



INFORMS Journal on Data Science

Publication details, including instructions for authors and subscription information:
<http://pubsonline.informs.org>

GIFAIR-FL: A Framework for Group and Individual Fairness in Federated Learning

Xubo Yue, Maher Nouiehed, Raed Al Kontar

To cite this article:

Xubo Yue, Maher Nouiehed, Raed Al Kontar (2023) GIFAIR-FL: A Framework for Group and Individual Fairness in Federated Learning. INFORMS Journal on Data Science 2(1):10-23. <https://doi.org/10.1287/ijds.2022.0022>

Full terms and conditions of use: <https://pubsonline.informs.org/Publications/Librarians-Portal/PubsOnLine-Terms-and-Conditions>

This article may be used only for the purposes of research, teaching, and/or private study. Commercial use or systematic downloading (by robots or other automatic processes) is prohibited without explicit Publisher approval, unless otherwise noted. For more information, contact permissions@informs.org.

The Publisher does not warrant or guarantee the article's accuracy, completeness, merchantability, fitness for a particular purpose, or non-infringement. Descriptions of, or references to, products or publications, or inclusion of an advertisement in this article, neither constitutes nor implies a guarantee, endorsement, or support of claims made of that product, publication, or service.

Copyright © 2022, INFORMS

Please scroll down for article—it is on subsequent pages



With 12,500 members from nearly 90 countries, INFORMS is the largest international association of operations research (O.R.) and analytics professionals and students. INFORMS provides unique networking and learning opportunities for individual professionals, and organizations of all types and sizes, to better understand and use O.R. and analytics tools and methods to transform strategic visions and achieve better outcomes.




For more information on INFORMS, its publications, membership, or meetings visit <http://www.informs.org>

GIFAIR-FL: A Framework for Group and Individual Fairness in Federated Learning

Xubo Yue,^a Maher Nouiehed,^b Raed Al Kontar^{a,*}

^aIndustrial and Operations Engineering, University of Michigan, Ann Arbor, Michigan 48109; ^bIndustrial Engineering and Management, American University of Beirut, Beirut 1107-2020, Lebanon

*Corresponding author

Contact: maxyxb@umich.edu,  <https://orcid.org/0000-0001-9929-8895> (XY); mn102@aub.edu.lb,  <https://orcid.org/0000-0001-8089-7011> (MN); alkontar@umich.edu,  <https://orcid.org/0000-0002-4546-324X> (RAK)

Received: February 28, 2022

Revised: August 16, 2022

Accepted: September 13, 2022

Published Online in Articles in Advance:
October 27, 2022

<https://doi.org/10.1287/ijds.2022.0022>

Copyright: © 2022 INFORMS

Abstract. In this paper, we propose GIFAIR-FL, a framework that imposes group and individual fairness (GIFAIR) to federated learning (FL) settings. By adding a regularization term, our algorithm penalizes the spread in the loss of client groups to drive the optimizer to fair solutions. Our framework GIFAIR-FL can accommodate both global and personalized settings. Theoretically, we show convergence in nonconvex and strongly convex settings. Our convergence guarantees hold for both independent and identically distributed (i.i.d.) and non-i.i.d. data. To demonstrate the empirical performance of our algorithm, we apply our method to image classification and text prediction tasks. Compared with existing algorithms, our method shows improved fairness results while retaining superior or similar prediction accuracy.

History: Kwok-Leung Tsui served as the senior editor for this article.

Funding: This work was supported by NSF CAREER [Grant 2144147].

Data Ethics & Reproducibility Note: The code capsule is available on Code Ocean at <https://codeocean.com/capsule/2590027/tree/v1> and in the e-Companion to this article (available at <https://doi.org/10.1287/ijds.2022.0022>).

Keywords: federated data analytics • fairness • global model • personalized model • convergence

1. Introduction

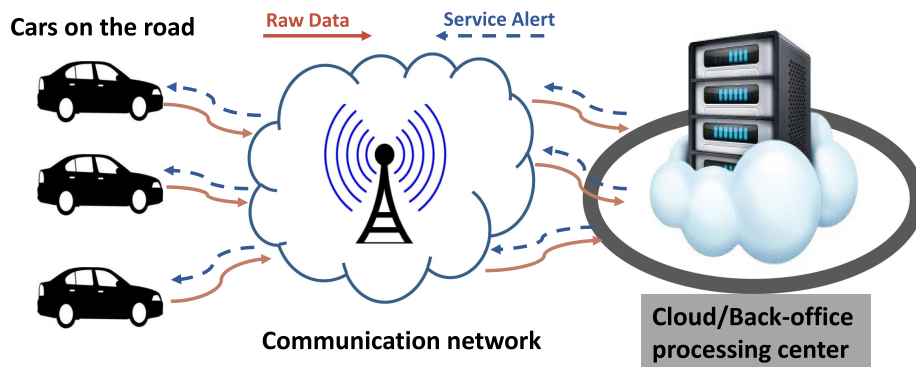
A critical change is happening in today's internet of things (IoT). The computational power of edge devices is steadily increasing. Artificial intelligence chips are rapidly infiltrating the market; today's smartphones have computing power comparable to that of everyday-use laptops (Samsung 2019); Tesla just boasted that its autopilot system has the computing power of more than 3,000 MacBook Pros (CleanTechnica 2021); and small local computers such as Raspberry Pis have become commonplace in many applications, especially manufacturing (Al-Ali et al. 2018). This opens a new paradigm for data analytics in the IoT, one that exploits local computing power to process more of the user's data where it is created. This future of the IoT has been recently termed "the internet of federated things" (Kontar et al. 2021), where the term *federated* refers to some autonomy for IoT devices and is inspired by the explosive recent interest in federated data science.

To give a microcosm of current the IoT and its future, consider the IoT teleservice system shown in Figure 1. Vehicles enrolled in this teleservice system often have their data in the form of condition monitoring signals uploaded to the cloud at regular intervals. The cloud acts as a data processing center that analyzes data for continuous improvement and to keep drivers informed

about the health of their vehicles. IoT companies and services, such as Ford's SYNC and General Motors' OnStar services, have long adopted this centralized approach to the IoT. However, this state where data are amassed on the cloud yields significant challenges. Uploading large amounts of data to the cloud incurs high communication and storage costs, demands large internet bandwidth (Jiang et al. 2020a, Yang et al. 2020), and leads to latency in deployment as well as reliability risks due to unreliable connections (Zhang et al. 2020c). Furthermore, such systems do not foster trust or privacy, as users need to share their raw data, which is often sensitive or confidential (Li et al. 2020a).

With the increasing computational power of edge devices, the discussed challenges can be circumvented by moving part of the model learning to the edge. More specifically, rather than processing the data at the cloud, each device performs small local computations and shares only the minimum information needed to allow devices to borrow strength from each other and collaboratively extract knowledge to build smart analytics. In turn, such an approach (i) improves privacy, as raw data are never shared; (ii) reduces cost and storage needs, as less information is transmitted; (iii) enables learning parallelization; and (iv) reduces latency in decisions, as many decisions can now be achieved locally. Hereafter, we will use *edge device* and *client* interchangeably, and the

Figure 1. (Color online) Example of the Traditional IoT-Enabled System



cloud or data processing center is referred to as the *central server*.

This idea of exploiting the computational power of edge devices by locally training models without recourse to data sharing gave rise to federated learning (FL). In particular, FL is a data analytics approach that allows distributed model learning without access to private data. Although the main concept of FL dates back to a while ago, it was brought to the forefront of data science in 2017 by a team at Google which proposed federated averaging (FedAvg; McMahan et al. 2017). In FedAvg, a central server distributes the model architecture (e.g., neural network, linear model) and a global model parameter (e.g., model weights) to selected devices. Devices run local computations to update model parameters and send updated parameters to the server. The server then takes a weighted average of the resulting local parameters to update the global model. This whole process is termed as one communication round, and the process is iterated over multiple rounds until an exit condition is met. Figure 2 provides one illustrative example of FedAvg. Since then, FL has seen immense success in various fields such as text prediction (Hard et al. 2018, Ramaswamy et al. 2019), Bayesian optimization (Dai et al. 2020, Khodak et al. 2021), multifidelity modeling (Yue and Kontar 2021), environment monitoring (Hu et al. 2018, Jiang et al. 2020b), and healthcare (Li et al. 2020a, Xu et al. 2021).

