# Federating for Learning Group Fair Models

**Afroditi Papadaki**
University College London
a.papadaki.17@ucl.ac.uk

**Natalia Martinez**
Duke University
natalia.martinez@duke.edu

**Martin Bertran**
Duke University
martin.bertran@duke.edu

**Guillermo Sapiro**
Duke University
guillermo.sapiro@duke.edu

**Miguel Rodrigues**
University College London
m.rodrigues@ucl.ac.uk

## Abstract

Federated learning is an increasingly popular paradigm that enables a large number of entities to collaboratively learn better models. In this work, we study minmax group fairness in paradigms where different participating entities may only have access to a subset of the population groups during the training phase. We formally analyze how this fairness objective differs from existing federated learning fairness criteria that impose similar performance across participants instead of demographic groups. We provide an optimization algorithm – FedMinMax – for solving the proposed problem that provably enjoys the performance guarantees of centralized learning algorithms. We experimentally compare the proposed approach against other methods in terms of group fairness in various federated learning setups.

## 1 Introduction

Machine learning models are being increasingly adopted to make decisions in a range of domains, such as finance, insurance, medical diagnosis, recruitment, and many more [2]. Therefore, we are often confronted with the need – sometimes imposed by regulatory bodies – to ensure that such machine learning models do not lead to decisions that discriminate individuals from a certain demographic group.

The development of machine learning models that are fair across different (demographic) groups has been well studied in traditional learning setups where there is a single entity responsible for learning a model based on a local dataset holding data from individuals of the various groups. However, there are various settings where the data representing different demographic groups is spread across multiple entities rather than concentrated on a single entity/server. For example, consider a scenario where various hospitals wish to learn a diagnostic machine learning model that is fair (or performs reasonably well) across different demographic groups but each hospital may only contain training data from certain groups because – in view of its geo-location – it serves predominantly individuals of a given demographic [4].

These emerging scenarios however bring about various challenges. The first challenge relates to the fact that each individual entity may not be able to learn locally by itself a fair model because it may not hold (or hold little) data from certain demographic groups; The second relates to that fact that each individual entity may also not be able to directly share their own data with other entities due

1st NeurIPS Workshop on New Frontiers in Federated Learning (NFFL 2021), Virtual Meeting.

to legal or regulatory challenges such as GDPR [3]. Therefore, the conventional machine learning fairness *ansatz* – relying on the fact that the learner has access to the overall data – does not generalize from the centralized data setup to the new distributed one.

It is possible to partially address these challenges by adopting federated learning (FL) approaches. These learning approaches enable multiple entities (or clients[1]) coordinated by a central server to iteratively learn in a decentralized manner a single global model to carry out some task [12, 13]. The clients do not share data with one another or with the server; instead the clients only share focused updates with the server, the server then updates a global model, and distributes the updated model to the clients, with the process carried out over multiple rounds or iterations. This learning approach enables different clients with limited local training data to learn better machine learning models.

However, with the exception of [4], which we will discuss later, FL is not typically used to learn models that exhibit performance guarantees for different demographic groups served by a client (i.e. *group fairness* guarantees); instead, it is primarily used to learn models that exhibit specific performance guarantees for each client involved in the federation (i.e. *client fairness* guarantees). Importantly, in view of the fact that a machine learning model that is *client fair* is not necessarily *group fair* (as we later demonstrate formally in this work), it becomes crucial to understand how to develop new federated learning techniques leading up to models that are also fair across different demographic groups.

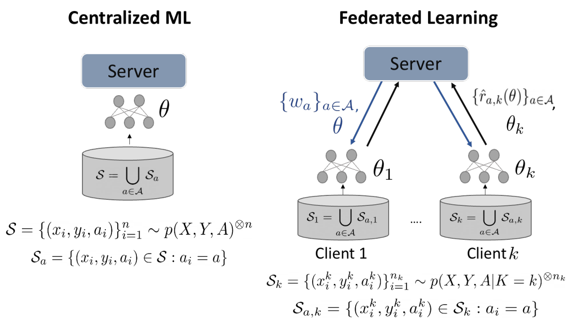


Figure 1: Centralized Learning vs. Federated Learning group fairness. *Left:* A single entity holds the dataset $\mathcal{S}$ in a single server that is responsible for learning a model $h$ parameterized by $\theta$. *Right:* Multiple entities hold different datasets $\mathcal{S}_k$, sharing restricted information with a server that is responsible for learning a model $h$ parameterized by $\theta$. See also Section 3.

This work develops a new federated learning algorithm that can be adopted by multiple entities coordinated by a single server to learn a global (minimax) group fair model. We show that our algorithm leads to the same (minimax) group fairness performance guarantees of centralized approaches such as [6, 18], which are exclusively applicable to settings where the data is concentrated in a single client. Interestingly, this also applies to scenarios where certain clients do not hold any data from some of the groups.

