A Reputation Mechanism Is All You Need: Collaborative Fairness and Adversarial Robustness in Federated Learning

Xinyi Xu [*1] **Lingjuan Lyu** [*2]

Abstract

Federated learning (FL) is an emerging practical framework for effective and scalable machine learning among multiple participants, such as end users, organizations and companies. However, most existing FL or distributed learning frameworks have not well addressed two important issues together: collaborative fairness and adversarial robustness (e.g. free-riders and malicious participants). In conventional FL, all participants receive the global model (equal rewards), which might be unfair to the high-contributing participants. Furthermore, due to the lack of a safeguard mechanism, free-riders or malicious adversaries could game the system to access the global model for free or to sabotage it. In this paper, we propose a novel *Robust and Fair Federated Learning* (RFFL) framework to achieve collaborative fairness and adversarial robustness simultaneously via a reputation mechanism. RFFL maintains a reputation for each participant by examining their contributions via their uploaded gradients (using vector similarity) and thus identifies non-contributing or malicious participants to be removed. Our approach differentiates itself by *not* requiring any auxiliary/validation dataset. Extensive experiments on benchmark datasets show that RFFL can achieve high fairness and is very robust to different types of adversaries while achieving competitive predictive accuracy.

1. Introduction

Federated learning (FL) (McMahan et al., 2017) provides a promising collaboration paradigm by enabling a multitude of participants to construct a joint model without exposing their private training data. Two emerging challenges in FL are collaborative fairness (participants with different contributions should be rewarded differently), and adversarial robustness (free-riders should not enjoy the global model for free, and malicious participants should not compromise system integrity) (Lyu et al., 2020c).

Most existing FL paradigms (McMahan et al., 2017; Kairouz et al., 2019; Yang et al., 2019a; Li et al., 2020a) allow all participants to receive the same model in the end regardless of their contributions (by their uploaded parameters/gradients). This may lead to an unfair outcome as the participants who contribute the most are rewarded equally with the ones who contribute nothing. In practice, there may be a number of reasons for why the contributions differ, one reason is the divergence in the quality of the local data of different participants (Zhao et al., 2019). (Yang et al., 2019b) presented a motivating example for collaborative fairness: larger banks (for fear of not being compensated fairly) may refuse to collaborate with smaller banks who have smaller client base and thus less high-quality data.

In terms of adversarial robustness, the conventional FL framework (McMahan et al., 2017) is potentially vulnerable to adversaries and free-riders as it does not offer any safeguard mechanisms. The follow-up works considered robustness from different lens (Blanchard et al., 2017; Fung et al., 2020; Bernstein et al., 2019; Yin et al., 2018), but none of them can provide comprehensive supports for all the three types of attacks (targeted poisoning, untargeted poisoning and free-riders) considered in this work.

In summary, our contributions include:

- We propose a *Robust and Fair Federated Learning* (RFFL) framework to simultaneously achieve collaborative fairness and adversarial robustness.
- RFFL utilizes a reputation system to iteratively calculate participants' contributions and reward participants accordingly with different models of performance com-

[*]Equal contribution [1]Department of Computer Science, University of Singapore, Singapore, Singapore [2]Ant financial. Correspondence to: Xinyi Xu <xinyi.xu@u.nus.edu>, Lingjuan Lyu <lingjuanlvsmile@gmail.com>.

This work was presented at the International Workshop on Federated Learning for User Privacy and Data Confidentiality in Conjunction with ICML 2021 (FL-ICML'21). This workshop does not have official proceedings and this paper is non-archival. Copyright 2021 by the author(s).

mensurate with their contributions.

- Extensive experiments on benchmark datasets demonstrate that our RFFL can achieve high fairness and is very robust against all the investigated attacks (eg. targeted poisoning, untargeted poisoning and free-riders) while maintaining competitive predictive accuracy.

2. Related work

Promoting collaborative fairness has attracted substantial attention in FL. One research direction uses incentive schemes combined with game theory, based on the rationale that participants should receive rewards commensurate with their contributions to incentivize good behaviour (Yang et al., 2017; Gollapudi et al., 2017; Richardson et al., 2019; Yu et al., 2020). Note that in all these works, all participants receive the same final model.

Another research direction addresses the egalitarian fairness notion, *i.e.*, equalizing the performance of all participants (Mohri et al., 2019), and a more generalized q-Fair FL (q-FFL) (Li et al., 2020b). The q-Fair gives participants with higher local losses higher weights (optimizing their local objectives more relative to others).

As opposed to the above mentioned works, the most recent works (Lyu et al., 2020d;b) are better aligned with collaborative fairness in FL, where model is used as rewards for FL participants, so the participants receive models of different performances commensurate with their contributions. Lyu et al. (2020d) adopted a mutual evaluation of local credibility mechanism, where each participant privately rates the other participants iteratively. However, their framework is mainly designed for a decentralized block-chain system, which may not be directly applicable to FL settings when a central server is deployed.To alleviate this obstacle, Lyu et al. (2020b) proposed a FL framework to achieve collaborative fairness via an additional validation dataset used by the server to determine the contributions of the participants.

In terms of robustness in FL, Blanchard et al. (2017) proposed the Multi-Krum method based on a Krum function which excludes a certain number of uploaded gradients furthest from the mean and demonstrates resilience against up to 33% *Gaussian Byzantine* participants and up to 45% omniscient Byzantine participants. Fung et al. (2020) presented FoolsGold to defend against Sybils. Bernstein et al. (2019) proposed a communication efficient approach called SignSGD, which is robust to arbitrary scaling. In this approach, participants only upload the element-wise signs of the gradients without the magnitudes. A similar method was proposed by Yin et al. (2018), based on the statistics of the gradients, specifically element-wise median, mean and trimmed mean.

3. Proposed RFFL Framework

Our *Robust and Fair Federated Learning* (RFFL) framework focuses on two important goals in FL: *collaborative fairness* and *adversarial robustness*. We address both goals simultaneously via a reputation mechanism.

Problem setting and notation. We adopt the standard optimization model for FL: $\min_{\boldsymbol{w}\in\mathcal{W}} F(\boldsymbol{w}) \coloneqq \sum_{i=1}^{N} p_i F_i(\boldsymbol{w})$ where N is the number of participants, p_i is the weight of i-th participant such that $p_i \geq 0$ and $\sum_{i=1}^{N} p_i = 1$. $F_i(\cdot)$ is the respective local objective. In round t, $\Delta\boldsymbol{w}_i^{(t)} \coloneqq \nabla F_i(\boldsymbol{w}^{(t-1)})$ and $\Delta\boldsymbol{w}^{(t)} = \sum_{i=1}^{N} p_i \Delta\boldsymbol{w}_i^{(t)}$. D denotes the number of parameters in the model $\boldsymbol{w}$. $\cos(\boldsymbol{u},\boldsymbol{v}) = \langle\boldsymbol{u},\boldsymbol{v}\rangle/(||\boldsymbol{u}|| \times ||\boldsymbol{v}||)$ is the cosine similarity between two flattened gradient vectors.

