SEMI-VARIANCE REDUCTION FOR FAIR FEDERATED LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Ensuring fairness in Federated Learning (FL) systems, i.e. a satisfactory performance for all of the diverse clients in the systems, is an important and challenging problem. There are multiple fair FL algorithms in the literature, which have been relatively successful in providing fairness. However, these algorithms mostly emphasize on the loss functions of worst-off clients to improve their performance, which often results in the suppression of well-performing ones. As a consequence, they usually sacrifice the system overall average performance for achieving fairness. Motivated by this and inspired by two well-known risk modeling methods in Finance, *Mean-Variance* and *Mean-Semi-Variance*, we propose and study two new fair FL algorithms, *Variance Reduction* (VRed) and *Semi-Variance Reduction* (Semi-VRed). VRed encourages equality between clients loss functions by penalizing their variance. In contrast, Semi-VRed penalizes the discrepancy of only the worst-off clients loss functions from the average loss. Through extensive experiments on multiple vision and language datasets, we show that, Semi-VRed achieves SoTA performance in scenarios with highly heterogeneous data distributions and improves both fairness and system overall average performance.

1 INTRODUCTION

Federated Learning McMahan et al. (2017) is a framework consisting of some clients and the private data that is distributed among them, and it allows training of a shared or personalized model based on the clients data. Since the invention of FL by proposing the well-known FedAvg algorithm (McMahan et al., 2017), it has attracted an intensive amount of attention and much progress has been made in its different aspects, including algorithmic innovations (Li et al., 2020b; Reddi et al., 2020a; Pathak & Wainwright, 2020; Huo et al., 2020; Wang et al., 2020; Reddi et al., 2020b; Qu et al., 2022), fairness (McMahan et al., 2017; Li et al., 2020c; Mohri et al., 2019; Li et al., 2020a; Yue et al., 2021; Zhang et al., 2022a), convergence analysis (Khaled et al., 2020; Li et al., 2020; Gorbunov et al., 2021), personalization (Zhang et al., 2021; Chen & Chao, 2022; Oh et al., 2022; Zhang et al., 2022b; Bietti et al., 2022).

Due to heterogeneity in clients data and their resources, performance fairness is an important challenge in FL systems. There have been some previous works addressing this problem. For instance, Mohri et al. (2019) proposed Agnostic Federated Learning (AFL), which aims at minimizing the largest loss function among clients through a minimax optimization framework. Similarly, Li et al. (2020a) proposed an algorithm called TERM using tilted losses. Ditto (Li et al., 2021) is another existing algorithm based on model personalization for clients[1]. Also, q-Fair Federated Learning (q-FFL) (Li et al., 2020c) is an algorithm inspired by α-fairness in wireless networks (Lan et al., 2010). Recently, Zhang et al. (2022a) proposed PropFair based on the concept of Proportional Fairness (PF). Interestingly, they also showed that all the aforementioned fair FL algorithms can be unified into a generalized mean framework. GiFair (Yue et al., 2021) is another recent algorithm which achieves fairness using a different mechanism than the previously mentioned algorithms: by penalizing the discrepancy between clients loss functions, i.e. encouraging equality.

Being designed for fair FL, the aforementioned algorithms usually result in the suppression of well-performing clients, due to the lower weights that the algorithms place on them or due to the equality

[1]In order to have fair comparison with our baseline algorithms, we do not use model personalization in this work.

that is encouraged between clients losses (GiFair). As a consequence, they achieve an overall average performance which is either smaller than or close to that of vanilla FedAvg. This is our motivation for proposing our two new algorithms.

Our inspiration in this paper is a concept in Finance called *risk modeling* used for portfolio selection. There are two vastly used methodologies for risk modeling: *Mean-Variance* (MV) (Zhang et al., 2018; Soleimani et al., 2009; Markowitz, 1952) and its expansion: *Mean-Semi-Variance* (MSV) (Boasson et al., 2017; Plà-Santamaria & Bravo, 2013; Ballestero, 2005; Stuart & Markowitz, 1959), which are used for quantifying investment return and investment risk. Motivated by the vast usage of these methodologies and their great success in financial planning, we bring the MV and MSV methods to FL by proposing our *Variance Reduction* (VRed) and *Semi-Variance Reduction* (Semi-VRed) algorithms, respectively. By conducting extensive experiments on popular vision and language datasets, we show that our VRed algorithm achieves a performance competitive to existing baseline fair FL algorithms. More importantly, Semi-VRed achieves state-of-the-art performance in terms of both fairness and system overall average performance.

2 BACKGROUND

With formal notations, we consider an FL setting with n clients for the task of multi-class classification. Let $x \in \mathcal{X} \subseteq \mathbb{R}^p$ and $y \in \mathcal{Y} = \{1, \ldots, C\}$ denote the input data point and its target label, respectively. Each client i has its own private data with data distribution $P_i(x, y)$. Let $h : \mathcal{X} \times \Theta \to \mathbb{R}^C$ be the used predictor function, which is parameterized by $\theta \in \Theta \subseteq \mathbb{R}^d$ shared among all clients. Also, let $\ell : \mathbb{R}^C \times \mathcal{Y} \to \mathbb{R}_+$ be the loss function, which we choose to be the cross entropy loss. Client i minimizes its local loss function $f_i(\theta)$, defined as $f_i(\theta) = \mathbb{E}_{(x,y)\sim P_i(x,y)}[\ell(h(x, \theta), y)]$.

There are various fair FL algorithms in the literature. In table 3 in the appendix, we have provided details of the most recent algorithms with their formulations. The existing fair FL algorithms can be grouped into two main categories:

2.1 ALGORITHMS BASED ON THE GENERALIZED MEAN

This category of algorithms includes FedAvg (McMahan et al., 2017), q-FFL (Li et al., 2020c), AFL (Mohri et al., 2019), TERM (Li et al., 2020a), PropFair (Zhang et al., 2022a). It was shown by Zhang et al. (2022a) that this set of existing fair FL algorithms can be unified into a generalized mean framework (Kolmogorov, 1930), where more attention is paid to the clients with larger losses.

