proposed method. For a dataset \( \mathcal{D} \), we put the data of each class \( C_i \) in the columns of its corresponding matrix \( \mathcal{Q}_i \). We then, perform truncated SVD on \( \mathcal{Q}_i \) and obtain \( U_p = [u_1, u_2, \ldots, u_p] \). These \( U_p \) vectors span the class subspace and provide a useful signature for distinguishing the underlying distributions of each class in \( \mathcal{D} \) since these principal bases characterize the main trends in the data of clients (like eigenfaces). Then according to the principal angle in between of the class subspaces, we understand how similar/dissimilar two classes are based on which we can generate Non-IID data partitioning. The more orthogonal two subspaces are the more heterogeneous the data of two classes will be.

Having the data signature of all classes of dataset \( \mathcal{D} \), in hand, we can obtain a proximity matrix \( A \) either as in (2) whose entries are the smallest principle angle between the pairs of \( U_p^i \) or as in (3) whose entries are the summation over the angle in between of the corresponding \( U_p \) vectors (in identical order) in each pairs of \( U_p^i \) (where \( tr(.) \) is the trace operator)

\[
A_{i,j} = \Theta_1 \left( U_p^i, U_p^j \right), \quad i, j = 1, \ldots, |C| \quad (2)
\]

\[
A_{i,j} = tr \left( \arccos \left( U_p^i \circ U_p^j \right) \right), \quad i, j = 1, \ldots, |C| \quad (3)
\]

where \( C \) is the total number of classes of \( \mathcal{D} \). Either of (2) and (3) can be employed in practice, however, theoretically (3) is more rigorous. Now, in order to capture the similarity/dissimilarity of different classes of \( \mathcal{D} \), we could form disjoint clusters of classes. For forming disjoint clusters, we can perform agglomerative hierarchical clustering [31] on the proximity matrix \( A \). The best number of clusters can easily be determined just by analyzing the proximity matrix. Each cluster contain classes, which are roughly identically distributed.

Before providing more details about the application of the proposed method in defining new Non-IID partition, we elucidate how the proposed method perfectly distinguishes different datasets based on their hidden data distribution by inspecting
There must be a translation protocol enabling the server to understand similarity and dissimilarity in the distribution of data across clients without sharing data. These 2-D figures intuitively demonstrate how the principal angle between the client data subspaces captures the statistical heterogeneity. In particular, it shows the subspaces spanned by the $U_p$s of four different datasets (left to right: CIFAR-10, SVHN, FMNIST, and USPS). As can be seen the principal angle between the corresponding $u$ vectors of CIFAR-10 and SVHN is smaller than that of CIFAR-10 and USPS. Table I shows the exact principal angles between every pairs of these subspaces.

**Fig. 2.** UMAP visualization of four different datasets including CIFAR-10 (orange), SVHN (blue), FMNIST (green), USPS (red).

**TABLE I**

| Dataset | CIFAR-10 | SVHN | FMNIST | USPS |
|---------|----------|------|--------|------|
| CIFAR-10 | 0 (0) | 6.13 (12.3) | 45.79 (91.6) | 66.26 (132.5) |
| SVHN | 6.13 (12.3) | 0 (0) | 43.42 (86.8) | 64.86 (129.7) |
| FMNIST | 45.79 (91.6) | 43.42 (86.8) | 0 (0) | 43.36 (86.7) |
| USPS | 66.26 (132.5) | 64.86 (129.7) | 43.36 (86.7) | 0 (0) |

This table shows the proximity matrix of four datasets whose UMAP visualization was shown in Fig. 3(c). Entries are $x(y)$, where $x$ and $y$ are obtained from (2), and (3), respectively. We let of $p$ in $U_p$ be 2.

the angle between their data subspaces \(^1\) spanned by their first $p$ left singular vectors. For a visual illustration of the result, we refer to Figs. 1 and 2, and Table I, where the similarity and dissimilarity of the four different datasets have been evaluated by the proposed measure in (2), and (3). Table I further confirmed with UMAP [32] visualization in Fig. 2. As can be seen from Fig. 2, CIFAR-10 is more similar to SVHN than USPS which reflected in a smaller principal angle between the subspaces of CIFAR-10 and SVHN than that of CIFAR-10 and USPS. It is noteworthy that the smallest principal angle between each pair of classes in each dataset is different as well. For instance, on FMNIST, by setting $p$ of $U_p$ to 3, the smallest principal angle between Trouser and Dress classes is 22.47° while that of Trousers and Bag classes is 51.7°, which stems from more similarity between the distribution of (Trouser, Dress) compared to that of (Trousers, Bag).

Making use of the proposed approach for measuring the similarity/dissimilarity of clients data and invoking the proximity matrix of all clients data under the current Non-IID partitioning method, we will uncover that the existing data partition strategies represent more like an IID partitioning or at most a light Non-IID that may not necessarily be considered as a challenge. Clients may have fewer or no label overlap but still contribute to each others’ performance because of the representation similarity of the classes. This hidden contribution of clients having no label overlap but rather similar classes is an unstudied factor in the existing literature. Consequently, our benchmark, along with similar ones, can serve as a new tool for generating heterogeneous data partitions and evaluating different algorithms recommended for addressing data heterogeneity.

**C. New Non-IID Partitioning Method**

Now we are in position to define a new Non-IID partitioning. We say that a federated network is heterogeneous if each client owns data only from one of super clusters of a dataset. In particular, we partition all the training samples in each super cluster into shards of $n$ examples and randomly assign two shards to each client only from one of the super clusters. Such Non-IID (2) partitioning is reasonable to expect in heterogeneous federated networks, due to the drastic disparities in the distribution of each client data. In contrast, the data partitioning proposed in previous papers [6], [18] may assign data from all the clusters.