3.1. Collaborative Fairness

The original FL framework (McMahan et al., 2017) can be viewed as adopting the *egalitarian* approach by giving everyone the same reward. Mohri et al. (2019); Li et al. (2020b) enforce the egalitarian concept through the lens of minimax optimization and fair resource allocation. However, the egalitarian approach might not be always desirable, if the participants are self-interested and not altruistic. For example in medicine, clinical trials data are time-consuming and expensive to collect, so researchers with limited resource may collaborate to conduct more extensive studies. Similar use cases for collaborative learning between competitors are present in finance (Yu et al., 2020). As such, the participants are self-interested and ultimately competing against each other, so it is not desirable to share the final reward with everyone as it will not be *fair* to the participants who have expended the most resources to collect data of higher qualities. If everyone is rewarded equally regardless of their contributions, then the participants may not have the motivation to contribute, by collecting data and uploading high-quality gradients.

The key idea for our reward design is that participants who contribute more should be rewarded better (Song et al., 2019; Wang et al., 2020; Sim et al., 2020). In addition to this qualitative relation, we propose a quantitative way to measure how 'fair' a set of rewards are with respect to the contributions of the participants, via the Pearson correlation coefficient, $\rho_{\mathrm{p}}(\cdot;\cdot)$. To see this, consider the following 3-participant example: suppose the contributions are $\boldsymbol{v} = [1, 2, 10]$, and two possible sets of rewards $\boldsymbol{\phi} = [2, 3, 4]$ and $\boldsymbol{\phi}' = [2, 4, 20]$. Both $\boldsymbol{\phi}, \boldsymbol{\phi}'$ correctly reflect the qualitative relations in the contributions but intuitively $\boldsymbol{\phi}'$ is 'fairer' as it better reflects the quantitative relations. This is indeed captured by the Pearson correlation coefficient where $\rho_{\mathrm{p}}(\boldsymbol{v}, \boldsymbol{\phi}) = 0.9122$ and $\rho_{\mathrm{p}}(\boldsymbol{v}, \boldsymbol{\phi}') = 1.0$. Note if $\boldsymbol{v}$ contains identical values, $\rho_{\mathrm{p}}(\boldsymbol{v}, \boldsymbol{\phi})$ (all participants contribute identically) is undefined, and in this case we reward all participants equally.

Definition 1 (Collaborative Fairness)**.** (Lyu et al., 2020a;b;d) Denote participants' real-valued contributions as $\boldsymbol{v}$, and a set of rewards as $\boldsymbol{\phi}$, then the quantitative collaborative fairness is: $\rho_{\text{p}}(\boldsymbol{v}, \boldsymbol{\phi})$ where $\rho_{\text{p}}(\cdot; \cdot)$ is the Pearson correlation coefficient.

Choice of Contributions and Rewards. The contributions $\boldsymbol{v}$ and rewards $\boldsymbol{\phi}$ should be quantitative and suitable to the FL setting. Intuitively, a participant with a larger and better dataset should be able to make higher contributions. Therefore, we adopt a simple empirical approach: the contributions are estimated by the standalone performance, *i.e.*, the test accuracy of a model trained only on the participant's *local* data. Similarly, the rewards are represented by the final performance, *i.e.*, the test accuracy of the model received by the participant at the end of FL. While in the original FL formulation (McMahan et al., 2017), the reward is the same for everyone because all participants synchronize with the server. In order to achieve collaborative fairness, we require the models to have different performance, commensurate with their contributions.

3.2. Adversarial Robustness

For adversarial robustness, we consider the threat model in Definition 2 (Blanchard et al., 2017; Yin et al., 2018).

Definition 2 (Threat Model)**.** In the t-th round, an honest participant uploads $\Delta \boldsymbol{w}_i^{(t)} := \nabla F_i(\boldsymbol{w}_i^{(t)})$ while a dishonest participant/adversary can upload arbitrary values.

In particular, we investigate three types of attacks:

- ***Targeted poisoning***. We consider the label-flipping attack, in which the labels of training examples are flipped to a target class (Biggio et al., 2011). For instance, in MNIST a '1-7' flip refers to training on images of '1' but using '7' as the labels.
- ***Untargeted poisoning***. We consider three types of untargeted poisoning defined in (Bernstein et al., 2019), where before uploading gradients, the adversary may (i) arbitrarily rescale gradients; or (ii) randomize the element-wise signs of the gradients; or (iii) randomly invert the element-wise values of the gradients.
- ***Free-riders***. Free-riders represent the participants unwilling to contribute their gradients due to data privacy concerns or computational costs, but want to access the jointly trained model for free (Yang et al., 2019b). They typically upload random gradients.

3.3. Robust and Fair FL (RFFL) via Reputation

Intuition for RFFL. The key to achieving collaborative fairness and adversarial robustness is in the uploaded gradients from the participants. A high-quality gradient (trained on high-quality local data) carries useful information for participants while a low-quality and possibly adversarial gradient can impair model performance. In gradient-based learning, a high-quality gradient moves the model towards lower loss quickly, while a low-quality or adversarial gradient can move the model very slowly or even move the model towards higher loss.

In particular, for a participant i, we manage the gradients i downloads so that i's model can move towards lower loss (mitigate adversarial gradients) at a rate commensurate with the quality of i's uploaded gradients (achieve collaborative fairness). To do so, we propose an iteratively updated *reputation* for each participant. This reputation is maintained by the server, and not seen by the participants.