2.2 ALGORITHMS BASED ON ENCOURAGING EQUALITY

The second category of fair FL algorithms, which includes GiFair, is based on encouraging equality between clients loss functions. GiFair adds a regularization term to the objective of FedAvg to penalize the discrepancy between clients loss functions (see table 3 in the appendix). In this way, it encourages equality between clients loss functions.

A common feature of all the aforementioned algorithms is their emphasis on the clients with relatively larger losses, which usually results in the suppression of the well-performing clients. This might result in the degradation of the overall average performance (measured by the mean test accuracy across clients). In the next sections, we will see that Semi-VRed can achieve fairness by regularizing the *semi-variance* of clients loss functions and improves both fairness *and* the system overall performance simultaneously. In the context of variance regularization, there have been some works in the literature: Maurer & Pontil (2009); Namkoong & Duchi (2017) propose regularizing the empirical risk minimization (ERM) by the empirical variance of losses across training samples to balance bias and variance and improve out-of-sample (test) performance and convergence rate. Similarly, Shivaswamy & Jebara (2010) propose boosting binary classifiers based on a variance penalty applied to exponential loss. Variance regularization has also been used for out-of-distribution (domain) generalization: assuming having access to data from multiple training domains, Krueger et al. (2021) propose penalizing variance of training risks across the domains, as a method of distributionally robust optimization, to provide domain generalization.

3 RISK MODELING METHODS IN FINANCE: *Mean-Variance* AND *Mean-Semi-Variance*

Mean-Variance (MV) and *Mean-Semi-Variance* (MSV) have been the most popular methods for Modeling risks and gains of an investment portfolio, which is a first step in financial planning.

***Mean-Variance* (MV)** (Zhang et al., 2018; Soleimani et al., 2009; Markowitz, 1952). This method treats the return of each security in an investment portfolio as a random variable and adopts its expected value and variance to quantify the return and risk of the portfolio, respectively. An investor either minimizes the risk for a fixed expected return level or maximizes the return for a given acceptable risk level. For instance, in the former case, MV results in the following problem:

$$\begin{aligned} \max_{x_1,\ldots,x_n} \quad & \mathbb{E}[x_1 S_1 + \ldots + x_n S_n] \\ \text{s.t.} \quad & \sigma^2[x_1 S_1 + \ldots + x_n S_n] \leq R, \quad \sum_i x_i = 1, \quad x_i \geq 0. \end{aligned} \tag{1}$$

Here $\mathbb{E}$ and σ^2 denote the expected value and variance operators, respectively. Also, S_i and x_i denote the random return from security i and the proportion of total wealth invested in security i, respectively. This example has provided a basic view of how MV model works. Other closely related measures of risk in the MV model include the standard deviation (σ) and coefficient of variation ($\sigma/\mathbb{E}$). However, the *Mean-Variance* modeling of risk is debatable: any uncertain return above the expectation is usually not considered as risk in the common sense, but the MV model does so. This shortcoming is resolved by the *Mean-Semi-Variance* model.

***Mean-Semi-Variance* (MSV)** (Boasson et al., 2017; Plà-Santamaria & Bravo, 2013; Ballestero, 2005; Stuart & Markowitz, 1959). Having recognized the importance of the (often) one-side nature of risks, MSV model proposed a *downside* risk measure known as *semi-variance*, which we denote by $\sigma^2_<$. Unlike variance, it is only concerned with the downside of the return. i.e. only the cases that the return drops *below* a predefined threshold. With this risk modeling method, problem 1 changes to the following:

$$\begin{aligned} \max_{x_1,\ldots,x_n} \quad & \mathbb{E}[x_1 S_1 + \ldots + x_n S_n] \\ \text{s.t.} \quad & \sigma^2_<[x_1 S_1 + \ldots + x_n S_n] \leq R, \quad \sum_i x_i = 1, \quad x_i \geq 0, \end{aligned} \tag{2}$$

where the operator semi-variance measures the *downsides* of the return: $\sigma^2_<[z] = \mathbb{E}[(\mathbb{E}[z] - z)^2_+]$. MSV is a preferable alternative for the MV model as its modeling of the risk is more consistent with our perception from an investment risk. Again, the problem above gives a basic understanding of how the MSV model works. More complex variations of MV and MSV models have been developed for complex and unpredictable financial markets (Rigamonti & Lučivjanská, 2022; Zhang et al., 2018; Ballestero, 2005).

4 MV AND MSV MODELS FOR FAIR FL

In this section, we propose two fair FL algorithms based on the MV and MSV models. We use the two models to quantify the inequality between clients performances. Inspired by Zhang et al. (2022a), we take a simple approach and define $u_i(\theta) = M - f_i(\theta)$ as the utility of client i, where M can be treated as a utility baseline. The smaller the loss function of a client becomes, the larger its utility becomes: *the utility of a client can be used to roughly represent the test accuracy of the shared model, parameterized by θ, on its local data*. With this definition, we propose to model the inequality between clients by the variance and semi-variance of their utilities, resulting in the VRed and Semi-VRed algorithm, respectively.