Consider an example. In Table II, we show the average final top-1 test accuracy of all clients on CIFAR-10 for FedAvg under IID (2), the conventional Non-IID (C-NIID) label skew (2), and our proposed Non-IID partitioning which we name it as super cluster based Non-IID (SC-NIID). As can be seen from Table II, the C-NIID data partitioning yield accuracy results close to IID data partitioning while our proposed SC-NIID partitioning accuracy results are far less than that of IID. This shows that the C-NIID data partitioning is not a severe Non-IID and it rather tends to be similar to IID. We will compare the performance of various global and personalized baselines under these Non-IID partitioning methods later on Experiment Section along with some intuitions and remarks.

We provide another empirical example to show that our proposed method can effectively produce a more challenging Non-IID data partitioning by leveraging the principal angle

**TABLE II**

| Dataset | IID (2) | C-NIID (2) | SC-NIID (2) |
|---------|--------|------------|-------------|
| CIFAR-10 | 88.15 ± 0.47 | 83.64 ± 1.57 | 75.53 ± 3.83 |

For each partitioning method, the average of final local test accuracy over all clients is reported. We run the federavg baseline for each partition 3 times for 100 communication rounds with 10 local epochs and a local batch size of 10.

\(^{1}\) Note that our method does not assign a specific subspace to the datasets stored in the FL devices, nor does it modify the subspace to increase its orthogonality. It rather works as a preprocessing step to find clusters of similar classes in order to make heterogeneous partitions that maximize the dissimilarity to reduce the hidden contribution of clients in the conventional heterogeneous partitioning used in previous works.
TABLE III

| Measure | IID  | C-NIID (2) | SC-NIID (2) |
|---------|------|------------|-------------|
| EMD     | 0.042| 0.17       | 0.29        |
| Eq. 2   | 0.072| 0.202      | 0.274       |
| Eq. 3   | 0.16 | 0.28       | 0.338       |
| CKA     | 0.94 | 0.889      | 0.825       |

Fig. 3. Main goal of this figure is to understand the cluster structure of different datasets based on which the Non-IID data partitioning of different datasets can be suggested. (a) Depicts the UMAP visualization of CIFAR-10 classes. As can be seen, CIFAR-10 naturally has two super clusters, namely animals (cat, dog, bird, deer, horse, frog) and vehicles (car, plane, ship, truck), which are shown in the purple and green regions, respectively. This means that within each super cluster, the distance between the distribution of the classes is small. While the distance between the distributions of the two super clusters is quite huge. Since the union of clients data is CIFAR-10, two clusters are enough to handle the Non-IIDness across clients. (b) We obtained the proximity matrix $A$ as in (2) and sketched it. The entries of $A$ are the smallest principle angle between all pairs of classes of CIFAR-10. This concurs with (a) showing the cluster structure of CIFAR-10. (c) The data of FMNIST are naturally clustered into three clusters. The structure of the three clusters is also perfectly suggested by our new proposed notion of heterogeneity for this dataset. (d) We did the same thing as in (b) for FMNIST as well and sketched the matrix.

between of the data subspaces spanned by the first $p$ significant left singular vectors of the data. In particular, as shown in Table III, we compare the average distance between the data of each partitioning method including IID, C-NIID, and our proposed SC-NIID. We employ the well-known distance measures including Earth Mover’s Distance (EMD) [33], Centered Kernel Alignment (CKA) [34], and our proposed method as in (2), and (3) in inspecting the distance between the data of each Non-IID partitioning method. In all of these measures, the smaller the entry is the more IID the data of the partition is. As can be seen, the entries of C-NIID is smaller than that of SC-NIID and are closer to that of IID. Tables II and III together reveal that first the C-NIID is more like an IID partitioning or at least it is not a challenging and severe Non-IID and second, they demonstrate that our proposed method as in (2) and (3) can accurately capture the similarity/dissimilarity between two distributions and its results are consistent with that of the well-known distance measures.

The UMAP [32] visualization in Fig. 3(a) confirms that the images of CIFAR-10 can naturally be clustered into two super clusters, i.e., cluster of animals (cat, dog, deer, frog, horse, bird) and cluster of vehicles (airplane, automobile, ship, and truck). This shows that the two clusters is the best case for training the local models on partitions of CIFAR-10 dataset in a Non-IID fashion. Fig 3(b) also depicts the proximity matrix of CIFAR-10 dataset, whose entries are the principal angle between the subspace of every pairs of ten classes (labels). This further confirms that our proposed notion perfectly captures the heterogeneity, thereby finding the best number of superclusters in each dataset. In particular, our experiments demonstrate that the clients that have the subclasses of these two big classes have common features and can improve the performance of other clients that own subclasses of the same big class if they are assigned to the same cluster. A similar discussion can be made about other datasets. In contrast, in almost all of the prior works [4], [9], [10], [18], [23], the data samples with the same label are divided into subsets and each client is only assigned two subsets with different labels, which produces a very light Non-IID partition as discussed above. Similar settings has been used where each party only has data samples with a single label [4].

Another method that has been adopted in the literature to simulate Non-IID label skew is allocating a proportion of the data of each label/class according to Dirichlet distribution (Dir($\alpha$)). In particular, random samples for class $c$ from Dirichlet distribution according to $p_c \sim \text{Dir}(\alpha)$ are selected from the whole dataset and to each client $j$ random samples of class $c$ according to $p_{c,j}$ proportion is assigned. While Dir($\alpha$) label skew can simulate label imbalance in the network, it may not simulate severe data heterogeneity across clients because the assigned samples are randomly selected from the whole dataset and they are not necessarily from only one of the super clusters of that dataset. We rather suggest random samples for each class be selected from each super cluster on a dataset according to Dirichlet distribution. This will provide a more severe and challenging Non-IIDness. We postpone the experiments on this new Dir($\alpha$) Non-IID setup to Sections III-A, III-B, and III-C.