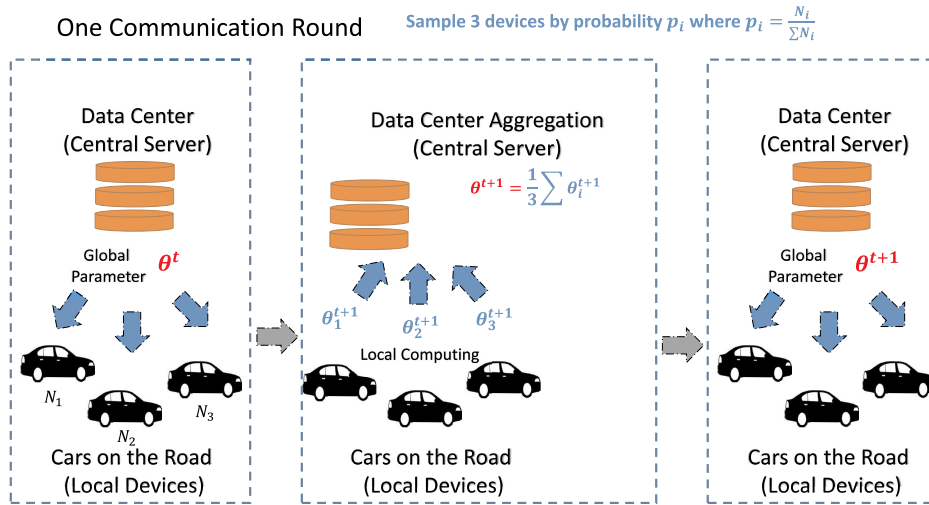
Over the last few years, literature has been proposed to improve the performance of FL algorithms, be it speeding up FL algorithms to enable faster convergence (Karimireddy et al. 2020, Nguyen et al. 2020, Yuan and Ma 2020), tackling heterogeneous data both in size and distribution (Li et al. 2018, Zhao et al. 2018, Ghosh et al. 2019, Li and Wang 2019, Sattler et al. 2019), improving the parameter aggregation strategies at the central server (Pillutla et al. 2019, Wang et al. 2020b), designing personalized FL algorithms (Jiang et al. 2019, Fallah et al. 2020, Mansour et al. 2020), protecting federated systems from adversarial attacks (Bhagoji et al. 2019, Wang et al. 2020a), or promoting fairness (Li et al. 2019a, Mohri et al. 2019, Du et al. 2020, Hu et al. 2020, Huang et al. 2020, Zhang

et al. 2020a). Please refer to Kontar et al. (2021) for a detailed literature review. Among those advances, fairness is a critical yet underinvestigated area.

In the training phase of FL algorithms, devices with few data, limited bandwidth/memory, or unreliable connection may not be favored by conventional FL algorithms. For instance, as shown in Figure 2, FedAvg samples devices using the weight coefficient p_k proportional to the sample size on the device k . Scant data on device k will render p_k insignificant and this device less favorable by the resulting global model. As a result, such device(s) can potentially incur higher error rate(s). This vicious cycle often relinquishes the opportunity of these devices to significantly contribute in the training process. Indeed, many recent papers have shown the large variety in model performance across devices under FL (Smith et al. 2017, Hard et al. 2018, Jiang et al. 2019, Kairouz et al. 2019, Wang et al. 2019), with some clients showing extremely bad model performance. Besides this aforementioned notion of individual fairness, group fairness also deserves attention in FL. As FL penetrates practical applications, it is important to achieve fair performance across groups of clients characterized by their gender, ethnicity, socioeconomic status, geographic location, etc. Despite the importance of this notion of group fairness, unfortunately, *no work exists along this line in FL*.

1.1. Contribution

We propose a framework, GIFAIR-FL, that aims for fairness in FL. GIFAIR-FL resorts to regularization techniques by penalizing the spread in the loss of clients/groups to drive the optimizer to fair solutions. We show that our regularized formulation can be viewed as a dynamic client reweighting technique that adaptively gives higher weights to low-performing individuals or groups. Our proposed method adapts the client weights at every communication round accordingly. One key feature of GIFAIR-FL is that it can handle both group-level and individual-level fairness. Also, GIFAIR-FL can be naturally tailored to either a global FL algorithm or a personalized FL algorithm. We then

Figure 2. (Color online) Illustrative Example of FL with FedAvg

prove that, under reasonable conditions, our algorithm converges to an optimal solution for strongly convex objective functions and to a stationary solution for non-convex functions under non-independent and identically distributed (i.i.d.) settings. Through empirical results on image classification and text prediction data sets, we demonstrate that GIFAIR-FL can promote fairness while achieving superior or similar prediction accuracy relative to recent state-of-the-art fair FL algorithms. Besides that, GIFAIR-FL can be easily plugged into other FL algorithms for different purposes.

1.2. Organization

The rest of this paper is organized as follows. In Section 2, we introduce important notations/definitions and briefly review FL. Related work is highlighted in Section 2.4. In Section 3, we present GIFAIR-FL-Global, which is a global modeling approach for fairness in FL. We then briefly discuss the limitation of GIFAIR-FL-Global and introduce GIFAIR-FL-Per, which is a personalized alternative for fairness, in Section 4, and we provide convergence guarantees for both methods. Experiments on image classification and text prediction tasks are then presented in Section 5. Finally, Section 6 concludes this paper with a brief discussion.

2. Background

We start by introducing needed background and notation for model development. Then we provide a brief overview of current literature.

2.1. Notation

Suppose there are $K \geq 2$ local devices and each device has N_k data points. Denote by $D_k = ((x_{k,1}, y_{k,1}), (x_{k,2}, y_{k,2}), \dots, (x_{k,N_k}, y_{k,N_k}))$ the data stored at device k ,

where $x \in \mathcal{X}$ is the input, \mathcal{X} is the input space, $y \in \mathcal{Y}$ is the output/label, and \mathcal{Y} is the output space. Denote by $\Delta_{\mathcal{Y}}$ the simplex over \mathcal{Y} , $h: \mathcal{X} \mapsto \Delta_{\mathcal{Y}}$ the hypothesis, and \mathcal{H} a family of hypotheses h . Let ℓ be a loss function defined over $\Delta_{\mathcal{Y}} \times \mathcal{Y}$. Without loss of generality, assume $\ell \geq 0$. The loss of h is therefore given by $\ell(h(x), y)$. Let $\theta \in \Theta$ be a vector of parameters defining a hypothesis h , and Θ is a parameter space. For instance, θ can be the model parameters of a deep neural network. In the following section, we use h_{θ} to represent the hypothesis.

2.2. Brief Background on FL with FedAvg

In FL, clients collaborate to learn a model that yields better performance relative to each client learning in isolation. This model is called the global model, where the global objective function is to minimize the average loss over all clients:

$$\min_{\theta} F(\theta) := \sum_{k=1}^K p_k F_k(\theta),$$

where $p_k = \frac{N_k}{\sum_k N_k}$, $F_k(\theta) = \mathbb{E}_{(x_{k,i}, y_{k,i}) \sim D_k} [\ell(h_{\theta}(x_{k,i}), y_{k,i})] \approx \frac{1}{N_k} \sum_{j=1}^{N_k} [\ell(h_{\theta}(x_{k,j}), y_{k,j})]$, and D_k indicates the data distribution of the k th device's data observations $(x_{k,i}, y_{k,i})$. During training, all devices collaboratively learn global model parameters θ to minimize $F(\theta)$. The most commonly used method to learn the global objective is FedAvg (McMahan et al. 2017). Details of FedAvg are highlighted in Algorithm 1, as our work will build upon it for fairness. As shown in Algorithm 1, FedAvg aims to learn a global parameter θ by iteratively averaging local updates θ_k learned by performing E steps of stochastic gradient descent (SGD) on each client's local objective F_k .

Algorithm 1 (FedAvg (McMahan et al. 2017))

Data: number of communication rounds C , number of local updates E , SGD learning rate schedule $\{\eta^{(t)}\}_t$, initial model parameter θ
for $c = 0 : (C-1)$ **do**
 | Select some clients by sampling probability p_k and denote by \mathcal{S}_c the set of selected clients;
 | Server broadcasts θ ;
 | **for all selected devices do**
 | | $\theta_k^{(cE)} = \theta$;
 | | **for** $t = cE : ((c+1)E-1)$ **do**
 | | | Randomly sample a subset of data and denote it by $\zeta_k^{(t)}$;
 | | | Local training $\theta_k^{(t+1)} = \theta_k^{(t)} - \eta^{(t)} g_k(\theta_k^{(t)}; \zeta_k^{(t)})$;
 | | **end**
 | **end**
 | Aggregation $\bar{\theta}_c = \frac{1}{|\mathcal{S}_c|} \sum_{k \in \mathcal{S}_c} \theta_k^{((c+1)E)}$, set $\theta = \bar{\theta}_c$;
end
 Return $\bar{\theta}_C$.

In Algorithm 1, ζ_k denotes the set of indices corresponding to a subset of training data on device k , and $g_k(\cdot; \zeta_k)$ denotes the stochastic gradient of $F_k(\cdot)$ evaluated on the subset of data indexed by ζ_k . Also, $|\mathcal{S}_c|$ denotes the cardinality of \mathcal{S}_c . One should note that it is also common for the central server to sample clients uniformly and then take a weighted average using p_k (Li et al. 2019c). Whichever method used, the resulting model may not be fair, as small a p_k implies a lower weight for client k .

2.3. Defining Fairness in FL

Suppose there are $d \in [2, K]$ groups and each client can be assigned to one of those groups $s \in [d] := \{1, \dots, d\}$. Note that clients from different groups are typically not i.i.d. Denote by $k^i, k \in [K], i \in [d]$, the index of k th local device in group i . Throughout this paper, we drop the superscript i unless we want to emphasize i explicitly. Group fairness can be defined as follows.

Definition 1. Denote by $\{a_1^i\}_{1 \leq i \leq d}$ and $\{a_2^i\}_{1 \leq i \leq d}$ the sets of performance measures (e.g., testing accuracy) of trained models 1 and 2, respectively. We say model 1 is more fair than model 2 if $Var(\{a_1^i\}_{1 \leq i \leq d}) < Var(\{a_2^i\}_{1 \leq i \leq d})$, where Var is variance.

Definition 1 is straightforward: a model is fair if it yields small discrepancies among testing accuracies of different groups. It can be seen that when $d = K$, Definition 1 is equivalent to individual fairness (Li et al. 2019a). Definition 1 is widely adopted in FL literature (Li et al. 2019a, 2020b, 2021; Mohri et al. 2019). This notion of fairness might be different from traditional definitions such as demographic disparity (Feldman et al. 2015), equal opportunity, and equalized odds (Hardt et al. 2016) in centralized systems. The reason is that those definitions cannot be extended to FL, as there is no clear notion of an outcome that is “good”

for a device (Kairouz et al. 2019). Instead, fairness in FL can be reframed as equal access to effective models (e.g., the accuracy parity (Zafar et al. 2017) or the representation disparity (Li et al. 2019a)). Specifically, the goal is to train models that incur uniformly good performance across all devices (Kairouz et al. 2019).