Algorithm 1: `VRed` and `Semi-VRed`

Input: global epoch T, client number n, loss function f_i, number of samples n_i for client i, , number of total samples N, initial global model θ_0, local step number K_i for client i, learning rate η

Let $p_i = \frac{n_i}{N}$ for $i \in \{0, 1, \ldots, n-1\}$
for $t = 0, 1 \ldots T-1$ **do**
  randomly select $\mathcal{S}_t \subseteq [n]$
  $\theta_t^{(i)} = \theta_t$ for $i \in \mathcal{S}_t$, $N = \sum_{i \in \mathcal{S}_t} n_i$
  for i *in* $\mathcal{S}_t$ **do** `// in parallel`
    starting from $\theta_t^{(i)}$, take K_i local SGD steps on $f_i(\theta_t^{(i)})$ with learning rate η to find $\theta_{t+1}^{(i)}$
    compute $\Delta_i^{(t)} = \theta_t^{(i)} - \theta_{t+1}^{(i)}$
  compute $\overline{f}(\theta) = \sum_i p_i f_i(\theta)$ and $\overline{\Delta}^{(t)} = \sum_i p_i \Delta_i^{(t)}$
  if `VRed` **then**
    compute $\Delta_t = \sum_i p_i \Delta_i^{(t)} + 2\beta \sum_i p_i (f_i(\theta) - \overline{f}(\theta))(\Delta_i^{(t)} - \overline{\Delta}^{(t)})$
  else if `Semi-VRed` **then**
    compute $\Delta_t = \sum_i p_i \Delta_i^{(t)} + 2\beta \sum_i p_i (f_i(\theta) - \overline{f}(\theta))_+(\Delta_i^{(t)} - \overline{\Delta}^{(t)})$
  update $\theta_{t+1} = \theta_t - \Delta_t$

Output: global model θ_T

4.1 THE VRED ALGORITHM

`VRed` models the inequality between clients utilities by their variance and aims to minimize the following objective function:

$$\min_\theta \; F(\theta) = \mathbb{E}[\{f_i(\theta)\}_{i=1}^n] + \beta\sigma^2[\{u_i(\theta)\}_{i=1}^n] = \sum_i p_i f_i(\theta) + \beta \sum_i p_i \left(u_i(\theta) - \sum_j p_j u_j(\theta) \right)^2$$
$$= \sum_i p_i f_i(\theta) + \beta \sum_i p_i \left(f_i(\theta) - \sum_j p_j f_j(\theta) \right)^2 . \tag{3}$$

This objective, in addition to minimizing the vanilla `FedAvg` mean loss, reduces the variance of clients utilities. Let us derive the `VRed` federated learning algorithm. By taking the gradient of equation 3 and multiplying it by the step size η, we have:

$$\eta \nabla F(\theta) = \sum_i p_i \eta \nabla f_i(\theta) + 2\beta \sum_i p_i \Big(f_i(\theta) - \sum_j p_j f_j(\theta) \Big) \Big(\eta \nabla f_i(\theta) - \sum_j p_j \eta \nabla f_j(\theta) \Big). \tag{4}$$

This equation immediately leads to an `FL` algorithm, by replacing the gradient $\eta \nabla f_i(\theta)$ with the pseudo-gradient (i.e., the opposite of the local update), denoted by $\Delta_i^{(t)}$:

$$\eta \nabla F(\theta) = \sum_i p_i \Delta_i^{(t)} + 2\beta \sum_i p_i \Big(f_i(\theta) - \overline{f}(\theta) \Big) \Big(\Delta_i^{(t)} - \overline{\Delta}^{(t)} \Big), \tag{5}$$

where $\overline{f}(\theta) = \sum_i p_i f_i(\theta)$ and $\overline{\Delta}^{(t)} = \sum_i p_i \Delta_i^{(t)}$. The corresponding algorithm is included in algorithm 1. There is a parameter β which tunes the effect of the regularization term, which needs to get tuned for better performance. Note that this is a new aggregation rule: instead of simply averaging the local models, it has an additional second term, which relates to the variance of clients losses. If all clients are identical, this term would vanish.

4.1.1 AN INTERPRETATION OF VRED

With the definition of utilities in the previous section ($u_i(\theta) = M - f_i(\theta)$), the objective function of `VRed` algorithm (equation 3) is aimed to penalize the variance of clients utilities. One potential drawback of this is that it might result in the suppression of well-performing clients (the ones

with small losses) for reducing the variance, which is the same drawback that GiFair (Yue et al., 2021) had. Hence, the final model overall performance averaged across clients might get sacrificed for achieving fairness. In fact, GiFair minimizes an upper bound of VRed objective function: assuming $p_i = \frac{1}{n}$ $(i = 1, \dots, n)$, i.e., all clients have the same number of data points, we have the following upper bound on VRed objective function (see equation 17 in the appendix for derivation):

$$\sum_i f_i + \beta \sum_i \left| f_i(\theta) - \frac{1}{n} \sum_j f_j(\theta) \right|^2 \leq \sum_i f_i + \frac{2\beta}{n^2} \sum_{j \neq i} \left| f_i(\theta) - f_j(\theta) \right|^2, \tag{6}$$

where the right hand side is the same as GiFair objective function (except the power 2 used for measuring the pairwise distances between clients losses, see table 3 in the appendix). Therefore, GiFair in fact minimizes an upper bound of VRed's objective function. This might be the reason explaining why our VRed usually outperforms GiFair in terms of fairness in our experiments.

In typical FL settings, the global objective function can be written as a weighted sum of clients loss functions, i.e. $F(\theta) := \sum_{i=1}^n w_i h_i(\theta)$, where $h_i(\theta)$ is used by client i as a surrogate of the global objective and is optimized using the client local data. Also, the weight w_i represents the importance of client i loss function in the global objective function $F(\theta)$. For example, FedAvg simply uses $h_i(\theta) = f_i(\theta)$ and $w_i = p_i$ ($p_i = \frac{n_i}{N}$, see algorithm 1) and q-FFL uses $h_i(\theta) = f_i^{q+1}(\theta)$ and $w_i = p_i$. A direct consequence of the above summation form for $F(\theta)$ is:

$$\nabla F(\theta) = \sum_{i=1}^n w_i \nabla h_i(\theta), \tag{7}$$

Again, the weight w_i represents the importance of the client i model updates. In lemma 1, we show that the gradient of the global objective of VRed in equation 3, can be written in the form of equation 7. *For simplicity and easier interpretation, we assume $p_i = \frac{1}{n}$ $(i = 1, \dots, n)$, i.e., all clients have the same number of data points.*