The rest of the paper is organized as follows: Section 2 overviews related work. Section 3 formulates our proposed distributed group fairness problem. Section 4 formally demonstrates that traditional federated learning approaches such as [4, 5, 16, 22] may not always solve group fairness. In Section 5 we propose a new federated learning algorithm to collaboratively learn models that are minimax group fair. Section 6 illustrates the performance of our approach in relation to other baselines. Finally, Section 7 draws various conclusions.

## 2 Related Work

The development of fair machine learning models in the standard *centralized learning setting* – where the learner has access to all the data – is underpinned by fairness criteria. One family of criteria – known as *group fairness* – requires the model to perform similarly on different demographic groups. Popular group fairness criteria include equality of odds, equality of opportunity [9], and demographic parity [17], that are usually imposed as a constraint within the learning problem. More recently, [18] introduced *minimax group fairness*; this criterion requires the model to optimize the prediction performance of the worst demographic group without unnecessarily impairing the performance of other demographic groups (also known as no-harm fairness) [6, 18].

[1]Clients are different user devices, organisations or even geo-distributed datacenters of a single company [11]. In this manuscript we use the terms participants, clients and entities, interchangeably.

The development of fair machine learning models in *federated learning settings* has been building upon the group fairness literature. However, the majority of these works has concentrated predominantly on the development of algorithms leading to models that exhibit similar performance across different clients rather than models that exhibit similar performance across different demographic groups [16].

One such approach is agnostic federated learning (AFL) [22], whose aim is to learn a model that optimizes the performance of the worst performing client. Another FL approach proposed in [16], extends AFL by adding an extra fairness constraint to flexibly control performance disparities across clients. Similarly, tilted empirical risk minimization [14] uses a hyperparameter called tilt to enable fairness or robustness by magnifying or suppressing the impact of individual client losses, respectively. FedMGDA+ is an algorithm that combines minimax optimization coupled with Pareto efficiency [19] and gradient normalization to ensure fairness across users and robustness against malicious clients. See also other related works in [15]. A very recent federated learning work, namely FCFL [4], focuses on improving the worst performing client while ensuring a certain level of local group fairness by employing gradient-based constrained multi-objective optimization in order to address the proposed challenge.

Our work concentrates on developing a federated learning algorithm for guaranteeing fairness across all demographic groups included across clients datasets. It therefore naturally departs from existing federated learning approaches such as AFL [22], FedMGDA+ [10] and $q$-FFL [16] that focus on across-client fairness since, as we prove in Section 4, a model that guarantees client fairness only guarantees group fairness under some special conditions. It also departs from FCFL [4], which considers group fairness to be a per-client objective associated only to the locally available groups. Our primary goal is to learn a model solving (demographic) group fairness across any groups included in the clients distribution, independently of the groups representation in a particular client.

## 3 Problem Formulation

### 3.1 Group Fairness in Centralized Machine Learning

We first describe the standard minimax group fairness problem in a centralized machine learning setting [6, 18], where there is a single entity/server holding all relevant data and responsible for learning a group fair model (see Figure 1). We concentrate on classification tasks, though our approach also applies to other learning tasks such as regression. Let the triplet of random variables $(X, Y, A) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{A}$ represent input features, target, and demographic groups. Let also $p(X, Y, A) = p(A) \cdot p(X, Y|A)$ represent the joint distribution of these random variables where $p(A)$ represents the prior distribution of the different demographic groups and $p(X, Y|A)$ their data conditional distribution.

Let $\ell : \Delta^{|\mathcal{Y}|-1} \times \Delta^{|\mathcal{Y}|-1} \rightarrow \mathbb{R}_+$ be a loss function where $\Delta$ represents the probability simplex. We now consider that the entity will learn an hypothesis $h$ drawn from an hypothesis class $\mathcal{H} = \{h : \mathcal{X} \rightarrow \Delta^{|\mathcal{Y}|-1}\}$, that solves the optimization problem given by

$$\min_{h\in\mathcal{H}} \max_{a\in\mathcal{A}} r_a(h),\, r_a(h) = \mathop{\mathbb{E}}_{(X,Y)\sim p(X,Y|A=a)} [\ell(h(X), Y)|A = a]. \tag{1}$$

Note that this problem involves the minimization of the expected risk of the worst performing demographic group.

