IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

FedArtML: A Tool to Facilitate the Generation of Non-IID Datasets in a Controlled Way to Support Federated Learning Research

DANIEL MAURICIO JIMENEZ G.[1], ARIS ANAGNOSTOPOULOS[2], IOANNIS CHATZIGIANNAKIS[3] AND ANDREA VITALETTI[4]

[1,2,3,4]Department of Computer, Control and Management Engineering, Sapienza University of Rome, Rome, Italy (e-mails: danielmauricio.jimenezgutierrez@uniroma1.it[1], aris@diag.uniroma1.it[2], ioannis.chatzigiannakis@uniroma1.it[3], andrea.vitaletti@uniroma1.it[4])

Corresponding author: Daniel Mauricio Jimenez G (e-mail: danielmauricio.jimenezgutierrez@uniroma1.it).

ABSTRACT

Federated Learning (FL) enables collaborative training of Machine Learning (ML) models across decentralized clients while preserving data privacy. One of the challenges that FL faces is when the clients' data is not independent and identically distributed (non-IID). It is, therefore, crucial to quantify how non-IID data impacts performance. However, due to the limited number of federated data available, it is not easy to carry out real-world simulations. In this work, we propose for the first time (1) the Hist-Dirichlet-based and Min-Size-Dirichlet methods for partitioning data into multiple nodes using the features and quantity distribution and the Dirichlet distribution. We use the (2) Jensen-Shannon and Hellinger distances for quantifying the degree of IID data. Moreover, we implemented (3) state-of-the-art partitioning methods based on the labels' distribution across clients. All our proposals are open-source in a library called FedArtML, publicly available on PyPI. It facilitates research on cross-silo and cross-device FL, allowing a systematic and controlled partition of centralized datasets using the label, features, and quantity skewness. To demonstrate the value of our proposed methods and the robustness of FedArtML, we experimented in the ECG arrhythmia detection field with Physionet 2020 data. Our results demonstrate that our tool generates federated datasets for multi-client model training and accurately measures client distribution heterogeneity. Our approach achieves 48% higher non-IID-ness than existing feature skew methods, providing more granularity. Furthermore, we validate our simulated federated datasets against real-world data, revealing only a 2% F1-Score difference, affirming the method's real-life applicability.

INDEX TERMS Centralized Datasets, Client's Heterogeneity, Federated Datasets, Federated Learning, Heterogeneity Metrics, Machine Learning, Non-IID-ness.

I. INTRODUCTION

In recent years, we have witnessed the development of impressive machine learning (ML) services [1], [2], [3]. The latter is witnessed with the introduction of Generative Artificial Intelligence (a.k.a GenAI) tools such as Chat GPT [4], Gemini [5] and Dall-E [6]. The success of Chat GPT by OpenAI is likely the most evident proof of the advancements made by the research in this field. GPT-3's model [7] was trained using a vast amount of text available on the internet collected by OpenAI (i.e., centralized ML - in the sequel CL). However, some relevant scenarios exist where data are sensible and private. For example, health [8], [9] or investment [10], [11] data are usually protected and cannot be easily shared. The General Data Protection Regulation (GDPR) [12] is a European Union (EU) regulation on data protection and privacy that defines several strict rules to handle sensitive data and limits data transfer across organizations and countries. Consequently, some are claiming that such strict rules will increase the emergence of data silos [13], namely the collection of data held by one group that is not quickly or thoroughly accessible by other groups, even in the same organization.

Furthermore, while most companies allow access to their models developed using CL by suitable API, they consider the model a company asset and then protect it. Finally, in CL settings, once data are shared, the owner loses control and has to trust the service provider to handle their data according to specific rules without guaranteeing indisputable technical means of verification. Consequently, a recent trend is emerging named Federated Learning (FL) [14], [15], [16] , a technique to run ML algorithms in a federation of nodes, each contributing with its dataset to the computation of a common model, without collecting them in a single place as CL. In FL, the data remains in the control of the data owner; all participants share only the model. The latter approach properly deals with *efficiency* [17], [18] (the global model converges to an optimal state despite the decentralized nature of training) and *privacy*, [19], [20] (local data remains on devices, addressing privacy concerns related to sharing sensitive information).

However, collecting data in multiple places presents severe challenges [21], [22]. Since local devices produce data, the usual assumption behind CL is that data are Identically and Independently Distributed (IID), which does not necessarily hold. In other words, non-IID data characterizes FL. Therefore, a natural research question is how CL and FL performance behave under different configurations of non-IID data. Thus, there is a need for *metrics that systematically evaluate the level of non-IID data*, as the current literature often relies on ad-hoc FL partitions [23], [24], [25]. In addition, given the widespread availability of centralized datasets, there is a necessity for a tool to *generate cross-silo and cross-device FL datasets* [26] with a desired level of non-IID-ness, starting from these widely available datasets.

A. MOTIVATION

Quantifying the degree of non-IID-ness becomes crucial as it provides insights into the heterogeneity of data distributions across clients. Understanding this variance aids in devising more effective FL algorithms, ensuring the robustness, fairness, and generalizability of models. In addition to the need for metrics to characterize the non-IID-ness of the data and a practical tool to create FL data, we think evaluating FL algorithms still needs a standard benchmark for a systematic study [24, 25]. In this perspective, and given the non-IID peculiar nature of FL, we do believe that the availability of non-IID data sets with a precise measure of the level of non-IID-ness is a fundamental ingredient to developing the benchmark mentioned above.

The availability of centralized datasets used to train and test CL algorithms is much more common than FL datasets. So, in this paper, we focus on using well-known CL datasets that are publicly available to generate FL ones with a measurable level of non-IID-ness. To the best of our knowledge, this work is the first attempt to propose metrics to control and evaluate the distribution heterogeneity among the multiple local datasets of the FL nodes. The latter allows us to quantify and control the number of local nodes participating in the federation and the imbalanced properties of local data to evaluate their effects on the federated algorithms. In addition, with our brand-new methods for partitioning centralized data using the feature and quantity distribution of the clients, we generate the datasets from centralized publicly available, and we do not have to face all the regulation and privacy concerns to access real-world federated data.

Thanks to the flexibility of the proposed metrics to evaluate the level of non-IID-ness, we can run experiments similar to the ones presented by Li et al. [24] but with a higher degree of granularity and consistency that allows us to have better confidence in the applicability of the results of our study to a broader context.

B. CONTRIBUTION

The following points summarize the contributions of our paper and tool:

1) We use the Jensen-Shannon (JSD) and Hellinger distance (HD) as reliable alternatives to quantify the level of data heterogeneity, systematically analyzing their performance in an FL setting.
2) We introduce two brand-new methods for partitioning centralized data into federated data by considering the feature and quantity distribution of the clients.
3) We provide FedArtML, a flexible, publicly available tool for creating cross-silo and cross-device FL datasets starting from any available centralized one, proving the tool's robustness by comparing the simulated FL data to a real-life FL dataset.

To our knowledge, no systematic analysis measures the non-IID-ness of clients in FL (see section III). In addition, as far as we know, a tool to create federated datasets from centralized datasets while controlling and quantifying the degree of heterogeneity (non-IID-ness) has yet to be developed.

The Python code employed for FedArtML is publicly available[1]. It contains the source code and getting started notebook examples for the users to get used to its functionalities. Moreover, users can access the package on PyPI and easily install it by running the following command: `pip install fedartml`.

C. ROADMAP

In the next Section, we provide the related literature. Section III presents metrics for quantifying the data's non-IID-ness. In Section IV, we provide techniques to create FL datasets. In Section V, we present the specifics of our tool. In Section VI, we introduce the data and models employed for the experiments; in Section VII, we depict the aggregation algorithm employed, and in Section VIII, we use it to showcase our tool. In Section IX, we conducted an ablation study of the methods and metrics presented in this

[1]Code available at: https://github.com/Sapienza-University-Rome/FedArtML

work. Section X exhibits the limitations of our tool. Finally, we conclude and provide some implications and future work in Section XI.