Lemma 1. *For any model parameter θ, the gradient of the global objective $F(\theta)$ defined in equation 3 can be expressed as*

$$\nabla F(\theta) = \sum_{i=1}^n w_i(\theta) \nabla f_i(\theta), \;\; w_i(\theta) = \frac{1}{n} + \frac{2\beta(f_i(\theta) - \overline{f}(\theta))}{n}, \;\; \overline{f}(\theta) = \frac{\sum_i f_i(\theta)}{n}. \tag{8}$$

The proof is deferred to §A in the appendix. Lemma 1 shows that, unlike FedAvg that would assign $w_i = \frac{1}{n}, i = 1, \dots, n$ to all clients, VRed assigns a relatively larger weight (w_i) to clients with larger loss functions, and dynamically updates the weights w_i at each communication round. We will provide an interpretation of this finding about VRed in the next sections. Importantly, based on equation 8, in order for all clients to get assigned a positive weight, the parameter β needs to satisfy the following inequalities: $0 \leq \beta < \beta_{\texttt{VRed}}^{max}(\theta) \triangleq \frac{1}{2(\overline{f}(\theta) - \min_i \{f_i(\theta)\})}$.

4.2 THE SEMI-VRED ALGORITHM

Inspired by the discussion on the superiority of MSV over MV in § 3 for risk modeling, we propose an extension of VRed. Consider the following objective function instead of equation 3:

$$\min_\theta F(\theta) = \mathbb{E}[\{f_i(\theta)\}_{i=1}^n] + \beta \sigma_<^2[\{u_i(\theta)\}_{i=1}^n] = \sum_i p_i f_i(\theta) + \beta \sum_i p_i \left(f_i(\theta) - \sum_j p_j f_j(\theta) \right)_+^2 \tag{9}$$

where $\sigma_<^2$ denotes the semi-variance of clients utilities. This objective, in addition to minimizing the mean loss, *reduces the semi-variance of clients losses*, meaning that only those clients that have relatively small utilities $u_i(\theta)$ (or equivalently large losses $f_i(\theta)$) contribute to the semi-variance regularization term in eq. (9).

Similar to what we did for VRed, if we take the gradient of equation 9, we have:

$$\eta \nabla F(\theta) = \sum_i p_i \Delta_i^{(t)} + 2\beta \sum_i p_i \Big(f_i(\theta) - \overline{f}(\theta)\Big)_+ \Big(\Delta_i^{(t)} - \overline{\Delta}^{(t)}\Big), \tag{10}$$

where $\Delta_i^{(t)}$ is the pseudo-gradient (i.e., the opposite of the local update) of user i. The corresponding algorithm is included in algorithm 1. Again, we have a tunable parameter β which sets the effect of the regularization term and needs to get tuned for better performance.

4.3 Can we interpret what Semi-VRed does?

4.3.1 Optimization aspect

We will show in lemma 2 that, in contrast to VRed (and GiFair) and thanks to its more efficient formulation, Semi-VRed does not suppress the well-performing clients to help the worst-off ones. *Similar to lemma 1, we assume $p_i = \frac{1}{n}$ $(i = 1, \dots, n)$,* for simplicity and easier interpretation.

Lemma 2. *In each communication round between the clients and the server, let $>_C$ denote the set of clients whose local loss function is greater than the average loss function $\overline{f}(\theta)$. For any model parameter θ, the gradient of the global objective $F(\theta)$ defined in equation 9 can be expressed as*

$$\nabla F(\theta) = \sum_{i=1}^{n} w_i(\theta) \nabla f_i(\theta), \tag{11}$$

where:

$$\overline{f}(\theta) = \frac{\sum_i f_i(\theta)}{n}, \quad w_i(\theta) = \begin{cases} \dfrac{1}{n} + \dfrac{2\beta(f_i(\theta) - \overline{f}(\theta))}{n} - \dfrac{2\beta \sum_{j \in >_C} (f_j(\theta) - \overline{f}(\theta))}{n^2}, \text{ if } i \in >_C \\ \dfrac{1}{n} - \dfrac{2\beta \sum_{j \in >_C} (f_j(\theta) - \overline{f}(\theta))}{n^2}, \text{ if } i \notin >_C \end{cases} \tag{12}$$

Similar to VRed, there is an upper-bound for β to ensure positive weights for all clients in equation 12: $0 \leq \beta < \beta^{max}_{\texttt{Semi-VRed}}(\theta) \triangleq \frac{n}{2\sum_{j \in >_C} (f_j(\theta) - \overline{f}(\theta))}$.

Remark 1. *Interesting points can be observed by comparing lemma 1 and lemma 2. First, both of the algorithms pay more attention to worst-off clients by assigning larger weights to their gradients. However, Semi-VRed assigns relatively larger weights to the well-performing clients. Also, for VRed, $w_i(\theta) = \frac{1}{n} + \frac{2\beta(f_i(\theta) - \overline{f}(\theta))}{n}$, so the better a client performs, the more it is suppressed by the algorithm. In contrast Semi-VRed assigns weights to well-performing clients depending on how bad the worst-off clients perform compared to the average performance (equation 12). As the performance of worst-off clients improves gradually, the algorithm also lets the well-performing ones for further improvement, instead of strictly suppressing them like VRed.*

4.3.2 Handling extreme label shifts

We now provide another interesting interpretation of Semi-VRed, related to data heterogeneity in FL. In order for an easier interpretability, we assume $P_i(x, y) = P_i(x|y)P_i(y) = P(x|y)P_i(y)$. This means that the class conditional distribution of input x is identical for all clients. Having made this assumption, we define $\overline{\ell}_j(\theta) = \mathbb{E}_{x \sim P(x|y=j)}[\ell(h(x, \theta), j)]$ as the average loss of predictor h on class j. We have lemma 3 about the objective function of Semi-VRed in equation 9.