Importantly, under the assumption that the loss is a convex function w.r.t the hypothesis[2] and the hypothesis class is a convex set, solving the minimax objective in Eq. 1 is equivalent to solving

$$\min_{h\in\mathcal{H}} \max_{a\in\mathcal{A}} r_a(h) \geq \min_{h\in\mathcal{H}} \max_{\mu\in\Delta^{|\mathcal{A}|-1}_{\geq\epsilon}} \sum_{a\in\mathcal{A}} \mu_a r_a(h). \tag{2}$$

where $\Delta^{|\mathcal{A}|-1}_{\geq\epsilon}$ represent the vectors in the simplex with all of their components larger than $\epsilon$. Note that if $\epsilon = 0$ the inequality in Eq. 2 becomes an equality, however, allowing zero value coefficients may lead to models that are weakly, but not strictly, Pareto optimal [8, 21].

[2] This is true for the most common functions in machine learning settings such as Brier score and cross entropy.

The minimax objective over the linear combination can be achieved by alternating between projected gradient ascent or multiplicative weight updates to optimize the weights, and stochastic gradient descent to optimize the model [1, 6, 18].

### 3.2 Group Fairness in Federated Learning

We now describe our proposed group fairness federated learning problem; this problem differs from the previous one because the data is now distributed across multiple clients but each client (or the server) do not have direct access to the data held by other clients. See also Figure 1.

In this setting, we incorporate categorical variable $K \in \mathcal{K}$ to our data tuple $(X, Y, A, K)$ to indicate the clients participating in the federation. The joint distribution of these variables is $p(X, Y, A, K) = p(K) \cdot p(A|K) \cdot p(X, Y|A, K)$, where $p(K)$ represents a prior distribution over clients – which in practice is the fraction of samples that are acquired by client $K$ relative to the total number of data samples –, $p(A|K)$, and $p(X, Y|A, K)$ represent the distribution of the groups and the distribution of the input and target variables conditioned on a client. We assume that the group-conditional distribution is the same across clients, meaning $p(X, Y|A, K) = p(X, Y|A)$. Note that our model explicitly allows for the distribution of the demographic groups to depend on the client, accommodating for the fact that certain clients may have a higher (or lower) representation of certain demographic groups over others.

We now aim to learn a model $h \in \mathcal{H}$ that solves the minimax fairness problem as presented in Eq. 1, but considering that the group loss estimates are split into $|\mathcal{K}|$ estimators associated to each client. We therefore re-express the linear weighted formulation of Eq. 2 using importance weights, allowing to incorporate the role of the different clients, as follows:

$$\begin{aligned}\min_{h\in\mathcal{H}} \max_{\mu\in\Delta^{|\mathcal{A}|-1}_{\geq\epsilon}} \sum_{a\in\mathcal{A}} \mu_a r_a(h) = \; & \min_{h\in\mathcal{H}} \max_{\mu\in\Delta^{|\mathcal{A}|-1}_{\geq\epsilon}} \sum_{a\in\mathcal{A}} p(A=a) w_a r_a(h) = \\ & \min_{h\in\mathcal{H}} \max_{\mu\in\Delta^{|\mathcal{A}|-1}_{\geq\epsilon}} \sum_{k\in\mathcal{K}} p(K=k) \sum_{a\in\mathcal{A}} p(A=a|K=k) w_a r_a(h) = \\ & \min_{h\in\mathcal{H}} \max_{\mu\in\Delta^{|\mathcal{A}|-1}_{\geq\epsilon}} \sum_{k\in\mathcal{K}} p(K=k) r_k(h, \boldsymbol{w})\end{aligned} \tag{3}$$

where $r_k(h, \boldsymbol{w}) = \sum_{a\in\mathcal{A}} p(A=a|K=k) w_a r_a(h)$ is the expected client risk and $w_a = \mu_a / p(A=a)$ denotes the importance weight for a particular group.

However, there is an immediate non-trivial challenge that arises within this proposed federated learning setting in relation to the centralized one described earlier: we need to devise an algorithm that solves the objective in Eq. 3 under the constraint that the different clients cannot share their local data with the server or with one another, but – in line with conventional federated learning settings [5, 16, 20, 22]– only local client updates of a global model (or other quantities such as local risks) are shared with the server.

## 4 Client Fairness vs. Group Fairness in Federated Learning

Prior proposing a federated learning algorithm to solve our proposed group fairness problem, we first reflect whether a model that solves the more widely used client fairness objective in federated learning settings given by [22]:

$$\min_{h\in\mathcal{H}} \max_{k\in\mathcal{K}} r_k(h) = \min_{h\in\mathcal{H}} \max_{\boldsymbol{\lambda}\in\Delta^{|\mathcal{K}|-1}} \mathop{\mathbb{E}}_{\mathcal{D}_{\boldsymbol{\lambda}}} [\ell(h(X), Y)] \tag{4}$$

