LoRA-FL: A Low-Rank Adversarial Attack for Compromising Group Fairness in Federated Learning

Anonymous authors
Paper under double-blind review

Abstract

Federated Learning (FL) enables collaborative model training without requiring participants to share raw data, and is increasingly deployed in regulated domains such as healthcare, finance, and large-scale personalization. FL offers privacy and governance benefits, it can also obscure fairness risks: heterogeneity in client data distributions may lead to models that systematically disadvantage minority groups. Ensuring fairness in such settings is not only an ethical concern but also a regulatory requirement under frameworks such as GDPR and anti-discrimination law. Existing adversarial manipulations in FL, such as noise injection or scaling attacks, typically degrade predictive performance or are mitigated by robust aggregation rules (e.g., KRUM or FLAME), limiting their practical relevance. In this work, we introduce LoRA-FL, a stealthy fairness attack that leverages low-rank adapters to inject group-level bias while preserving accuracy. By constraining adversarial updates to a compact subspace that aligns with benign client variation, LoRA-FL evades both standard and robust aggregators, even under heterogeneous (non-IID) data distributions. We provide empirical results, across widely used fairness benchmarks, including tabular datasets such as Adult and Bank. With LoRA-FL as few as 10–20% adversarial clients can increase violations of demographic parity and equalized odds by over 40%, while maintaining comparable predictive performance.

1 Introduction

Federated Learning (FL) (McMahan et al., 2017) enables multiple data holders to collaboratively train deep learning models without sharing raw data, thereby respecting data ownership and supporting regulatory compliance (Parliament & of the European Union, 2016). FL has been widely adopted in high-stakes domains such as healthcare (Sheller et al., 2020; Xu et al., 2021), finance (Yang et al., 2019), and personalized language modeling (Hard et al., 2018), where centralized data collection is either infeasible or undesirable. As FL systems increasingly influence decisions about individuals and communities, concerns about fairness have become central to their responsible deployment (Barocas & Selbst, 2016; Barocas et al., 2023).

Real-world Instances (Salazar et al., 2024). In federated credit scoring or risk assessment systems, banks and financial institutions collaboratively train models while retaining control over proprietary customer data. Such institutions may have economic incentives to preserve lending practices that disadvantage certain demographic groups, so long as overall predictive performance remains strong. Similarly, large digital platforms participating in federated training for ranking, recommendation, or personalization systems may prioritize revenue or engagement objectives that conflict with fairness constraints, resulting in models that encode systematic representational biases.

From a standard learning viewpoint, success in FL is typically measured by the convergence of the global loss and overall predictive accuracy. In the idealized IID setting—where each clients local data are drawn independently from a common distribution—such metrics often suffice to characterize model performance. In contrast, in realistic deployments, FL operates under non-IID data distributions, where the joint distribution of features and labels varies significantly across clients. In decision-making systems such as credit approval, however, model behavior across demographic groups, such as gender, age, or race, constitutes a

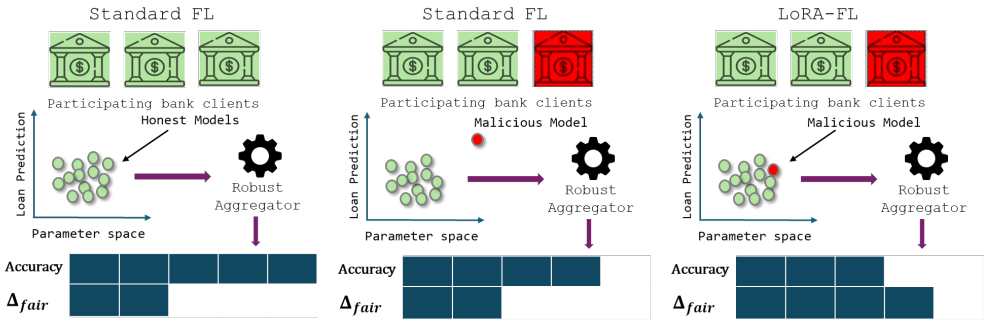

Figure 1: LoRA-FL constrains adversarial updates to a low-rank subspace, making malicious updates appear similar to benign ones and allowing robust aggregators to accept them. Consequently, overall accuracy remains stable, while group-wise bias increases.

first-order concern (Chouldechova, 2017; Hardt et al., 2016a). A converged FL model may achieve strong aggregate accuracy while approving loans for one group at substantially higher rates than another, even among applicants with comparable financial profiles (Angwin et al., 2016; Fabris et al., 2025). In practice, such fairness degradation is rarely treated as anomalous; instead, it is often viewed as a natural byproduct of maximizing accuracy under client heterogeneity, particularly in non-IID FL settings. This tension between performance and bias has motivated a formal approach in quantifying and enforcing fairness constraints.

Fair Federated Learning. A substantial body of work has formalized group fairness through criteria such as Demographic Parity (Chouldechova, 2017), Equalized Odds (Hardt et al., 2016a), and Equal Opportunity (Hardt et al., 2016a), and studied their enforcement and evaluation in machine learning systems (Bilal Zafar et al., 2015; Agarwal et al., 2018; Madras et al., 2018; Friedler et al., 2019; Pessach & Shmueli, 2022; Rabonato & Berton, 2025). In FL, however, achieving such fairness guarantees is fundamentally more challenging. Even small and individually insignificant disparities at the client level can accumulate and amplify into significant group-level unfairness in the global model (Chang & Shokri; Wang et al.; Salazar et al., 2024). The fairness challenge becomes substantially more complex when clients are *malicious*. In FL, participants retain control over their local training procedures and may intentionally deviate from the intended optimization process by manipulating local updates, causing adversarial attacks (Bhagoji et al., 2019; Shejwalkar & Houmansadr, 2021; Xia et al., 2023). Such attacks can further worsen disparities across sensitive groups (Solans et al., 2020).