Lemma 3. *Assuming $P_i(x, y) = P_i(x|y)P_i(y) = P(x|y)P_i(y)$ for $i \in \{1, \dots, n\}$, for any model parameter θ, Semi-VRed global objective $F(\theta)$ defined in equation 9 can be expressed as*

$$F(\theta) = \sum_{j=1}^{C} \overline{P}(j) \overline{\ell}_j(\theta) + \frac{\beta}{n} \sum_{i=1}^{n} \Big(\sum_{j=1}^{C} [P_i(j) - \overline{P}(j)] \overline{\ell}_j(\theta) \Big)_+^2, \tag{13}$$

where $\overline{P}(j) = \frac{\sum_{i=1}^{n} P_i(j)}{n}$ is the marginal distribution of class j in the global dataset.

$P_i(j)$ and $\overline{P}(j)$ show the ratio of class j in the client i local dataset and the global dataset, respectively. Based on equation 13, `Semi-VRed` is capable of improving the performance of the predictor h in extreme class imbalance scenarios: consider when a label j is over-represented in a client i's data (i.e. $P_i(j) \approx 1$) and under-represented in the global data (i.e. $\overline{P}(j) \approx 0$). For a better understanding of how `Semi-VRed` does so, see example 1 in the appendix.

5 Convergence Results

In this section, we prove the convergence of our proposed `VRed` algorithm. We make some standard assumptions on the objective functions f_i. Specifically, we assume the functions are Lipschitz smooth and their gradients have bounded global and local variance:

Assumption 1 (**smoothness and bounded gradient variances**). *Each objective function f_i is L-Lipschitz smooth: for any $\theta, \theta' \in \mathbb{R}^d$ and any $i \in [n]$, we have $\|\nabla f_i(\theta) - \nabla f_i(\theta')\| \leq L\|\theta - \theta'\|$. Also, For any $\theta \in \mathbb{R}^d$ and any $i, j \in [n]$, we have $\|\nabla f_i(\theta) - \nabla f_j(\theta)\|^2 \leq \sigma_g^2$ (bounded global variance). Also, for any batch $S \sim \mathcal{B}_i^b$ of b i.i.d samples from client i local data, the following inequality holds (bounded local variance):*

$$\mathop{\mathbb{E}}_{S\sim\mathcal{B}_i^b} \left\| \tfrac{1}{|S|} \sum\nolimits_{(x,y)\in S} \nabla \ell(\theta, (x, y)) - \nabla f_i(\theta) \right\|^2 \leq \sigma_{l,i}^2,$$

Note that, according to algorithm 1, each client might have a different number (K_i) of mini batches of size b, but all clients use the same learning rate η. In order to prove the convergence of `VRed`, we also additionally assume the Lipschitzness and bounded variances for the client losses:

Assumption 2 (**boundedness, Lipschitz continuity and bounded variances**). *For any $i \in [n]$, $\theta \in \mathbb{R}^d$ and any batch $S \sim \mathcal{B}_i^b$ of b i.i.d. samples, we have:*

$$0 \leq \ell_S(\theta) := \frac{1}{|S|} \sum\nolimits_{(x,y)\in S} \ell(\theta, (x, y)) \leq \frac{M}{2},$$

and $\|f_i(\theta) - f_i(\theta')\| \leq L_0\|\theta - \theta'\|$. We also assume that for any $i, j \in [n]$ and $\theta \in \mathbb{R}^d$, $\|f_i(\theta) - f_j(\theta)\|^2 \leq \sigma_{0,g}^2$ and $\mathbb{E}_{S\sim\mathcal{B}_i^b} \left\| \frac{1}{|S|} \sum_{(x,y)\in S} \ell(\theta, (x, y)) - f_i(\theta) \right\|^2 \leq \sigma_{0,l,i}^2$ hold.

With the above assumptions, we prove that `VRed` algorithm converges to the neighborhood of a stationary point.

Theorem 1 (**`VRed`**). *Denote $\widetilde{L} = (L + \beta ML + 2\beta L_0^2)$ and $p_i = \frac{n_i}{N}$. Given Assumptions 1 and 2, assume that the local learning rate satisfies $\eta \leq \frac{1}{6\widetilde{L}K_i}$ for any $i \in [n]$ and:*

$$\eta \leq \frac{1}{L}\sqrt{\frac{1}{24(e-2)(\sum_i p_i^2)(\sum_i K_i^4)}} \tag{14}$$

By running Algorithm 1 for T global epochs we have:

$$\min_{0\leq t\leq T-1} \mathbb{E}\|\nabla F(\theta_t)\|^2 \leq \frac{12}{(11\gamma - 9)\eta}\left(\frac{F_0 - F^*}{T} + \widetilde{\Psi}_\sigma\right),$$

with $\gamma = \sum_i p_i K_i$ for full participation and $\gamma = \min_i K_i$ for partial participation, $F_0 = F(\theta_0)$, $F^* = \min_\theta F(\theta)$ the optimal value, and

$$\widetilde{\Psi}_\sigma = \frac{\eta}{2}\Big(\sum_{i=1}^n p_i^2\Big)\Big[\sum_{i=1}^n K_i^2(\widetilde{\sigma}_{l,i}^2 + 2\widetilde{\sigma}_g^2) + 2(e-2)\eta^2 L^2 \sum_{i=1}^n K_i^3\left(\widetilde{\sigma}_{l,i}^2 + 6K_i\widetilde{\sigma}_g^2\right)\Big],$$

where $\widetilde{\sigma}_{l,i}^2 = \big(2(2\beta\mu + 1)^2\sigma_{l,i}^2 + 8\beta^2M^2L_0^2\big)$ and $\widetilde{\sigma}_g = \big((2\beta\mu + 1)\sigma_g + 2\beta ML_0\big)$ and $\mu = \Sigma_i p_i f_i(\theta_0)$.