High-level overview. RFFL makes two important modifications to the conventional FL framework: in the gradient aggregation, and in the downloading of the gradients for the participants. In addition, by keeping a reputation for each participant and a pre-determined threshold, we can achieve collaborative fairness (rewarding participants commensurately according to their reputations) and adversarial robustness (identifying and removing adversaries). The detailed realization of RFFL is given in Algorithm 1. Our code is available at: https://github.com/XinyiYS/Robust-and-Fair-Federated-Learning

Algorithm 1 Robust and Fair Federated Learning (RFFL)

```
Input: moving average coefficient $\alpha$, reputation threshold $\beta$; gradient normalizing constant $\gamma$.
Notations: $r_i^{(t)}$ is $i$'s reputation in round $t$ ; $R := \{i | r_i^{(t)} \geq \beta\}$ is the reputable set and w.l.o.g $\sum_{i \in R} r_i^{(t)} = 1$; $\boldsymbol{w}_i$ and $\boldsymbol{w}_g$ denote participant $i$ and server model parameters, respectively
                    Participant $i$
Upload local gradients $\Delta \boldsymbol{w}_i^{(t)} := \nabla F_i(\boldsymbol{w}_i^{(t)})$ to server
Download the allocated gradients $\Delta \boldsymbol{w}_{*i}^{(t)}$, and integrate with local gradients:
$\boldsymbol{w}_i^{(t+1)} = \boldsymbol{w}_i^{(t)} + \Delta \boldsymbol{w}_i^{(t)} + \Delta \boldsymbol{w}_{*i}^{(t)}$
                    Server
Aggregation:
$\Delta \boldsymbol{w}_g^{(t)} = \sum_{i \in R} r_i^{(t-1)} \Delta \boldsymbol{w}_i^{(t)} \times \gamma / ||\Delta \boldsymbol{w}_i^{(t)}||$
for $i \in R$ do
    $\tilde{r}_i^{(t)} = \cos(\Delta \boldsymbol{w}_g^{(t)}, \Delta \boldsymbol{w}_i^{(t)})$
    $r_i^{(t)} = \alpha r_i^{(t-1)} + (1 - \alpha) \tilde{r}_i^{(t)}$
    if $r_i^{(t)} < \beta$ then
        $R = R \setminus \{i\}$ Remove too low reputations
    end if
end for
Download:
for $i \in R$ do
    $\text{quota}_i = D \times r_i^{(t)} / (\max_i r_i^{(t)})$
    $\Delta \boldsymbol{w}_{*i}^{(t)} = \texttt{sparsify}(\Delta \boldsymbol{w}_g^{(t)}, \text{quota}_i) - r_i^{(t-1)} \Delta \boldsymbol{w}_i^{(t)}$
end for
```

Aggregation step. During gradient aggregation step, the server adopts reputation-weighted aggregation:

$$\Delta \boldsymbol{w}_g^{(t)} = \textstyle\sum_{i \in R} r_i^{(t-1)} \Delta \boldsymbol{w}_i^{(t)} \times \gamma / ||\Delta \boldsymbol{w}_i^{(t)}|| \qquad (1)$$

where $\Delta \boldsymbol{w}_i^{(t)} := \nabla F_i(\boldsymbol{w}_i^{(t-1)})$ is the uploaded gradient by participant i and γ is a normalization coefficient to prevent gradient explosion (Lin et al., 2018; Pascanu et al., 2013). R is the set of reputable participants, i.e., those whose reputations are higher than a pre-determined threshold β. The reputation for each round is calculated as follows,

$$r_i^{(t)} = \alpha r_i^{(t-1)} + (1 - \alpha) \tilde{r}_i^{(t)} \qquad (2)$$

where $\tilde{r}_i^{(t)} = \cos(\Delta \boldsymbol{w}_g^{(t)}, \Delta \boldsymbol{w}_i^{(t)})$ is i's reputation in the current round and α is a settable weight coefficient. Cosine similarity has previously been used in determining the quality of gradients to improve model performance (Cao et al., 2020; Fung et al., 2020). In (2), we integrate the reputation in both the current round and the previous round, in order to update the reputations in a smooth way and mitigate noise incurred by the training process and random model initializaiton (Song et al., 2019; Wang et al., 2020).

Download step. During the download step, the server determines the gradient i can download based on $r_i^{(t)}$ as follows,

$$\Delta \boldsymbol{w}_{*i}^{(t)} = \texttt{sparsify}(\Delta \boldsymbol{w}_g^{(t)}, \text{quota}_i) - r_i^{(t-1)} \Delta \boldsymbol{w}_i^{(t)} \qquad (3)$$

where $\text{quota}_i = D \times r_i^{(t)} / (\max_j r_j^{(t)})$ is the number of parameters to be downloaded and determined by the relative reputation, $r_i^{(t)} / (\max_j r_j^{(t)})$. After quota_i is calculated, the server first constructs a sparsified version of the aggregated gradient $\Delta \boldsymbol{w}_g^{(t)}$ by retaining only the largest quota_i values, then removes i's own gradient from it. $\Delta \boldsymbol{w}_{*i}^{(t)}$ refers to the gradient for i to download.

Sparsifying gradient vectors by retaining only the largest values gradually reduces the information and thus the quality of the gradient (Alistarh et al., 2018; Yan et al., 2020), which allows us to design rewards based on the contributions of the participants. Simply put, i with a higher contribution downloads a less sparsified gradient.

Reputation threshold β. We introduce a reputation threshold β as a settable coefficient to impose a requirement for the least amount of contribution from the participants. It can also be used to identify and remove adversaries as their contributions are usually low. In each round t, the updated reputations $r_i^{(t)}$ are compared against β and participants with reputations less than β are removed from the subsequent rounds. Specifically, R denotes the participants with reputations higher than β. The removed participants will *not* be added back in later.

4. Experiments

4.1. Datasets

We conduct experiments on various datasets including: (1) image classifications datasets: MNIST (LeCun et al., 1998) and CIFAR-10 (Krizhevsky et al., 2009); (2) text classifications datasets: Movie review (MR) (Pang & Lee, 2005) and Stanford sentiment treebank (SST) (Kim, 2014). We use a 2-layer convolutional neural network (CNN) for MNIST (LeCun et al., 1990), a 3-layer CNN for CIFAR-10 (Krizhevsky et al., 2017) and a text embedding CNN for MR and SST (Kim, 2014).

4.2. Baselines

We examine performance via three metrics : 1) predictive performance; 2) fairness and 3) robustness. For predictive performance, we include FedAvg (McMahan et al., 2017), and the *Standalone* framework where participants train locally without collaboration. For fairness performance, we focus our comparison with q-FFL (Li et al., 2020b) and CFFL (Lyu et al., 2020b). For robustness performance, we include several Byzantine-tolerant and/or robust FL frameworks including Multi-Krum (Blanchard et al., 2017), FoolsGold (Fung et al., 2020), SignSGD (Bernstein et al., 2019) and Median (Yin et al., 2018). FoolsGold was designed to mitigate sybils attacks and we adapt it for comparison.