It is noteworthy that the number of formed super clusters in each dataset can be controlled by the distance threshold (clustering threshold), which is a hyperparameter in hierarchical clustering. The smaller the clustering threshold the larger number of super clusters will be formed and in turn the more heterogeneous the partitioning will be. Therefore, the level of data heterogeneity across clients can be easily controlled by the clustering threshold. We emphasize that the proposed method finds applicability in particular tasks such as image and video classification tasks, and may not work for some of the other data domains. Furthermore, the proposed method is intended for supervised partitioning or classification tasks, where the training data are labeled.

D. Closer Look at FL Under Heterogeneity

To understand which of the Non-IID setups is a better benchmark to be considered in heterogeneous FL scenarios, we perform an experimental study on heterogeneous local models. We choose CIFAR-10 with 10 clients and LeNet-5 as convolutional neural network with 5 layers. We then train the model two times independently with the same random seeds with FedAvg, where in the first training we partition the data according to the C-NIID (2) and in the second time we partition the data according to our new notion of Non-IID-ness, i.e., SC-NIID (2). We train both cases for 100 rounds and each client optimizes for 10 local epochs at each round. For each layer in the models, we use CKA [34] and our proposed measure in (2) to evaluate the similarity of the output features between two local models,
Fig. 4. Similarities of the outputted feature representation of three different layers of different partitions obtained by CKA (left) and by our proposed measure as in (2) (right) when trained on CIFAR-10. This plot is sketched once federation finished.

Fig. 5. Means of the similarities of different layers in different local models obtained by CKA (left) and by our proposed measure as in (2) (middle) and (3) (right).

given the same input testing samples for each of the cases independently. CKA outputs a similarity score between 0 (not similar at all) and 1 (identical).

We first show the pairwise CKA features similarity of the first three layers of LeNet-5 across local models in Fig. 4. Interestingly, as can be seen, we find that features outputted from IID data and our proposed Non-IID setup show lower CKA similarity compared to that of the C-NIID. It indicates that our proposed benchmark provide a more severe heterogeneity across different clients. By averaging the pairwise CKA features similarity in Fig. 4, we can obtain a single value to approximately represent the similarity of the feature outputs by each layer across different clients for IID, C-NIID, and our SC-NIID setups. We demonstrated the approximated layerwise features similarity in Fig. 5. These results witness that the models trained on our proposed Non-IID setup consistently produced features across clients for all layers which are less similar to IID in comparison to that of C-NIID.

III. EXPERIMENTS

We perform an extensive empirical analysis using a standard image classification task for multiple popular federated learning datasets and various statistical heterogeneity setups.

A. Experimental Setup

Datasets and Models: We use image classification task and three popular datasets, i.e., CIFAR-10 [35], CIFAR-100 [35], STL-10 [36] to employ our novel partitioning method. Typically, datasets that are harder often necessitate more advanced architectures capable of handling increased complexity. Conversely, simpler datasets can be effectively trained using less complex models. In our experiments, we adhered to this convention by employing ResNet-9 [37] for CIFAR-100 and STL-10, while opting for LeNet-5 [38] for the simpler CIFAR-10 task. Further, it is worth highlighting that the proposed technique is architecture-agnostic, meaning it does not depend on the specific architecture backbone. Rather, the approach focuses on utilizing the dataset to create super clusters, which are then utilized to generate a new data partitioning. Details of the architectures can be found in Section IV.

Baselines and Implementation: To assess the performance of our novel Non-IID partitioning method against the conventional partitioning, we compare the results over a set of baselines. For SOTA personalized FL methods, the baselines include LG-FedAvg [24], Per-FedAvg [6], Clustered-FL (CFL) [26], and IFCA [25]. Besides, we compare with FedAvg$^+$ [4], FedProx$^+$ [10], FedNova$^+$ [16], and SCAFFOLD$^+$ [11]. It is noteworthy that the superscript + sign on global baselines means that these baselines has been fine-tuned by the clients and thus are considered personalized ones. We report the average results performance over three independent trials.

C-NIID ($\rho\%$) Label Skew and SC-NIID ($\rho\%$). In this setting, the conventional method first randomly assigns $\rho\%$ of the total available labels of a dataset to each client and then randomly distributes the samples of each label among clients own those labels as in [39]. In our SC-NIID method, we first form the super clusters according to our proposed method explained in

2We would like to point that our focus is on the image classification task and the well-established label distribution skew partitioning in the FL community [5], [6], [18], [39], [40], [41], [42], [43], [44], [45]. We aim to provide a better understanding of the limitations and propose new notions for it.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
TABLE IV
EVALUATING DIFFERENT PERSONALIZED FL METHODS UNDER DIFFERENT DATA PARTITIONS. WE EVALUATE ON RESNET-9 WITH CIFAR-100 AND STL-10 AS WELL AS LEUNET-5 ON CIFAR-10. FOR EACH COMMUNICATION ROUND, A FRACTION 10%, 30%, 10% OF THE TOTAL 100 CLIENTS ARE RANDOMLY SELECTED. WE SET LOCAL EPOCH AND BATCH SIZE TO 10.