II. RELATED WORK

The heterogeneity in the clients' distribution in FL is a well-known challenge [27], [28], [29], [30], [31] that can impact the performance and convergence of the learning process. These studies are based on the non-IID distribution, where clients have very different data (like a hospital specializing in one disease), which leads to local updates that diverge during training, ultimately hurting the overall model's accuracy and convergence. This is further complicated because parametric and non-parametric models react differently to non-IID data due to their distinct training mechanisms. Therefore, quantifying data heterogeneity is crucial to acting against heterogeneous scenarios. Chen et al. [32] introduced a general Bayesian Personalized FL framework to decompose and jointly learn shared and personalized uncertainty representations on statistically heterogeneous client data over time. They utilized the HD to investigate the average generalization error of their solution. Fang et al. [33] used the JSD to constrain the loss model updates to improve the model stability. The main idea was to constrain the output consistency of the classifier on different augmentations of the same image. Zhang et al. [34] employed the JSD, expressed as the summation of two Kullback-Leibler (KL) divergences, to quantify the trade-offs between privacy leakage, utility loss, and efficiency reduction, which led them to the No-Free-Lunch (NFL) theorem for the FL system. Abay et al. [35] utilized the HD between the training data-induced distribution and the trained model-induced distribution to measure the similarity between the truth and model predictions, where a lower HD indicates greater similarity. Notice that previous works employed JSD and HD to measure the generalization error of their proposals and constrain loss of the FL models. However, as far as we know, *JSD and HD metrics have not been employed to quantify the degree of heterogeneity of the clients' distribution in an FL setting.* The latter exemplifies how our proposal differs from the state-of-the-art uses of JSD and HD.

Researchers have proposed relevant approaches to create federated datasets using a centralized dataset as input while varying the heterogeneity in the distribution of the client's labels in FL. Zeng et al. [36] showcased FedLab, a lightweight open-source framework for the simulation of FL. Its design focuses on FL algorithm effectiveness and communication efficiency. It allows server optimization, client optimization, communication agreement, and communication compression customization. Lai et al. [37] introduced the FedScale framework with datasets encompassing a wide range of critical FL tasks, ranging from image classification and object detection to language modeling and speech recognition. Each dataset has a unified evaluation protocol using real-world data splits. Ogier et al. [38] presented a novel cross-silo dataset suite focused on healthcare, FLamby (Federated Learning AMple Benchmark of Your cross-silo strategies), to bridge the gap between theory and practice of cross-silo FL. FLamby encompasses seven healthcare datasets with natural splits, covering multiple tasks, modalities, and data volumes, each accompanied by a baseline training code. Nevertheless, notice that the previous frameworks only have algorithms and FL datasets to simulate label skew, but *they do not include feature and quantity skewness partition methods*.

Hsieh et al. [25] proposed a federated dataset creation approach explicitly partitioning centralized datasets using label distribution. However, for the feature skew partition, only one relevant method has been introduced by Li et al. [24], in which the features are transformed by adding Gaussian Noise to its local dataset to achieve different feature distributions where users can change σ to increase the feature dissimilarity among the parties. The limitation of this approach is that injecting high levels of variation, i.e., high variance leads to a poor model's performance. We tackle this issue by including a *new-brand algorithm* for partitioning centralized data into federated data by controlling the desired level of *non-IID-ness of the features* present in the local nodes. Li et al. [24] also proposed a method to simulate quantity skew using the Dirichlet distribution to allocate different amounts of data samples to each party. They used the parameter β to control the imbalance level of the quantity skew. However, their approach is not applicable when the data is too small or the number of clients is too big. We propose an *alternative algorithm that successfully tackles this limitation*.

Moreover, Li et al. [24] and Hsieh et al. [25] provided different techniques for mimicking non-IID data situations, which aided researchers in investigating and comprehending the non-IID data setting in FL. They suggested various data partitioning approaches encompassing the most common non-IID data cases. However, researchers selected those strategies arbitrarily despite being thoroughly examined and defined. Our tool offers a solution by *allowing the selection of non-IID scenarios in a systematic and regulated manner*. Additionally, the mentioned works [24], [25] employed techniques such as Dirichlet distribution, Percent of non-IID method and Gaussian Noise to create FL datasets. Yet, none of them used any metrics for measuring the degree of heterogeneity among clients' distributions, an aspect our proposed tool includes.

Table 1 compares FedArtML to the other popular FL simulation tools presented above (FedLab, FedScale, FLamby, and NIID-Bench). While all these tools can simulate label skew, and some offer functionalities for quantity skew and varied datasets, FedArtML stands out in its ability to generate data with controlled feature skew. Additionally, FedArtML incorporates non-IID metrics (i.e., JSD and HD) to quantify the degree of non-IID-ness in the generated datasets for each type of skewness. Furthermore, FedArtML allows users to upload custom datasets and provides an

interactive UI for a more user-friendly experience compared to some of the other tools.

Attribute	FedLab [36]	FedScale [37]	FLamby [38]	NIID-Bench [24]	FedArtML
non-IID metrics	✗	✗	✗	✗	✓
Label skew	✓	✓	✓	✓	✓
Feature skew	✗	✗	✗	✓	✓
Quantity skew	✓	✗	✗	✓	✓
Varied datasets	✓	✓	✗	✓	✓
Custom data use	✓	✗	✗	✗	✓
Interactive UI	✗	✓	✗	✗	✓

Table 1. Comparison of FedArtML to existing tools. Notice that at the date of publication of this paper, our tool is the only one offering metrics to quantify the non-IID-ness in FL.

III. METRICS FOR DISTRIBUTION HETEROGENEITY

A generic dataset (a.k.a. *centralized dataset*[2]) $\mathcal{D}$ is a collection of n tuples $\mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$, where $\mathbf{x}_i = [(x_i)_1, \ldots, (x_i)_m]$ is the feature representation of the ith element (sample) in the dataset, and $y_i \in \{1, \ldots, \ell\}$ is the (true) label of the ith element.

In the FL setting, the dataset $\mathcal{D}$ is distributed over K clients. We let $\mathcal{D}_i$ be the set of elements of the ith client. That is:

$$\mathcal{D} = \cup_{i=1}^{K} \mathcal{D}_i \qquad \text{and for } i \neq j: \qquad \mathcal{D}_i \cap \mathcal{D}_j = \emptyset.$$

Defining the type of non-IID-ness in FL is relevant since it can drastically influence the performance of the models. We follow the settings of previous work [39, 40]. For a supervised learning task on client i (local node i), we assume that each data sample $(x, y) \in \mathcal{D}_i$, where x is the input attributes or features, and y is the label, following a local distribution $P_i(\mathbf{x}, y)$. Let us define: with $P_i^Y(y)$, the

$$P_i^Y(y) = \sum_{\substack{(\mathbf{x},z)\in\mathcal{D}_i \\ z=y}} P_i(\mathbf{x}, z) \qquad P_i^{X_\ell}(x) = \sum_{\substack{(\mathbf{x},y)\in\mathcal{D}_i \\ x_\ell=x}} P_i(\mathbf{x}, y)$$

ith client labels' distribution and $P_i^{X_\ell}(x)$ the distribution over the ℓth input feature of the ith client. Then, the classification for non-IID is as follows:

- Regarding the concept of *identically distributed*:
  1) **Label skew**: Means that the label distribution $P_i^Y(y)$ of different clients is different.
  2) **Feature skew**: Occurs when the distribution of the features $P_i^{X_\ell}(x)$ varies from client to client.
  3) **Quantity skew**: Refers to the significant difference in the number of examples of different client data $P_i(\mathbf{x}, y)$.
- Regarding the concept of *independent*:
  4) **Spatiotemporal skewness**: Refers to the inner correlation of data in the time (or space) domain. In other words, the distribution $P_i(\mathbf{x}, y)$ is not stationary but depends on the time or space.

[2]Notice that this definition of a centralized dataset includes tabular data, images, medical data, graph data, and any type of dataset that can be expressed as a collection of arrays.

Notice that the spatiotemporal skewness is not included in this work. The latter is due to the complexity of incorporating location and time-based parameters during data partitioning, and it is currently under development. We plan to explore methods for simulating spatiotemporal skewness in future versions of FedArtML, allowing researchers to investigate the impact of these factors on model performance. Given the above definitions, we introduce two metrics to measure the non-IID-ness across two or more clients.

A. JENSEN-SHANNON DISTANCE (JSD)

The Jensen-Shannon distance (JSD) is a metric that measures the similarity between two probability distributions, calculated by taking the square root of the Jensen-Shannon divergence, a smoothed and symmetrical version of the Kullback-Leibler (KL) divergence. The formula [41] for the Jensen-Shannon distance between two probability distributions, $P_1^Y(y)$ and $P_2^Y(y)$, is given by Equation (1):

$$\text{JSD}\left(P_1^Y(y), P_2^Y(y)\right) = \sqrt{0.5 \cdot \text{KL}(P_1^Y(y), M) + 0.5 \cdot \text{KL}(P_2^Y(y), M)} \tag{1}$$

where $\text{KL}\left(P_1^Y(y), M\right)$ is the KL divergence between $P_1^Y(y)$ and the mean distribution $M = \left(P_1^Y(y) + P_2^Y(y)\right)/2$.