4.3. Experimental Setup

Data splits. In addition to the standard I.I.D data sampling regime ('uniform' split, denoted as UNI), we consider two heterogeneous data splits by varying the data set sizes and the class numbers respectively. We follow a power law to randomly partition total {3000,6000,12000} MNIST examples among {5,10,20} participants respectively. In this way, each participant has a distinctly different number of examples, with the first participant has the least and the last participant has the most (on average 600 (McMahan et al., 2017)). We refer to this as the 'powerlaw' split (POW). Data splits for CIFAR-10, MR and SST datasets follow a similar way, with details in the appendix. Next, we investigate 'classimbalance' split (CLA), for which we vary the number of distinct classes in each participant's dataset, increasing from the first participant to the last. For example, for MNIST with total 10 classes and 5 participants, participant-{1,2,3,4,5} owns {1,3,5,7,10} classes of digits respectively. All participants have the same data size, but different class numbers. We only investigate MNIST and CIFAR-10 dataset as they both contain 10 classes.

Adversaries. We consider three types of adversaries on MNIST: targeted poisoning as in label-flipping (Biggio et al., 2011), untargeted poisoning as in the blind multiplicative adversaries (Bernstein et al., 2019), and free-riders. In each experiment, we evaluate RFFL against one type of adversary, and we test two proportions of the adversaries (to the honest participants), 20% and 110%. For targeted poisoning, the adversary uses '7' as labels for actual '1' images, during their local training to produce 'crooked' gradients. For untargeted poisoning, we consider three sub-cases separately, the adversary: 1) re-scales the gradients by -100; 2) randomizes the element-wise signs; or 3) randomly takes the element-wise reciprocals. For free-riders, they upload gradients randomly drawn from the $[-1, 1]$ uniform distribution. We conduct experiments with adversaries under UNI and POW for illustration purpose.

Hyper-Parameters. We set the reputation threshold to $\beta = 1/(3N)$, the moving average coefficient $\alpha = 0.95$ and the gradient normalizing constant $\gamma = 0.5$ for MNIST, 0.15 for CIFAR-10, and 1 for MR and SST. The interpretation for $\beta = 1/(3N)$ is that each participant (in order not to be removed from R) should contribute at least $1/3$ of their individual proportion which is $1/N$ as there are N participants. Further details on hyperparameters, hardware resource, and runtime statistics are included in Appendix A.1.

4.4. Experimental Results

Predictive performance. Table 1 reports the average and maximum accuracy of participants' final local models. RFFL outperforms other methods by a noticeable margin, especially for more heterogeneous data splits. It may be attributed to the reputation-weighted aggregation which can dynamically up-weight the participants who contribute more (implying they have better local data) (Li et al., 2020c).

Collaborative Fairness. Table 2 shows the calculated fairness results (the Pearson correlation coefficient between the standalone performance and the final performance). We use the standalone performance because it can estimate the contributions of the participants and more importantly because it is independent of the methods so can be used to compare 'fairness' results across methods. The results indicate that in RFFL, participants who have better local data (contribute more) get better models. Fig. 1 in Appendix A.2 provides an illustration where in MNIST and CIFAR-10, the agent with larger index has better final performance (because their local data are better in quantity and/or quality, under POW and CLA). While CFFL outperforms RFFL in some cases, CFFL requires an additional auxiliary dataset for validation, to determine the contributions of the participants. The results for the 5-participant case on both MNIST and CIFAR-10 are included in Appendix A.2.

Adversarial robustness. We first demonstrate RFFL's effectiveness in identifying and isolating the untargeted poisoning adversaries and free-riders as shown in Figs. 2 and 3 in Appendix A.2. The figures show the reputations of free-riders and the untargeted poisoning adversaries quickly decrease to below the threshold $\beta = 1/(3N)$ and get removed from subsequent rounds. For targeted poisoning, the adversaries are not completely identified. It is possible since an adversary intentionally mislabelling only one digit out of ten may still meet the reputation threshold of $\beta = 1/(3N)$.

For targeted poisoning, we consider two additional metrics (Fung et al., 2020): targeted class accuracy and attack success rate. Targeted class accuracy in our experiment corresponds to the test accuracy on digit '1' images. Attack success rate corresponds to the proportion of '1' images incorrectly classified as '7'. The results are in Tables 3 and 4. Table 3 illustrates that FedAvg, Multi-Krum and RFFL perform well in all three metrics. FedAvg and Multi-Krum are robust against 20% label flipping adversaries because these introduced 'crooked' gradients that are outweighed by the gradients from the honest participants. RFFL performs well by reducing the negative effect from these adversaries. Note Table 4 shows results for the case of 10 honest participants with 11 adversaries. This is an interesting scenario as common methods consider the majority to be honest participants (Blanchard et al., 2017). Additional and corresponding results for such an extreme scenario with other types of adversaries are provided in Appendix A.2.

For untargeted poisoning, the results are in Tables 5, 6 and 7. These results demonstrate that RFFL is overall the most robust. We observe that Multi-Krum and FoolsGold are not robust against untargeted poisoning. Multi-Krum utilizes the mean vector of the gradients, and is thus not robust to rescaling attacks. FoolsGold was designed to be robust against adversaries with a common objective, which is *not* the case for untargeted poisoning. Both SignSGD and Median demonstrate some degree of robustness for re-scaling attack. SignSGD is robust against re-scaling attack as it preserves the signs of gradients. Median utilizes the median statistic and is robust against extreme outliers as in re-scaling attack.

For the free-rider scenario, only FedAvg and RFFL are consistently robust. FedAvg is robust because the gradients from the free-riders have an expected value of zero, so the additional noise does not affect the asymptotic unbiasedness. Among the others, Multi-Krum exhibits some degree of robustness but compromises the accuracy. FoolsGold is not robust against free-riders as it assumes that the honest participants produce gradients that are more random than the adversaries who share a common attack objective function. For SignSGD, the free-riders are exactly the sign-randomizing adversaries, so the behavior is consistent. For Median, it is possible that the honest gradients are small and thus close to random noisy gradients, and as a result the random noisy gradients get updated to the model.

Additional experiments for robustness under POW are in Appendix A.2.

Table 1. Average and Maximum Test Accuracy[%]. Values in brackets denote maximum accuracy among N participants.

	MNIST						CIFAR-10			MR	SST
N	10			20			10			5	5
Data Split	UNI	POW	CLA	UNI	POW	CLA	UNI	POW	CLA	POW	POW
Standalone	91(91)	88(92)	53(92)	91(91)	89(92)	48(90)	46 (47)	43 (49)	31 (44)	47(56)	31(34)
FedAvg	93(94)	92(94)	53(93)	93(93)	92(94)	49(92)	48 (48)	47 (50)	32 (47)	51(63)	33(35)
q-FFL	85(91)	27(45)	44(64)	88(91)	48(53)	40(59)	41 (46)	36 (36)	22 (28)	12(18)	23(25)
CFFL	90(92)	85(90)	34(44)	91(93)	88(91)	39(46)	39 (41)	35 (45)	22 (40)	44(53)	31(32)
RFFL	**96(96)**	**95**(**96**)	**73(94)**	**97**(**97**)	**95(96)**	**66**(**95**)	**61 (62)**	**59 (61)**	**35 (54)**	**57(76)**	**35(37)**

Table 2. Fairness results[%] as in Definition 1 between the standalone performance and the final performance of N participants.