| Dataset  | Algorithm      | C-NIID(2)  | SC-NIID(2) | C-Dir(0.5) | SC-Dir(0.5) |
|----------|----------------|------------|------------|------------|-------------|
| SOLO     | FedAvg         | 83.62 ± 0.72 | 75.68 ± 0.47 | 48.45 ± 0.57 | 43.44 ± 0.43 |
| CIFAR-10 | FedAvg         | 83.46 ± 1.15 | 77.02 ± 0.73 | 53.31 ± 0.85 | 43.43 ± 1.81 |
|          | FedProx        | 85.01 ± 0.59 | 76.54 ± 1.15 | 52.69 ± 0.60 | 44.61 ± 1.44 |
|          | FedNova        | 83.99 ± 0.68 | 77.36 ± 0.46 | 53.21 ± 0.82 | 46.09 ± 0.33 |
|          | Scafold+       | 82.69 ± 2.93 | 78.88 ± 0.44 | 41.55 ± 0.82 | 16.45 ± 2.71 |
|          | L              | 83.18 ± 0.46 | 76.40 ± 0.46 | 31.35 ± 7.42 | 41.36 ± 0.75 |
|          | PerFedAvg      | 83.60 ± 0.48 | 75.70 ± 0.74 | 54.90 ± 0.25 | 42.63 ± 1.08 |
| IFCA     |                | 86.59 ± 1.16 | 80.49 ± 0.86 | 57.78 ± 1.14 | 51.54 ± 0.98 |
| SOLO     |                | 76.71 ± 1.12 | 73.16 ± 0.17 | 45.24 ± 1.74 | 40.43 ± 0.85 |
| CIFAR-100| FedAvg         | 87.99 ± 0.97 | 81.55 ± 0.59 | 66.15 ± 2.79 | 53.41 ± 1.16 |
|          | FedProx        | 87.68 ± 0.82 | 80.70 ± 0.81 | 67.67 ± 0.98 | 53.86 ± 0.25 |
|          | FedNova        | 87.22 ± 0.45 | 80.75 ± 1.28 | 63.15 ± 2.32 | 53.77 ± 0.52 |
|          | Scafold+       | 55.97 ± 13.29 | 15.61 ± 9.85 | 39.04 ± 26.73 | 11.43 ± 3.90 |
|          | L              | 78.27 ± 1.31 | 72.87 ± 0.80 | 44.43 ± 1.40 | 39.44 ± 1.00 |
|          | PerFedAvg      | 77.47 ± 1.30 | 58.93 ± 0.90 | 58.02 ± 2.38 | 37.96 ± 1.24 |
| IFCA     |                | 88.06 ± 0.19 | 82.23 ± 0.73 | 69.89 ± 1.64 | 55.66 ± 0.86 |
| SOLO     |                | 78.14 ± 2.27 | 69.88 ± 0.62 | 51.12 ± 2.16 | 42.17 ± 0.68 |
| CIFAR-100| FedAvg         | 85.67 ± 2.23 | 77.23 ± 1.45 | 56.80 ± 5.64 | 49.69 ± 0.66 |
|          | FedProx        | 86.83 ± 1.83 | 83.03 ± 1.40 | 64.97 ± 5.86 | 56.35 ± 1.85 |
|          | FedNova        | 88.45 ± 0.27 | 83.18 ± 2.28 | 60.00 ± 6.77 | 52.84 ± 1.03 |
|          | Scafold+       | 31.74 ± 1.80 | 27.71 ± 4.12 | 50.31 ± 2.90 | 31.28 ± 7.52 |
|          | L              | 84.42 ± 0.77 | 75.52 ± 0.91 | 52.56 ± 2.94 | 46.82 ± 1.47 |
|          | PerFedAvg      | 52.91 ± 2.12 | 51.64 ± 0.57 | 30.10 ± 2.74 | 31.80 ± 3.14 |
| IFCA     |                | 88.99 ± 0.45 | 81.13 ± 0.46 | 67.99 ± 1.66 | 60.73 ± 1.27 |

We evaluate on ResNet-9 with CIFAR-100 and STL-10 as well as LeNet-5 on CIFAR-10. For each communication round, a fraction 10%, 30%, 10% of the total 100 clients are randomly selected. We set local epoch and batch size to 10.

Section II. We then randomly assign all clients to only one of the formed super clusters. The number of assigned clients to each super cluster is proportional to the size (number of samples that cluster contains) of the super cluster, which means that if the size of a super cluster is bigger, more clients are assigned to that particular super cluster. Next, the total samples of each super cluster are divided into shards and each client picks $\varphi\%$ of the total labels contained in the super cluster to which the client belongs to.

Conventional Dir (C-Dir) and SC-Dir: In this setting, the conventional method distributes the training data between the clients based on the Dirichlet distribution. In particular, for $N$ clients data it samples $N$ random numbers $p_i \sim Dir(N)(\alpha)$ from Dir$(\alpha)$ distribution and allocates the $p_{i,j}$ proportion of the training data of class $i$ to client $j$ as in [39]. In our proposed SC-Dir(.) method, we again constitute the super clusters with the help of (2) or (3) as explained in Section II. We then assign a certain number of clients randomly to only one of these super clusters depending upon the size of each super cluster. We then let the data within each super cluster be assigned according to the Dir(.) distribution as in C-Dir(.) to the clients that belong to each cluster.

B. Comparing the Performance of SOTA Baselines on the Conventional and Newly Proposed Non-IID Method

We conduct experiments to compare the above four Non-IID partitioning method, i.e., C-NIID (2), C-Dir(.), SC-NIID (2), and SC-Dir(.) methods on CIFAR-10, CIFAR-100, and STL-10 datasets and present the results in Table IV. It can be observed that all SOTA baselines present a significant performance drop when the clients data are distributed according to the newly proposed Non-IID setup compared to the conventional Non-IID setups used in prior arts. Based on the results of Table IV and through extensive studies, we have the following key findings: 1) the newly proposed Non-IID partitioning is more challenging compared to the C-NIID. Since effectively addressing data heterogeneity is of paramount concern in federated learning, we suggest that the challenging tasks like SC-NIID ($\varphi\%$), and SC-Dir($\alpha$) should be included in the benchmark for future FL setups. 2) Under the new Non-IID setup none of the existing SOTA FL algorithms outperforms others in all cases. This further indicates the importance of having a more comprehensive Non-IID distribution benchmark.

C. Comparing the Level of Data Heterogeneity of the C-NIID and SC-NIID

Data heterogeneity will impact the convergence of a federated model and hurts its performance. This necessitate a deeper investigation of data heterogeneity which has been missing. Herein, we provide some visualizations to facilitate understanding the heterogeneity level of the setup that has been used as an standard one in the prior works. To this end, we distribute CIFAR-10 according to three data partitioning method, i.e., IID, C-NIID, and SC-NIID among the clients and then we leverage three distribution similarity/dissimilarity measures namely EMD [33], CKA [34], and our proposed method as in (2) to monitor the significance of data heterogeneity across 100 participant clients in the federation. Figs. 6–8 visualize the proximity matrix whose entries are the distance between clients data computed by EMD, CKA, and our proposed method, respectively. These figures further confirm that the C-NIID partitioning is more like IID.