2.4. Literature Overview

Now we briefly review existing state-of-the-art fair and personalized FL algorithms.

2.4.1. Fair FL. Mohri et al. (2019) proposed a minimax optimization framework named agnostic FL (AFL). AFL optimizes the worst weighted combination of local devices and is demonstrated to be robust to unseen testing data. Du et al. (2020) further refined the notation of AFL and proposed the *AgnosticFair* algorithm. Specifically, they linearly parametrized weight parameters by kernel functions and showed that AFL can be viewed as a special case of *AgnosticFair*. Upon that, Hu et al. (2020) combined minimax optimization with gradient normalization techniques to produce a fair algorithm, *FedMGDA+*. Motivated by fair resource allocation problems, Li et al. (2019a) proposed q -Fair FL (q -FFL), which reweights loss functions such that devices with poor performance will be given relatively higher weights. The q -FFL objective is proved to encourage individual fairness in FL. However, this algorithm requires accurate estimation of a local Lipschitz constant L . Later, Li et al. (2020b) developed a tilted empirical risk minimization algorithm, *TERM*, to handle outliers and class imbalance in statistical estimation procedures. *TERM* has been shown to be superior to q -FFL in many FL applications. Along this line, Huang et al. (2020) proposed using training accuracy and frequency to adjust weights of devices to promote fairness. Zhang et al. (2020a) developed an algorithm to minimize the discrimination index of the global model to encourage fairness. Here we note that recent work has studied collaborative fairness in FL (Lyu et al. 2020, Xu and Lyu 2020, Zhang et al. 2020b). The goal of this literature, which is perpendicular to our purpose, is to provide more rewards to high-contributing participants while penalizing free riders.

2.4.2. Personalized FL. One alternative to global modeling is personalized FL, which allows each client to retain their own individualized parameters $\{\theta_k\}_{k=1}^K$. For instance, in Algorithm 1 and after training is done, each device i can use $\theta = \bar{\theta}_C$ as the initial weight and run additional SGD steps to obtain a personalized solution θ_i . Though personalization techniques do not directly target fairness, recent papers have shown that personalized FL algorithms may improve fairness. Arivazhagan et al. (2019) and Liang et al. (2020) used different layers of a network to represent global and personalized

solutions. Specifically, they fit personalized layers to each local device such that each device will return a task-dependent solution based on its own local data. Wang et al. (2019), Yu et al. (2020), Dinh et al. (2020), and Li et al. (2021) resorted to fine-tuning techniques to learn personalized models. Notably, Li et al. (2021) developed a multitask personalized FL algorithm, *Ditto*. After optimizing a global objective function, *Ditto* allows local devices to run more steps of SGD, subject to some constraints, to minimize their own losses. Li et al. (2021) showed that *Ditto* can significantly improve testing accuracy among local devices and encourage fairness.

2.4.3. Features of GIFAIR-FL. We here give a quick comparison that highlights the features of our proposed algorithm. The detailed formulation of GIFAIR-FL will be presented in the following section. GIFAIR-FL resorts to regularization to penalize the spread in the loss of client groups. Interestingly, GIFAIR-FL can be seen as a dynamic reweighting strategy based on the statistical ordering of client/group losses at each communication round. As such, our approach aligns with FL literature that uses reweighting client schemes, yet existing work faces some limitations. Specifically, AFL and its variants (Li et al. 2019a, Mohri et al. 2019, Du et al. 2020, Hu et al. 2020) exploit minimax formulations that optimize the worst-case distribution of weights among clients to promote fairness. Such approaches lead to overly pessimistic solutions, as they focus only on the device with the largest loss. As will be shown in our case studies, GIFAIR-FL significantly outperforms such approaches. Adding to this key advantage, GIFAIR-FL enjoys convergence guarantees even for non-i.i.d. data and is amenable to both global and personalized modeling.

3. GIFAIR-FL-Global: A Global Model for Fairness

We start by detailing our proposed global modeling approach, GIFAIR-FL-Global. In this approach, all local devices collaborate to learn one global model parameter θ . Our fair FL formulation aims at imposing group fairness while minimizing the training error. More specifically, our goal is to minimize the discrepancies in the average group losses while achieving a low training error. By penalizing the spread in the loss among client groups, we propose a regularization framework for computing optimal parameters θ that balances learning accuracy and fairness. This translates to solving the following optimization problem:

$$\min_{\theta} H(\theta) \triangleq \sum_{k=1}^K p_k F_k(\theta) + \lambda \sum_{1 \leq i < j \leq d} |L_i(\theta) - L_j(\theta)|, \quad (1)$$

where λ is a positive scalar that balances fairness and goodness-of-fit, and

$$L_i(\theta) \triangleq \frac{1}{|\mathcal{A}_i|} \sum_{k \in \mathcal{A}_i} F_k(\theta)$$

is the average loss for client group i , \mathcal{A}_i is the set of indices of devices that belong to group i , and $|\mathcal{A}|$ is the cardinality of the set \mathcal{A} .

Remark 1. Objective (1) aims at ensuring fairness by reducing client loss spread when losses are evaluated at a single global parameter θ . This achieves fairness from the server perspective. Specifically, the goal is to find a single solution that yields small discrepancies among $\{L_i(\theta)\}_{i=1}^d$.

In typical FL settings, the global objective is given as

$$H(\theta) = \sum_{k=1}^K p_k H_k(\theta),$$

where each client uses local data to optimize a surrogate of the global objective function. For instance, FedAvg simply uses the local objective function $H_k(\theta) = F_k(\theta)$ for a given client k . Interestingly, our global objective in (1) can also be written as $H(\theta) = \sum_{k=1}^K p_k H_k(\theta)$, as shown in Lemma 1.

Lemma 1. Let $s_k \in [d]$ denote the group index of device k . For any given θ , the global objective function $H(\theta)$ defined in (1) can be expressed as

$$H(\theta) = \sum_{k=1}^K p_k \left(1 + \frac{\lambda}{p_k |\mathcal{A}_{s_k}|} r_k(\theta) \right) F_k(\theta), \quad (2)$$

where

$$r_k(\theta) = \sum_{1 \leq j \neq s_k \leq d} \text{sign}(L_{s_k}(\theta) - L_j(\theta)).$$

Consequently,

$$H(\theta) = \sum_{k=1}^K p_k H_k(\theta)$$

such that the local client objective is

$$H_k(\theta) \triangleq \left(1 + \frac{\lambda}{p_k |\mathcal{A}_{s_k}|} r_k(\theta) \right) F_k(\theta). \quad (3)$$

In Lemma 1, $r_k(\theta) \in \{-d+1, -d+3, \dots, d-3, d-1\}$ is a scalar directly related to the statistical ordering of L_{s_k} among client group losses. To illustrate that in a simple example, suppose that at a given θ , we have $L_1(\theta) \geq L_2(\theta) \geq \dots \geq L_d(\theta)$. Then,

$$r_k(\theta) = \begin{cases} d-1 & \text{if } s_k = 1, \\ d-3 & \text{if } s_k = 2, \\ \vdots & \\ -d+1 & \text{if } s_k = d. \end{cases}$$

According to Lemma 1, one can view our global objective as a parameter-based weighted sum of the client loss functions. Particularly, rather than using uniform weighting for clients, our assigned weights are functions of the parameter θ . More specifically, for a given parameter θ , our objective yields higher weights for groups with higher average group loss, hence, imposing group fairness. To illustrate this idea, we provide a simple concrete example.

Example 1. Without loss of generality and for a given θ , consider four different groups each having 10 clients with $L_1(\theta) > L_2(\theta) > L_3(\theta) > L_4(\theta)$. Then our global objective function $H(\theta)$ in (2) can be expressed as

$$\sum_{k=1}^{40} p_k F_k(\theta) + \lambda(|L_1(\theta) - L_2(\theta)| + |L_1(\theta) - L_3(\theta)| + |L_1(\theta) - L_4(\theta)| + |L_2(\theta) - L_3(\theta)| + |L_2(\theta) - L_4(\theta)| + |L_3(\theta) - L_4(\theta)|),$$

which is equivalent to

$$\sum_{k \in \mathcal{A}_1} p_k \left(1 + \frac{3\lambda}{10p_k}\right) F_k(\theta) + \sum_{k \in \mathcal{A}_2} p_k \left(1 + \frac{\lambda}{10p_k}\right) F_k(\theta) + \sum_{k \in \mathcal{A}_3} p_k \left(1 - \frac{\lambda}{10p_k}\right) F_k(\theta) + \sum_{k \in \mathcal{A}_4} p_k \left(1 - \frac{3\lambda}{10p_k}\right) F_k(\theta).$$

The objective clearly demonstrates a higher weight applied to clients that belong to a group with a higher average loss.

According to (3), the optimization problem solved by every selected client is a weighted version of the local objective in FedAvg. The objective imposes a higher weight for clients that belong to groups with higher average losses. These weights will be dynamically updated at every communication round. To assure positive weights for clients, we require the following bounds on λ :

$$0 \leq \lambda < \lambda_{max} \triangleq \min_k \left\{ \frac{p_k |A_{S_k}|}{d-1} \right\}.$$

When $\lambda = 0$, our approach is exactly FedAvg. Moreover, a higher value of λ imposes more emphasis on fairness.