	MNIST						CIFAR-10			MR	SST
N	10			20			10			5	5
Data Split	UNI	POW	CLA	UNI	POW	CLA	UNI	POW	CLA	POW	POW
FedAvg	−31.2	77.33	64.53	3.85	−3.58	70.83	−42.9	40.58	79.34	22.22	64.18
q-FFL	-44.73	39.00	22.38	-22.01	38.71	48.07	-17.64	51.33	94.06	56.43	-75.92
CFFL	**83.57**	91.80	81.24	**82.52**	94.70	85.71	78.25	72.55	81.31	96.85	**93.34**
RFFL	83.36	**98.33**	**99.81**	75.19	**97.88**	**99.64**	**81.93**	**98.78**	**99.89**	**99.59**	65.88

Table 3. Maximum accuracy [%], Attack success rate [%] and Target accuracy [%] for MNIST under UNI with 10 honest participants and additional 20% **label-flipping** adversaries.

	Max accuracy	Attack success rate	Target accuracy
FedAvg	**96.8**	0.2	98.8
FoolsGold	9.8	**0**	0
Multi-Krum	95.6	0.2	**99.0**
SignSGD	9.1	41.9	18.8
Median	0.3	0.5	0.1
RFFL	93.4	**0**	98.9

Table 4. Maximum accuracy [%], Attack success rate [%] and Target accuracy [%] for MNIST under UNI with 10 honest participants and additional 110% **label-flipping** adversaries. 10 honest participants and 11 adversaries.

	Max accuracy	Attack success rate	Target accuracy
FedAvg	90.9	48.6	49.3
FoolsGold	19.2	**0**	55.0
Multi-Krum	**96.3**	0	98.8
SignSGD	9.1	0	18.8
Median	8.2	0	72.3
RFFL	93.5	**0**	**99.1**

Table 5. Individual test accuracies [%] over MNIST under UNI with 10 honest participants and additional 20% **sign-randomizing** adversaries. Adversaries omitted.

	1	2	3	4	5	6	7	8	9	10
FedAvg	10	10	10	10	10	10	10	10	10	10
FoolsGold	11	11	11	11	11	11	11	11	11	11
Multi-Krum	10	10	10	10	10	10	10	10	10	10
SignSGD	9	9	9	9	9	9	9	9	9	9
Median	1	1	1	1	1	1	1	1	1	1
RFFL	92	92	94	91	92	93	92	92	92	92

Table 6. Individual test accuracies [%] over MNIST under UNI with 10 honest participants and additional 20% **re-scaling** adversaries. Adversaries omitted.

	1	2	3	4	5	6	7	8	9	10
FedAvg	10	10	10	10	10	10	10	10	10	10
FoolsGold	10	10	10	10	10	10	10	10	10	10
Multi-Krum	10	10	10	10	10	10	10	10	10	10
SignSGD	50	58	62	58	59	64	66	57	57	57
Median	11	10	39	28	20	40	48	27	35	28
RFFL	93	93	94	92	92	94	94	93	93	92

Table 7. Individual test accuracies [%] over MNIST under UNI with 10 honest participants and additional 20% **value-inverting** adversaries. Adversaries omitted.

	1	2	3	4	5	6	7	8	9	10
FedAvg	9	9	9	9	9	9	9	9	9	9
FoolsGold	8	8	8	8	8	8	8	8	8	8
Multi-Krum	17	17	17	17	17	17	17	17	17	17
SignSGD	9	9	9	9	9	9	9	9	9	9
Median	1	1	1	1	1	1	1	1	1	1
RFFL	92	93	94	92	92	93	93	93	93	92

5. Discussion & Conclusion

We propose a Robust and Fair Federated Learning (RFFL) framework to address both *collaborative fairness* and *adversarial robustness* in FL. RFFL achieves these two goals by introducing reputations and iteratively evaluating the contribution of each participant, via the cosine similarity between the uploaded local gradients and the aggregated global gradients. Extensive experiments on various datasets demonstrate that RFFL achieves higher accuracy than FedAvg, and is robust against various types of adversaries under various settings. For future work, we plan to explore and theoretically formalize the potential trade-off among these three metrics: predictive performance, collaborative fairness and adversarial robustness.

References

Alistarh, D., Hoefler, T., Johansson, M., Khirirat, S., Konstantinov, N., and Renggli, C. The convergence of sparsified gradient methods. In *Advances in neural information processing systems*, pp. 5977–5987, 2018.

Bernstein, J., Zhao, J., Azizzadenesheli, K., and Anandkumar, A. SignSGD with majority vote is communication efficient and fault tolerant. In *Proceedings of the International Conference on Learning Representations*, 2019.

Biggio, B., Nelson, B., and Laskov, P. Support vector machines under adversarial label noise. In *Proceedings of the Asian Conference on Machine Learning*, pp. 97–112, 2011.

Blanchard, P., Guerraoui, R., Stainer, J., et al. Machine learning with adversaries: Byzantine tolerant gradient descent. In *Advances in Neural Information Processing Systems*, pp. 119–129, 2017.

Cao, X., Fang, M., Liu, J., and Gong, N. Z. FLTrust: Byzantine-robust federated learning via trust bootstrapping. In *Proceedings of the Network and Distributed Systems Security*, 2020.

Fung, C., Yoon, C. J. M., and Beschastnikh, I. The Limitations of Federated Learning in Sybil Settings. In *Proceedings of the Symposium on Research in Attacks, Intrusion, and Defenses*, 2020.

Gollapudi, S., Kollias, K., Panigrahi, D., and Pliatsika, V. Profit sharing and efficiency in utility games. In *Proceedings of the Annual European Symposium on Algorithms*, volume 87, pp. 43:1–43:14, 2017.

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.

Kim, Y. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., and Jackel, L. Handwritten digit recognition with a back-propagation network. In *Advances in neural information processing systems*, volume 2, 1990.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Li, T., Sahu, A. K., Talwalkar, A., and Smith, V. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020a.

Li, T., Sanjabi, M., Beirami, A., and Smith, V. Fair resource allocation in federated learning. In *Proceedings of the International Conference on Learning Representations*, 2020b.