The JSD can be extended by calculating the average distance between each pair of distributions to include more than two distributions. The JSD has the advantage of being bounded between 0 and 1, with 0 indicating identical distributions and 1 indicating entirely different distributions. Moreover, it satisfies the triangular inequality [41].

B. HELLINGER DISTANCE (HD)

The Hellinger Distance (HD) is a metric for measuring the separation between two probability distributions calculated as in Equation (2).

$$\text{HD}(P_1^Y(y), P_2^Y(y)) = \frac{1}{\sqrt{2}}\sqrt{\sum_{y\in Y}\left(\sqrt{P_1^Y(y)} - \sqrt{P_2^Y(y)}\right)^2} \tag{2}$$

To include more than two distributions, the HD can be extended by first calculating the average distance between each pair of distributions. The Hellinger Distance is between 0 and 1, with 0 denoting similar distributions and 1 indicating completely different distributions. It is symmetric and satisfies the triangular inequality [42].

IV. METHODS FOR SYNTHETIC PARTITIONING

This section considers various methods for creating synthetic partitions of a centralized dataset into multiple clients (federating datasets) for label, features, and quantity skew simulation. In the following, we introduce methods in which one parameter controls heterogeneity among client distributions.

A. LABEL SKEW PARTITION

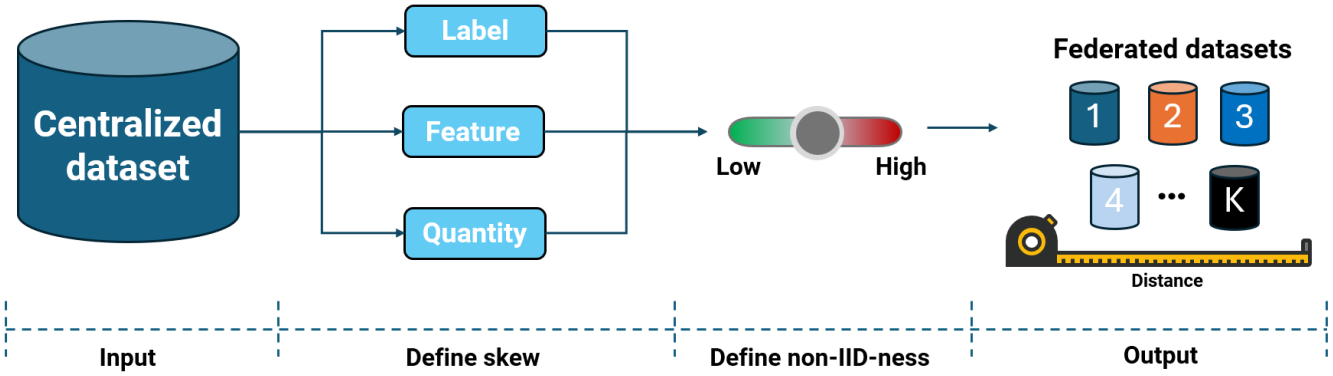

Figure 1. High-level partition process for FedArtML

1) Dirichlet-based method

This technique uses the Dirichlet distribution (DD) to partition the data. The DD is a probability distribution that produces a set of random numbers that sum up to one [43]. In federating datasets, the DD generates a set of weights to partition the data into various subsets. We can control the skewness of the dataset by adjusting the parameter α. The value of α manages the degree of non-IID-ness: $\alpha = 100$ mimics identical local data distributions. The smaller α is, the more likely the clients hold examples from only one class (randomly chosen) [23].

2) Percentage of nonIID-ness-based method

It is a technique that partitions the centralized data based on the desired Percentage of non-IID data points from the centralized dataset [25] by adjusting the Percentage of the non-IID parameter, allowing control of the skewness by changing the fraction of non-IID data. For example, 20% non-IID indicates 20% of the dataset gets partitioned by labels, while the remaining 80% is partitioned uniformly at random [25]. For the sake of space, this method is not showcased in this paper but belongs to the FedArtML tool.

B. FEATURE SKEW PARTITION

1) Gaussian noise method

In this method, for each client, different levels of Gaussian noise are added to the local dataset to achieve different feature distributions [24]. Precisely, given user-defined noise level σ, they add noises $\hat{x} \sim \text{Gau}\left(\sigma \cdot \frac{i}{K}\right)$ for client i, where $\hat{x}$ represents the resulting features after adding the noise level to the original features $\text{Gau}\left(\sigma \cdot \frac{i}{K}\right)$ is a Gaussian distribution with mean 0 and variance $\sigma \cdot \frac{i}{K}$ and K the number of clients. Users can change σ to increase the feature dissimilarity among the local nodes. One drawback of this approach is that injecting high σ levels (i.e., high variance) directs to a poor model's performance, leading to inaccurate conclusions about the FL model's performance.

2) Hist-Dirichlet-based method

It is *our proposed method* to tackle the mentioned problem of the Gaussian Noise approach. The algorithm begins by characterizing the features of each client and then applying a binning process to categorize them. Then, the participation of each label's class inside each client is defined using the DD with a given α. The latter guarantees that the CL data gets split into disjoint datasets and, as a result, the performance of the models will not be affected, given that the dataset is not transformed, as happens in the Gaussian Noise method, but it still gets distributed among the clients.

C. QUANTITY SKEW PARTITION

1) Dirichlet method

In quantity skew, the size of the local dataset varies across parties. Although data distribution may still be consistent among the parties, it is interesting to see the effect of the quantity imbalance in FL. Like label skew imbalance, the DD allocates different data samples to each party. It is sampled $q \sim Dir(\alpha)$ and allocates a q_j proportion of the total data samples to each client. The parameter α controls the imbalance level of the quantity skew [24]. The mentioned method has one drawback: it is not usable when the data is too small, or the number of clients is too big.

2) Min-Size-Dirichlet method

It is *our proposed method* to tackle the mentioned problem of the Dirichlet approach for quantity skew. The algorithm begins by defining an α for the DD distribution and generating the desired participation proportions for each client. Then, a minimum required size (a.k.a, number of examples) gets defined for each client. Thus, the minimum proportion size $MinSize$ is $MinSize = \frac{MinRequiredSize}{n}$, where n is the total number of examples in the centralized dataset. If the defined proportions are smaller than $MinSize$, the proportions get replaced by $MinSize$. Finally, the proportions get normalized between 0 and 1. The latter approach guarantees that the method will converge even if the data is too small or there are many clients.

V. THE FEDARTML TOOL

The FedArtML tool is an FL library compatible with Python versions 3.5 and above. The current version of the package is 0.1.32, including all the functionalities introduced in section IV to create federated datasets. It also incorporates the metrics used for non-IID-ness (JSD, HD). The interactive User Interface (UI) has sliders to select the desired values (number of clients and degree of non-IID-ness), with plots that automatically adjust to depict the clients' distributions. Thus, stacked barplot, scatterplot, and barplot divided by client distributions showcase the label's distributions for each client in real time after setting the desired percentage of heterogeneity and the number of clients.

The high-level partition process of FedArtML is depicted in Figure 1. It takes a centralized dataset as input; then, the user defines the skew desired (label, feature, or quantity) to split the data. Afterward, the user can select the desired level of non-IID-ness and the number of clients (K). The tool then partitions the data, allocating specific quantities to each client while ensuring a desired level of dissimilarity between partitions using metrics like JDS and HD. This way, researchers can generate federated datasets with characteristics they define, facilitating more realistic simulations for their FL research.

Figure 2 depicts a screenshot of the FedArtML tool for a partition with $\alpha = 0.01$ and four clients. When the user selects the α parameter of the desired DD, the stacked barplot distribution automatically updates to reflect the label distributions for each client. A similar behavior occurs when the user controls the sliders to choose the number of partitions.

VI. DATA AND MODELS

Data. In this work, we consider high-quality 12-lead electrocardiography (ECG) data [44] obtained from different health centers that use different ECG recording devices with varying levels of recording accuracy to classify different types of cardiac anomalies. The latter belongs to the Physionet 2020 competition that integrates six diverse datasets [45]. We leveraged some experiments with a random oversampling technique to balance the classes (arrhythmias) and improve the model's performance.