Now, the above formulation can be readily extended to individual fairness simply through considering each client to be a group. This translates to the global objective in (2) to be given as

$$H(\theta) = \sum_{k=1}^K \left(1 + \frac{\lambda}{p_k} r_k(\theta)\right) F_k(\theta).$$

In essence, our approach falls in line with FL literature that exploits the reweighting of clients. For instance, AFL proposed by Mohri et al. (2019) computes at every communication round the worst-case distribution of weights among clients. This approach promotes robustness but may be overly conservative in the sense that it focuses on the largest loss and thus causes very pessimistic

performance to other clients. Our algorithm, however, adaptively updates the weight of clients at every communication round based on the statistical ordering of client/group losses. Moreover, the dynamic update of the weights can potentially avoid overfitting by impeding updates for clients with low loss. We will further demonstrate the advantages of our algorithm in Section 5. In the next subsection, we provide our detailed algorithm for solving our proposed objective.

3.1. Algorithm

Algorithm 2 (GIFAIR-FL-Global)

Data: number of devices K , fraction α , number of communication rounds C , number of local updates E , SGD learning rate schedule $\{\eta^{(t)}\}_t$, initial model parameter θ , regularization parameter λ , initial loss $\{L_i\}_{1 \leq i \leq d}$

for $c = 0 : (C-1)$ **do**

Select clients by sampling probability p_k , and denote by \mathcal{S}_c the indices of these clients;

Server broadcasts $\left(\theta, \left\{\frac{\lambda}{p_k |A_{S_k}|} r_k^c(\theta)\right\}_{k \in \mathcal{S}_c}\right)$;

for $k \in \mathcal{S}_c$ **do**

$\theta_k^{(cE)} = \theta$;

for $t = cE : ((c+1)E-1)$ **do**

Randomly sample a subset of data and denote it by $\zeta_k^{(t)}$;

$\theta_k^{(t+1)} = \theta_k^{(t)} - \eta^{(t)} \left(1 + \frac{\lambda}{p_k |A_{S_k}|} r_k^c(\theta)\right) g_k(\theta_k^{(t)}; \zeta_k^{(t)})$;

//Note that $r_k^c(\theta)$ is fixed during local update (see Remark 4)

end

end

Aggregation $\bar{\theta}_c = \frac{1}{|\mathcal{S}_c|} \sum_{k \in \mathcal{S}_c} \theta_k^{((c+1)E)}$, set $\theta = \bar{\theta}_c$;

Calculate $L_i = \frac{1}{|A_i|} \sum_{k \in A_i} F_k(\theta_k^{((c+1)E)})$ for all $i \in [d]$ and update $r_k^{c+1}(\theta)$;

$c \leftarrow c + 1$;

end

Return $\bar{\theta}_C$.

In this section, we describe our proposed algorithm, GIFAIR-FL-Global, which is detailed in Algorithm 2. We highlight the differences between GIFAIR-FL-Global and FedAvg in red. At every communication round c , our algorithm selects a set of clients to participate in the training and shares r_k^c with each selected client. For each client, multiple SGD steps are then applied to a weighted client loss function. The updated parameters are then passed to the server that aggregates these results and computes r_k^{c+1} .

Computationally, our approach requires evaluating the client loss function at every communication round to compute r_k^c . Compared with existing fair FL approaches, GIFAIR-FL-Global is simple and computationally efficient. For instance, q-FFL proposed by Li

et al. (2019a) first runs FedAvg to obtain a well-tuned learning rate and uses this learning rate to roughly estimate the Lipschitz constant L . Another example is AFL, which requires running two gradient calls at each iteration to estimate the gradients of model and weight parameters. Similarly, Ditto requires running additional steps of SGD, at each communication round, to generate personalized solutions. In contrast, our proposed method can be seen as a fairness-aware weighted version of FedAvg.

Remark 2. In Algorithm 2, we sample local devices by sampling probability p_k and aggregate model parameters by an unweighted average $\frac{1}{|\mathcal{S}_c|} \sum_{k \in \mathcal{S}_c} \theta_k^{((c+1)E)}$. Alternatively, one may choose to uniformly sample clients. Then, the aggregation strategy should be replaced by $\bar{\theta}_c = \frac{K}{|\mathcal{S}_c|} \sum_{k \in \mathcal{S}_c} p_k \theta_k^{((c+1)E)}$ (Li et al. 2019c).

Remark 3. Instead of broadcasting p_k and $|\mathcal{A}_{s_k}|$ separately to local devices, the central server broadcasts the product $\frac{\lambda}{p_k |\mathcal{A}_{s_k}|} r_k^c(\theta)$ to client k . Hence, the local device k cannot obtain any information about p_k , $|\mathcal{A}_{s_k}|$, and $r_k^c(\theta)$. This strategy can protect privacy of other devices.

Remark 4. Notice that in (3), $H_k(\theta)$ is not differentiable because of the $r_k(\theta)$ component. However, r_k^c is fixed during local client training, as it is calculated on the central server. Also, local devices do not have any information about other devices; hence, they cannot update r_k^c during local training.

Remark 5. Despite introducing $O(d^2)$ regularizers to the main objective, our algorithm only requires computing group losses and r_k^c values, which require sorting the losses. More specifically, once the central server collects the selected clients' losses $\{F_k\}_k$, it first calculates group losses $\{L_i\}_i$. This step involves only the summation of scalars. Afterward, the server runs a sort algorithm to rank loss values. One can use many built-in sort functions in the Python library, and this sorting step is very fast even with millions of groups.

3.2. Convergence Guarantees

In this section, we first show that, under mild conditions, GIFAIR-FL-Global converges to the global optimal solution at a rate of $O\left(\frac{E^2}{T}\right)$ for strongly convex functions, and to a stationary point at a rate of $O\left(\frac{(E-1)\log(T+1)}{\sqrt{T}}\right)$, up to a logarithmic factor, for nonconvex functions. Here, $T := CE$ denotes the total number of iterations across all devices. Our theorems hold for both i.i.d. and non-i.i.d. data. Because of space limitations, we defer proof details to the online appendix.

3.2.1. Strongly Convex Functions. We assume each device performs E steps of local updates and make the following assumptions. Here, our assumptions are

based on F_k rather than H_k . These assumptions are very common in many FL papers (Li et al. 2018, 2019b, c).

Assumption 1. We assume F_k is L -smooth and μ -strongly convex for all $k \in [K]$.

Assumption 2. The variance of stochastic gradient is bounded. Specifically,

$$\mathbb{E}\{\|g_k(\theta_k^{(t)}; \zeta_k^{(t)}) - \nabla F_k(\theta_k^{(t)})\|^2\} \leq \sigma_k^2, \quad \forall k \in [K].$$

Assumption 3. The expected squared norm of the stochastic gradient is bounded. Specifically,

$$\mathbb{E}\{\|g_k(\theta_k^{(t)}; \zeta_k^{(t)})\|^2\} \leq G^2, \quad \forall k \in [K].$$

Typically, data from different groups are non-i.i.d.. We modify the definition in Li et al. (2019c) to roughly quantify the degree of non-i.i.d.-ness. Specifically,

$$\Gamma_K = H^* - \sum_{k=1}^K p_k H_k^* = \sum_{k=1}^K p_k (H^* - H_k^*),$$

where $H^* \triangleq H(\theta^*) = \sum_{k=1}^K H_k(\theta^*)$ is the optimal value of the global objective function, and $H_k^* \triangleq H_k(\theta_k^*)$ is the optimal value of the local loss function. If data are i.i.d., then $\Gamma_K \rightarrow 0$ as the number of samples grows. Otherwise, $\Gamma_K \neq 0$ (Li et al. 2019c). Given all aforementioned assumptions, we next prove the convergence of our proposed algorithm. We first assume all devices participate in each communication round (i.e., $|\mathcal{S}_c| = K$, for all c).

Theorem 1. Assume Assumptions 1–3 hold and $|\mathcal{S}_c| = K$. If $\eta^{(t)}$ is decreasing at a rate of $O\left(\frac{1}{t}\right)$ and $\eta^{(t)} \leq O\left(\frac{1}{t}\right)$, then, for $\gamma, \mu, \epsilon > 0$, we have

$$\mathbb{E}\{H(\bar{\theta}_c)\} - H^* \leq \frac{L}{2\gamma + T} \left\{ \frac{4\xi}{\epsilon^2 \mu^2} + (\gamma + 1) \|\bar{\theta}^{(0)} - \theta^*\|^2 \right\},$$

where $\xi = 8(E-1)^2 G^2 + 4L\Gamma + 2\frac{\Gamma_{\max}}{\eta^{(t)}} + 4\sum_{k=1}^K p_k^2 \sigma_k^2$, and $\Gamma_{\max} := \sum_{k=1}^K p_k (H^* - H_k^*) \geq |\sum_{k=1}^K p_k (H^* - H_k^*)| = |\Gamma_K|$. Here, $\bar{\theta}^{(0)} := \theta^{(0)}$, where $\theta^{(0)}$ is the initial model parameter in the central server.

Remark 6. Theorem 1 shows an $O\left(\frac{E^2}{T}\right)$ convergence rate, which is similar to that obtained from FedAvg. However, the rate is also affected by ξ , which contains the degree of non-i.i.d.-ness. Under fully i.i.d. settings where $\Gamma_K = \Gamma_{\max} = 0$, we retain the typical FedAvg for strongly convex functions.