3The value of $\alpha$ controls the degree of Non-IID-ness. A big value of $\alpha$, e.g., $\alpha = 100$ mimics identical label distribution (IID), while $\alpha = 0.1$ results in a split, where the vast majority of data on every client are Non-IID.
We obtained a proximity matrix of $100 \times 100$ dimension, which corresponds to 100 clients’ data distribution by the EMD measure for three different data partitioning method, i.e., IID (left), C-NIID (middle), and SC-NIID (right). This concurs with Fig. 3 showing that CIFAR-10 naturally have two Non-IID clusters as well with Figs. 7 and 8.

We obtained a proximity matrix of $100 \times 100$ dimension, which corresponds to 100 clients’ data distribution by the CKA measure for three different data partitioning method, i.e., IID (left), C-NIID (middle), and SC-NIID (right). This concurs with Fig. 3 showing that CIFAR-10 naturally have two Non-IID clusters as well with Figs. 6 and 8.

We obtained a proximity matrix of $100 \times 100$ dimension, which corresponds to 100 clients’ data distribution by our own proposed measure as in (2) for three different data partitioning method, i.e., IID (left), C-NIID (middle), and SC-NIID (right). This concurs with Fig. 3 showing that CIFAR-10 naturally have two Non-IID clusters as well with Figs. 6 and 7.

This is while as can be seen in these three figures, our newly proposed SC-NIID distribute the data across all clients, which is very different from IID. This corresponds to the fact that our method takes the intrinsic structure of the dataset into account and from a certain number of Non-IID super clusters according to the entries of the proximity matrix and then distribute the data among the clients from only one of these Non-IID super clusters. According to what we discussed, it is therefore natural to ask the following questions: How do prior FL approaches tackle the newly proposed Non-IID partitioning?

Remark 1: It is not clear how the prior personalized FL algorithms, which has been proposed to alleviate the statistical data heterogeneity should be extended to tackle the new Non-IID setup. For instance, as mentioned in [29], the proposed regularization method is effective only for light data heterogeneity but would not be beneficial or even lead to drop in the performance with the increase of the heterogeneity. This could be true about many of the prior arts as also confirmed by the results reported in Table IV. This further emphasizes that the performance of the prior arts should be evaluated under severe and challenging Non-IID setups to judge their efficacy.

Remark 2: While many works in the literature have highlighted detrimental impacts of statistical data heterogeneity, showing that heterogeneity can lead to poor performance in federated optimization, which necessitates novel forms of personalization, some works [30] discussed distinct benefits of data heterogeneity in their FL analyses. In contrast to this work [30], we believe to judge whether the impact of the corresponding heterogeneity issue is positive or negative, we should first provide a true notion of data heterogeneity as well as standard Non-IID benchmarks without which it will be hard to attest measurable benefits of heterogeneity.

Remark 3: In Table IV, we provided the results of SOLO training as well. Considering SC-Dir(0.5) on CIFAR-10, as is evident from Table IV, except for a few baselines, the accuracy

\[ \text{In SOLO baseline the client trains a model locally on its own local data without taking part in federation.} \]
results of other baselines are worse than SOLO training which means that it is not justified for clients to participate in federation under the newly proposed Non-IID setting. Because the clients not only do not gain from federation but federation also has degraded their performance. This phenomenon can be explained by the fact that depending upon the dataset the proposed Non-IID partitioning distributes highly heterogeneous data to clients under which federation is not beneficial. This is while considering C-Dir(0.5) most of the baselines benefited from federation and yielded better results compared to SOLO training. The same behavior can be seen on other datasets and on SC-NIID(2).

D. Under What Heterogeneity Environment Clients Benefit From Collaboration?

In this section, by an empirical pseudoexample we demonstrate that the widely used Non-IID setup, i.e., C-NIID in the literature is not a real Non-IID partition. In doing so, we sample two clients named as C1, and C2 and manually assign two labels to each as in Table V and let them to do federation with vanilla FedAvg [4] for 20 communication rounds. When C1 owns labels “ship+truck,” and C2 owns labels “plane+car” as in row 4, according to the common belief in the FL community we should have expected that due to the existence of C-NIID the average final accuracy will be worse than SOLO training as in row 1. But surprisingly it is not the case and even though these clients have no label overlap, their performance improved through federation compared to SOLO. In contrast, as can be seen from row 5, even though C1 and C2 have 50% labels overlap (50% distribution similarity according to C-NIID), by taking part in federation, the accuracy of C1 drops compared to SOLO. This is while row 3 with the same condition (50% labels overlap) improved the results of C1 through federation. This further confirms that C-NIID cannot represent a true view of non-IIDness in FL. These anomalies can be justified by our newly proposed Non-IID partition, which is not based on the simple label skew but based on the angle in between of the clients’ data subspaces. In particular, two clients data are considered heterogeneous if their data are drawn from two different super clusters formed by our approach in (2) or (3). The result of C1 in row 4 improved because both of the clients own data from the same super cluster. The results of other rows can be justified in the same fashion.

The authors hope the reader to take the preceding discussion as an example showing that the adopted Non-IID partition in the prior works may not represent a true data heterogeneity setup and also the authors do not claim that the proposed method can justify all anomalies. We rather like to encourage the researchers to design more comprehensive alternatives to the current Non-IID setups.