Models. We select two of the most high-achieving models of the PhysioNet 2020 competition [44]. The first is a Deep Neural Network (DNN), selected because they are widely employed in ECG arrhythmia predictions due to their capacity to learn hierarchical representations of data through multiple layers of interconnected neurons [46]. It comprises one input layer, three hidden layers, and one output layer [47]. The input layer uses as many units as the number of features used in the training set. The three layers contain 500 hidden units each, while the last layer is formed by considering the neurons equal to the number of classes to predict. Additionally, the hidden layers employed the ReLu activation and the SoftMax for the output layer. The mentioned activation functions arose after the fine-tuning method. The second uses a Long-Short-Term-Memory (LSTM) methodology [47], chosen because those models can effectively model the dynamic nature of heart rhythm variations over time, allowing for accurate detection and classification of arrhythmias [48]. The input layer uses as many neurons as the number of variables to predict, while the LSTM cell and the output layer are composed of the same number of neurons as the number of classes to predict. Regarding the output layer's activation function, their values appeared after the fine-tuning process, where the ReLu and SoftMax functions provided the best performance. While Convolutional Neural Networks (CNNs) and Transformers are commonly utilized in ECG arrhythmia detection, it is essential to note that they are not considered in this work. This is because they are primarily designed for processing signals and image-like datasets, whereas our tool is showcased using tabular data. Nevertheless, our tool can split centralized datasets such as images, medical data, graph data, etc.

Remark that our paper's main goal is to develop a tool to investigate the effect of non-IID-ness on FL algorithms. For this reason, we decided to study the impact of non-IID-ness on two relatively simple models where the impact of hyperparameter tuning is limited. Indeed, in our reference DNN and LSTM models, we do not have to set up any hyperparameter. In this scenario, coherently with our main goal, the experimental results can be exclusively associated with the non-IID-ness of the input. In future work, we plan to exploit this tool to thoroughly investigate the effects of non-IID-ness on state-of-the-art algorithms.

The hardware specification used to train and evaluate the models is shown in Table 2.

Component	Specification
Disk size	108 GB
Processors' model	Intel(R) Xeon(R) CPU @ 2.20GHz
Number of processors	2
Memory	51.0 GB
Operating System	(Linux) Ubuntu 18.04.5 LTS
GPU	GeForce RTX 3070 8GB
Python version	3.7.13

Table 2. Hardware specification used to train models

VII. AGGREGATION ALGORITHM

Researchers have proposed several algorithms for constructing a single global model based on the local models trained by each client using the locally available data. In this work, we use FedAvg [49], the most well-studied algorithm in FL. Each round briefly starts with the central server that transmits the global model to a random set of clients. In the sequel, each client receiving a model updates it using the local dataset. The updated models are transmitted back to the central server, which computes the averages of the received local models and updates the global model.

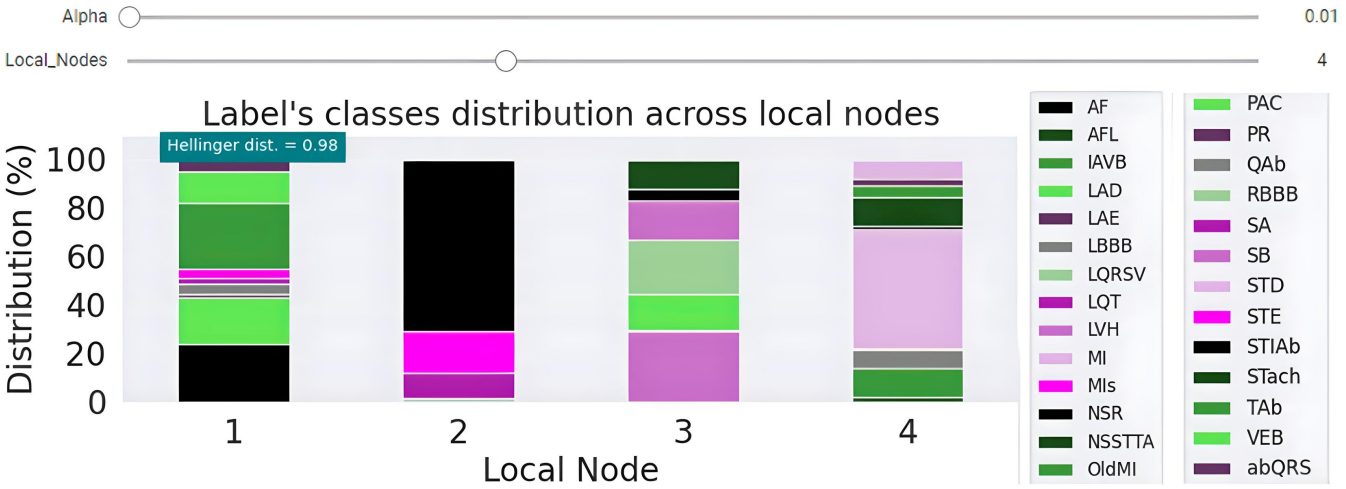

Figure 2. Stacked distribution of labels partitioned into four clients, $\alpha = 0.01$, with an $HD = 0.98$.

VIII. EXPERIMENTS AND RESULTS

This section provides three examples of how FedArtML can facilitate experimentation for federated learning research considering label, features, and quantity skew partitions. In addition, it showcases the robustness of the tool. We merge the six datasets and then partition them for a desired level of skewness. Given the newly partitioned dataset, we evaluate the performance of the DNN and LSTM models using the FedAvg algorithm. Based on the datasets, models, and algorithms presented, we showcase the Dirichlet-based method leveraged on the labels' distribution, the Hist-Dirichlet-based and Gaussian Noise methods to create synthetic partitions leveraged on the distribution of the features, and the Min-Size-Dirichlet method for quantity skew. For the sake of the extension, the Percentage of non-IID-ness-based method is not showcased in this work but remains available in the FedArtML tool.

A. LABEL SKEW PARTITION SHOWCASE

This experiment demonstrates how the Dirichlet-based method can partition the Physionet FL dataset based on the label's distribution. We first start by merging all the partitions into a centralized dataset and, in the sequel, split into disjoint clients, considering two, four, six, eight, ten, one hundred and five hundred clients and employing $\alpha = \{0.03, 0.3, 1, 6, 1000\}$ to control the label skewness of the resulting partition. We use the JSD and the HD for each resulting partition to measure the similarity between the different clients' distributions. Table 3 shows the resulting metrics for each combination. Observe how the distance of the partitions increases as we increase the value of α. In addition, notice that it is impossible (NaN) to get pathologically high non-IID-ness levels when the number of clients is 500 since the local nodes' size (proportion) becomes much smaller. Thus, the algorithm can't find a way to split the data.

Table 3. Achieved label skewness as measured using the JSD and HD metrics when partitioning the Physionet dataset into different numbers of clients (CLS) for $\alpha = 0.03, 0.3, 1, 6, 1000$.

	JSD					HD				
CLS \ α	**1000**	**6**	**1**	**0.3**	**0.03**	**1000**	**6**	**1**	**0.3**	**0.03**
2	0.108	0.530	0.564	0.810	0.994	0.107	0.505	0.513	0.764	0.978
4	0.050	0.334	0.557	0.726	0.957	0.062	0.350	0.577	0.732	0.955
6	0.036	0.322	0.491	0.700	0.963	0.049	0.374	0.541	0.746	0.972
8	0.029	0.262	0.449	0.662	0.933	0.042	0.314	0.514	0.733	0.959
10	0.035	0.294	0.432	0.644	0.914	0.052	0.368	0.513	0.737	0.953
100	0.030	0.172	0.329	0.492	0.729	0.056	0.293	0.531	0.753	0.946
500	0.092	0.171	0.299	0.422	NaN	0.195	0.344	0.572	0.762	NaN

Figure 3. F1-Score of the DNN and LSTM models on the Physionet dataset for CL, and the Physionet dataset partitioned for 4 clients to achieve different label skewness as measured based on JSD, for FL using FedAvg.

Consider the case where we wish to evaluate the performance of four clients. After using FedArtML with the resulting partitions, we measure the performance of the two models for the CL and FL versions using the FedAvg aggregation algorithm. Figure 3 depicts the performance achieved in terms of the F1-score for the resulting partitions based on the measured distance using the JSD metric. Remark that the JSD value of the original dataset is included to compare the performance quickly. Notice that when the label skewness increases, the achieved performance drops.

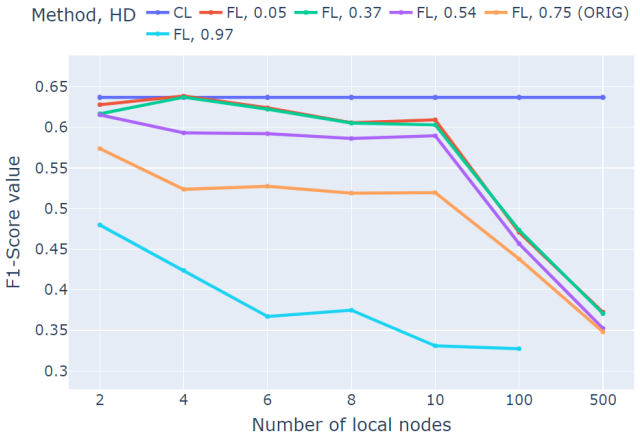

Figure 4. F1-Score of the DNN model on the Physionet dataset for CL and FL, partitioned for $2 \ldots 500$ clients for different label skewness as measured based on HD $= \{0.05, 0.37, 0.54, 0.75, 0.97\}$, for FL using FedAvg.