Next, we assume only a fraction of devices participate in each communication round (i.e., $|\mathcal{S}_c| = \alpha K$, for all $c, \alpha \in (0, 1)$). As per Algorithm 2, all local devices are sampled according to the sampling probability p_k (Li et al. 2018). Our theorem can similarly be extended to the scenario where devices are sampled uniformly (i.e., with the same probability). Recall that the aggregation strategy becomes $\bar{\theta}_c = \frac{K}{|\mathcal{S}_c|} \sum_{k \in \mathcal{S}_c} p_k \theta_k$ (Li et al. 2019c).

Theorem 2. Assume that at each communication round, the central server samples a fraction $|\mathcal{S}_c|$ of devices according to the sampling probability p_k . Additionally, assume

Assumptions 1–3 hold. If $\eta^{(t)}$ is decreasing at a rate of $\mathcal{O}(\frac{1}{t})$ and $\eta^{(t)} \leq \mathcal{O}(\frac{1}{t})$, then, for $\gamma, \mu, \epsilon > 0$, we have

$$\mathbb{E}\{H(\bar{\theta}_c)\} - H^* \leq \frac{L}{2} \frac{1}{\gamma + T} \left\{ \frac{4(\xi + \tau')}{\epsilon^2 \mu^2} + (\gamma + 1) \|\bar{\theta}^{(0)} - \theta^*\|^2 \right\},$$

where $\tau' = \frac{4G^2 E^2}{|\mathcal{S}_c|}$.

Remark 7. Under the partial device participation scenario, the same convergence rate $\mathcal{O}(\frac{E^2}{T})$ holds. The only difference is that there is a term $\tau' = \frac{4G^2 E^2}{|\mathcal{S}_c|}$ that appears in the upper bound. This ratio slightly impedes the convergence rate when the number of sampled devices $|\mathcal{S}_c|$ is small.

3.2.2. Nonconvex Functions. To prove the convergence result on nonconvex functions, we replace Assumption 1 by the following assumption.

Assumption 4. We assume F_k is L -smooth for all $k \in [K]$.

Theorem 3. Assume Assumptions 2–4 hold and $|\mathcal{S}_c| = K$. If $\eta^{(t)} = \mathcal{O}(\frac{1}{\sqrt{t}})$ and $\eta^{(t)} \leq \mathcal{O}(\frac{1}{t})$, then our algorithm converges to a stationary point. Specifically,

$$\begin{aligned} \min_{t=1, \dots, T} \mathbb{E}\{\|\nabla H(\bar{\theta}^{(t)})\|^2\} \\ \leq \frac{\{2(1 + 2L^2 \log(T + 1))\mathbb{E}\{H(\bar{\theta}^{(0)}) - H^*\} + 2\xi_{\Gamma_K}\}}{\sqrt{T}}, \end{aligned}$$

where

$$\xi_{\Gamma_K} = \mathcal{O}\left(\left(2L^2 \Gamma_K + 8(E-1)LG^2 + 10L \sum_{k=1}^K p_k \sigma_k^2\right) \log(T + 1)\right),$$

$$\text{and } \bar{\theta}^{(t)} = \frac{1}{|\mathcal{S}_c|} \sum_{k \in \mathcal{S}_c} \theta_k^{(t)}.$$

Remark 8. Our results show that GIFAIR-FL converges to a stationary point at a rate of $\mathcal{O}(\frac{(E-1)\log(T+1)}{\sqrt{T}})$. Similar to the strongly convex setting, this convergence rate is affected by the degree of non-i.i.d.-ness Γ_K .

3.3. Discussion and Limitations

We here note our theoretical results require exact computation of r_k . However, Algorithm 2 uses an estimate of r_k at every communication round using the local client loss prior to aggregation. This procedure might generate an inexact estimate of r_k , as one cannot guarantee $F_k(\bar{\theta}_c) = F_k(\theta_k^{(c+1)E})$ at each communication round. Here recall that r_k in Equation (2) is calculated based on the order of $F_k(\bar{\theta}_c)$. To guarantee an exact r_k , the server can ask clients to share the local losses evaluated at the global parameters. Specifically, after sharing $\bar{\theta}_c$ to selected local devices, those devices calculate $\{F_k(\bar{\theta}_c)\}_k$ and send loss values back to the central server to update r_k^{c+1} . This, however, requires more communication rounds. One approach to remedy the limitation of GIFAIR-FL-Global is to develop a personalized counterpart to

circumvent additional communication rounds. We will detail this idea in the coming section.

4. GIFAIR-FL-Per: A Personalized Model for Fairness

In this section, we slightly tailor GIFAIR-FL-Global to a personalized fair algorithm, GIFAIR-FL-Per. While still aiming to minimize the spread in the loss among client groups, our proposed objective evaluates the loss at the client-specific (i.e., personalized) solution. Formally speaking, our objective function is

$$\begin{aligned} \min_{\theta} H(\theta, \theta_1, \dots, \theta_K) \triangleq \\ \sum_{k=1}^K p_k F_k(\theta) + \lambda \sum_{1 \leq i < j \leq d} |L_i(\{\theta_k\}_{k \in \mathcal{A}_i}) - L_j(\{\theta_k\}_{k \in \mathcal{A}_j})|, \end{aligned} \quad (4)$$

where

$$L_i(\{\theta_k\}_{k \in \mathcal{A}_i}) \triangleq \frac{1}{|\mathcal{A}_i|} \sum_{k \in \mathcal{A}_i} F_k(\theta_k)$$

is the average loss for client group i , and $\sum_{k=1}^K p_k \theta_k = \theta$.

Remark 9. Different from (1), objective (4) achieves fairness from the device perspective. By optimizing (4), we can obtain device-specific solutions $\{\theta_k\}_{k=1}^K$ that yield small discrepancies among $\{L_i(\{\theta_k\}_{k \in \mathcal{A}_i})\}_{i=1}^d$. Although (1) and (4) have different perspectives, their ultimate goals are aligned with Definition 1.

Objective (4) has many notable features: (i) First, compared with (1), the new objective function (4) evaluates group losses $\{L_i\}_{i=1}^d$ with respect to personalized solutions $\{\theta_k\}_{k=1}^K$. This formulation circumvents the need to collect losses evaluated at the global parameter and therefore requires no extra communication rounds to calculate r_k exactly. (ii) Second, the global parameter $\theta = \sum_{k=1}^K p_k \theta_k$ ensures aggregation happens at every communication round. This can safeguard against overfitting on local devices. Otherwise, each device will simply minimize its own local loss, without communication, and obtain a small loss value. (iii) Before discussing the third property, we first need to present the convergence results of GIFAIR-FL-Per.

Theorem 4. Assume Assumptions 1–3 hold and $|\mathcal{S}_c| = K$. If $\eta^{(t)}$ is decreasing at a rate of $\mathcal{O}(\frac{1}{t})$ and $\eta \leq \mathcal{O}(\frac{1}{t})$, then, for $\gamma, \mu, \epsilon > 0$, we have

$$\mathbb{E}\left\{H(\bar{\theta}_c, \{\theta_k^{(T)}\}_{k=1}^K)\right\} - H^* \leq \mathcal{O}\left(\frac{E^2}{T}\right),$$

where $\sum_{k=1}^K p_k \theta_k^{(T)} = \bar{\theta}_c$ and $H^* := H(\theta^*, \{\theta_k\}_{k=1}^K)$ such that $\sum_{k=1}^K p_k \theta_k = \theta^*$. A same convergence rate holds for the partial device participation scenario.

Under the nonconvex condition, we have

$$\min_{t=1, \dots, T} \mathbb{E}\left\{\|\nabla H(\bar{\theta}^{(t)})\|^2\right\} \leq \mathcal{O}\left(\frac{(E-1)\log(T+1)}{\sqrt{T}}\right).$$

The proof here follows a scheme similar to that for GIFAIR-FL-Global. Theorem 4 implies that GIFAIR-FL-Per drives aggregated parameter $\bar{\theta}_c$ to the global

optimal solution θ^* at a rate of $\mathcal{O}\left(\frac{E^2}{T}\right)$. This aggregated parameter is obtained from taking the weighed average of personalized solutions $\{\theta_k^{(T)}\}_{k=1}^K$. This leads to a new interpretation of GIFAIR-FL-Per: once the optimizer reaches $\bar{\theta}_C$, device k retains personalized solution $\theta_k^{(T)}$ that stays in the vicinity of the global model parameter to balance each client's shared knowledge and unique characteristics. One can link this idea to Ditto (Li et al. 2021), the recent state-of-the-art personalized FL algorithm. Ditto allows local devices to run more steps of SGD, subject to some constraints such that local solutions will not move far away from the global solution. GIFAIR-FL-Per, on the other hand, scales the magnitude of gradients based on the statistical ordering of client/group losses.

Algorithm 3 (GIFAIR-FL-Per)

Data: number of devices K , fraction α , number of communication rounds C , number of local updates E , SGD learning rate schedule $\{\eta^{(t)}\}_{t=1}^E$, initial model parameter θ , regularization parameter λ , initial loss $\{L_i\}_{1 \leq i \leq d}$

for $c = 0 : (C-1)$ **do**

Select $|S_c|$ clients by sampling probability p_k , and denote by S_c the indices of these clients;

Server broadcasts $\left(\theta, \left\{\frac{\lambda}{p_k |A_{S_k}|} r_k^c(\{\theta_k\}_{k=1}^K)\right\}_{k \in S_c}\right)$;

for $k \in S_c$ **do**

$\theta_k^{(cE)} = \theta$;

for $t = cE : ((c+1)E-1)$ **do**

$\theta_k^{(t+1)} = \theta_k^{(t)} - \eta^{(t)} \left(1 + \frac{\lambda}{p_k |A_{S_k}|} r_k^c(\{\theta_k\}_{k=1}^K)\right) \nabla F_k(\theta_k^{(t)})$;

end

end

Aggregation $\bar{\theta}_c = \frac{1}{|S_c|} \sum_{k \in S_c} \theta_k^{((c+1)E)}$, set $\theta = \bar{\theta}_c$;

Calculate $L_i = \frac{1}{|A_i|} \sum_{k \in A_i} F_k(\theta_k^{((c+1)E)})$ for all $i \in [d]$;

Set $\theta_k = \theta_k^{((c+1)E)}$ for all $k \in S_c$. Remain θ_k unchanged otherwise; update $r_k^{c+1}(\{\theta_k\}_{k=1}^K)$;

$c \leftarrow c + 1$;

end

Return $\{\theta_k\}_{k=1}^K$.