where we let $\mathcal{D}_{\boldsymbol{\lambda}} = \sum_{k=1}^{|\mathcal{K}|} \lambda_k p(X, Y|K=k)$ denote a joint data distribution over the clients, also solves our proposed minimax group fairness objective given by:

$$\min_{h\in\mathcal{H}} \max_{a\in\mathcal{A}} r_a(h) = \min_{h\in\mathcal{H}} \max_{\boldsymbol{\mu}\in\Delta^{|\mathcal{A}|-1}} \mathop{\mathbb{E}}_{\mathcal{D}_{\boldsymbol{\mu}}} [\ell(h(X), Y)] \tag{5}$$

where we let $\mathcal{D}_{\boldsymbol{\mu}} = \sum_{a=1}^{|\mathcal{A}|} \mu_a p(X, Y|A=a)$ denote a joint data distribution over sensitive groups.

The following lemma illustrates that a model that is minimax fair with respect to the clients is equivalent to a relaxed minimax fair model with respect to the (demographic) groups.

**Lemma 1.** *Let $\mathbf{P}_\mathcal{A}$ denote a matrix whose entry in row $a$ and column $k$ is $p(A=a|K=k)$ (i.e. the prior of group $a$ in client $k$). Then, given a solution to the minimax problem across clients*

$$h^*, \boldsymbol{\lambda}^* \in \arg\min_{h\in\mathcal{H}} \max_{\boldsymbol{\lambda}\in\Delta^{|\mathcal{K}|-1}} \mathop{\mathbb{E}}_{\mathcal{D}_{\boldsymbol{\lambda}}}[\ell(h(X),Y)], \tag{6}$$

*$\exists\, \boldsymbol{\mu}^* = \mathbf{P}_\mathcal{A}\boldsymbol{\lambda}^*$ that is solution to the following constrained minimax problem across sensitive groups*

$$h^*, \boldsymbol{\mu}^* \in \arg\min_{h\in\mathcal{H}} \max_{\boldsymbol{\mu}\in\mathbf{P}_\mathcal{A}\Delta^{|\mathcal{K}|-1}} \mathop{\mathbb{E}}_{\mathcal{D}_{\boldsymbol{\mu}}}[\ell(h(X),Y)], \tag{7}$$

*where the weighting vector $\boldsymbol{\mu}$ is constrained to belong to the simplex subset defined by $\mathbf{P}_\mathcal{A}\Delta^{|\mathcal{K}|-1} \subseteq \Delta^{|\mathcal{A}|-1}$. In particular, if the set $\Gamma = \left\{\boldsymbol{\mu}' \in \mathbf{P}_\mathcal{A}\Delta^{|\mathcal{K}|-1} \colon \boldsymbol{\mu}' \in \arg\min_{h\in\mathcal{H}} \max_{\boldsymbol{\mu}\in\Delta^{|\mathcal{A}|-1}} \mathop{\mathbb{E}}_{\mathcal{D}_{\boldsymbol{\mu}}}[\ell(h(X),Y)]\right\} \neq \emptyset$, then $\boldsymbol{\mu}^* \in \Gamma$, and the minimax fairness solution across clients is also a minimax fairness solution across demographic groups.*

Lemma 1 proves that being minimax with respect to the clients is equivalent to finding the group minimax model constraining the weighting vectors $\boldsymbol{\mu}$ to be inside the simplex subset $\mathbf{P}_\mathcal{A}\Delta^{|\mathcal{K}|-1}$. Therefore, if this set already contains a group minimax weighting vector, then the group minimax model is equivalent to client minimax model. Another way to interpret this result is that being minimax with respect to the clients is the same as being minimax for any group assignment $\mathcal{A}$ such that linear combinations of the groups distributions are able to generate all clients distributions, and there is a group minimax weighting vector in $\mathbf{P}_\mathcal{A}\Delta^{|\mathcal{N}|-1}$.

Being minimax at the client and group level relies on $\mathbf{P}_\mathcal{A}\Delta^{|\mathcal{K}|-1}$ containing the minimax weighting vector. In particular, if for each sensitive group there is a client comprised entirely of this group ($\mathbf{P}_\mathcal{A}$ contains a identity block), then $\mathbf{P}_\mathcal{A}\Delta^{|\mathcal{K}|-1} = \Delta^{|\mathcal{A}|-1}$ and group and client level fairness are guaranteed to be fully compatible. Another trivial example is when at least one of the client's group priors is equal to a group minimax weighting vector. This result also suggests that client level fairness may also differ from group level fairness. This motivates us to develop a new federated learning algorithm to guarantee group fairness that – where the conditions of the lemma hold – also results in client fairness. We experimentally validate the insights deriving from Lemma 1 in Section 6.