6 Experiments

In this section, we evaluate our proposed algorithms for training fair models. From the obtained experimental results, we can observe that `VRed` achieves competitive fairness performance and `Semi-VRed` beats almost all the existing algorithms in terms of multiple fairness metrics. Furthermore, `Semi-VRed` achieves the state of the art performance in terms of the system overall average performance too.

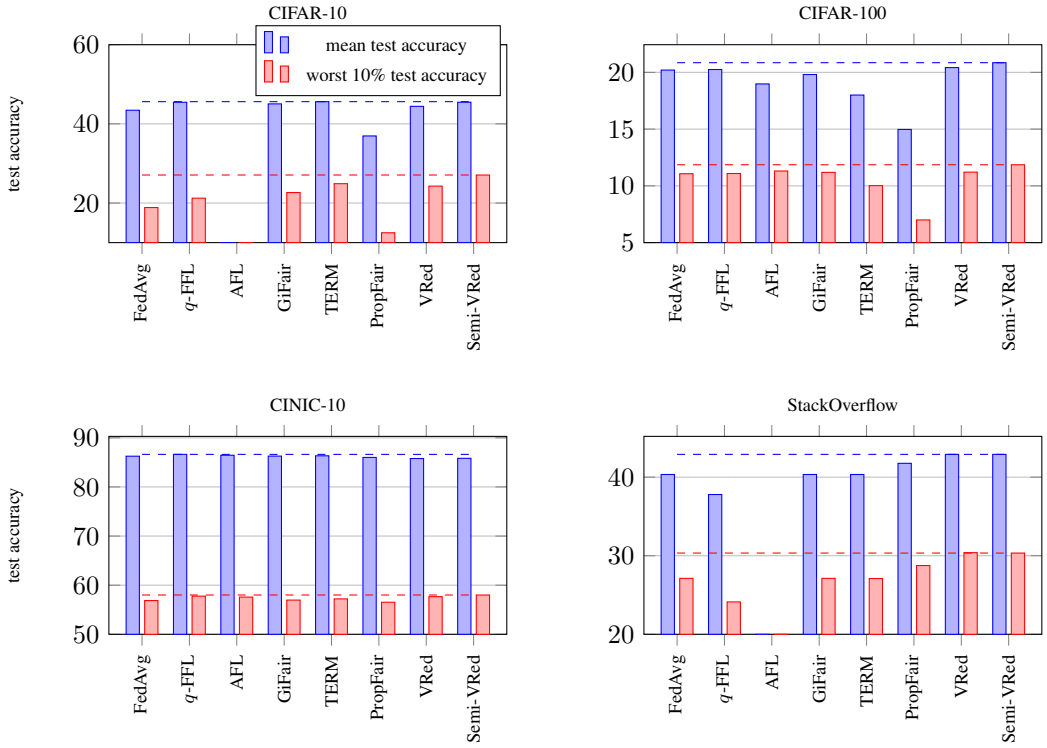

Figure 1: Average and worst 10% test accuracies. **top left:** CIFAR-10, **top right:** CIFAR-100, **bottom left:** CINIC-10, **bottom right:** StackOverflow. Due to divergence, results for `AFL` on CIFAR-10 and StackOverFlow are not shown. All subfigures share the same legends and axis labels.

6.1 EXPERIMENTAL SETUP

In this section, we provide some details about the experiments we conducted to evaluate our algorithms: the details of the datasets, models and their hyperparameters, and the metrics we use to evaluate our algorithms. For detailed explanation of the experiments, see §C in the appendix.

Datasets We use four benchmark datasets existing in the literature. The datasets we use include: CIFAR-10/100 (Krizhevsky et al., 2009), CINIC-10 (Darlow et al., 2018) (task of image classification) and StackOverflow (The Tensorflow Federated Authors, 2019) (task of next word prediction). In order to split the data among clients, we use Dirichlet distribution Wang et al. (2019). StackOverflow has a default realistic partition for each client. We follow the same default data distribution.

Train-Test dataset splitting After partitioning the dataset among clients, we split the data of each client into train and test sets with a ratio for each dataset. Each client uses its test data to evaluate the common trained model. For more details of the data splittings, see §C in the appendix.

Models, optimizers and loss functions For the CIFAR-10/100 and CINIC-10 datasets, we use ResNet-18 He et al. (2016). For the language dataset (StackOverflow), we use LSTMs Hochreiter & Schmidhuber (1997). In order to optimize the models parameters, We use SGD for minimizing average cross entropy loss. For further details, see §C in the appendix.

Baseline algorithms We compare our `VRed` and `Semi-VRed` algorithms with various fair `FL` algorithm existing in the literature including: `FedAvg` McMahan et al. (2017), `AFL` Mohri et al. (2019), `q-FFL` Li et al. (2020c) , `PropFair` Zhang et al. (2022a), `TERM` Li et al. (2020a), `GiFair` Yue et al. (2021) and `Ditto` Li et al. (2021).

Other hyperparameters We implement an `FL` setting where different clients participate in all communication rounds with one local epoch at each round. We use 200 communication rounds for all algorithms on the datasets to ensure their convergence. For CIFAR-10/100 and CINIC-10, we partition the data into 50 clients and for language datasets, we partition the data into 20 clients.