Adversarial Attacks in FL. Prior work on adversarial behavior in FL has largely examined model poisoning strategies in which malicious participants manipulate local updates to influence the global modeleither by degrading overall predictive accuracy or by inducing targeted misclassification that benefits specific objectives (Cao et al., 2020; So et al., 2020). Such attacks are particularly relevant in settings where participants have incentives to shape model outcomes while remaining indistinguishable from benign clients. To defend against malicious participants, a substantial literature has proposed Byzantine-resilient aggregation rules that suppress updates deviating from the majority (Blanchard et al., 2017a; Yin et al., 2018; Nguyen et al., 2022).

Robust FL blind to fairness manipulation. The proposed robust aggregators, both in IID and non-IID, are designed to detect *accuracy-disruptive* behavior. Classical defenses such as `KRUM` (Blanchard et al., 2017a) and trimmed-mean (`TM`) (Yin et al., 2018) operate under IID assumptions by selecting or averaging updates that are closest in parameter space, while more recent mechanisms such as `FLAME` (Nguyen et al., 2022) explicitly account for non-IID heterogeneity by clustering or reweighting updates before aggregation. Updates that remain statistically consistent with benign optimization, yet systematically influence fairness-relevant directions in parameter space, do not violate the criteria enforced by these aggregators. In fact, under non-IID assumptions, the natural variance of benign updates further enlarges the set of directions that such defenses must tolerate, creating room for fairness manipulation that preserves both convergence and accuracy.

Our key insight. The above observations suggest a structural gap between how robustness is enforced in FL and how fairness violations manifest. Robust aggregators are designed to detect deviations that disrupt accuracy, implicitly treating all directions in parameter space as equally relevant. In contrast, fairness-relevant behavior often depends on a restricted subspace tied to features correlated with sensitive attributes, whose influence on global performance may be subtle. We hypothesize that confining adversarial perturbations to this subspace allows fairness manipulations to remain hidden and preserves accuracy.

Our Approach and Contributions. Motivated by this hypothesis, we propose `LoRA-FL` that leverages low-rank parameterizations to restrict adversarial updates to a compact subspace (Figure 1). Using `LoRA-FL` clients jointly optimize for task performance and fairness degradation with minimal additional computational overhead. As we demonstrate empirically, this design allows adversarial behavior to blend seamlessly with benign updates, even under robust aggregation and non-IID data heterogeneity. Our contributions are as follows:

1. We introduce `LoRA-FL`, a fairness-attack strategy that augments standard FL training by leveraging low-rank adapters to inject fairness-oriented bias into federated models without compromising predictive performance (Algorithm 1). Operating within a low-dimensional subspace, `LoRA-FL` allows adversarial clients to craft updates that remain statistically consistent with benign ones (Section 4.2).
2. We demonstrate the effectiveness of how `LoRA-FL` reliably degrades Demographic Parity, Equalized Odds, and Equal Opportunity on standard benchmarks (Adult Dua & Graff (2017), Bank Moro et al. (2014), and Dutch liobaite et al. (2011)). With only 10–20% adversarial clients, `LoRA-FL` reduces fairness metrics (DP, EO, and EOpp) by over 40%, while preserving high accuracy under both IID (Table 1) and non-IID (Table 2), robust aggregators such as `KRUM` (Blanchard et al., 2017a) and `FLAME` (Nguyen et al., 2022) (Section 5). An ablation study further confirms `LoRA-FL`'s robustness to varying numbers of agents. Additionally, increasing the adapter rank makes adversarial updates more detectable, highlighting the importance of low-rank constraints.
3. We analyze why low-rank adversarial updates evade detection by studying their alignment with dominant gradient subspaces and aggregation geometry (Section 6). A detailed interpretability study shows how low-rank adapters are effective. Lower-rank perturbations closely align with benign update distributions, thereby evading detection. Moreover, these adapters disrupt internal neuron-level representations, thereby systematically skewing predictions across demographic groups. Finally, we show that even higher-rank adapters concentrate changes along a few principal directions – explaining why low-rank updates suffice to achieve the attack's effect while remaining covert.

Together, our results show that fairness manipulation can be embedded directly into the optimization dynamics of FL, remaining invisible to accuracy-centric robust aggregators. This exposes a fundamental gap between robustness and fairness in decentralized optimization.

2 Related Work

Data poisoning attacks corrupt local training datasets by injecting false or misleading examples, thereby influencing the learned model (Biggio et al., 2012; Rubinstein et al., 2009; Li et al., 2016; Fang et al., 2020; Dai & Li, 2023). Prior work has shown that algorithmic fairness can be compromised through data poisoning in centralized learning. Solans et al. (2020) propose a gradient-based poisoning attack that induces disparities across demographic groups, degrading fairness metrics such as Demographic Parity and Equalized Odds.

In contrast, local model poisoning is a much more severe threat in FL. Byzantine-robust stochastic gradient descent and aggregation methods include distance-based and coordinate-wise defenses (Blanchard et al., 2017a; Yin et al., 2018; Nguyen et al., 2022). Following the robustness definition of Li et al. (2021), these approaches aim to preserve convergence and predictive performance in the presence of adversarial clients. Studies have demonstrated that targeted manipulation of training data can exacerbate bias while maintaining high predictive accuracy (Van et al., 2022; Mehrabi et al., 2021). These results establish that fairness can be manipulated independently of accuracy, but they focus on instance-level attacks in centralized settings. Unlike these, we propose an attack for the FL setting that introduces a substantially different threat model.