Li, X., Huang, K., Yang, W., Wang, S., and Zhang, Z. On the convergence of FedAvg on non-IID data. In *Proceedings of the International Conference on Learning Representations*, 2020c.

Lin, Y., Wang, Y., Han, S., Dally, W. J., and Mao, H. Deep gradient compression: Reducing the communication bandwidth for distributed training. In *Proceedings of the International Conference on Learning Representations*, 2018.

Lyu, L., Li, Y., Nandakumar, K., Yu, J., and Ma, X. How to democratise and protect ai: Fair and differentially private decentralised deep learning. *IEEE Transactions on Dependable and Secure Computing*, 2020a.

Lyu, L., Xu, X., Wang, Q., and Yu, H. Collaborative fairness in federated learning. In Yang, Q., Fan, L., and Yu, H. (eds.), *Federated Learning*, volume 12500 of *Lecture Notes in Computer Science*, pp. 189–204. Springer, Cham, 2020b.

Lyu, L., Yu, H., and Yang, Q. Threats to federated learning: A survey. *arXiv preprint arXiv:2003.02133*, 2020c.

Lyu, L., Yu, J., Nandakumar, K., Li, Y., Ma, X., Jin, J., Yu, H., and Ng, K. S. Towards fair and privacy-preserving federated deep models. *IEEE Transactions on Parallel and Distributed Systems*, 31(11):2524–2541, 2020d.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the Artificial Intelligence and Statistics*, volume 54, pp. 1273–1282, 2017.

Mohri, M., Sivek, G., and Suresh, A. T. Agnostic federated learning. In *Proceedings of the International Conference on Machine Learning*, volume 97, pp. 4615–4625, 2019.

Pang, B. and Lee, L. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the Annual Meeting on Association for Computational Linguistics*, pp. 115–124, 2005.

Pascanu, R., Mikolov, T., and Bengio, Y. On the difficulty of training recurrent neural networks. In *Proceedings of the International Conference on Machine Learning*, volume 28, pp. 1310–1318, 2013.

Richardson, A., Filos-Ratsikas, A., and Faltings, B. Rewarding high-quality data via influence functions. *arXiv preprint arXiv:1908.11598*, 2019.

Sim, R. H. L., Zhang, Y., Chan, M. C., and Low, B. K. H. Collaborative machine learning with incentive-aware model rewards. In *Proceedings of the International Conference on Machine Learning*, volume 119, pp. 8927–8936, 2020.

Song, T., Tong, Y., and Wei, S. Profit allocation for federated learning. In *Proceedings of the IEEE International Conference on Big Data*, pp. 2577–2586, 2019.

Wang, T., Rausch, J., Zhang, C., Jia, R., and Song, D. A principled approach to data valuation for federated learning. In Yang, Q., Fan, L., and Yu, H. (eds.), *Federated Learning*, volume 12500 of *Lecture Notes in Computer Science*, pp. 153–167. Springer, Cham, 2020.

Yan, Z., Xiao, D., Chen, M., Zhou, J., and Wu, W. Dual-way gradient sparsification for asynchronous distributed deep learning. In *Proceedings of the International Conference on Parallel Processing*, 2020.

Yang, Q., Liu, Y., Chen, T., and Tong, Y. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology*, 10(2), 2019a.

Yang, Q., Liu, Y., Cheng, Y., Kang, Y., Chen, T., and Yu, H. *Federated Learning*. Morgan & Claypool Publishers, 2019b.

Yang, S., Wu, F., Tang, S., Gao, X., Yang, B., and Chen, G. On designing data quality-aware truth estimation and surplus sharing method for mobile crowdsensing. *IEEE Journal on Selected Areas in Communications*, 35(4): 832–847, 2017.

Yin, D., Chen, Y., Kannan, R., and Bartlett, P. Byzantine-robust distributed learning: Towards optimal statistical rates. In *Proceedings of the International Conference on Machine Learning*, volume 80, pp. 5650–5659, 2018.

Yu, H., Liu, Z., Liu, Y., Chen, T., Cong, M., Weng, X., Niyato, D., and Yang, Q. A fairness-aware incentive scheme for federated learning. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pp. 393–399, 2020.

Zhao, L., Wang, Q., Zou, Q., Zhang, Y., and Chen, Y. Privacy-preserving collaborative deep learning with unreliable participants. *IEEE Transactions on Information Forensics and Security*, 15:1486–1500, 2019.

A. Additional Experimental Results

A.1. Experimental Setup

Imbalanced dataset sizes. For CIFAR-10, we follow a power law to randomly partition total $\{10000, 20000\}$ examples among $\{5, 10\}$ participants respectively. For MR (SST), we follow a power law to randomly partition 9596 (8544) examples among 5 participants.

Hyper-Parameters. We provide the framework-independent hyperparameters used for different datasets in Table 8. q-FFL: fairness coefficient $q = 0.1$ and participants sampling ratio is 0.8; SignSGD: momentum coefficient is 0.8 and parameter weight decay is 0.977. FoolsGold: confidence $K = 1$. Multi-Krum: participant clip ratio is 0.2. For the hyperparameters, we either use the default values introduced in their respective papers or apply grid search to empirically find the values.

Table 8. Framework-independent Hyperparameters. Batch size B, learning rate η, exponential learning rate decay γ, total communication rounds/epochs T, local epochs E. Note that for experiments with more than 5 participants for MNIST and CIFAR-10, the learning rate η is 0.25 and 0.025, respectively

Dataset	B	η (γ)	$T(E)$
MNIST	16	0.15 (0.977)	60 (1)
CIFAR-10	64	0.015 (0.977)	200 (1)
MR	128	1e-4 (0.977)	100 (1)
SST	128	1e-4 (0.977)	100 (1)

Runtime Statistics, Hardware and Software. We conduct our experiments on a machine with 12 cores (Intel(R) Xeon(R) CPU E5-2650 v4 @ 2.20GHz), 110 GB RAM and 4 GPUs (P100 Nvidia). Execution time for the experiments including only RFFL (all) frameworks: for MNIST (10 participants) approximately 0.6 (0.7) hours; for CIFAR-10 (10 participants) approximately 0.7 (4.3) hours; for MR and SST (5 participants) approximately 1.5 (2) hours.

Our implementation mainly uses PyTorch, torchtext, torchvision and some auxiliary packages such as Numpy, Pandas and Matplotlib. The specific versions and package requirements are provided together with the source code. To reduce the impact of randomness in the experiments, we adopt several measures: fix the model initilizations (we initialize model weights and save them for future experiments); fix all the random seeds; and invoke the deterministic behavior of PyTorch. As a result, given the same model initialization, our implementation is expected to produce consistent results on the same machine over experimental runs.