Another approach is to evaluate the models' performance by varying the number of partitions from two to five hundred and different non-IID characteristics. In this experiment, we use FedArtML to partition the Physionet dataset into multiple datasets with a label skewness of $HD = \{0.05, 0.37, 0.54, 0.75, 0.97\}$. Figure 4 depicts the resulting performance of the DNN model for the CL and FL versions using the FedAvg algorithm. Thus, increasing the HD among the partitions (increasing the degree of non-IID-ness) highly and consistently impacts the model's performance. Additionally, when considering 100 and 500 clients, the gap for different non-IID-ness levels gets smaller while the overall performance drastically decreases.

B. FEATURE SKEW PARTITION SHOWCASE

This experiment shows how our Hist-Dirichlet-based method can partition the Physionet FL dataset based on the feature distribution. We first start by merging all the partitions into a centralized dataset and next we split into disjoint clients, considering two, four, six, eight, ten, one hundred and five hundred clients and employing $\alpha = \{0.03, 0.3, 1, 6, 1000\}$ to control the feature skewness of the resulting partition. We use the JSD and the HD for each resulting division to measure the similarity between the different clients' distributions. Table 4 shows the resulting metrics for each combination. Observe how the distance of the partitions increases as we decrease the value of α. Moreover note that achieving pathologically high levels of non-IID-ness (NaN) is unattainable when the number of clients is 500. This is because the size or proportion of local nodes becomes significantly smaller, making it challenging for the algorithm to divide the data effectively.

Consider the case where we wish to evaluate the performance of four clients. After using FedArtML with the resulting partitions, we measure the performance of the DNN model in the CL and FL versions using the FedAvg aggregation algorithm. Moreover, we compare the performance of the Gaussian Noise method vs. the Hist-Dirichlet-based approach. Figure 5 depicts the performance achieved in terms of the F1-score for the resulting partitions based on the measured distance using the HD metric, demonstrating the superiority of our proposed method. Notice that when the feature skewness increases, the achieved performance for the FL fashion, in terms of the F1-Score, does not diminish significantly compared to the CL version. In addition, the performance of CL and FL approaches using the Hist-Dirichlet-based method does not decrease when the HD increases, as occurs for the Gaussian Noise method. This is because, over a given threshold, the noise becomes the most significant component, making 1) difficult to distinguish differences among the samples in the clients and 2) impacting the performance of the models.

Table 4. Achieved feature skewness as measured using the JSD and HD metrics when dividing the Physionet dataset into different numbers of clients for Hist-Dirichlet with $\alpha = \{0.03, 0.3, 1, 6, 1000\}$.

	JSD					HD				
CLS \ α	**1000**	**6**	**1**	**0.3**	**0.03**	**1000**	**6**	**1**	**0.3**	**0.03**
2	0.017	0.235	0.447	0.845	0.996	0.014	0.197	0.38	0.808	0.987
4	0.013	0.179	0.510	0.723	0.967	0.013	0.174	0.524	0.750	0.962
6	0.013	0.194	0.459	0.711	0.969	0.014	0.212	0.495	0.758	0.973
8	0.013	0.195	0.444	0.686	0.944	0.014	0.226	0.498	0.762	0.965
10	0.013	0.203	0.442	0.660	0.942	0.015	0.252	0.521	0.754	0.970
100	0.011	0.175	0.375	0.535	0.79	0.016	0.263	0.531	0.737	0.944
500	0.032	0.136	0.289	0.428	NaN	0.059	0.262	0.551	0.769	NaN

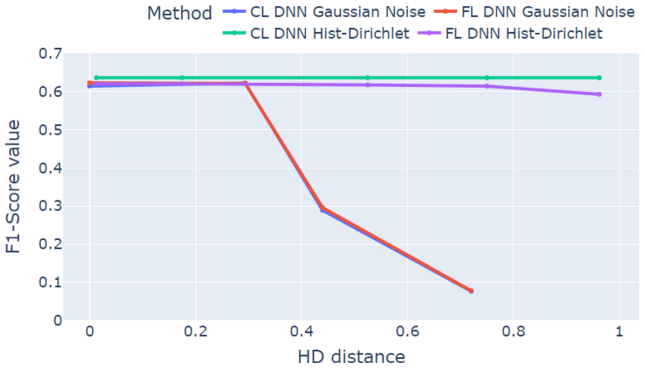

Figure 5. F1-Score of the DNN model for CL and FL using FedAvg; the dataset partitioned for 4 clients achieving different feature skewness measured by HD, comparing Gaussian Noise and Hist-Dirichlet methods.

C. QUANTITY SKEW PARTITION SHOWCASE

This experiment shows how our Min-Size-Dirichlet method can partition the Physionet FL dataset based on the quantity skew (number of examples per client). Following the same approach as in the previous chapters, we split the CL dataset into two, six, ten, one hundred, and five hundred clients. We set $\alpha = \{0.03, 0.3, 1000\}$ to control the quantity skewness of the resulting partition. We use the JSD and the HD for each resulting partition to measure the similarity between the different clients' distributions. Table 5 shows the resulting metrics for each combination. Observe how the distance of the partitions increases as we decrease the value of α.

Consider the case where we wish to evaluate the per-

Table 5. Achieved quantity skewness as measured using the JSD and HD metrics when dividing the Physionet dataset into different numbers of clients (CLS) for Mis-Size-Dirichlet with $\alpha = \{0.03, 0.3, 1000\}$.

CLS \ α	JSD			HD		
	1000	**0.3**	**0.03**	**1000**	**0.3**	**0.03**
2	0.008	0.301	0.699	0.007	0.254	0.673
6	0.005	0.233	0.346	0.006	0.245	0.353
10	0.004	0.171	0.274	0.005	0.191	0.282
100	0.001	0.03	0.07	0.001	0.05	0.08
500	0.001	0.009	0.01	0.001	0.009	0.02

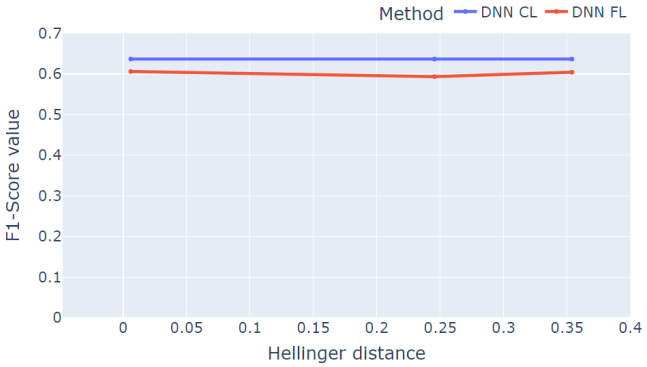

Figure 6. F1-Score of the DNN model for CL and FL using FedAvg; the dataset partitioned for six clients achieving different quantity skewness measured by HD.

formance of six clients. After using FedArtML with the resulting partitions, we measure the performance of the DNN model in the CL and FL versions using the FedAvg aggregation algorithm. Figure 6 depicts the performance achieved in terms of the F1-score for the resulting partitions based on the measured distance using the HD metric. Notice that when the quantity skewness increases, the achieved performance for the FL fashion, in terms of the F1-Score, remains almost constant compared to the CL version. Notice that the previous result was also obtained by Li et al. [24], leveraging the significance of such behavior.

D. COMPARING REAL-WORLD VS. SIMULATED FL PARTITIONS.

When designing non-IID data simulation methods, one important thing is figuring out what kind of generation is analogous to the real-world heterogeneous statistical distribution. The latter leads to evaluating the similarity between our proposed partition methods and a real-life federated scenario. To achieve the last, we perform the following experiment:

1) Start from the Physionet decentralized dataset (a.k.a. original FL)
2) Compute the level of non-IID-ness (Hellinger distance) on the original FL
3) Train the DNN and LSTM models on the original FL
4) Create a centralized dataset (a.k.a. CL) made of the union of the clients of the original FL
5) Train the same classification models on the CL
6) Generate a new decentralized dataset from the CL (a.k.a. Simul. FL (FedArtML) with the same HD distance computed in 2), using the FedArtML library
7) Train the same classification models on the Simul. FL (FedArtML)

As depicted in Table 6, the DNN model trained on the original FL dataset reaches an F1-Score = 0.44. Using FedArtML to re-partition the dataset in six clients with FedArtML, the DNN model obtained an average F1-Score = 0.46. In addition, for the LSTM model, the behavior is similar since the F1-Score of the real-world and simulated datasets is almost identical. The latter demonstrates that FedArtML can generate analogous results to a real-life FL scenario. It is a reliable tool for creating new synthetic FL datasets from a centralized one with different non-IID characteristics that resemble real-world settings.