In FL, adversaries can control entire clients, influence local optimization procedures, and repeatedly affect the global model through aggregated updates. To mitigate fairness concerns in this setting, prior work has proposed fairness-aware aggregation and optimization mechanisms, such as FairFed (Ezzeldin et al., 2023) and GIFAIR-FL (Yue et al., 2023), which explicitly incorporate fairness objectives into federated training. Unlike our work, these approaches typically assume benign participants and do not consider adversarial manipulation of local updates.

Only a limited number of works study adversarial attacks on fairness in FL. PFAttack (Gao et al., 2024) demonstrates that fairness-aware aggregators can be subverted without degrading accuracy, but the attack is tailored to demographic parity and requires careful empirical tuning to remain stealthy. He et al. (2025) propose robust aggregation while maintaining *performance fairness*. This notion of fairness differs fundamentally from group fairness definitions we consider, which focus on disparities across individuals or demographic groups. EAB-FL (Meerza & Liu, 2024) proposes a model poisoning strategy that increases bias while preserving utility, but is ineffective against robust aggregation rules. DiTTO (Li et al., 2021) explores targeted fairness manipulation through client-level optimization, yet remains sensitive to aggregation mechanisms and data heterogeneity. In contrast, our proposed `LoRA-FL` attack is model-agnostic and effective against robust aggregators as it preserves accuracy.

3 Background

We consider a standard classification setting in which each data point is a tuple (x, y, a) drawn from an unknown distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y} \times \mathcal{A}$. The instance space is $\mathcal{X} \subseteq \mathbb{R}^d$, where $x \in \mathcal{X}$ denotes a d-dimensional feature vector. The label space is $\mathcal{Y} = [C]$ for a C-class classification problem, and $\mathcal{A}$ denotes the space of sensitive attributes. Each $a \in \mathcal{A}$ encodes a sensitive group membership (e.g., gender, age, caste), and is observed together with the input-label pair.

3.1 Federated Learning (FL)

In a typical Federated Learning (FL) setup, we consider (i) a set of agents $[K] = \{1, \ldots, K\}$; and (ii) a central aggregator. Each agent k holds a local dataset $\mathcal{D}^{(k)} = \{(x_i, y_i, a_i)\}_{i=1}^{n_k}$ formed by sampling from $\mathcal{D}$ according to a client-specific distribution. To model statistical heterogeneity across agents, we assume that class proportions at each agent are drawn from a Dirichlet distribution parameterized by a concentration parameter $\rho > 0$. Larger values of ρ correspond to more homogeneous data distributions across agents, with the limit $\rho \to \infty$ recovering the i.i.d. setting. Conversely, smaller values of ρ induce increasing non-i.i.d. behavior, resulting in highly skewed local datasets dominated by a subset of classes or sensitive groups. At the outset, the aggregator initializes global model parameters, denoted by Θ_0. In each training round t, every agent k updates its local model parameters $\theta_{k,t}$ using its dataset $\mathcal{D}^{(k)}$ and sends the updated parameters to the aggregator for aggregation.

At each communication round t, the central aggregator computes the global model parameters Θ_t by aggregating local updates $\{\theta_{k,t}\}_{k=1}^K$ according to an aggregation rule `Agg`:

$$\Theta_t = \texttt{Agg}\big(\{\theta_{k,t}\}_{k=1}^K, \{w_k\}_{k=1}^K\big),$$

where $\theta_{k,t}$ denotes the local model parameters of agent k, and w_k is the aggregation weight assigned to agent k. A widely used instantiation is the *Federated Averaging* (`FedAvg`) algorithm McMahan et al. (2017), where the weights are proportional to the number of data points held by each agent. The global model is updated as:

$$\Theta_t^{\texttt{FedAvg}} = \sum_{k \in \mathcal{S}_t} \frac{|\mathcal{X}^{(k)}|}{\sum_{j \in \mathcal{S}_t} |\mathcal{X}^{(j)}|} \cdot \theta_{k,t} \tag{1}$$

Here, $\mathcal{S}_t \subseteq [K]$ represents the (random) set of participating agents in round t, and $|\mathcal{X}^{(k)}|$ denotes the size of agent k's dataset. The weights, $w_k = \frac{|\mathcal{X}^{(k)}|}{\sum_{j \in \mathcal{S}_t} |\mathcal{X}^{(j)}|}$ for each agent k. This weighting ensures that agents with larger datasets have a proportionally greater influence on the global model. The process repeats over multiple rounds until convergence, resulting in a final global model Θ^* at round T.

3.1.1 Robust Aggregators

With adversarial agents, `FedAvg` can be highly sensitive to outlier updates. This sensitivity to outliers has motivated the development of robust aggregation rules such as `KRUM` , `TM` and `FLAME` , which aim to limit the influence of anomalous or adversarial.

m-`KRUM` Blanchard et al. (2017b). m-`KRUM` selects updates that are most consistent with the majority. Given K agents in any FL round, each providing an update θ_k for $k = 1, 2, \ldots, K$, and assuming up to $\tilde{q} = \lfloor qK \rfloor$ adversarial agents, m-`KRUM` proceeds as follows. First, it computes pairwise distances $d_{i,j} = \left\|\theta_i - \theta_j\right\|_2$ for all $i \neq j$. For each agent i, it then selects the set $\mathcal{N}_i$ of the $K - \tilde{q} - 2$ updates closest to θ_i and computes the outlier score $s_i = \sum_{j \in \mathcal{N}_i} d_{i,j}^2$. Next, it chooses the m updates with the smallest scores – ensuring $m \geq K - \tilde{q}$ – and aggregates them via a weighted average proportional to each agents sample size. By excluding highscore (outlying) updates, m-`KRUM` effectively filters adversarial contributions, yielding a more robust global model Θ. The formal aggregation is summarized in Appendix B.1