Table 1: Comparison between the performance of different algorithms on CIFAR-100. **Second column:** the percentage (%) of suffering clients with improved test accuracy. The value in parentheses shows the amount of test accuracy improvement averaged over suffering clients. **Third column:** the percentage (%) of well-performing clients with degraded test accuracy. The value in parentheses shows the amount of test accuracy improvement averaged over well-performing clients. **Fourth column:** the amount of improvement in the overall mean test accuracy

Algorithm	Improved suffering clients	Degraded well-performing clients	Overall accuracy improvement
q-FFL	$52.08_{\pm 11.95}$ (+0.69)	$54.21_{\pm 15.38}$ (-0.79)	$+0.04_{\pm 0.41}$
AFL	$51.18_{\pm 9.26}$ (+0.56)	$77.25_{\pm 12.85}$ (-3.50)	$-1.22_{\pm 0.95}$
GiFair	$60.55_{\pm 6.17}$ (+0.86)	$68.22_{\pm 16.13}$ (-2.03)	$-0.40_{\pm 0.61}$
TERM	$23.66_{\pm 5.34}$ (-1.12)	$87.93_{\pm 1.81}$ (-3.57)	$-2.20_{\pm 0.66}$
PropFair	$8.33_{\pm 1.69}$ (-4.05)	$92.41_{\pm 4.28}$ (-6.74)	$-5.23_{\pm 0.96}$
VRed	$60.50_{\pm 12.52}$ (+1.11)	$60.62_{\pm 7.53}$ (-0.94)	$+0.21_{\pm 0.06}$
Semi-VRed	**$65.40_{\pm 6.29}$ (+1.47)**	**$53.17_{\pm 6.75}$ (-0.42)**	**$+0.64_{\pm 0.30}$**

Evaluation metrics As we discussed in §4, the ultimate goal of proposing our novel algorithms is to achieve fairness without compromising the system overall average performance. We measure the overall performance with the *mean test accuracy* across clients. In order to measure the fairness in the system, we use the worst 10% test accuracies among clients, which is a standard metric for fairness in FL (Li et al., 2020a;c). In the appendix, we also use other common metrics in the literature for measuring fairness, e.g. the standard deviation of test accuracies (see table 6 in appendix C).

6.2 Comparison of VRed and Semi-VRed with other baseline algorithms

From fig. 1, Semi-VRed outperforms almost all the existing baseline algorithms in terms of the fairness in the system. Furthermore, Semi-VRed improves the system overall average performance (mean test accuracy) for three of the datasets as well. For instance, as can be observed from the results obtained for StackOverflow (see table 6 in § C in the appendix for evaluation in terms of various fairness metrics), Semi-VRed improves both fairness and mean test accuracy by 3% and 2.7%, respectively. Also, we can observe the competitive performance of VRed.

6.3 Superiority of Semi-VRed over VRed and the other baseline algorithms

As discussed in § 2, the existing fair FL algorithms usually suppress the well-performing clients in order to improve the worst-off clients performance. However, Semi-VRed, thanks to its efficient formulation, tries to avoid this. In order to get a better understanding of this, in table 1, we have compared different algorithms based on the amount of performance improvement that they provide: after running the simple vanilla FedAvg on CIFAR-100, we divide the existing 50 clients into two sets: 1. *suffering clients*: those with test accuracies below the FedAvg mean accuracy (22 clients in our experiment) 2. *well-performing clients*: those with test accuracies above the FedAvg mean accuracy (28 clients). Then, we run each of the other algorithms and compare their performance improvement with each other. The results clearly delivers two important messages: 1. the existing algorithms either more or less suppress the well-performing clients or cannot improve them, due to the more attention that they pay to the worst-off clients 2. Semi-VRed has the least suppression of well-performing clients (53.17% with a small average degradation of -0.42), and the highest average improvement of suffering clients (65.40% with an average accuracy improvement of +1.47), which results in improving both the fairness and the system overall average performance simultaneously.

7 Conclusion

In this work, we introduced two novel fair FL algorithms: VRed and Semi-VRed. In order to resolve the drawback of most of the existing fair FL algorithms, which is suppression of well-performing clients, we propose Semi-VRed, which uses an efficient method for measuring performance inequality in a FL system. Our experimental results show that Semi-VRed not only improves the worst-off clients performance, but also improves the system overall average performance as well. Accordingly, Semi-VRed achieves SoTA performance in terms of both the overall average accuracy and fairness, measured in terms of various common fairness metrics.

REFERENCES

Enrique Ballestero. Mean-semivariance efficient frontier: A downside risk model for portfolio selection. *Applied Mathematical Finance*, 12:1 – 15, 2005.

Alberto Bietti, Chen-Yu Wei, Miroslav Dudik, John Langford, and Steven Wu. Personalization improves privacy-accuracy tradeoffs in federated learning. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 1945–1962. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/bietti22a.html.

Vigdis Boasson, Emil Bóasson, and Zhaoli Zhou. Portfolio optimization in a mean-semivariance framework. *Investment management & financial innovations*, 8, 2017.

Hong-You Chen and Wei-Lun Chao. On bridging generic and personalized federated learning for image classification. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=I1hQbx10Kxn.

Luke N Darlow, Elliot J Crowley, Antreas Antoniou, and Amos J Storkey. CINIC-10 is not ImageNet or CIFAR-10. *arXiv preprint arXiv:1810.03505*, 2018.

Eduard Gorbunov, Filip Hanzely, and Peter Richtárik. Local SGD: Unified theory and new efficient methods. In *AISTATS*, volume 130, pp. 3556–3564, 2021. URL http://proceedings.mlr.press/v130/gorbunov21a.html.

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

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Zhouyuan Huo, Qian Yang, Bin Gu, Lawrence Carin, and Heng Huang. Faster on-device training using new federated momentum algorithm. arXiv:2002.02090, 2020. URL https://arxiv.org/abs/2002.02090.

Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. First analysis of local gd on heterogeneous data. arXiv:1909.04715, 2020. URL https://arxiv.org/abs/1909.04715.

Andrey Kolmogorov. On the notion of mean. *Atti della Academia Nazionale dei Lincei*, 12(9):388–391, 1930. URL https://link.springer.com/book/9789027727961. reprinted in "Selected Works I of Andrey Kolmogorov: Mathematics and Mechanics", pp. 144–146, 1991.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images, 2009. Technical report.