E. New Benchmark for Non-IID FL

As mentioned earlier, existing studies have been evaluated on simple partitioning strategies, i.e., Non-IID label skew (20%) and Non-IID label skew (30%). In data partitioning with a% label skew, the union of client data will only be one dataset. Focusing on CIFAR-10 and with 20% label skew, most of the 100 clients can have either 50% label overlap or 100%. This simulates a partially Non-IID setting and cannot represent a full view of Non-IIDness. Because the data distributions of clients are the subdistributions of a unique dataset such as CIFAR-10. This is the reason that statistical data heterogeneity has never been a big issue in the proposed personalized FL algorithms.

In order to better assess the potential of the SOTA baselines under a real world and challenging Non-IID task, where the local data of clients have strong statistical heterogeneity, and the data distributions of clients are not the subdistributions of a unique dataset, we design the following benchmark naming it as MIX-4. We hope Mix-4 become an standard benchmark for comparing different SOTA against each other in the FL community. We assume that each client owns data samples from one of the four datasets, i.e., USPS [46], CIFAR-10, SVHN, and FMNIST. In particular, we distribute CIFAR-10, SVHN, FMNIST, USPS among x, y, z, v clients, respectively (x+y+z+v= total clients), where each client receives a certain number of samples from all classes but only from one of these datasets. This is a very challenging Non-IID task. Under this circumstance, the union of the clients data is not a single dataset and in each round of communications there will be some clients whose data distribution vary drastically. In Table VI, we present the
test accuracy of the SOTA baselines on Mix-4. It can be seen from Table VI that the accuracy performance of all the methods drops significantly compared to the ones reported for C-NIID (the widely used Non-IID setup in prior works) in Table IV. These sorts of realistic assumption has never been adopted in the literature. That is why it is a common belief that all users can benefit from heterogeneity. We hope designing Mix-4 opens up a new avenue to design more standard real-world benchmark for comparing different personalized SOTA in the FL community.

### IV. IMPLEMENTATION

We have released our implemented code at https://github.com/MMorafah/FL-SC-NIID. To be consistent, we adapt the official public codebase of Qinbin et al. [39] to implement our proposed method and all the other baselines with the same codebase in PyTorch V.1.9. We used the public codebase of LG [24], Per-FedAvg [6], and IFCA [25] in our implementation. For all the global benchmarks, including FedAvg [4], FedProx [10], FedNova [16], Scaffold [11] we used the official public codebase of Qinbin et al. [39]. It is worth noting that, unlike the original paper and the official implementation of LG [24], for the sake of fair comparison we initialized the models randomly with the same random seed just like all other baselines.

The formed super clusters on each dataset: The details of formed super clusters on each dataset are presented in Table VII.

| Dataset      | Formed Super Clusters                                      |
|--------------|------------------------------------------------------------|
| CIFAR-10     | (0.18, 8.9, (2.3, 4.5, 6, 9, 7))                           |
| CIFAR-100    | (0.83, 53, 82), (1.34, 43, 51, 70, 92, 62), (2.36, 93, 30, 95, 67, 73), (4.76, 59, 53), (2.97, 27, 65, 64, 36, 28, 61, 99, 18, 77, 79, 80, 34, 88, 42, 38, 44, 60, 78, 66, 84, 8, 39, 55, 72, 93, 91, 3, 4, 29, 31, 7, 24, 70, 36, 45, 74, 5, 25, 15, 19, 32, 9, 16, 10, 22, 40, 11, 33, 98, 46, 6, 14, 57, 84, 56, 13, 58, 37, 81, 90, 89, 85, 21, 48, 86, 47, 41, 75, 12, 71, 49, 17, 60, 76, 33, 68) |
| STL-10       | (2, 8, 9), (0, 1, 3, 4, 5, 6)                              |

1 The numbers correspond to the labels according to the standard naming of labels in each dataset.

The details of the convolutional neural network used for CIFAR-10, CIFAR-100, and STL-10 dataset are presented in Table VIII.

| Layer  | Details                                      |
|--------|----------------------------------------------|
| layer 1 | Conv2d(i=3, o=6, k=(5, 5), s=(1, 1))       |
|        | ReLU()                                      |
|        | MaxPool2d(k=(2, 2))                         |
| layer 2 | Conv2d(i=6, o=16, k=(5, 5), s=(1, 1))      |
|        | ReLU()                                      |
|        | MaxPool2d(k=(2, 2))                         |
| layer 3 | Linear(i=400 (256 for MNIST), o=120)       |
|        | ReLU()                                      |
| layer 4 | Linear(i=120, o=84)                        |
|        | ReLU()                                      |
| layer 5 | Linear(i=84, o=10 (100 for CIFAR-100, and 40 for Mix-4)) |

The details of the LeNet-5 architecture used for the FMNIST, SVHN, CIFAR-10, and Mix-4 datasets are presented in Table IX.

| Block | Details                                      |
|-------|----------------------------------------------|
| block 1 | Conv2d(i=3, o=64, k=(3, 3), s=(1, 1))       |
|        | GroupNorm(g=32, o=64)                       |
|        | ReLU()                                      |
| block 2 | Conv2d(i=64, o=128, k=(3, 3), s=(1, 1))     |
|        | GroupNorm(g=32, o=128)                     |
|        | ReLU()                                      |
| block 3 | Conv2d(i=128, o=128, k=(3, 3), s=(1, 1))   |
|        | GroupNorm(g=32, o=128)                     |
|        | ReLU()                                      |
| block 4 | Conv2d(i=256, o=256, k=(3, 3), s=(1, 1))   |
|        | GroupNorm(g=32, o=256)                     |
|        | ReLU()                                      |
| block 5 | Conv2d(i=256, o=512, k=(3, 3), s=(1, 1))   |
|        | GroupNorm(g=32, o=512)                     |
|        | ReLU()                                      |
| block 6 | Conv2d(i=512, o=512, k=(3, 3), s=(1, 1))   |
|        | GroupNorm(g=32, o=512)                     |
|        | ReLU()                                      |
| classifier | MaxPool2d(k=(4, 4))                  |
|         | Linear(i=512, o=100)                      |