Trimmed-Mean (`TM`) (Yin et al., 2018). f-`Trimmed-Mean` assumes up to f of these may be adversarial updates and the f largest and f smallest values among $\{\theta_k\}_{k=1}^K$ are removed. Specifically, for each parameter θ, the aggregator collects all corresponding agent values, sorts them element-wise, discards the extreme $2f$ values, and takes the mean of the remaining $K - 2f$ entries to compute the aggregated parameter $\Theta^{\texttt{TM}}$. This process is repeated for all parameters in the model. The formal algorithm is available in Appendix B.2.

`FLAME` (Nguyen et al., 2022). `FLAME` computes pairwise cosine similarities between updates and clusters them to identify a majority benign group. Updates outside this cluster are discarded or down-weighted, and the benign updates are averaged with adaptive Gaussian noise calibrated to their variance, providing robustness under adversarial behavior and non-IID data. The formal procedure is summarized in Appendix B.3.

3.2 Group Fairness

Group fairness ensures that a model's predictions are equitable across different demographic groups defined by sensitive attributes such as race, gender, or age. We consider a parameterized classifier $f_\Theta : \mathcal{X} \to \mathcal{Y}$, where θ are the (learned) model parameters and $\mathcal{X}$ is the input space. For each input sample $x \in \mathcal{X}$, the predicted label is given by $\hat{y} = f_\Theta(x)$. We denote by $\hat{Y}$ the set of predicted labels for a dataset, i.e., $\hat{Y} = \{f_\Theta(x_i)\}_{i=1}^n$ for samples $\{x_i\}_{i=1}^n \subseteq \mathcal{X}$. Formally, we focus on the following notions of fairness.

Demographic Parity (DP) Dwork et al. (2012). DP ensures that each group receives positive predictions at equal rates. In our running example of loan approval, DP looks only at the overall rate of approvals: it requires that f_Θ approves individuals at equal rates across groups a & b, regardless of actual qualification.

Definition 1 (Demographic Parity (DP) Dwork et al. (2012))**.** *A classifier f_Θ satisfies DP if the probability of a positive prediction is the same across all groups, regardless of the actual outcomes. Formally, for all groups $a, b \in \mathcal{A}$:*

$$\Pr(\hat{Y} = 1 \mid \mathcal{A} = a) = \Pr(\hat{Y} = 1 \mid \mathcal{A} = b)$$

Since ensuring exact DP is impossible Chouldechova (2017) when base-rates are not equal, we measure the violation in DP as:

$$\Delta_{DP} := |\Pr(\hat{Y} = 1 \mid \mathcal{A} = a) - \Pr(\hat{Y} = 1 \mid \mathcal{A} = b)| \tag{2}$$

Equalized Odds (EO) Hardt et al. (2016b). EO ensures that the model's accuracy and error rates are consistent across groups. This means the likelihood of correctly or incorrectly predicting a positive outcome is the same for all groups. In our loan approval example, EO ensures that qualified and unqualified individuals are treated similarly across groups a and b, i.e., both true positive and false positive rates align.

Definition 2 (Equalized Odds (EO) Hardt et al. (2016b))**.** *A classifier f_Θ satisfies EO if all groups have equal true positive rates (TPR) and false positive rates (FPR). Formally, for all groups $a, b \in \mathcal{A}$:*

$$\Pr(\hat{Y} = 1 \mid \mathcal{Y} = 1, \mathcal{A} = a) = \Pr(\hat{Y} = 1 \mid \mathcal{Y} = 1, \mathcal{A} = b)$$
$$\Pr(\hat{Y} = 1 \mid \mathcal{Y} = 0, \mathcal{A} = a) = \Pr(\hat{Y} = 1 \mid \mathcal{Y} = 0, \mathcal{A} = b)$$

We define the violation in EO as:

$$\Delta_{EO} := \max\{\Delta_{\texttt{TPR}}, \Delta_{\texttt{FPR}}\}, \text{where} \tag{3}$$

$$\Delta_{\texttt{TPR}} := |\Pr(\hat{Y} = 1 \mid \mathcal{Y} = 1, \mathcal{A} = a) - \Pr(\hat{Y} = 1 \mid \mathcal{Y} = 1, \mathcal{A} = b)|$$
$$\Delta_{\texttt{FPR}} := |\Pr(\hat{Y} = 1 \mid \mathcal{Y} = 0, \mathcal{A} = a) - \Pr(\hat{Y} = 1 \mid \mathcal{Y} = 0, \mathcal{A} = b)|$$

Equal Opportunity (EOpp) Hardt et al. (2016b). EOpp focuses on ensuring that qualified individuals (i.e., those with $\mathcal{Y} = 1$) have an equal chance of being correctly identified by the model, regardless of their group membership. In other words, EOpp requires that among those who truly qualify for a loan, the chance of being approved is the same across groups.

Definition 3 (Equal Opportunity (EOpp) Hardt et al. (2016b))**.** *A classifier f_θ satisfies EOpp if it has equal true positive rates (TPR) across all groups. Formally, for all groups $a, b \in \mathcal{A}$:*

$$\Pr(\hat{Y} = 1 \mid Y = 1, \mathcal{A} = a) = \Pr(\hat{Y} = 1 \mid Y = 1, \mathcal{A} = b)$$

The violation in EOpp is straightforward from Definition 3, and implies that EOpp is a weaker fairness notion than EO.

$$\Delta_{EOpp} := \Delta_{\texttt{TPR}} \tag{4}$$

Accuracy Parity (AP). AP ensures that the classifier achieves equal predictive accuracy across sensitive groups. Unlike EO and EOpp, which condition on specific outcomes, AP enforces fairness in terms of overall correctness.