Table 6. Metrics achieved for the scenarios explained in the experiment using six clients. Results for Simul. FL (FedArtML) using five trials and taking the average. The standard deviation is shown after the $\pm$ symbol.

		DNN		LSTM	
Scenario	**HD**	**F1-Score**	**Accuracy**	**F1-Score**	**Accuracy**
Original FL	0.73	**0.44**	0.39	**0.46**	0.42
CL	NA	0.46	0.46	0.52	0.49
Simul. FL (FedArtML)	0.72 ± 0.03	$\mathbf{0.46 \pm 0.01}$	0.44 ± 0.01	$\mathbf{0.48 \pm 0.01}$	0.41 ± 0.005

Figure 7 demonstrates how the examples of the original FL dataset rolled to the clients generated in the simulated FL dataset. It is evident that with FedArtML, we can get similar datasets (in terms of HD or non-IID-ness), but it does not mean we reproduce exactly the original dataset.

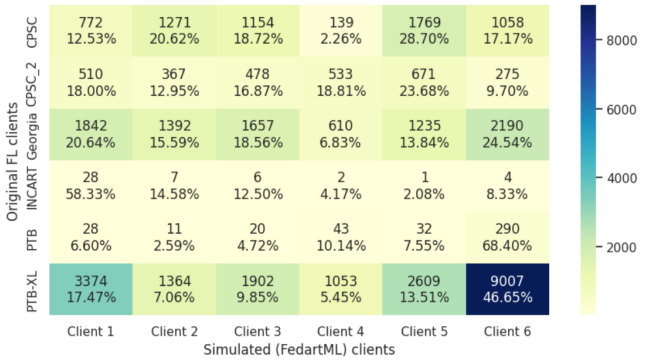

Figure 7. Rolling matrix of original vs simulated FL datasets.

IX. ABLATION STUDY

In this section, we further conduct ablation studies to investigate the impact of JSD and HD under different non-IID scenarios and varying the number of clients participating in the FL training process. Notice that separate analyses are performed for each type of skewness (label, feature, quantity) to understand the differences in each metric's behavior properly.

A. LABEL SKEW

In this analysis, we start by setting the number of clients as $K = \{4, 100, 500\}$ to evaluate the metrics changes under diverse devices participating in the FL process. Additionally, we set as a baseline the cases where partitions are created with FedArtML using the Dirichlet-based method with $\alpha = 1000$, simulating a completely IID case obtaining JSD and HD close to 0. Then, for each number of clients, we simulate cases increasing the non-IID-ness level for the labels across the clients using $\alpha = \{6, 1, 0.3, 0.03\}$, reaching JSD and HD that range from 0 to 1. Finally, we train the DNN model defined in Section VI for all the partitions generated and extract the F1-Score.

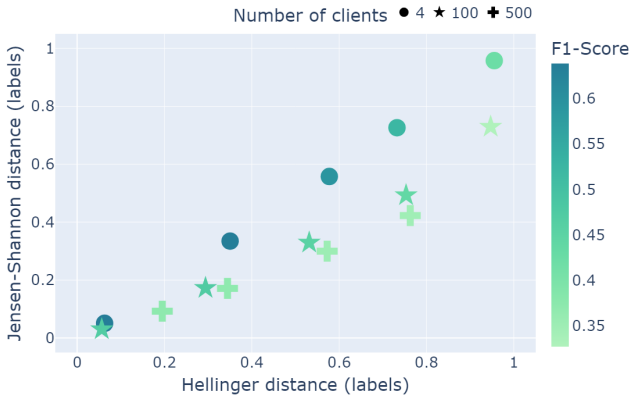

Figure 8. HD, JSD and F1-Score comparison for $K = \{4, 100, 500\}$ clients simulating label skew.

As depicted in Figure 8 JSD and HD are pretty similar when considering four clients. Nevertheless, when increasing the number of devices, the JSD tends to provide a smaller measurement (reaching a maximum of 0.8) than HD, which still reaches levels close to 1. In addition, notice that for the case of 500 clients, it is impossible to partition when the non-IID-ness is pathologically high. The mentioned behaviors permit us to infer that when dealing with a few clients, the users can use indistinctly JSD of HD, but when considering a large number of clients, HD is a more granular choice. Another interesting highlight is that the F1-Score decreases as the number of clients increases, regardless of the distance metric used. This suggests that as data gets partitioned among more clients, the overall performance of the FL model tends to decline.

B. FEATURE SKEW

The setting for feature skew is similar to the one presented in the previous subsection. The only difference is that we employed the Hist-Dirichlet-based method from FedArtML to create the partitions in this case.

As occurred for label skew, in feature skew also, JSD and HD are similar for few clients but more diverse for a higher number of devices as shown in Figure 9. Besides, partitioning the data into more clients leads to a decrease in the F1-Score. But, in this case, increasing the JSD or HD

Figure 9. HD, JSD and F1-Score comparison for $K = \{4, 100, 500\}$ clients simulating feature skew.

distances among the feature distributions does not cause a drop in the model's performance.

C. QUANTITY SKEW

The scenario regarding quantity skew closely resembles those shown in the preceding subsections. However, one distinction lies in utilizing the Min-Size-Dirichlet-based technique from FedArtML to generate the partitions in this particular instance. In addition, we set the number of clients to $K = \{6, 100, 500\}$.

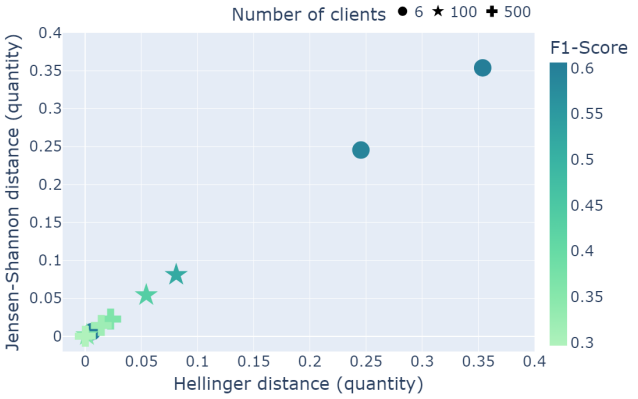

Figure 10. HD, JSD and F1-Score comparison for $K = \{6, 100, 500\}$ clients simulating quantity skew.

Figure 10 illustrates that the maximum value obtained in quantity skew for JSD and HD is close to 0.4, much smaller than the obtained for label and feature skew. The latter is understandable since the data itself might not inherently differ between clients because only the amount of data varies. Another relevant point is that the scenario with six clients is the only one that reaches JSD and HD higher than 0.2. The cases with more devices provided a similarity smaller than 0.1. The latter arises because as the number of clients increases, the data needs to be further divided, resulting in smaller and potentially more similar subsets for each client.

X. LIMITATIONS AND DRAWBACKS

Although FedArtML offers a suite of valuable techniques to simulate federated data controlling label, feature, and quantity skewness using as input a centralized dataset, it has the following limitations and drawbacks that the users need to take into consideration when using the implemented methods:

- **Multi-label data:** FedArtML's label skew simulation methods are currently limited to scenarios where labels are vectors. It cannot handle multi-label data, where a data point can have multiple labels simultaneously (e.g., an image tagged with both "cat" and "dog," a patient with more than one type of arrhythmia at the same time).
- **Categorical feature skew:** The feature skew simulation methods focus on numerical features since they are based on average and addition operations. The current techniques of FedArtML do not apply to categorical features (e.g., occupations with values like "doctor," "engineer," etc.), which are prevalent in real-world data.
- **Task-Specific applicability:** Supervised learning (SL) algorithms are composed of two varieties: regression and classification tasks. Regression-based SL methods try to predict numerical outputs based on input variables. Classification-based SL tasks identify which category a set of data items belongs to. The current version of FedArtML only supports classification tasks.
- **Small alpha and large Clients:** The label, feature, and quantity skewness simulation method using the Dirichlet distribution might not work well when the Dirichlet's alpha parameter is minimal (i.e., smaller than 0.1) and the number of participating clients is high (i.e., higher than 300). The latter specific case can lead to long splitting times or non-converging partitions.
- **Spatiotemporal skew simulation:** The current version of the tool permits partitioning the data focusing on label, feature, and quantity skew. Nevertheless, it does not support partitioning the centralized data using space or temporal variables.