Finally, we detail GIFAIR-FL-Per in Algorithm 3. In the algorithm, r_k is defined as

$$r_k(\{\theta_k\}_{k=1}^K) = \sum_{1 \leq j \neq s_k \leq d} \text{sign}(L_{s_k}(\{\theta_m\}_{m \in A_{s_k}}) - L_j(\{\theta_m\}_{m \in A_j})). \quad (5)$$

In other words, r_k is computed based on the ordering of losses evaluated on the personalized solutions.

5. Experiments

In this section, we test GIFAIR-FL on image classification and text prediction tasks.

We benchmark our model with the following algorithms: q-FFL (Li et al. 2019a), TERM and TERM-Group (Li et al. 2020b), FedMGDA+ (Hu et al. 2020), AFL (Mohri et al. 2019), and FedMGDA+ (Hu et al. 2020). To the best of our knowledge, those are the well-known current state-of-the-art FL algorithms for fairness. We also benchmark our model with Ditto (Li et al. 2021), which is a personalized FL approach using multitask learning.

5.1. Image Classification

We start by considering a federated image classification data set, FEMNIST (Federated Extended Modified National Institute of Standards and Technology (MNIST) database; Caldas et al. 2018). FEMNIST consists of images of digits (zero to nine) and English characters (A to Z and a to z) with 62 classes (Figure 3) written by different people. Images are 28 by 28 pixels. All images are partitioned and distributed to 3,550 devices by the data set creators (Caldas et al. 2018).

5.1.1. Individual Fairness. Following the setting in Li et al. (2018), we first sample 10 lowercase characters ("a" through "j") from Extended MNIST (EMNIST; Cohen et al. 2017) and distribute five classes of images to each device (FEMNIST-Skewed; $d = 100$). Each local device has 500 images. There are 100 devices in total. Results are reported in Table 1. Then, following the setting in (Li et al. 2021), we sample 500 devices and train models using the default data stored in each device

Figure 3. Example of Images from FEMNIST

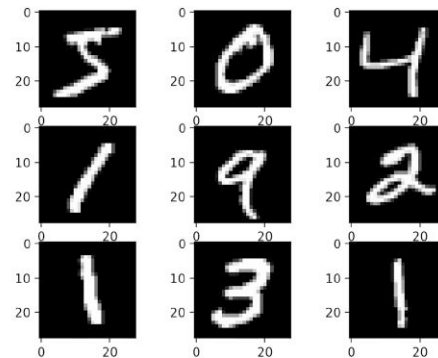


Table 1. Empirical Results on FEMNIST-Skewed

	Algorithm						
	FedAvg	q-FFL	TERM	FedMGDA+	Ditto	GIFAIR-FL-Global	GIFAIR-FL-Per
\bar{a}	79.2 (1.0)	84.6 (1.9)	84.2 (1.3)	85.0 (1.7)	92.5 (3.1)	87.9 (0.9)	93.0 (1.1)
$\sqrt{\text{Var}(a)}$	22.3 (1.1)	18.5 (1.2)	13.8 (1.0)	14.9 (1.6)	14.3 (1.0)	5.7 (0.8)	6.2 (0.9)

Note. Each experiment is repeated five times.

(FEMNIST-Original; $d = 500$). Results are reported in Table 2.

5.1.2. Group Fairness. We manually divide FEMNIST data into three groups (FEMNIST-3-Groups; $d = 3$). See Table 3 for the detailed assignment. This assignment is inspired by the statistic that most people prefer to write in lowercase letters, but a small amount of people prefer to write in capital letters or a mix of the two types (Jones and Mewhort 2004). In such cases, it is important to assure that an FL algorithm is capable of achieving similar performance between such groups. Results are reported in Table 4.

5.1.3. Implementation. For all tasks, we randomly split the data on each local device into a 70% training set, a 10% validation set, and a 20% testing set. This is a common data splitting strategy used in many FL papers (Chen et al. 2018, Li et al. 2018, Reddi et al. 2020). The batch size is set to be 32. We use the tuned initial learning rate 0.1 and decay rate 0.99 for each method. During each communication round, 10 devices are randomly selected, and each device will run two epochs of SGD. We use a convolutional neural network model with two convolution layers followed by two fully connected layers. All benchmark models are well tuned. Specifically, we solve q-FFL with $q \in \{0, 0.001, 0.01, 0.1, 1, 2, 5, 10\}$ (Li et al. 2019a) in parallel and select the best q . Here, the best q is defined as the q value where the variance decreases the most while the averaged testing accuracy is superior or similar to FedAvg. This definition is borrowed from the original q-FFL paper (Li et al. 2019a). Similarly, we train TERM with $t \in \{1, 2, 5\}$ and select the best t (Li et al. 2020b). For Ditto, we tune the regularization parameter $\lambda_{\text{Ditto}} \in \{0.01, 0.05, 0.1, 0.5, 1, 2, 5\}$. In GIFAIR-FL, we tune the parameter $\lambda \in \{0, 0.1\lambda_{\max}, 0.2\lambda_{\max}, \dots, 0.8\lambda_{\max}, 0.9\lambda_{\max}\}$. Here, kindly note that λ_{\max} is a function of $p_k, |A_{s_k}|$ and d (i.e., data dependent).

Table 2. Test Accuracy on FEMNIST-Original

	Algorithm						
	FedAvg	q-FFL	TERM	AFL	Ditto	GIFAIR-FL-Global	GIFAIR-FL-Per
\bar{a}	80.4 (1.3)	80.9 (1.1)	81.0 (1.0)	82.4 (1.0)	83.7 (1.9)	83.2 (0.7)	84.1 (1.2)
$\sqrt{\text{Var}(a)}$	11.1 (1.4)	10.6 (1.3)	10.3 (1.2)	9.85 (0.9)	10.1 (1.6)	5.2 (0.8)	4.5 (0.8)

Note. Each experiment is repeated five times.

5.1.4. Performance Metrics. Denote by a_k the prediction accuracy on device k . We define (1) individual-level mean accuracy as $\bar{a} := \frac{1}{K} \sum_{k=1}^K a_k$ and (2) individual-level variance as $\text{Var}(a) := \frac{1}{K} \sum_{k=1}^K (a_k - \bar{a})^2$.

5.2. Text Data

5.2.1. Individual Fairness. We train a recurrent neural network (RNN) to predict the next character using text data built from *The Complete Works of William Shakespeare* (<https://github.com/TalwalkarLab/leaf/tree/master/data/shakespeare>). In this data set, there are about 1,129 speaking roles. Naturally, each speaking role in each play is treated as a device. Each device stored several text data, and those information were used to train a RNN on each device. The data set is available on the LEAF website (Caldas et al. 2018).

Following the setting in McMahan et al. (2017) and Li et al. (2019a), we subsample 31 roles ($d = 31$). The RNN model takes an 80-character sequence as the input and outputs one character after two long short-term memory layers and one densely connected layer. For FedAvg, q-FFL, and Ditto, the best initial learning rate is 0.8 and best decay rate is 0.95 (Li et al. 2021). We also adopt this setting for GIFAIR-FL-Global and GIFAIR-FL-Per. The batch size is set to be 10. The number of local epochs is fixed to be one, and all models are trained for 500 epochs. Results are reported in Table 5.

5.2.2. Group Fairness. We obtain gender information from <https://shakespeare.folger.edu/> and group speaking roles based on gender ($d = 2$). It is known that the majority of characters in Shakespearean dramas are males. Simply training a FedAvg model on this data set will cause implicit bias toward male characters. On par with this observation, we subsample 25 males and 10 females from *The Complete Works of William Shakespeare*. Here we note that each device in the male group implicitly has more text data. The setting of hyperparameters is same as for individual fairness. Results are reported in Table 6.

Table 3. Data Structure of FEMNIST-3-Groups

Group	Data type	Number of images	Number of devices
Group 1	Capital letters + digits	800	60
Group 2	Lowercase letters + digits	1,000	100
Group 3	Capital/lowercase letters + digits	600	40

Table 4. Test Accuracy on FEMNIST-3-Groups

	Algorithm							
	FedAvg	q-FFL	TERM	FedMGDA+	Ditto	TERM-Group	GIFAIR-FL-Global	GIFAIR-FL-Per
Group 1	79.72 (2.08)	81.15 (1.97)	81.29 (1.45)	81.03 (2.28)	82.37 (2.06)	82.01 (1.95)	83.41 (1.34)	83.96 (1.22)
Group 2	90.93 (2.35)	88.24 (2.13)	88.08 (1.09)	89.12 (1.74)	92.05 (2.00)	89.13 (1.00)	88.29 (1.22)	91.05 (1.31)
Group 3	80.21 (2.91)	80.93 (1.86)	81.84 (1.44)	81.33 (1.59)	83.03 (2.18)	81.75 (2.04)	84.37 (1.85)	84.98 (0.99)
Discrepancy	11.21	7.31	6.79	8.09	9.02	7.38	6.07	7.09

Notes. Each experiment is repeated five times. Discrepancy is the difference between the largest accuracy and the smallest accuracy.