Definition 4 (Accuracy Parity (AP))**.** *A classifier f_θ satisfies Accuracy Parity (AP) if it has equal accuracy across all groups. Formally, for all groups $a, b \in \mathcal{A}$:*

$$\Pr(\hat{Y} = Y \mid \mathcal{A} = a) = \Pr(\hat{Y} = Y \mid \mathcal{A} = b)$$

We define the violation in Accuracy Parity as:

$$\Delta_{\texttt{AP}} := \Big|\Pr(\hat{Y} = Y \mid \mathcal{A} = a) - \Pr(\hat{Y} = Y \mid \mathcal{A} = b)\Big| \tag{5}$$

4 Methodology

In this section, we formalize the overall optimization objective in the FL setting for classification. We then describe the optimization problem solved by a strategic adversary aiming to amplify group-level bias in the resulting global model. Specifically, the adversary seeks to perform a model poisoning attack that circumvents state-of-the-art robust aggregation mechanisms.

4.1 FL Optimization

Recall that our goal is to train a global classifier $f_\theta : \mathcal{X} \to \mathcal{Y}$ that optimizes for global accuracy across a set of $\mathcal{K}$ agents each having a private dataset $\mathcal{D}^{(k)} = \left\{(x_i^{(k)}, y_i^{(k)}, a_i^{(k)})\right\}_{i=1}^{n_k}$. Each agent $k \in \mathcal{K}$ defines a local objective $F_k(\theta)$ given by,

$$F_k(\theta) = \mathbb{E}_{x^{(k)}, y^{(k)} \sim \mathcal{D}^{(k)}}\left[\ell_{CE}(\theta; x^{(k)}, y^{(k)})\right] \tag{6}$$

where $\ell_{CE}(\cdot)$ is the standard cross-entropy loss. The global objective is to minimize the **weighted aggregation** of local losses: $\min_\Theta F(\Theta) = \sum_{k=1}^{K} w_k F_k(\Theta)$. Here, Θ denotes the model parameters to be optimized, $w_k = \frac{n_k}{n}$ is the weight for agent k, with $n_k = |\mathcal{D}_k|$ being the size of agent k's dataset, and $n = \sum_{k=1}^{K} n_k$ representing the total number of data points across all agents.

4.1.1 FL Optimization: Adversarial agent

When a subset of agents is malicious, they aim to compromise the fairness of the global model by launching model poisoning attacks. Let $K_A \subset K$ denote the set of *adversarial agents* among the K total agents. While honest agents minimize the standard local objective defined in Equation 6, adversarial agents aim to increase the demographic bias of the aggregated model while maintaining acceptable predictive accuracy. Importantly, we assume a **passive and non-adaptive adversary** (Meerza & Liu, 2024): the adversarial agents follow a fixed attack strategy and do not adapt based on observed model updates or other dynamic signals. Each adversarial agent $k \in K_A$ solves the following:

$$\begin{array}{ll} \max\limits_{\theta} \ \ell_F(\theta; \mathcal{D}^{(k)}) & \triangleright \text{ Maximize Bias} \\ \text{s.t. } \ \mathbb{E}[\ell_{CE}(\theta, x^{(k)}, y^{(k)})] \leq \epsilon & \triangleright \text{ Maintain Accuracy} \end{array}$$

Here, $\ell_F(\cdot)$ is a differentiable surrogate objective designed to increase group-level fairness violations (e.g., for Demographic Parity or Equal Opportunity), and ℓ_{CE} is the standard cross-entropy loss. The threshold ϵ defines the maximum allowable performance degradation – controlling the *stealthiness* of the attack. Without this constraint, the adversary's update could be easily flagged by robust aggregators due to poor accuracy.

Surrogate for Equalized Odds (Padala & Gujar, 2020). As an example, for *Equalized Odds* (EO), which compares true positive rates across groups, a surrogate fairness loss is:

$$\ell_{EO} = \left| \frac{\sum_i (1-p_i) a_i y_i}{\sum_i a_i y_i} - \frac{\sum_i (1-p_i)(1-a_i) y_i}{\sum_i (1-a_i) y_i} \right| + \left| \frac{\sum_i p_i a_i (1-y_i)}{\sum_i a_i (1-y_i)} - \frac{\sum_i p_i (1-a_i)(1-y_i)}{\sum_i (1-a_i)(1-y_i)} \right|$$

Where $p_i = f_\theta(x_i)$ denotes the predicted logits (or probability scores), $a_i \in \{0, 1\}$ indicates binary group membership (e.g., gender), and $y_i \in \{0, 1\}$ is the ground-truth label. Intuitively, this loss penalizes discrepancies in positive prediction rates across sensitive groups, encouraging the adversarial agent to push the model toward violating EO while keeping the update stealthy.