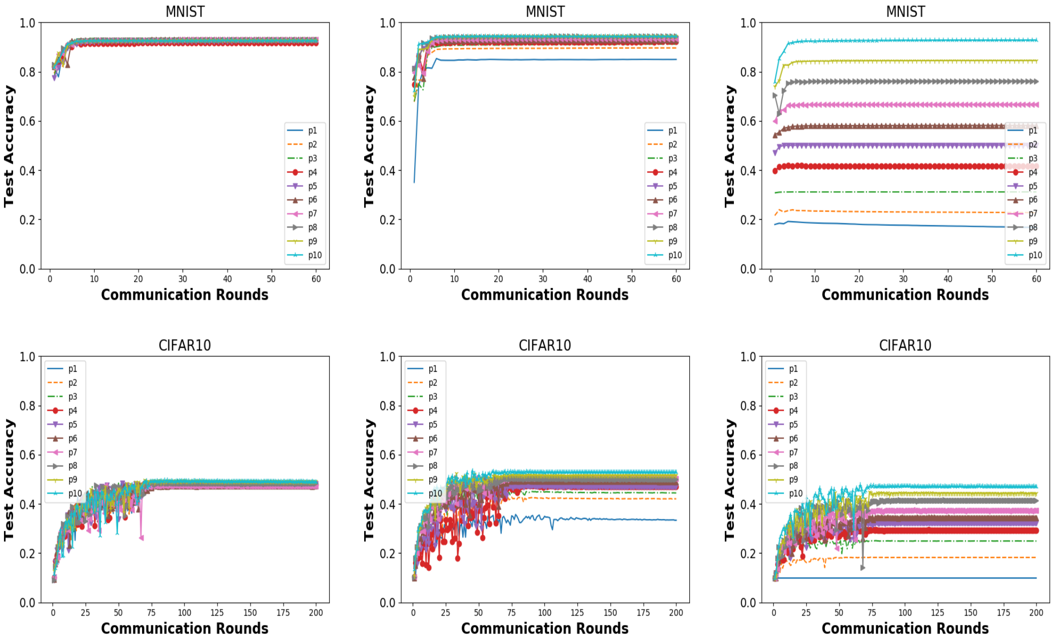

Figure 1. Participants final performance for MNIST and CIFAR10. From left to right {UNI, POW, CLA}.

A.2. Experimental Results

Comprehensive experimental results below demonstrate that RFFL is the *only* framework that performs consistently well over all the investigated situations, though may not perform the best in all of them.

5-participant Case for MNIST and CIFAR-10. We include the fairness and accuracy results for the 5-participant case for MNIST and CIFAR-10 under the three data splits in Tables 9 and 10, respectively.

Free-riders. For better illustration and coherence, we include here the experimental results together with the participants' reputation curves. Table 11 demonstrates the performance results for 20% free-riders in the 10-participant case for MNIST over UNI. Figure 2 demonstrates the reputations of the participants. It can be clearly observed that free-riders are isolated from the federated system at the early stages of collaboration (within 5 rounds).

Table 9. Additional predictive performance results. Average and Maximum Test Accuracy[%]. Values in brackets denote maximum accuracy among the participants.

	MNIST			CIFAR-10		
	UNI	POW	CLA	UNI	POW	CLA
Standalone	91(91)	87(94)	50(91)	44(46)	42(52)	29(44)
FedAvg	93(93)	91(95)	50(92)	46(47)	46(52)	30(45)
q-FFL	82(85)	59(78)	49(84)	31(32)	31(34)	19(24)
CFFL	24(39)	21(37)	27(28)	44(45)	40(49)	26(43)
RFFL	**97(97)**	**96(97)**	**79(94)**	**57(57)**	**56(58)**	**31(48)**

Table 10. Additional Fairness results[%] as in Definition 1 between the standalone performance and the final performance.

	MNIST			CIFAR-10		
	UNI	POW	CLA	UNI	POW	CLA
FedAvg	20.27	95.10	55.86	16.92	84.76	86.20
q-FFL	66.49	−38.48	−54.85	17.23	60.47	28.07
CFFL	30.76	18.06	−23.04	66.21	63.35	−13.94
RFFL	**85.12**	**98.45**	**99.64**	**95.99**	**99.58**	**99.93**

Adversarial Experiments with the POW. We conduct experiments with adversaries under two data splits, the UNI and the POW. We have included the experimental results with respect to the UNI in the main paper and supplement here the experimental results with respect to the POW. Table 12, Table 13, Table 14, Table 15 and Table 16 show the respective results for the targeted poisoning adversaries, three untargeted poisoning adversaries and free-riders.

Adversarial Experiments with Adversaries as the Majority. For extension, we also conduct experiments by increasing the number of adversaries to test RFFL's Byzantine tolerance. Our experimental results in Table 4, Table 17, Table 18, Table 19, and Table 20 demonstrate that RFFL consistently achieves competitive performance over various types of adversaries even when the adversaries are the majority in the system.

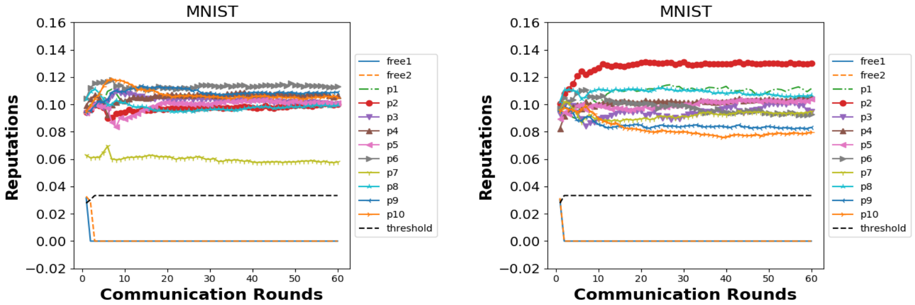

Figure 2. Reputations of the participants including free-riders for the UNI (left) and POW (right) splits in RFFL. The reputations of these two free-riders are very quickly decayed lower than the reputation threshold, thus free-riders are identified and isolated from the system at the beginning.