## 5 MinMax Group Fairness Federating Learning Algorithm

We now propose an algorithm – Federated Minimax (FedMinMax) – to solve the group fairness problem in Eq. 3.

We let each client $k$ have access to a dataset $\mathcal{S}_k = \{(x_i^k, y_i^k, a_i^k); i=1,\dots,n_k\}$ containing various data points drawn i.i.d according to $p(X,Y,A|K=k)$. We also define three additional sets: (a) $\mathcal{S}_{a,k} = \{(x_i^k, y_i^k, a_i^k) \in \mathcal{S}_k : a_i = a\}$ is a set containing all data examples associated with group $a$ in client $k$; (b) $\mathcal{S}_a = \bigcup_{k\in\mathcal{K}} \mathcal{S}_{k,a}$ is the set containing all data examples associated with group $a$ across the various clients; and (c) $\mathcal{S} = \bigcup_{a\in\mathcal{A}} \mathcal{S}_a = \bigcup_{k\in\mathcal{K}} \bigcup_{a\in\mathcal{A}} \mathcal{S}_{a,k}$ is containing all data examples across groups and across clients.

Note again that – in view of our modelling assumptions – it is possible that $\mathcal{S}_{a,k}$ can be empty for some $k$ and some $a$ implying that such a client does not have data realizations for such group.

We will also let the model $h$ be parameterized via a vector of parameters $\boldsymbol{\theta} \in \Theta$, i.e. $h(\cdot) = h(\cdot;\boldsymbol{\theta})$. [3] Then, one can approximate the relevant statistical risks appearing in Eq. 3 using empirical risks as follows:

$$\hat{r}_k(\boldsymbol{\theta}, \boldsymbol{w}) = \textstyle\sum_{a\in\mathcal{A}} \frac{n_{a,k}}{n_k} \hat{w}_a \hat{r}_{a,k}(\boldsymbol{\theta}), \quad \hat{r}_a(\boldsymbol{\theta}) = \sum_{k\in\mathcal{K}} \frac{n_{a,k}}{n_a} \hat{r}_{a,k}(\boldsymbol{\theta}) \tag{8}$$

where $\hat{r}_{a,k}(\boldsymbol{\theta}) = \frac{1}{n_{a,k}} \sum_{(x,y)\in\mathcal{S}_{a,k}} \ell(h(x;\boldsymbol{\theta}),y)$, $\hat{w}_a = \mu_a/(n_a/n)$, $n_k = |\mathcal{S}_k|$, $n_a = |\mathcal{S}_a|$, $n_{a,k} = |\mathcal{S}_{a,k}|$, and $n = |\mathcal{S}|$. Note that $\hat{r}_k(\boldsymbol{\theta}, \boldsymbol{w})$ is an estimate of $r_k(\boldsymbol{\theta}, \boldsymbol{w})$, $\hat{r}_a(\boldsymbol{\theta})$ is an estimate of $r_a(\theta)$, and $\hat{r}_{a,k}(\boldsymbol{\theta})$ is an estimate of $r_{a,k}(\theta)$.

We consider the importance weighted empirical risk $\hat{r}_k$ since the clients do not have access to the data distribution but instead to a dataset with finite samples.

[3] This vector of parameters could correspond to the set of weights / biases in a neural network.

Therefore, the clients in coordination with the central server attempt to solve the optimization problem given by:

$$\min_{\boldsymbol{\theta}\in\Theta}\ \max_{\boldsymbol{\mu}\in\Delta_{\geq\epsilon}^{|\mathcal{A}|-1}} \hat{r}_a(\boldsymbol{\theta}) \coloneqq \sum_{a\in\mathcal{A}} \mu_a \hat{r}_a(\boldsymbol{\theta}) \text{ or equivalently, } \min_{\boldsymbol{\theta}\in\Theta}\ \max_{\boldsymbol{\mu}\in\Delta_{\geq\epsilon}^{|\mathcal{A}|-1}} \sum_{k\in\mathcal{K}} \frac{n_k}{n} \hat{r}_k(\boldsymbol{\theta}, \boldsymbol{w}). \tag{9}$$

**Algorithm 1** FEDERATED MINIMAX (FEDMINMAX)

**Input:** $\mathcal{K}$: Set of clients, $T$ : total number of communication rounds, $\eta_{\boldsymbol{\theta}}$: model learning rate, $\eta_{\boldsymbol{\mu}}$: global adversary learning rate, $\mathcal{S}_{a,k}$: set of examples for group $a$ in client $k$, $\forall a \in \mathcal{A}$ and $\forall k \in \mathcal{K}$.

```
Server initializes μ^0 ← ρ = {|S_a|/|S|}_{a∈A} and θ^0 randomly.
for t = 1 to T do
    Server computes w^{t−1} ← μ^{t−1}/ρ
    Server broadcasts θ^{t−1}, w^{t−1}
    for each client k ∈ K in parallel do
        θ_k^t ← θ^{t−1} − η_θ ∇_θ r̂_k(θ^{t−1}, w^{t−1})
        Client-k obtains and sends {r̂_{a,k}(θ^{t−1})}_{a∈A} and θ_k^t to server
    end for
    Server computes: θ^t ← Σ_{k∈K} (n_k/n) θ_k^t
    Server updates
        μ^t ← Π_{Δ^{|A|−1}} ( μ^{t−1} + η_μ ∇_μ ⟨ μ^{t−1}, r̂_a(θ^{t−1}) ⟩ )
end for
```