4.1.2 Naïve Model Poisoning Attack

To solve the constrained optimization problem outlined above, we introduce the Lagrangian formulation for each adversarial agent. The Lagrangian multiplier $\lambda \in \mathbb{R}_{\geq 0}$ will enforce the constraint on $\mathbb{E}[\ell_{CE}(\cdot)] \leq \epsilon$ while allowing the adversarial agent to optimize for fairness (or bias) maximization. For all $k \in \mathcal{K}_A$,

$$F_k(\theta) = -\ell_F(\theta; \mathcal{D}^{(k)}) + \lambda \cdot \left(\mathbb{E}\left[\ell_{CE}(\theta, x^{(k)}, y^{(k)}) \right] - \epsilon \right) \tag{7}$$

The empirical version of the above objective for $\epsilon = 0$ is given by,

$$F_k(\theta) = -\ell_F(\theta; \mathcal{D}^{(k)}) + \lambda \cdot \frac{1}{n_k} \sum_{i=1}^{n_k} \ell_{CE}(\theta; x_i^{(k)}, y_i^{(k)}) \tag{8}$$

Equation 8 optimizes for maximizing fairness violation and minimizing accuracy simultaneously. The optimal parameter updates deviate significantly from an honest agent that solves Equation 6. Thus, the scores computed in m-KRUM (Algorithm B.1 (Line 4)) would easily help detect the adversarial agent. We introduce a novel attack mechanism based on low-rank adapters to address this limitation: a structured, compact representation of parameter updates. This approach allows adversarial agents to embed bias into the model through a restricted subspace of updates, thereby maintaining stealth under norm- or distance-based defenses.

Algorithm 1 `LoRA-FL`

Require: Global model parameters Θ_0, number of rounds T, local epochs E, adversarial local epochs $E_{\mathcal{A}}$, number of agents m, agent optimizer `OPT`, adversarial optimizers $\texttt{OPT}_{\texttt{REG}}, \texttt{OPT}_{\texttt{F}}$, scaling factor $\alpha \in (0, 1]$, aggregator function `Agg`

Ensure: Aggregated global model Θ_T

1: **for** each round $t = 1, 2, \ldots, T$ **do**
2: Server samples a subset of agents $\mathcal{S}_t \subseteq \{1, \ldots, K\}$
3: **for** each agent $i \in \mathcal{S}_t$ **in parallel do**
4: $\theta_{i,t} \leftarrow \Theta_t$
/ Agent i updates $\theta_{i,t}$ locally using optimizer `OPT` */*
5: **for** each local epoch $e = 1$ to E **do**
6: $\theta_{i,t} \leftarrow \texttt{OPT}(\theta_{i,t}, \nabla \ell_{\texttt{CE}}(\theta_{i,t}; \mathcal{D}_i))$
7: **end for**
8: **if** i is Adversarial **then**
9: Initialize adapter parameters $A_{i,t}, B_{i,t}$
10: **for** each adversarial local epoch $e = 1$ to $E_{\mathcal{A}}$ **do**
11: **for** each batch in adversarial data **do**
/ Phase 1: Train adapters for Accuracy */*
12: $(A_{i,t}, B_{i,t}) \leftarrow \texttt{OPT}_{\texttt{REG}}(A_{i,t}, B_{i,t}, \nabla \ell_{\texttt{REG}}(\cdot))$
/ Phase 2: Train Adapters to Compromise Fairness*/*
13: $(A_{i,t}, B_{i,t}) \leftarrow \texttt{OPT}_{\texttt{F}}(A_{i,t}, B_{i,t}, -\nabla \ell_{\texttt{F}}(\cdot))$
14: **end for**
15: **end for**
16: $\theta_{i,t} \leftarrow \Theta_t + \alpha \cdot A_{i,t} B_{i,t}^{\top}$
17: **end if**
18: Agent sends updated model $\theta_{i,t}$ to server
19: **end for**
20: Server aggregates agent updates:

$$\Theta_{t+1} \leftarrow \texttt{Agg}(\{\theta_{i,t}\}_{i \in [n]})$$

21: **end for**
22: **return** Final global model Θ_T

4.2 `LoRA-FL`: Achieving Unfairness through Low-rank Adapters in FL

Notice that the update to agent k's local model after training at round t is the decomposition $\theta_{k,t} = \Theta_{t-1} + \Delta\theta$, where $\Delta\theta$ represents the parameter change during the local update. Intuitively, an adversary's objective (from the discussion in Section 4.1.2) is that the update $\Delta\theta$ encodes information that compromises fairness, while remaining close (in the parameter space) to Θ to avoid detection by robust aggregators.

In our attack, namely `LoRA-FL`, an adversary achieves its objective by training low-rank matrices (aka *adapters*) that replace $\Delta\theta$, ensuring the desired behavior. Algorithm 1 presents the formal attack. For our discussion, consider the local model $\theta \in \mathbb{R}^{d \times k}$ with adapters $A \in \mathbb{R}^{d \times r}$ and $B \in \mathbb{R}^{k \times r}$ as low-rank matrices such that $r \ll \min(d, k)$.

Local Training. Each agent $k \in [K]$ (whether adversarial or benign) performs standard local updates as in conventional FL (Line 6, Algorithm 1). Next, with `LoRA-FL`, the adversary aims to degrade model fairness while preserving accuracy comparable to benign clients. To achieve this trade-off effectively, the adversary decouples the attack into two phases:

Phase 1: Train Adapters for Accuracy. The first phase focuses on stealth part of the attack, by improving the adapters' accuracy, using a regularizer that constrains the adapters to be close to $\Delta\theta$. Formally, for $\tilde{\theta}_{k,t}$ as the current model (Line 12, Algorithm 1),

$$\ell_{\texttt{REG}}(A_{k,t}, B_{k,t}; \cdot) := \|A_{k,t} \cdot B_{k,t}^{\top} - (\Theta_{t-1} - \tilde{\theta}_{k,t})\|_2 \tag{9}$$

The optimizer $\texttt{OPT}_{\texttt{REG}}$ minimizes $\ell_{\texttt{REG}}$ during Phase 1, effectively driving the low-rank update $A_{k,t} \cdot B_{k,t}^{\top}$ to be close to $\Delta\theta$ – providing performance gains and avoiding detection.