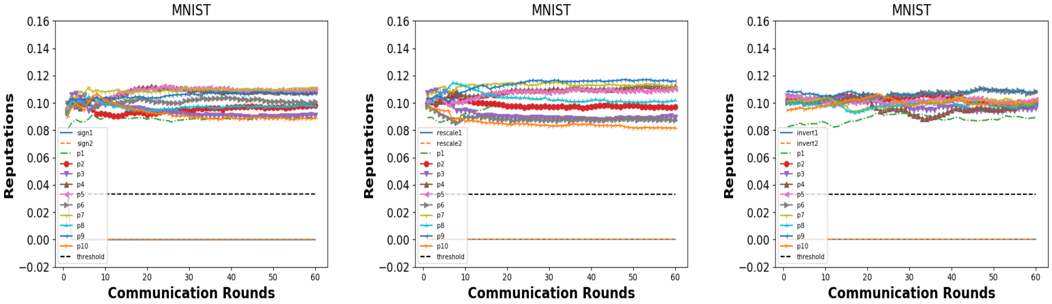

Figure 3. Reputations for MNIST 10 participants with 2 adversaries of untargeted poisoning. From left to right, {**sign-randomizing**, **re-scaling**, **value-inverting**}. The adversaries have clearly lower reputations and are removed.

Table 11. Individual test accuracies [%] over MNIST under UNI with 10 honest participants and additional 20% **free-riders**. Free-riders omitted.

	1	2	3	4	5	6	7	8	9	10
FedAvg	97	97	97	97	97	97	97	97	97	97
FoolsGold	11	11	10	11	10	10	11	11	10	11
Multi-Krum	61	61	64	57	60	62	62	60	62	57
SignSGD	9	9	9	9	9	9	9	9	9	9
Median	1	1	1	1	1	1	1	1	1	1
RFFL	92	93	94	92	93	93	91	92	93	92

Table 12. Maximum accuracy [%], Attack success rate [%] and Target accuracy [%] over MNIST under POW with 10 honest participants and additional 20% **label-flipping** adversaries.

	Max accuracy	Attack success rate	Target accuracy
FedAvg	**97.22**	0.20	98.80
SignSGD	9.11	41.90	18.80
FoolsGold	9.80	**0**	0.00
Multi-Krum	96.13	**0**	**98.90**
Median	0.09	0.20	0.20
RFFL	95.01	**0**	98.70

Table 13. Individual test accuracies [%] over MNIST under POW with 10 honest participants and additional 20% **sign-randomizing** adversaries. Adversaries omitted.

	1	2	3	4	5	6	7	8	9	10
FedAvg	97	97	97	97	97	97	97	97	97	97
SignSGD	9	9	9	9	9	9	9	9	9	9
FoolsGold	80	78	81	83	84	86	86	87	87	88
Multi-Krum	96	96	96	96	96	96	96	96	96	97
Median	1	1	1	1	1	1	1	1	1	1
RFFL	86	88	91	92	93	93	94	94	95	94

Table 14. Individual test accuracies [%] over MNIST under POW with 10 honest participants and additional 20% **re-scaling** adversaries. Adversaries omitted.

	1	2	3	4	5	6	7	8	9	10
FedAvg	10	10	10	10	10	10	10	10	10	10
SignSGD	9	9	9	9	9	9	9	9	9	9
FoolsGold	93	93	93	93	93	93	93	93	93	93
Multi-Krum	10	10	10	10	10	10	10	10	10	10
Median	1	1	1	1	1	1	1	1	1	1
RFFL	86	88	92	92	93	93	94	94	95	94

Table 15. Individual test accuracies [%] over MNIST under POW with 10 honest participants and additional 20% **value-inverting** adversaries. Adversaries omitted.

	1	2	3	4	5	6	7	8	9	10
FedAvg	10	10	10	10	10	10	10	10	10	10
SignSGD	9	9	9	9	9	9	9	9	9	9
FoolsGold	10	10	10	10	10	10	10	10	10	10
Multi-Krum	9	9	9	9	9	9	9	9	9	9
Median	0	0	0	0	0	0	0	0	0	0
RFFL	73	83	91	91	93	93	94	94	95	94

Table 16. Individual test accuracies [%] over MNIST under POW with 10 honest participants and additional 20% **free-riders**. Adversaries omitted.

	1	2	3	4	5	6	7	8	9	10
FedAvg	97	97	97	97	97	97	97	97	97	97
SignSGD	9	9	9	9	9	9	9	9	9	9
FoolsGold	10	10	10	9	10	10	11	11	10	10
Multi-Krum	53	57	58	58	55	53	56	59	58	61
Median	1	1	1	1	1	1	1	1	1	1
RFFL	86	89	90	92	93	93	94	94	95	95

Table 17. Individual test accuracies [%] over MNIST under UNI with 10 honest participants and additional 110% **sign-randomizing** adversaries. 10 honest participants with 11 adversaries.

	1	2	3	4	5	6	7	8	9	10
FedAvg	96	96	96	96	96	96	96	96	96	96
SignSGD	9	9	9	9	9	9	9	9	9	9
FoolsGold	61	58	64	60	62	66	54	58	60	58
Multi-Krum	95	94	96	95	96	95	96	95	95	95
Median	1	1	1	1	1	1	1	1	1	1
RFFL	93	93	94	91	92	93	93	92	92	92

Table 18. Individual test accuracies [%] over MNIST under UNI with 10 honest participants and additional 110% **re-scaling** adversaries. 10 honest participants with 11 adversaries.

	1	2	3	4	5	6	7	8	9	10
FedAvg	10	10	10	10	10	10	10	10	10	10
SignSGD	9	9	9	9	9	9	9	9	9	9
FoolsGold	11	11	11	11	11	11	11	11	11	11
Multi-Krum	10	10	10	10	10	10	10	10	10	10
Median	93	93	93	93	93	93	93	93	93	93
RFFL	93	92	94	92	93	93	93	92	93	93

Table 19. Individual test accuracies [%] over MNIST under UNI with 10 honest participants and additional 110% **value-inverting** adversaries. 10 honest participants with 11 adversaries.

	1	2	3	4	5	6	7	8	9	10
FedAvg	9	9	9	9	9	9	9	9	9	9
SignSGD	9	9	9	9	9	9	9	9	9	9
FoolsGold	10	10	10	10	10	10	10	10	10	10
Multi-Krum	18	18	18	18	18	18	18	18	18	18
Median	9	9	9	9	9	9	9	9	9	9
RFFL	93	92	94	92	93	93	93	92	93	93

Table 20. Individual test accuracies [%] over MNIST under UNI with 10 honest participants and additional 110% **free-riders**. 10 honest participants and 11 free-riders.

	1	2	3	4	5	6	7	8	9	10
FedAvg	97	97	97	97	97	97	97	97	97	97
SignSGD	9	9	9	9	9	9	9	9	9	9
FoolsGold	10	10	10	10	10	10	10	10	10	10
Multi-Krum	51	52	45	46	41	47	43	46	47	47
Median	1	1	1	1	1	1	1	1	1	1
RFFL	92	94	93	92	92	93	93	93	92	92