**Outputs**: $\frac{1}{T}\sum_{t=1}^{T} \boldsymbol{\theta}^t$

The objective in Eq. 9 can be interpreted as a zero-sum game between two players: the learner aims to minimize the objective by optimizing the model parameters $\boldsymbol{\theta}$ and the adversary seeks to maximize the objective by optimizing the weighting coefficients $\boldsymbol{\mu}$.

We use a non-stochastic variant of the stochastic-AFL algorithm introduced in [22]. Our version, provided in Algorithm 1, assumes that all clients are available to participate in each communication round $t$. In particular, in each round $t$, the clients receive the latest model parameters $\boldsymbol{\theta}^{t-1}$, the clients then perform one gradient descent step using all their available data, and the clients then share the updated model parameters along with certain empirical risks with the server. The server (learner) then performs a weighted average of the client model parameters $\boldsymbol{\theta}^t = \sum_{k\in\mathcal{K}} \frac{n_k}{n}\boldsymbol{\theta}_k^t$. The server also updates the weighting coefficient using a projected gradient ascent step in order to guarantee that the weighting coefficient updates are consistent with the constraints. We use the Euclidean algorithm proposed in [7] in order to implement the projection operation ($\prod_{\Delta^{|\mathcal{A}|-1}}(\cdot)$).

We can also show that our algorithm can exhibit convergence guarantees.

**Lemma 2.** *Consider our federated learning setting (Figure 1, right) where each entity $k$ has access to a local dataset $\mathcal{S}_k = \bigcup_{a\in\mathcal{A}} \mathcal{S}_{a,k}$ and a centralized machine learning setting (Figure 1, left) where there is a single entity that has access to a single dataset $\mathcal{S} = \bigcup_{k\in\mathcal{K}} \mathcal{S}_k = \bigcup_{k\in\mathcal{K}} \bigcup_{a\in\mathcal{A}} \mathcal{S}_{a,k}$ (i.e. this single entity in the centralized setting has access to the data of the various clients in the distributed setting). Then, Algorithm 1 and Algorithm 2 (in supplementary material, Appendix B) lead to the same global model provided that learning rates and model initialization are identical.*

This lemma shows that our federated learning algorithm inherits any convergence guarantees of existing centralized machine learning algorithms. In particular, assuming that one can model the single gradient descent step using a $\delta$-approximate Bayesian Oracle [1], we can show that a centralized algorithm converges and hence our FedMinMax one also converges too (under mild conditions on the loss function, hypothesis class, and learning rates). See Theorem 7 in [1].

## 6 Experimental Results

To validate the benefits of the proposed FedMinMax approach, we consider three federated learning scenarios: (1) *Equal access to Sensitive Groups (ESG)*, where every client has access to all sensitive groups but does not have enough data to train a model individually, to examine a case where group and client fairness are not equivalent; (2) *Partial access to Sensitive Groups (PSG)* where each client has access to a subset of the available groups memberships, to compare the performances when there is low or no local representation of particular groups; (3) *access to a Single Sensitive Group (SSG)*, each client holds data from one sensitive group, for showcasing the group and client fairness objectives equivalence derived from Lemma 1.

| Setting | Method | Worst Group Risk | Best Group Risk |
|---|---|---|---|
| ESG | AFL | 0.485±0.0 | 0.216±0.001 |
| | FedAvg | 0.487±0.0 | 0.214±0.002 |
| | $q$-FedAvg ($q$=0.2) | 0.479±0.002 | 0.22±0.002 |
| | $q$-FedAvg ($q$=5.0) | 0.478±0.002 | 0.223±0.004 |
| | FedMinMax (ours) | **0.451±0.0** | **0.31±0.001** |
| SSG | AFL | **0.451±0.0** | **0.31±0.001** |
| | FedAvg | 0.483±0.002 | 0.219±0.001 |
| | $q$-FedAvg ($q$=0.2) | 0.476±0.001 | 0.221±0.002 |
| | $q$-FedAvg ($q$=5.0) | 0.468±0.005 | 0.274±0.004 |
| | FedMinMax (ours) | **0.451±0.0** | **0.309±0.003** |
| Centalized Minmax Baseline | | **0.451±0.0** | **0.308±0.001** |

Table 1: Testing brier score risks for FedAvg, AFL, $q$-FedAvg and FedMinmax across different federated learning scenarios on the synthetic dataset for binary classification involving two sensitive groups. PSG scenario is not included because for $|\mathcal{A}| = 2$ it is equivalent to SSG.