XI. CONCLUSIONS AND FUTURE WORK

The experiments in this study demonstrated the effectiveness of the FedArtML library in producing cross-silo and cross-device decentralized datasets generated from a widely available centralized dataset to facilitate the comparisons between CL and FL research. The results showed that the library successfully generated consistent datasets with different degrees of heterogeneity through a measurable and controlled level of non-IID-ness, even offering similar performance to real-life FL scenarios.

We used the Physionet centralized dataset for our experiments and considered JSD and HD as metrics to measure the differences among the clients' distributions. The experiments conducted using the Dirichlet-based method for label skew and the brand-new Hist-Dirichlet-based and Min-Size-Dirichlet methods showed we could effectively control the degree of skewness of the generated FL datasets. Notice that FedArtML is not limited only to tabular datasets. It can be used to split centralized images, medical datasets, graph datasets, etc., since it is designed to receive arrays as inputs.

All the methods allowed us to increase the distance among partitions (measured by the earlier metrics) by decreasing the parameter α. Furthermore, as the label skewness increases, the performance of the FL degrades in terms of the F1-Score. When there is no label skewness, namely, data are IID, the effectiveness of the CL and FL, in terms of the F1-Score, is almost identical. Regarding features and quantity skewness, high levels of non-IID-ness in the features or quantity distributions do not significantly impact the models' performance.

Overall, the results indicate that the tool is helpful and robust for researchers in the field of FL, enabling them to create compelling and diverse datasets for comparing centralized and FL algorithms, guaranteeing that the results obtained with the simulated FL datasets are comparable to the real-world FL data.

A. IMPLICATIONS

FedArtML offers significant advancements for researchers and developers in the field of FL. Here's how this work can have a lasting impact:

- **Reducing the gap between simulation and reality:** By enabling the creation of controlled, non-IID datasets, FedArtML allows researchers to simulate real-world FL scenarios more effectively. This connection between simulation and reality accelerates the development of robust FL algorithms that can handle the inherent heterogeneity of FL data.
- **Fairer comparisons between CL and FL:** The ability to generate datasets with measurable non-IID levels allows for a more refined comparison between traditional CL and FL approaches. Researchers can directly evaluate the impact of data distribution on model performance, leading to a better understanding of when each approach is best suited.
- **Standardization and benchmarking:** The HD and JSD metrics included in FedArtML provide a way to standardize experiments in FL research. Researchers can utilize consistent methods for data partitioning while quantifying the degree of non-IID-ness, facilitating the comparison and replication of results across different studies. The latter promotes faster innovation and a deeper understanding of FL's potential.
- **Real-world applicability:** The ability to create datasets that mimic real-world conditions empowers developers to build FL models that are generalizable and function effectively across diverse data silos and devices. The latter contributes to practical applications of FL in various sectors, from healthcare and finance to manufacturing and the IoT.

B. FUTURE WORK

FedArtML offers a foundation for further exploration in non-IID data generation based on label, feature, and quantity skew for FL research. Here are some relevant avenues for future development:

- **Simulating spatiotemporal skewness:** Real-world federated data often exhibits spatiotemporal skewness, where data distribution varies geographically and over time. Future work can focus on incorporating approaches to simulate such skewness within FedArtML. This could involve introducing location or time-based parameters during data partitioning, allowing researchers to evaluate the impact of these factors on FL performance.
- **Mixed non-IID types:** FedArtML currently simulates a kind of non-IID data at a time. An interesting extension would be implementing or modifying methods for generating data with mixed non-IID characteristics. This could involve combining label, feature, and quantity skewness within the partitioning process, creating more complex and realistic data distributions for researchers to explore.
- **Regression tasks:** FedArtML's current functionalities primarily focus on classification tasks. Future development could also involve extending the implemented techniques to handle regression tasks. This would allow researchers to investigate the effects of non-IID data on FL models predicting continuous values.
- **Categorical variables and mixed data:** The current feature skew methods primarily address data partition based on numerical features. Future work could involve extending them to handle categorical variables, a common data type, in real-world scenarios. Exploring methods for handling mixed-type datasets containing numerical and categorical features would further enhance FedArtML's capabilities.
- **Multi-label Data:** The current metrics JSD and HD for label skew methods are limited to single-label data. Extending these functionalities to handle multi-label data, where each data point can have multiple labels, would broaden the applicability of FedArtML to a broader range of FL research problems.
- **Alternative metrics for non-IID-ness:** The state of the tool includes JSD and HD metrics to quantify non-IIDness. Nevertheless, it is worth including and analyzing more metrics that can measure the distance of the data distribution of the clients in FL.

XII. ACKNOWLEDGMENTS

Daniel Mauricio Jimenez G. was partially supported by PNRR351 TECHNOPOLE – NEXT GEN EU Roma Technopole – Digital Transition, FP2 – Energy transition and digital transition in urban regeneration and construction and Sapienza Ateneo Research grant "La disintermediazione della Pubblica Amministrazione: il ruolo della tecnologia blockchain e le sue implicazioni nei processi e nei ruoli della PA." Aris Anagnostopoulos was supported by the ERC Advanced Grant 788893 AMDROMA, the EC H2020RIA project "SoBigData++" (871042), the PNRR MUR project PE0000013-FAIR, the PNRR MUR project IR0000013-SoBigData.it, and the MUR PRIN project 2022EKNE5K "Learning in Markets and Society." Ioannis Chatzigiannakis was supported by PE07-SERICS (Security and Rights in the Cyberspace) – European Union Next-Generation-EU-PE0000014 (Piano Nazionale di Ripresa e Resilienza – PNRR). Andrea Vitaletti was supported by PE11 - MICS (Made in Italy – Circular and Sustainable) – European Union Next-Generation-EU (Piano Nazionale di Ripresa e Resilienza – PNRR).

References

[1] Nazish Khalid et al. "Privacy-preserving artificial intelligence in healthcare: Techniques and applications". In: *Computers in Biology and Medicine* (2023), p. 106848.

[2] Adam Bohr and Kaveh Memarzadeh. "The rise of artificial intelligence in healthcare applications". In: *Artificial Intelligence in healthcare*. Elsevier: Elsevier, 2020, pp. 25–60.

[3] C Krittanawong. "The rise of artificial intelligence and the uncertain future for physicians". In: *European journal of internal medicine* 48 (2018), e13–e14.

[4] Som S Biswas. "Role of chat gpt in public health". In: *Annals of biomedical engineering* 51.5 (2023), pp. 868–869.

[5] Gemini Team et al. "Gemini: a family of highly capable multimodal models". In: *arXiv preprint arXiv:2312.11805* (2023).

[6] Ivan Kapelyukh, Vitalis Vosylius, and Edward Johns. "Dall-e-bot: Introducing web-scale diffusion models to robotics". In: *IEEE Robotics and Automation Letters* (2023).

[7] Luciano Floridi and Massimo Chiriatti. "GPT-3: Its nature, scope, limits, and consequences". In: *Minds and Machines* 30 (2020), pp. 681–694.

[8] Mariam Al Zaabi and Saadat M Alhashmi. "Big data security and privacy in healthcare: A systematic review and future research directions". In: *Information Development* (2024), p. 02666669241247781.

[9] Metty Paul et al. "Digitization of healthcare sector: A study on privacy and security concerns". In: *ICT Express* 9.4 (2023), pp. 571–588.

[10] Hari Prasad Josyula et al. "A Review on Security and Privacy Considerations in Programmable Payments". In: *International Journal of Intelligent Systems and Applications in Engineering* 12.9s (2024), pp. 256–263.

[11] Adedoyin Tolulope Oyewole et al. "Data privacy laws and their impact on financial technology companies: a review". In: *Computer Science & IT Research Journal* 5.3 (2024), pp. 628–650.

[12] Protection Regulation. "General data protection regulation". In: *Intouch* 25 (2018), pp. 1–5.

[13] Alberto S Ortega-Calvo et al. "Aimdp: An artificial intelligence modern data platform. use case for Spanish national health service data silo". In: *Future Generation Computer Systems* 143 (2023), pp. 248–264.

[14] Rizwana Naz Asif et al. "Detecting Electrocardiogram Arrhythmia Empowered With Weighted Federated Learning". In: *IEEE Access* (2023).

[15] Bala Siva Prakash Thummisetti and Haritha Atluri. "Advancing Healthcare Informatics for Empowering Privacy and Security through Federated Learning Paradigms". In: *International Journal of Sustainable Development in Computing Science* 1.1 (2024), pp. 1–16.

[16] Jie Wen et al. "A survey on federated learning: challenges and applications". In: *International Journal of Machine Learning and Cybernetics* 14.2 (2023), pp. 513–535.