Table 5. Means and Standard Deviations of Test Accuracy on Shakespeare ($d = 31$)

	Algorithm					
	FedAvg	q-FFL	AFL	Ditto	GIFAIR-FL-Global	GIFAIR-FL-Per
\bar{a}	53.21 (0.31)	53.90 (0.30)	54.58 (0.14)	60.74 (0.42)	57.04 (0.23)	61.58 (0.14)
$\sqrt{\text{Var}(\bar{a})}$	9.25 (6.17)	7.52 (5.10)	8.44 (5.65)	8.32 (4.77)	3.14 (1.25)	4.33 (1.25)

Note. Each experiment is repeated five times.

Table 6. Test Accuracy on Shakespeare ($d = 2$)

	Algorithm						
	FedAvg	q-FFL	FedMGDA+	Ditto	TERM-Group	GIFAIR-FL-Global	GIFAIR-FL-Per
Male	72.95 (1.70)	67.14 (2.18)	67.07 (2.11)	74.19 (3.75)	72.87 (1.01)	67.42(0.98)	73.95 (0.59)
Female	40.39 (1.49)	43.26 (2.05)	43.85 (2.32)	45.73 (4.01)	44.31 (0.96)	52.04 (1.10)	54.88 (1.12)

Note. Each experiment is repeated five times.

5.3. Analysis of Results

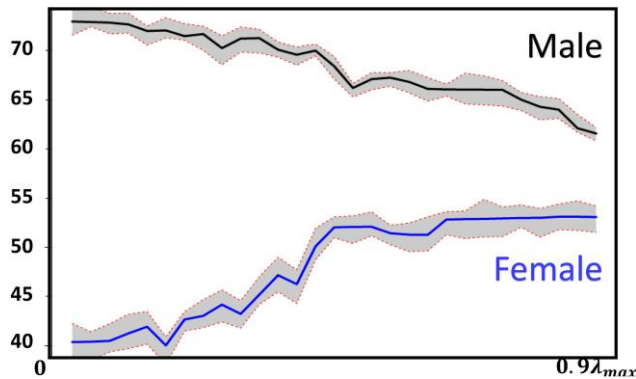
Based on Tables 1, 2, 4–6, we can obtain important insights. First, compared with other benchmark models, GIFAIR-FL-Global and GIFAIR-FL-Per lead to significantly more fair solutions. As shown in Tables 1, 2, and 5, our algorithm significantly reduces the variance of testing accuracy of all devices (i.e., $\text{Var}(\bar{a})$), while the average testing accuracy remains consistent. Second, from Tables 4 and 6, it can be seen that GIFAIR-FL-Global and GIFAIR-FL-Per boosted the performance of the group with the worst testing accuracy and achieved the smallest discrepancy. Notably, this boost did not affect the performance of other groups. This indicates that GIFAIR-FL-Global and GIFAIR-FL-Per are capable of ensuring fairness among different groups while retaining superior or similar prediction accuracy compared with existing benchmark models. Finally, we note

that GIFAIR-FL-Global sometimes achieves lower prediction performance than Ditto. This is understandable, as Ditto provides a personalized solution to each device k , whereas our model returns only a global parameter $\bar{\theta}$. Yet, as shown in the last column, if we use GIFAIR-FL-Per, then the prediction performance can be significantly improved without sacrificing fairness. However, even without personalization, GIFAIR-FL-Global achieves superior testing performance compared with existing fair FL benchmark models.

5.4. Sensitivity Analysis

In this section, we use GIFAIR-FL-Global to study the effect of the tuning parameter $\lambda \in [0, \lambda_{max}]$ using the Shakespeare data set. A similar conclusion holds for GIFAIR-FL-Per, and we therefore omit it. Results are reported in Figure 4. It can be seen that as λ increases,

Figure 4. (Color online) Sensitivity with Respect to λ (Shakespeare Data Set)



the discrepancy between male and female groups decreases accordingly. However, after λ passes a certain threshold, the averaged testing accuracy of the female group remains flat, yet the performance of the male group significantly drops. Therefore, in practice, it is recommended to consider a moderate λ value. Intuitively, when $\lambda = 0$, GIFAIR-FL becomes FedAvg. When λ is close to λ_{max} , the coefficient (i.e., $(1 + \lambda \frac{1}{p_k |A_{s_k}|} r_k)$) of devices with good performance will be close to zero, and the updating is, therefore, impeded. A moderate λ balances those two situations well. Besides this example, we also conducted additional sensitivity analysis. Because of space limitations, we defer those results to the online appendix.

6. Conclusion

In this paper, we propose GIFAIR-FL, a framework that imposes group and individual fairness to FL. Experiments show that GIFAIR-FL can lead to more fair solutions compared with recent state-of-the-art fair and personalized FL algorithms while retaining similar testing performance. To the best of our knowledge, fairness in FL is an underinvestigated area, and we hope our work will help inspire continued exploration into fair FL algorithms.

Also, real-life FL data sets for specific engineering or health science applications are still scarce. This is understandable, as FL efforts have mainly focused on mobile applications. As such, we test only on image classification and text prediction data sets. However, as FL is expected to infiltrate many applications, we hope that more real-life data sets will be generated to provide a means for model validation within different domains. We plan to actively pursue this direction in future research.

Acknowledgments

This paper abides by data ethics requirements. All data are publicly available online.

References

- Al-Ali A, Gupta R, Nabulsi AA (2018) Cyber physical systems role in manufacturing technologies. *AIP Conf. Proc.*, vol. 1957 (AIP Publishing, Melville, NY), 050007.
- Arivazhagan MG, Aggarwal V, Singh AK, Choudhary S (2019) Federated learning with personalization layers. Preprint, submitted December 2, <https://arxiv.org/abs/1912.00818>.
- Bhagoji AN, Chakraborty S, Mittal P, Calo S (2019) Analyzing federated learning through an adversarial lens. *Proc. 36th Internat. Conf. Machine Learn.* Proceedings of Machine Learning Research, vol. 97, 634–643.
- Caldas S, Duodu SMK, Wu P, Li T, Konečný J, McMahan HB, Smith V, Talwalkar A (2018) LEAF: A benchmark for federated settings. Preprint, submitted December 3, <https://arxiv.org/abs/1812.01097>.
- Chen F, Luo M, Dong Z, Li Z, He X (2018) Federated meta-learning with fast convergence and efficient communication. Preprint, submitted February 22, <https://arxiv.org/abs/1802.07876>.
- CleanTechnica (2021) Tesla FSD hardware has 150 million times more computer power than Apollo 11 computer. Accessed May 24, <https://cleantechnica.com/2021/05/24/tesla-fsd-hardware-has-150-million-times-more-computer-power-than-apollo-11-computer/>.
- Cohen G, Afshar S, Tapson J, Van Schaik A (2017) EMNIST: Extending MNIST to handwritten letters. *2017 Internat. Joint Conf. Neural Networks* (Institute of Electrical and Electronics Engineers, Piscataway, NJ), 2921–2926.
- Dai Z, Low KH, Jaillet P (2020) Federated Bayesian optimization via Thompson sampling. Preprint, submitted October 20, <https://arxiv.org/abs/2010.10154>.
- Dinh CT, Tran NH, Nguyen TD (2020) Personalized federated learning with Moreau envelopes. Larochelle H, Ranzato M, Hadsell R, Balcan MF, Lin H, eds. *Proc. 34th Internat. Conf. Neural Inform. Processing Systems* (Curran Associates, Red Hook, NY), 21394–21405.
- Du W, Xu D, Wu X, Tong H (2020) Fairness-aware agnostic federated learning. Preprint, October 10, <https://arxiv.org/abs/2010.05057>.
- Fallah A, Mokhtari A, Ozdaglar A (2020) Personalized federated learning: A meta-learning approach. Larochelle H, Ranzato M, Hadsell R, Balcan MF, Lin H, eds. *Proc. 34th Internat. Conf. Neural Inform. Processing Systems* (Curran Associates, Red Hook, NY).
- Feldman M, Friedler SA, Moeller J, Scheidegger C, Venkatasubramanian S (2015) Certifying and removing disparate impact. *Proc. 21th ACM SIGKDD Internat. Conf. Knowledge Discovery Data Mining* (Association for Computing Machinery, New York), 259–268.
- Ghosh A, Hong J, Yin D, Ramchandran K (2019) Robust federated learning in a heterogeneous environment. Preprint, submitted June 16, <https://arxiv.org/abs/1906.06629>.
- Hard A, Rao K, Mathews R, Ramaswamy S, Beaufays F, Augenstein S, Eichner H, Kiddon C, Ramage D (2018) Federated learning for mobile keyboard prediction. Preprint, submitted November 8, <https://arxiv.org/abs/1811.03604>.
- Hardt M, Price E, Srebro N (2016) Equality of opportunity in supervised learning. Lee D, Sugiyama M, Luxburg U, Guyon I, Garnett R, eds. *Proc. 30th Internat. Conf. Neural Inform. Processing Systems* (Curran Associates Inc., Red Hook, NY).
- Hu B, Gao Y, Liu L, Ma H (2018) Federated region-learning: An edge computing based framework for urban environment sensing. *2018 IEEE Global Comm. Conf.* (Institute of Electrical and Electronics Engineers, Piscataway, NJ), 1–7.
- Hu Z, Shaloudegi K, Zhang G, Yu Y (2020) FedMGDA+: Federated learning meets multi-objective optimization. Preprint, submitted June 20, <https://arxiv.org/abs/2006.11489>.
- Huang W, Li T, Wang D, Du S, Zhang J (2020) Fairness and accuracy in federated learning. Preprint, submitted December 18, <https://arxiv.org/abs/2012.10069>.