In all experiments we consider a federation consisting of 40 clients and a single server that orchestrates the training procedure per Algorithm 1. We benchmark our approach against AFL [22], $q$-FedAvg [16], and FedAvg [20]. Further, as a baseline, we also run FedMinMax with one client (akin to centralized ML) to confirm Lemma 2.

We generated a synthetic dataset for binary classification involving two sensitive groups (i.e. $|\mathcal{A}| = 2$), details available in Appendix B. We provide the performance on ESG and SSG scenarios[4] in Table 1. FedMinMax performs similarly to Centralized Minmax Baseline for both sensitive groups in all setups, as proved in Lemma 2. AFL yields the same solution as FedMinMax and Centralized Minmax Baseline only in SSG where group fairness is implied by client fairness. Both FedAvg and $q$-FedAvg fail to achieve minimax group fairness.

| Setting | Method | T-shirt | Pullover | Shirt |
|---|---|---|---|---|
| ESG | AFL | 0.239±0.003 | 0.262±0.001 | 0.494±0.004 |
| | FedAvg | 0.243±0.003 | 0.262±0.001 | 0.492±0.003 |
| | FedMinMax (ours) | 0.261±0.006 | 0.256±0.027 | **0.307±0.01** |
| SSG | AFL | 0.267±0.009 | 0.236±0.013 | **0.307±0.003** |
| | FedAvg | 0.227±0.003 | 0.236±0.004 | 0.463±0.003 |
| | FedMinMax (ours) | 0.269±0.012 | 0.238±0.017 | **0.309±0.011** |
| PSG | AFL | 0.244±0.007 | 0.257±0.066 | 0.425±0.019 |
| | FedAvg | 0.229±0.008 | 0.236±0.004 | 0.464±0.011 |
| | FedMinMax (ours) | 0.263±0.013 | 0.228±0.011 | **0.31±0.008** |
| Centalized Minmax Baseline | | 0.259±0.01 | 0.239±0.051 | **0.311±0.006** |

Table 2: Testing brier score risks for the top-3 worst groups in FashionMNIST dataset. The risks for all the available classes are available in Table 4, in Appendix B.

For FashionMNIST we use all ten clothing target categories, which we assign both as targets and sensitive groups (i.e. $|\mathcal{A}| = 10$). In Table 2 we examine the performance on the three worst categories: *T-shirt*, *Pullover*, and *Shirt* (the risks for all classes are available in Table 4, in Appendix B). In all settings, FedMinMax improves the risk of the worst group, *Shirt*, more than it degrades the performance on the *T-shirt* class, all while maintaining the same risk on *Pullover* as FedAvg. AFL performs akin to FedMinMax for the SSG setup but not on the other settings as expected by Lemma 1. Note that FedMinMax has the best worst group performance in all settings as expected. More details about datasets, models, experiments and results are provided in Appendix B.

## 7 Conclusion

In this work, we formulate (demographic) group fairness in federated learning setups where different participating entities may only have access to a subset of the population groups during the training phase (but not necessarily the testing phase), exhibiting minmax fairness performance guarantees akin to those in centralized machine learning settings.

We formally show how our fairness definition differs from the existing fair federated learning works, offering conditions under which conventional client-level fairness is equivalent to group-level fairness. We also provide an optimization algorithm, FedMinMax, to solve the minmax group fairness problem in federated setups that exhibits minmax guarantees akin to those of minmax group fair centralized machine learning algorithms. We empirically confirm that our method outperforms existing federated learning methods in terms of group fairness in various learning settings and validate the conditions that the competing approaches yield the same solution as our objective.

[4]Note that PSG scenario is valid only for datasets where $|\mathcal{A}| > 2$, else its equivalent to SSG setting.

## Acknowledgments

University College London was supported by Cisco under grand #217462. Duke University was partially supported by the Department of Defense, NSF, NIH, and gifts from Cisco, Google, Amazon.

## References

[1] Robert S. Chen, Brendan Lucier, Yaron Singer, and Vasilis Syrgkanis. Robust optimization for non-convex objectives. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[2] Alexandra Chouldechova and Aaron Roth. A snapshot of the frontiers of fairness in machine learning. *Commununications of the ACM*, 63(5):82–89, April 2020.