Table 1: Comparison of different aggregators on accuracy and fairness metrics across the three tabular datasets for the **IID** setting. Here, $r = 4$ for Adult and Dutch, and $r = 2$ for Bank. **Acc** denotes accuracy, **Adv** denotes adversarial clients, and results are reported as $\texttt{mean}_{\texttt{std}}$ over four independent runs. **Bold** values indicate highest accuracy and highest fairness violation, highlighting the strongest performance and most severe fairness degradation, respectively.

Agg	% Adv	Dataset											
		Adult				Bank				Dutch			
		Acc (↑)	Δ_{EO} (↓)	Δ_{EOpp} (↓)	Δ_{DP} (↓)	**Acc** (↑)	Δ_{EO} (↓)	Δ_{EOpp} (↓)	Δ_{DP} (↓)	**Acc** (↑)	Δ_{EO} (↓)	Δ_{EOpp} (↓)	Δ_{DP} (↓)
FedAvg	–	$85.00_{0.10}$	$0.118_{0.007}$	$0.096_{0.016}$	$0.185_{0.006}$	$91.68_{0.09}$	$0.151_{0.020}$	$0.111_{0.026}$	$\mathbf{0.211}_{0.043}$	$\mathbf{82.24}_{0.29}$	$0.061_{0.010}$	$0.051_{0.012}$	$\mathbf{0.188}_{0.009}$
KRUM	–	$84.76_{0.08}$	$\mathbf{0.129}_{0.019}$	$\mathbf{0.106}_{0.014}$	$\mathbf{0.195}_{0.011}$	$91.49_{0.08}$	$\mathbf{0.156}_{0.026}$	$\mathbf{0.122}_{0.018}$	$0.203_{0.039}$	$82.26_{0.20}$	$0.061_{0.011}$	$0.052_{0.015}$	$0.187_{0.011}$
TM	–	$\mathbf{85.05}_{0.08}$	$0.101_{0.012}$	$0.073_{0.011}$	$0.185_{0.015}$	$\mathbf{91.70}_{0.02}$	$0.149_{0.020}$	$0.117_{0.027}$	$0.199_{0.036}$	$80.07_{0.75}$	$\mathbf{0.073}_{0.009}$	$\mathbf{0.054}_{0.017}$	$0.179_{0.016}$
FedAvg	10%	$\mathbf{84.73}_{0.19}$	$0.152_{0.016}$	$0.131_{0.018}$	$0.218_{0.008}$	$\mathbf{91.68}_{0.11}$	$0.172_{0.0167}$	$0.140_{0.0189}$	$0.238_{0.0430}$	$\mathbf{80.50}_{0.20}$	$0.072_{0.002}$	$0.029_{0.006}$	$0.207_{0.009}$
FedAvg	20%	$84.39_{0.14}$	$0.187_{0.030}$	$0.171_{0.035}$	$0.238_{0.009}$	$91.38_{0.16}$	$0.199_{0.028}$	$0.135_{0.020}$	$0.269_{0.040}$	$78.18_{0.55}$	$0.125_{0.015}$	$0.063_{0.009}$	$0.254_{0.006}$
FedAvg	30%	$83.51_{0.25}$	$0.266_{0.030}$	$0.261_{0.032}$	$0.285_{0.008}$	$90.83_{0.24}$	$0.262_{0.0150}$	$0.166_{0.0314}$	$0.338_{0.0377}$	$74.36_{0.92}$	$0.189_{0.014}$	$0.125_{0.020}$	$0.290_{0.011}$
FedAvg	40%	$81.95_{0.43}$	$\mathbf{0.374}_{0.052}$	$\mathbf{0.371}_{0.055}$	$\mathbf{0.344}_{0.016}$	$89.86_{0.30}$	$\mathbf{0.312}_{0.015}$	$\mathbf{0.174}_{0.029}$	$\mathbf{0.384}_{0.016}$	$70.08_{0.48}$	$\mathbf{0.244}_{0.010}$	$\mathbf{0.216}_{0.021}$	$\mathbf{0.326}_{0.009}$
KRUM	10%	$\mathbf{84.56}_{0.28}$	$0.157_{0.017}$	$0.135_{0.019}$	$0.213_{0.016}$	$\mathbf{91.39}_{0.21}$	$0.174_{0.028}$	$0.136_{0.025}$	$0.222_{0.044}$	$\mathbf{82.11}_{0.22}$	$0.065_{0.011}$	$0.053_{0.010}$	$0.188_{0.009}$
KRUM	20%	$84.14_{0.56}$	$0.181_{0.068}$	$0.157_{0.070}$	$0.234_{0.043}$	$91.21_{0.36}$	$0.196_{0.055}$	$0.150_{0.037}$	$0.258_{0.068}$	$78.85_{0.41}$	$0.091_{0.008}$	$0.029_{0.007}$	$0.220_{0.006}$
KRUM	30%	$82.87_{1.25}$	$\mathbf{0.372}_{0.119}$	$\mathbf{0.367}_{0.125}$	$\mathbf{0.310}_{0.052}$	$90.97_{0.15}$	$0.225_{0.047}$	$0.157_{0.039}$	$0.287_{0.052}$	$77.84_{0.13}$	$0.115_{0.007}$	$0.055_{0.015}$	$0.242_{0.010}$
KRUM	40%	$81.76_{2.12}$	$0.318_{0.130}$	$0.304_{0.143}$	$0.304_{0.068}$	$87.92_{0.70}$	$\mathbf{0.396}_{0.036}$	$\mathbf{0.211}_{0.019}$	$\mathbf{0.463}_{0.046}$	$76.85_{0.29}$	$\mathbf{0.122}_{0.008}$	$\mathbf{0.076}_{0.007}$	$\mathbf{0.250}_{0.006}$
TM	10%	$\mathbf{84.73}_{0.26}$	$0.108_{0.010}$	$0.091_{0.013}$	$0.170_{0.019}$	$\mathbf{91.60}_{0.13}$	$0.165_{0.027}$	$0.139_{0.025}$	$0.215_{0.065}$	$79.12_{1.39}$	$0.081_{0.028}$	$0.027_{0.007}$	$0.207_{0.025}$
TM	20%	$84.33_{0.55}$	$0.197_{0.033}$	$0.195_{0.035}$	$0.183_{0.028}$	$91.37_{0.13}$	$0.197_{0.023}$	$0.142_{0.019}$	$0.255_{0.042}$	$\mathbf{79.79}_{0.37}$	$0.087_{0.008}$	$0.025_{0.006}$	$0.220_{0.009}$
TM	30%	$84.24_{0.33}$	$0.201_{0.052}$	$0.192_{0.060}$	$0.224_{0.011}$	$91.16_{0.13}$	$0.216_{0.006}$	$0.159_{0.024}$	$0.278_{0.031}$	$79.43_{0.32}$	$0.087_{0.011}$	$0.033_{0.004}$	$0.227_{0.009}$
TM	40%	$82.82_{1.05}$	$\mathbf{0.281}_{0.076}$	$\mathbf{0.280}_{0.077}$	$\mathbf{0.270}_{0.053}$	$90.21_{0.35}$	$\mathbf{0.299}_{0.030}$	$\mathbf{0.173}_{0.017}$	$\mathbf{0.358}_{0.053}$	$79.18_{0.64}$	$\mathbf{0.100}_{0.013}$	$\mathbf{0.036}_{0.013}$	$\mathbf{0.235}_{0.010}$