- Jiang JC, Kantarci B, Oktug S, Soyata T (2020b) Federated learning in smart city sensing: Challenges and opportunities. *Sensors (Basel)* 20(21):6230.
- Jiang Y, Konečný J, Rush K, Kannan S (2019) Improving federated learning personalization via model agnostic meta learning. Preprint, submitted September 27, <https://arxiv.org/abs/1909.12488>.
- Jiang J, Hu L, Hu C, Liu J, Wang Z (2020a) BACoMo-bandwidth-aware decentralized federated learning. *Electronics (Basel)* 9(3):440.
- Jones MN, Mewhort DJ (2004) Case-sensitive letter and bigram frequency counts from large-scale English corpora. *Behav. Res. Methods Instruments Comput.* 36(3):388–396.
- Kairouz P, McMahan HB, Avent B, Bellet A, Bennis M, Bhagoji AN, Bonawitz K, et al. (2019) Advances and open problems in federated learning. Preprint, submitted December 10, <https://arxiv.org/abs/1912.04977>.
- Karimireddy SP, Kale S, Mohri M, Reddi S, Stich S, Suresh AT (2020) SCAFFOLD: Stochastic controlled averaging for federated learning. *Proc. 37th Internat. Conf. Machine Learn.* Proceedings of Machine Learning Research, vol. 119, 5132–5143.
- Khodak M, Tu R, Li T, Li L, Balcan MF, Smith V, Talwalkar A (2021) Federated hyperparameter tuning: Challenges, baselines, and connections to weight-sharing. Preprint, submitted June 8, <https://arxiv.org/abs/2106.04502>.
- Kontar R, Shi N, Yue X, Chung S, Byon E, Chowdhury M, Jin J, et al. (2021) The internet of federated things (IoFT). *IEEE Access* 9:156071–156113.
- Li D, Wang J (2019) FedMD: Heterogenous federated learning via model distillation. Preprint, submitted October 8, <https://arxiv.org/abs/1910.03581>.
- Li T, Beirami A, Sanjabi M, Smith V (2020b) Tilted empirical risk minimization. Preprint, submitted July 2, <https://arxiv.org/abs/2007.01162>.
- Li L, Fan Y, Tse M, Lin KY (2020a) A review of applications in federated learning. *Comput. Indust. Engrg.* 149(November): 106854.
- Li T, Hu S, Beirami A, Smith V (2021) Ditto: Fair and robust federated learning through personalization. *Proc. 38th Internat. Conf. Machine Learn.* Proceedings of Machine Learning Research, vol. 139, 6357–6368.
- Li T, Sanjabi M, Beirami A, Smith V (2019a) Fair resource allocation in federated learning. *Internat. Conf. Learn. Representations*.
- Li X, Yang W, Wang S, Zhang Z (2019b) Communication-efficient local decentralized SGD methods. Preprint, submitted October 21, <https://arxiv.org/abs/1910.09126>.
- Li X, Huang K, Yang W, Wang S, Zhang Z (2019c) On the convergence of FedAvg on non-IID data. Preprint, submitted July 4, <https://arxiv.org/abs/1907.02189>.
- Li T, Sahu AK, Zaheer M, Sanjabi M, Talwalkar A, Smith V (2018) Federated optimization in heterogeneous networks. Dhillon I, Papailiopoulos D, Sze V, eds. *Proc. Machine Learn. Systems*, vol. 2, 429–450.
- Liang PP, Liu T, Ziyin L, Allen NB, Auerbach RP, Brent D, Salakhutdinov R, Morency LP (2020) Think locally, act globally: Federated learning with local and global representations. Preprint, submitted January 6, <https://arxiv.org/abs/2001.01523>.
- Lyu L, Xu X, Wang Q, Yu H (2020) Collaborative fairness in federated learning. Goebel R, Tanaka Y, Wahlster W, eds. *Federated Learning* (Springer, Cham, Switzerland), 189–204.
- Mansour Y, Mohri M, Ro J, Suresh AT (2020) Three approaches for personalization with applications to federated learning. Preprint, submitted February 25, <https://arxiv.org/abs/2002.10619>.
- McMahan B, Moore E, Ramage D, Hampson S, Arcas BA (2017) Communication-efficient learning of deep networks from decentralized data. *Proc. 20th Internat. Conf. Artificial Intelligence Statist.* Proceedings of Machine Learning Research, vol. 54, 1273–1282.
- Mohri M, Sivek G, Suresh AT (2019) Agnostic federated learning. *Proc. 36th Internat. Conf. Machine Learn.* Proceedings of Machine Learning Research, vol. 97, 4615–4625.
- Nguyen HT, Sehswag V, Hosseinalipour S, Brinton CG, Chiang M, Poor HV (2020) Fast-convergent federated learning. *IEEE J. Selected Areas Comm.* 39(1):201–218.
- Pillutla K, Kakade SM, Harchaoui Z (2019) Robust aggregation for federated learning. Preprint, submitted December 31, <https://arxiv.org/abs/1912.13445>.
- Ramaswamy S, Mathews R, Rao K, Beaufays F (2019) Federated learning for emoji prediction in a mobile keyboard. Preprint, submitted June 11, <https://arxiv.org/abs/1906.04329>.
- Reddi S, Charles Z, Zaheer M, Garrett Z, Rush K, Konečný J, Kumar S, McMahan HB (2020) Adaptive federated optimization. Preprint, submitted February 29, <https://arxiv.org/abs/2003.00295>.
- Samsung (2019) Your phone is now more powerful than your PC. Accessed February 19, 2019, <https://insights.samsung.com/2021/08/19/your-phone-is-now-more-powerful-than-your-pc-3/>.
- Sattler F, Wiedemann S, Müller KR, Samek W (2019) Robust and communication-efficient federated learning from non-iid data. *IEEE Trans. Neural Networks Learn. Systems* 31(9):3400–3413.
- Smith V, Chiang CK, Sanjabi M, Talwalkar AS (2017) Federated multi-task learning. Guyon I, Von Luxburg U, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R, eds. *Advances in Neural Information Processing Systems*, vol. 30 (Curran Associates, Red Hook, NY), 4424–4434.
- Wang H, Yurochkin M, Sun Y, Papailiopoulos D, Khazaeni Y (2020b) Federated learning with matched averaging. *Internat. Conf. Learn. Representations*.
- Wang K, Mathews R, Kiddon C, Eichner H, Beaufays F, Ramage D (2019) Federated evaluation of on-device personalization. Preprint, submitted October 22, <https://arxiv.org/abs/1910.10252>.
- Wang H, Sreenivasan K, Rajput S, Vishwakarma H, Agarwal S, Sohn Jy, Lee K, Papailiopoulos D (2020a) Attack of the tails: Yes, you really can backdoor federated learning. Larochelle H, Ranzato M, Hadsell R, Balcan MF, Lin H, eds. *Proc. 34th Internat. Conf. Neural Inform. Processing Systems* (Curran Associates, Red Hook, NY), 16070–16084.
- Xu X, Lyu L (2020) A reputation mechanism is all you need: Collaborative fairness and adversarial robustness in federated learning. Preprint, submitted November 20, <https://arxiv.org/abs/2011.10464>.
- Xu J, Glicksberg BS, Su C, Walker P, Bian J, Wang F (2021) Federated learning for healthcare informatics. *J. Healthcare Informatics Res.* 5(1):1–19.
- Yang K, Jiang T, Shi Y, Ding Z (2020) Federated learning via over-the-air computation. *IEEE Trans. Wireless Comm.* 19(3):2022–2035.
- Yu T, Bagdasaryan E, Shmatikov V (2020) Salvaging federated learning by local adaptation. Preprint, submitted February 12, <https://arxiv.org/abs/2002.04758>.
- Yuan H, Ma T (2020) Federated accelerated stochastic gradient descent. *Proc. 34th Internat. Conf. Neural Inform. Processing Systems* (Curran Associates Inc., Red Hook, NY), 5332–5344.
- Yue X, Kontar RA (2021) Federated Gaussian process: Convergence, automatic personalization and multi-fidelity modeling. Preprint, submitted November 28, <https://arxiv.org/abs/2111.14008>.
- Zafar MB, Valera I, Gomez Rodriguez M, Gummadi KP (2017) Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. *Proc. 26th Internat. Conf. World Wide Web* (International World Wide Web Conferences Steering Committee, Geneva, Switzerland), 1171–1180.
- Zhang DY, Kou Z, Wang D (2020a) FairFL: A fair federated learning approach to reducing demographic bias in privacy-sensitive classification models. *2020 IEEE Internat. Conf. Big Data* (Institute of Electrical and Electronics Engineers, Piscataway, NJ), 1051–1060.

Zhang J, Li C, Robles-Kelly A, Kankanhalli M (2020b) Hierarchically fair federated learning. Preprint, submitted April 22, <https://arxiv.org/abs/2004.10386>.

Zhang X, Zhu X, Wang J, Yan H, Chen H, Bao W (2020c) Federated learning with adaptive communication compression under

dynamic bandwidth and unreliable networks. *Inform. Sci.* 540 (November):242–262.

Zhao Y, Li M, Lai L, Suda N, Civan D, Chandra V (2018) Federated learning with non-IID data. Preprint, submitted June 2, <https://arxiv.org/abs/1806.00582>.