[17] Omair Rashed Abdulwareth Almanifi et al. "Communication and computation efficiency in federated learning: A survey". In: *Internet of Things* 22 (2023), p. 100742.

[18] Xinchen Lyu et al. "Secure and efficient federated learning with provable performance guarantees via stochastic quantization". In: *IEEE Transactions on Information Forensics and Security* (2024).

[19] Jie Ling, Junchang Zheng, and Jiahui Chen. "Efficient federated learning privacy preservation method with heterogeneous differential privacy". In: *Computers & Security* 139 (2024), p. 103715.

[20] Truc Nguyen and My T Thai. "Preserving privacy and security in federated learning". In: *IEEE/ACM Transactions on Networking* (2023).

[21] Mang Ye et al. "Heterogeneous federated learning: State-of-the-art and research challenges". In: *ACM Computing Surveys* 56.3 (2023), pp. 1–44.

[22] Ahmed M Abdelmoniem et al. "A comprehensive empirical study of heterogeneity in federated learning". In: *IEEE Internet of Things Journal* (2023).

[23] Tao Lin et al. "Ensemble distillation for robust model fusion in federated learning". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 2351–2363.

[24] Qinbin Li et al. "Federated learning on non-iid data silos: An experimental study". In: *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE. IEEE: IEEE, 2022, pp. 965–978.

[25] Kevin Hsieh et al. "The non-iid data quagmire of decentralized machine learning". In: *International Conference on Machine Learning*. PMLR. unknown: PMLR, 2020, pp. 4387–4398.

[26] Qinbin Li et al. "A survey on federated learning systems: Vision, hype and reality for data privacy and protection". In: *IEEE Transactions on Knowledge and Data Engineering* (2021).

[27] Zijian Li et al. "Feature matching data synthesis for non-iid federated learning". In: *IEEE Transactions on Mobile Computing* (2024).

[28] Zhongyuan Zhao et al. "Ensemble federated learning with non-IID data in wireless networks". In: *IEEE Transactions on Wireless Communications* (2023).

[29] Yanmeng Wang, Qingjiang Shi, and Tsung-Hui Chang. "Why batch normalization damage federated learning on non-iid data?" In: *IEEE Transactions on Neural Networks and Learning Systems* (2023).

[30] Hangyu Zhu et al. "Federated learning on non-IID data: A survey". In: *Neurocomputing* 465 (2021), pp. 371–390.

[31] Xiaodong Ma et al. "A state-of-the-art survey on solving non-IID data in Federated Learning". In: *Future Generation Computer Systems* 135 (2022), pp. 244–258.

[32] Hui Chen et al. "Bayesian Personalized Federated Learning with Shared and Personalized Uncertainty Representations". In: *arXiv preprint arXiv:2309.15499* (2023).

[33] Xiuwen Fang, Mang Ye, and Xiyuan Yang. "Robust heterogeneous federated learning under data corruption". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 5020–5030.

[34] Xiaojin Zhang et al. "Trading off privacy, utility and efficiency in federated learning". In: *arXiv preprint arXiv:2209.00230* 0.0 (2022), p. 38.

[35] Annie Abay et al. "Mitigating bias in federated learning". In: *arXiv preprint arXiv:2012.02447* 0.0 (2020), p. 10.

[36] Dun Zeng et al. "Fedlab: A flexible federated learning framework". In: *Journal of Machine Learning Research* 24.100 (2023), pp. 1–7.

[37] Fan Lai et al. "Fedscale: Benchmarking model and system performance of federated learning at scale". In: *International conference on machine learning*. PMLR. 2022, pp. 11814–11827.

[38] Jean Ogier du Terrail et al. "Flamby: Datasets and benchmarks for cross-silo federated learning in realistic healthcare settings". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 5315–5334.

[39] Xiaodong Ma et al. "A state-of-the-art survey on solving non-IID data in Federated Learning". In: *Future Generation Computer Systems* 135 (2022), pp. 244–258.

[40] Hangyu Zhu et al. "Federated learning on non-IID data: A survey". In: *Neurocomputing* 465 (2021), pp. 371–390.

[41] Richard Connor et al. "Evaluation of Jensen-Shannon distance over sparse data". In: *Similarity Search and Applications: 6th International Conference, SISAP 2013, A Coruña, Spain, October 2-4, 2013, Pro-*

ceedings 6. Springer. Springer Nature Switzerland: Springer, 2013, pp. 163–168.

[42] Roma Goussakov. "Hellinger Distance-based Similarity Measures for Recommender Systems". PhD thesis. Umea University, 2020.

[43] Thomas Minka. *Estimating a Dirichlet distribution*. 2000.

[44] Shenda Hong et al. "Practical lessons on 12-lead ECG classification: Meta-analysis of methods from PhysioNet/computing in cardiology challenge 2020". In: *Frontiers in Physiology* 12 (2022).

[45] Erick Andres Perez Alday et al. "Classification of 12-lead ECGs: the PhysioNet/Computing in Cardiology Challenge 2020". In: *medRxiv* 0.0 (2020). DOI: 10.1101/2020.08.11.20172601. eprint: https://www.medrxiv.org/content/early/2020/08/14/2020.08.11.20172601.full.pdf. URL: https://www.medrxiv.org/content/early/2020/08/14/2020.08.11.20172601.

[46] Maryam Saeed et al. "ECG Classification with Event-Driven Sampling". In: *IEEE Access* (2024).

[47] Fatma Murat et al. "Application of deep learning techniques for heartbeats detection using ECG signals-analysis and review". In: *Computers in Biology and Medicine* 120 (2020), p. 103726. ISSN: 0010-4825. DOI: https://doi.org/10.1016/j.compbiomed.2020.103726. URL: https://www.sciencedirect.com/science/article/pii/S0010482520301104.

[48] Mariya R Kiladze et al. "Multimodal Neural Network for Recognition of Cardiac Arrhythmias Based on 12-Load Electrocardiogram Signals". In: *IEEE Access* 11 (2023), pp. 133744–133754.

[49] Brendan McMahan et al. "Communication-efficient learning of deep networks from decentralized data". In: *Artificial intelligence and statistics*. PMLR. PMLR: PMLR, 2017, pp. 1273–1282.

DANIEL MAURICIO JIMENEZ GUTIERREZ is a Ph.D. student in Data Science with current research in Federated Machine Learning leveraged on data-driven problems at the Sapienza University of Rome. He holds an MSc. in Data Science magna cum laude (2022) and received a BSc. in Statistics at the National University of Colombia (2013). He was accepted to the Student's Honours program at Sapienza for outstanding performance in the master's course. In addition, he belonged to the top 1% ranking of the Wiraki start-up, based on the most talented and skilled students. He has applied Machine Learning and AI techniques for almost ten years, mainly oriented to Credit Risk and banking analytical solutions. He was the Bureau Models' Manager at Experian-Colombia, one of the biggest and most significant data bureaus worldwide. He has also worked as Lead Data Scientist in ALTO group, a multinational technology company focused on preventing and reducing capital losses.

ARIS ANAGNOSTOPOULOS is an Professor in the Department of Computer, Control, and Management Engineering, Sapienza University of Rome, Italy. Before Sapienza, he was at Yahoo Research, Santa Clara, CA, USA. He received the Ph.D. degree in computer science from Brown University, Providence, RI, USA.

IOANNIS CHATZIGIANNAKIS holds a Ph.D. from the University of Patras (2003) in the area of ad-hoc wireless mobile networks and a BEng from the University of Kent (1997) in Computer Systems Engineering. He is an Associate Professor at the Sapienza University of Rome in the Computer, Control, and Management Engineering Department. He has co-authored over 150 scientific publications in areas related to dynamic distributed computing, the Internet of Things, algorithm engineering, and software systems. He has been a project manager and site leader for numerous research & development projects funded by the EU in the context of H2020, FP7, FP6, and EDA. He has participated in the research & development teams of industrial projects. He actively participates in many open-source projects and regularly participates in open-source international events. He has started several technology-based start-ups related to the Internet of Things. He has served as the Secretary of the European Association for Theoretical Computer Science (EATCS).

ANDREA VITALETTI holds a Ph.D. in Computer Engineering from SAPIENZA University of Rome (2002). He visited renowned international research centers such as ETHZ (CH) and AT&T Research Labs (USA). He is an Associate professor in networking and algorithmic topics at (DIAG) Dipartimento di Ingegneria informatica automatica e gestionale Antonio Ruberti University of Rome "La Sapienza". He has (co-)authored more than 80 papers in journals and international conferences, mainly in algorithms and protocols for wireless and sensor networks and IoT. His current research interests concern the design and analysis of efficient IoT solutions and blockchain technologies. He has been involved in several EU projects as a researcher and PI and has founded 3 start-ups.