[3] European Commission. Reform of eu data protection rules 2018. `https://ec.europa.eu/commission/sites/beta-political/files/data-protection-factsheet-changes_en.pdf`.

[4] Sen Cui, Weishen Pan, Jian Liang, Changshui Zhang, and Fei Wang. Fair and consistent federated learning. *CoRR*, abs/2108.08435, 2021.

[5] Yuyang Deng, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. Distributionally robust federated averaging. *Advances in Neural Information Processing Systems*, 33, 2020.

[6] Emily Diana, Wesley Gill, Michael Kearns, Krishnaram Kenthapadi, and Aaron Roth. Convergent algorithms for (relaxed) minimax fairness. *arXiv preprint arXiv:2011.03108*, 2020.

[7] John Duchi, Shai Shalev-Shwartz, Yoram Singer, and Tushar Chandra. Efficient projections onto the l1-ball for learning in high dimensions. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, page 272–279, New York, NY, USA, 2008. Association for Computing Machinery.

[8] Arthur M Geoffrion. Proper efficiency and the theory of vector maximization. *Journal of mathematical analysis and applications*, 22(3):618–630, 1968.

[9] Moritz Hardt, Eric Price, and Nathan Srebro. Equality of opportunity in supervised learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 3323–3331, Red Hook, NY, USA, 2016. Curran Associates Inc.

[10] Zeou Hu, Kiarash Shaloudegi, Guojun Zhang, and Yaoliang Yu. Fedmgda+: Federated learning meets multi-objective optimization. *CoRR*, abs/2006.11489, 2020.

[11] Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, Rafael G. L. D'Oliveira, Salim El Rouayheb, David Evans, Josh Gardner, Zachary Garrett, Adrià Gascón, Badih Ghazi, Phillip B. Gibbons, Marco Gruteser, Zaïd Harchaoui, Chaoyang He, Lie He, Zhouyuan Huo, Ben Hutchinson, Justin Hsu, Martin Jaggi, Tara Javidi, Gauri Joshi, Mikhail Khodak, Jakub Konecný, Aleksandra Korolova, Farinaz Koushanfar, Sanmi Koyejo, Tancrède Lepoint, Yang Liu, Prateek Mittal, Mehryar Mohri, Richard Nock, Ayfer Özgür, Rasmus Pagh, Mariana Raykova, Hang Qi, Daniel Ramage, Ramesh Raskar, Dawn Song, Weikang Song, Sebastian U. Stich, Ziteng Sun, Ananda Theertha Suresh, Florian Tramèr, Praneeth Vepakomma, Jianyu Wang, Li Xiong, Zheng Xu, Qiang Yang, Felix X. Yu, Han Yu, and Sen Zhao. Advances and open problems in federated learning. *CoRR*, abs/1912.04977, 2019.

[12] Jakub Konecný, H. Brendan McMahan, Daniel Ramage, and Peter Richtárik. Federated optimization: Distributed machine learning for on-device intelligence. *CoRR*, abs/1610.02527, 2016.

[13] Jakub Konecný, H. Brendan McMahan, Felix X. Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. In *NeurIPS Workshop on Private Multi-Party Machine Learning*, 2016.

[14] Tian Li, Ahmad Beirami, Maziar Sanjabi, and Virginia Smith. Tilted empirical risk minimization. In *International Conference on Learning Representations*, 2021.

[15] Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Federated multi-task learning for competing constraints. *CoRR*, abs/2012.04221, 2020.

[16] Tian Li, Maziar Sanjabi, Ahmad Beirami, and Virginia Smith. Fair resource allocation in federated learning. In *International Conference on Learning Representations*, 2020.

[17] Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard S. Zemel. The variational fair autoencoder. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016.

[18] Natalia Martinez, Martin Bertran, and Guillermo Sapiro. Minimax pareto fairness: A multi objective perspective. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6755–6764. PMLR, 13–18 Jul 2020.

[19] Andreu Mas-Colell, Michael Whinston, and Jerry Green. *Microeconomic Theory*. Oxford University Press, 1995.

[20] H. Brendan McMahan, Eider Moore, Daniel Ramage, and Blaise Agüera y Arcas. Federated learning of deep networks using model averaging. *CoRR*, abs/1602.05629, 2016.

[21] Kaisa Miettinen. *Nonlinear Multiobjective Optimization*, volume 12. Springer Science & Business Media, 2012.

[22] Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. Agnostic federated learning. In *36th International Conference on Machine Learning, ICML 2019*, 36th International Conference on Machine Learning, ICML 2019, pages 8114–8124. International Machine Learning Society (IMLS), January 2019. 36th International Conference on Machine Learning, ICML 2019 ; Conference date: 09-06-2019 Through 15-06-2019.

[23] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.