The details of the ResNet-9 architecture used for CIFAR-100, and STL-10 dataset are presented in Table IX.

| Block | Details                                      |
|-------|----------------------------------------------|
| block 1 | Conv2d(i=3, o=64, k=(3, 3), s=(1, 1))       |
|        | GroupNorm(g=32, o=64)                       |
|        | ReLU()                                      |
| block 2 | Conv2d(i=64, o=128, k=(3, 3), s=(1, 1))     |
|        | GroupNorm(g=32, o=128)                     |
|        | ReLU()                                      |
| block 3 | Conv2d(i=128, o=128, k=(3, 3), s=(1, 1))   |
|        | GroupNorm(g=32, o=128)                     |
|        | ReLU()                                      |
| block 4 | Conv2d(i=256, o=256, k=(3, 3), s=(1, 1))   |
|        | GroupNorm(g=32, o=256)                     |
|        | ReLU()                                      |
| block 5 | Conv2d(i=256, o=512, k=(3, 3), s=(1, 1))   |
|        | GroupNorm(g=32, o=512)                     |
|        | ReLU()                                      |
| block 6 | Conv2d(i=512, o=512, k=(3, 3), s=(1, 1))   |
|        | GroupNorm(g=32, o=512)                     |
|        | ReLU()                                      |
| classifier | MaxPool2d(k=(4, 4))                  |
|         | Linear(i=512, o=100)                      |

Appendices: [Online]. Available. https://github.com/Xtra-Computing/NIID-Bench [Online]. Available. https://github.com/pliang279/LG-FedAvg [Online]. Available. https://github.com/CharlieDinh/pFedMe [Online]. Available. https://github.com/jichan3751/ifca
different shards. Each client then is assigned a certain number of shards from only one of the super clusters. The proposed method addresses a broad range of data heterogeneity issues beyond simpler forms of Non-IIDness like label skews. Extensive evaluations show that our approach generates a more severe setup than that, via experiments we demonstrated that under the newly proposed Non-IID setup, none of the prior arts consistently beat others. Further, we showed that in contrast to the common belief of classifiers calibrated for federated learning with non-IID data distribution, clients can benefit from model fusion in heterogeneous networks.

### REFERENCES

[1] J. Zhang et al., “Towards building the federated GPT: Federated instruction tuning,” CoRR, 2023, doi: 10.48550/ArXiv.2305.05644.

[2] N. Hynes, D. Tao, D. Yan, R. Cheng, and D. Song, “A demonstration of selling: A privacy-preserving data marketplace,” in Proc. VLDB Endow., 2018, pp. 2086–2089.

[3] B. McMahan and D. Ramage, “Federated learning: Collaborative machine learning without centralized training data,” Google Res. Blog, vol. 3, 2017.

[4] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, “Communication-efficient learning of deep networks from decentralized data,” in Proc. 20th Int. Conf. Artif. Intell. Statist., 2017, pp. 1273–1282.

[5] S. Vahidian et al., “Efficient distribution similarity identification in clustered federated learning via principal angles between client data subspaces,” 2022, arXiv:2209.10526.

[6] A. Fallah, A. Mokhtari, and A. Ozdaglar, “Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach,” Adv. Neural Inf. Process. Syst., vol. 33, pp. 5557–5568, 2020.

[7] M. Morafah, S. Vahidian, W. Wang, and B. Lin, “FLIS: Clustered federated learning via inference similarity for non-IID data distribution,” IEEE Open J. Comput. Soc., vol. 4, pp. 109–120, 2023.

[8] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, “Federated optimization in heterogeneous networks,” 2018, arXiv:1812.06127.

[9] Q. Li, B. He, and D. Song, “Model-contrastive federated learning,” CVPR, vol. abs/2103.16257, 2021.

[10] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, “Federated optimization in heterogeneous networks,” in Proc. Mach. Learn. Syst., 2020, pp. 523–530.

[11] S. P. Karimireddy, S. Kale, M. Mohri, S. J. Reddi, S. U. Stich, and A. T. Suresh, “SCAFFOLD: Stochastic controlled averaging for federated learning,” in Proc. 57th Int. Conf. Mach. Learn., 2020, pp. 5132–5143.

[12] H. Wang, M. Yurochkin, Y. Sun, D. S. Papailiopoulos, and Y. Khazaeni, “Federated learning with matched averaging,” in Proc. 5th Int. Conf. Learn. Representations, 2020, Addis Ababa, Ethiopia, 2020, pp. 1–7.

[13] D. A. E. Acar, Y. Zhao, R. M. Navarro, M. Mattina, P. N. Whatmough, and V. Saligrama, “Federated learning based on dynamic regularization,” in Proc. Int. Conf. Learn. Representations, 2021, pp. 732–741.

[14] T. H. Hsu, H. Qi, and M. Brown, “Measuring the effects of non-identical data distribution for federated visual classification,” CoRR, vol. abs/1909.06335, 2019.

[15] T. Lin, L. Song, S. U. Stich, and M. Jaggi, “Ensemble distillation for robust model fusion in federated learning,” CoRR, vol. abs/2006.07242, 2020.

[16] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, “Tackling the objective inconsistency problem in heterogeneous federated optimization,” in Adv. Neural Inf. Process. Syst., 2020, pp. 7611–7623.

[17] M. Yurochkin, M. Agarwal, S. Ghosh, K. H. Greenewald, T. N. Hoang, and Y. Khazaeni, “Bayesian nonparametric federated learning of neural networks,” in Proc. 36th Int. Conf. Mach. Learn., 2019, pp. 7252–7261.

[18] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, “Federated learning with non-IID data,” 2018, arXiv:1806.00582.

[19] W. Hao et al., “Towards fair federated learning with zero-shot data augmentation,” CoRR, vol. abs/2104.13417, 2021.