David Krueger, Ethan Caballero, Jörn-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Rémi Le Priol, and Aaron C. Courville. Out-of-distribution generalization via risk extrapolation (rex). In *ICML*, 2021.

Tian Lan, David Kao, Mung Chiang, and Ashutosh Sabharwal. An axiomatic theory of fairness in network resource allocation. In *2010 Proceedings IEEE INFOCOM*, pp. 1–9, March 2010. doi: 10.1109/INFCOM.2010.5461911. ISSN: 0743-166X.

Tian Li, Ahmad Beirami, Maziar Sanjabi, and Virginia Smith. Tilted empirical risk minimization. In *International Conference on Learning Representations*, 2020a.

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems*, volume 2, pp. 429–450, 2020b. URL https://proceedings.mlsys.org/paper/2020/file/38af86134b65d0f10fe33d30dd76442e-Paper.pdf.

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

Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6357–6368. PMLR, 2021.

Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. In *ICLR*, 2020. URL https://openreview.net/forum?id=HJxNAnVtDS.

Harry Markowitz. Portfolio slection. *Journal of Finance*, 1952.

Andreas Maurer and Massimiliano Pontil. Empirical bernstein bounds and sample-variance penalization. In *COLT*, 2009.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *AISTATS*, volume 54, pp. 1273–1282, 2017. URL http://proceedings.mlr.press/v54/mcmahan17a/mcmahan17a.pdf.

Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. Agnostic federated learning. In *ICML*, volume 97, pp. 4615–4625, 2019. URL http://proceedings.mlr.press/v97/mohri19a.html.

Hongseok Namkoong and John C. Duchi. Variance-based regularization with convex objectives. *J. Mach. Learn. Res.*, 20:68:1–68:55, 2017.

Jaehoon Oh, SangMook Kim, and Se-Young Yun. FedBABU: Toward enhanced representation for federated image classification. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=HuaYQfggn5u.

Reese Pathak and Martin J. Wainwright. Fedsplit: An algorithmic framework for fast federated optimization. In *NeurIPS*, 2020. URL https://proceedings.neurips.cc//paper/2020/hash/4ebd440d99504722d80de606ea8507da-Abstract.html.

David Plà-Santamaria and Mila Bravo. Portfolio optimization based on downside risk: a mean-semivariance efficient frontier from dow jones blue chips. *Annals of Operations Research*, 205: 189–201, 2013.

Zhe Qu, Xingyu Li, Rui Duan, Yao Liu, Bo Tang, and Zhuo Lu. Generalized federated learning via sharpness aware minimization. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 18250–18280. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/qu22a.html.

Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečny, Sanjiv Kumar, and H Brendan McMahan. Adaptive Federated Optimization. arXiv:2003.00295, 2020a. URL https://arxiv.org/abs/2003.00295.

Sashank J Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and Hugh Brendan McMahan. Adaptive federated optimization. In *International Conference on Learning Representations*, 2020b.

Andrea Rigamonti and Katarína Lučivjanská. Mean-semivariance portfolio optimization using minimum average partial. *Annals of Operations Research*, pp. 1–19, 2022.

Zebang Shen, Juan Cerviño, Hamed Hassani, and Alejandro Ribeiro. An agnostic approach to federated learning with class imbalance. In *ICLR*, 2022.

Pannagadatta Shivaswamy and Tony Jebara. Empirical bernstein boosting. In Yee Whye Teh and Mike Titterington (eds.), *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pp. 733–740, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR. URL https://proceedings.mlr.press/v9/shivaswamy10a.html.

Hamed Soleimani, Hamid Reza Golmakani, and Mohammad Hossein Salimi. Markowitz-based portfolio selection with minimum transaction lots, cardinality constraints and regarding sector capitalization using genetic algorithm. *Expert Syst. Appl.*, 36:5058–5063, 2009.

Alan L. Stuart and Harry M. Markowitz. Portfolio selection: Efficient diversification of investments. *A Quarterly Journal of Operations Research*, 10:253, 1959.

The Tensorflow Federated Authors. Tensorflow federated StackOverflow dataset. 2019. URL https://www.tensorflow.org/federated/api_docs/python/tff/simulation/datasets/stackoverflow.

Hongyi Wang, Mikhail Yurochkin, Yuekai Sun, Dimitris Papailiopoulos, and Yasaman Khazaeni. Federated learning with matched averaging. In *ICLR*, 2020. URL https://openreview.net/forum?id=BkluqlSFDS.

Kangkang Wang, Rajiv Mathews, Chloé Kiddon, Hubert Eichner, Françoise Beaufays, and Daniel Ramage. Federated evaluation of on-device personalization. *CoRR*, abs/1910.10252, 2019. URL http://arxiv.org/abs/1910.10252.

Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, 2018.

Xubo Yue, Maher Nouiehed, and Raed Al Kontar. Gifair-fl: An approach for group and individual fairness in federated learning. *ArXiv*, abs/2108.02741, 2021.

Guojun Zhang, Saber Malekmohammadi, Xi Chen, and Yaoliang Yu. Equality is not equity: Proportional fairness in federated learning. *CoRR*, abs/2202.01666, 2022a. URL https://arxiv.org/abs/2202.01666.

Michael Zhang, Karan Sapra, S. Fidler, Serena Yeung, and Jose M. Alvarez. Personalized federated learning with first order model optimization. In *ICLR*, 2021. URL https://openreview.net/forum?id=ehJqJQk9cw.

Xu Zhang, Yinchuan Li, Wenpeng Li, Kaiyang Guo, and Yunfeng Shao. Personalized federated learning via variational Bayesian inference. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 26293–26310. PMLR, 17–23 Jul 2022b. URL https://proceedings.mlr.press/v162/zhang22o.html.

Yuanyuan Zhang, Xiang Li, and Sini Guo. Portfolio selection problems with markowitz's mean–variance framework: a review of literature. *Fuzzy Optimization and Decision Making*, 17:125–158, 2018.