[20] E. Jeong, S. Oh, H. Kim, J. Park, M. Bennis, and S. Kim, “Communication-efficient on-device machine learning: Federated distillation and augmentation under non-IID private data,” CoRR, vol. abs/1811.11479, 2018. [Online]. Available: http://arxiv.org/abs/1811.11479

[21] J. Goetz and A. Tewari, “Federated learning via synthetic data,” CoRR, vol. abs/2008.04489, 2020. [Online]. Available: https://arxiv.org/abs/2008.04489

[22] Y. Deng, M. M. Kamani, and M. Mahdavi, “Adaptive personalized federated learning,” CoRR, vol. abs/2003.13461, 2020. [Online]. Available: https://arxiv.org/abs/2003.13461

[23] S. Vahidian, M. Morafah, and B. Lin, “Personalized federated learning by structured and unstructured pruning under data heterogeneity,” Proc. 31st IEEE ICDCS, 2021, pp. 27–34.

[24] P. P. Liang, T. Liu, L. Ziyin, R. Salakhutdinov, and L. P. Morency, “Think through the experiments,” 2020, vol. 3, 2017.

[25] S. Vahidian et al., “When do curricula work in federated learning?,” in Proc. 34th Int. Conf. Neural Inf. Process. Syst., 2020, pp. 19586–19597.

[26] F. Sattler, K. Müller, and W. Samek, “Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints,” IEEE Trans. Neural Netw. Learn. Syst., vol. 32, no. 8, pp. 3710–3722, Aug. 2021.

[27] Y. Mansour, M. Mohri, J. Ro, and A. T. Suresh, “Three approaches for personalization with applications to federated learning,” CoRR, vol. abs/2002.10619, 2020.

[28] S. Vahidian et al., “When do curricula work in federated learning?,” CoRR, vol. abs/2212.12712, 2022. [Online]. Available: https://doi.org/10.48550/arxiv.2212.12712

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

[30] D. K. Dennis, T. Li, and V. Smith, “Heterogeneity for the win: One-shot federated clustering,” in Proc. 38th Int. Conf. Mach. Learn., 2021, pp. 2611–2620.
Saeed Vahidian was born in Ghochan, Iran. He received the B.Sc. degree from the Ferdowsi University of Mashhad, Mashhad, Iran, in 2012, the M.Sc. degree from the K. N. Toosi University of Technology, Tehran, Iran, in 2014, and the Ph.D. degree from the University of California San Diego, CA, USA, in 2023, all in electrical and computer engineering.

He is currently a Postdoctoral Associate with Duke University, NC, USA. In the summer of 2021, he was an intern at Qualcomm AI Research. His research interests include the area of federated learning, learning, column subset selection, and convex and non-convex optimization.

Mahdi Morafah received the B.S. degree in electrical engineering from the Tehran Polytechnic University, Tehran, Iran, in 2019, and the M.S. degree in electrical and computer engineering from the University of California, San Diego, CA, USA, in 2021. He is currently working toward the Ph.D. degree in electrical and computer engineering with the University of California, San Diego.

His research interests include machine learning and deep learning, distributed learning, federated learning, continual learning, and optimization.

Chen Chen received the Ph.D. degree from the Department of Electrical Engineering, University of Texas at Dallas, Richardson, TX, USA, in 2016.

He is currently an Assistant Professor with the Center for Research in Computer Vision, University of Central Florida, Orlando, FL, USA. His research interests include computer vision, efficient deep learning, and federated learning.

Dr. Chen is an Associate Editor for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, Journal of Real-Time Image Processing, and IEEE JOURNAL ON MINIATURIZATION FOR AIR AND SPACE SYSTEMS. He was the recipient of the David Daniel Fellowship (Best Doctoral Dissertation Award) at the University of Texas at Dallas.

Mubarak Shah (Life Fellow, IEEE) received the Ph.D. degree in computer science from Wayne State University, Detroit, MI, USA, in 1986. He has published extensively on topics related to visual surveillance, visual tracking, human activity and action recognition, object detection and categorization, shape from shading, geo registration, visual crowd analysis, etc. He has been ACM and IEEE Distinguished Visitor Program speaker and is often invited to present seminars, tutorials and invited talks all over the world. He is the UCF Trustee Chair Professor, is the founding Director of Center for Research in Computer Vision at University of Central Florida (UCF).

Dr. Shah is a fellow of ACM, NAI, IAPR, AAAAS, AIAA and SPIE; and a member of Academy of Science, Engineering, and Medicine of Florida (ASEMFL). He is a recipient of ACM SIGMM Technical Achievement award; ACM SIGMM Test of Time Honorable Mention Award for his paper in Proceedings of the 14th ACM International Conference on Multimedia, MM’06; International Conference on Pattern Recognition (ICPR) 2020 Best Scientific Paper Award; IEEE Outstanding Engineering Educator Award; Harris Corporation Engineering Achievement Award; an honorable mention for the ICCV 2005 Where Am I? Challenge Problem; 2013 NGA Best Research Poster Presentation; 2nd place in Grand Challenge at the ACM Multimedia 2013 conference; and runner-up for the best paper award in ACM Multimedia Conference in 2005 and 2010. At UCF, he has received Pegasus Professor Award; University Distinguished Research Award; Faculty Excellence in Mentoring Doctoral Students; Scholarship of Teaching and Learning Award; Teaching Incentive Program award; Research Incentive Award.

Bill Lin (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering and computer sciences from the University of California, Berkeley, CA, USA, in 1985, 1988, and 1991, respectively.

He is a Professor in Electrical and Computer Engineering, University of California, San Diego, where he is actively involved with the Center for Wireless Communications (CWC), the Center for Networked Systems (CNS), and the Qualcomm Institute in industry-sponsored research efforts. His research has led to over 200 journal and conference publications, including a number of Best Paper awards and nominations.

Dr. Lin holds 5 awarded patents. He has served as the General Chair and on the executive and technical program committee of many IEEE and ACM conferences, and he has served as an Associate or Guest Editor for several IEEE and ACM journals as well.