Phase 2: Train Adapters to Compromise Fairness. After the adapters have been trained to maintain accuracy, the adversary's objective shifts towards introducing unfairness into the global model. The adversary aims to minimize the fairness loss $\ell_{\texttt{UF}}$ concerning the adapter parameters. Formally, for $\tilde{\theta}_{k,t}$ as the current model (Line 13, Algorithm 1), this loss function is designed to maximize the bias in the model's predictions. Specifically,

$$\ell_{\texttt{UF}}(A_{i,t}, B_{i,t}; \cdot) := -\ell_F \quad \text{s.t.} \quad F \in \{\text{EO}, \text{EOpp}, \text{DP}, \text{AP}\} \tag{10}$$

where ℓ_F represents a surrogate fairness loss for the chosen fairness metric. The optimizer $\texttt{OPT}_{\texttt{F}}$ minimizes $\ell_{\texttt{UF}}$ during this phase, effectively guiding the adapter parameters to introduce unfairness.

Communication & Parameter Complexity. In `LoRA-FL`, an adversary incurs no additional communication cost compared to the standard FL setup. From Algorithm 1 (Line 16), the adversary fuses the adapters into the base model and communicates the resulting model, identical to that of honest agents, to the Aggregator. For a base model with dimensions $d \times k$ and adapter parameters $d \times r$ and $r \times k$, the total number of parameters is $\mathcal{O}(d \times k + r \times (d + k))$. Since the adapters are low-rank, i.e., $r \ll \min(d, k)$, the parameter increase is slight and *independent* in terms of d and k. Hence, for both honest and adversarial agents, the effective count is $\mathcal{O}(d \times k)$, with minimal overhead from the adapters.

5 Experiments & Results

Datasets. We evaluate our approach on four datasets spanning both tabular and image domains. Three are tabular binary classification datasets: **Adult** (Dua & Graff, 2017), which predicts whether an individuals income exceeds \$50K using demographic attributes with `sex` and `race` treated as sensitive attributes; **Bank** (Moro et al., 2014), which predicts subscription to a term deposit based on demographic and contact-related features, where `age` (individuals aged 25-60 as the privileged group) is the sensitive attribute; **Dutch Census** (liobaite et al., 2011), which predicts occupation and uses `gender` as the binary sensitive attribute. These tabular datasets contain approximately 40K–60K samples each. We also consider image-based classification using **UTKFace** (Zhang et al., 2017), a dataset of approximately 20K face images. The task is *multi-class* ethnicity classification and `gender` is treated as the sensitive attribute. Together, these datasets enable evaluation of fairness and robustness across diverse data modalities and classification settings.

IID and Non-IID Client Data. We evaluate our approach under both IID and non-IID FL settings. In the IID setting, each client's local dataset is formed by uniformly sampling from the global dataset, so that all clients share the same distribution over features and labels. In the non-IID setting, clients possess partitioned data using a Dirichlet distribution (Ng et al., 2011) over *class labels* with concentration parameter $\rho \in \{0.25, 0.5, 0.75\}$, where smaller values of ρ induce stronger label and feature skew across clients.

5.1 Model Architectures and Adapter Placement

5.1.1 Adapters in Dense Layers (Tabular Datasets)

In the first FL setup, corresponding to tabular datasets, each honest client employs a two-layer multilayer perceptron (MLP) with hidden layer sizes of 64 and 32, together with dataset-specific input and output heads. Adversarial agents augment this architecture by inserting low-rank adapters into the hidden layers of the MLP, with adapter rank set to $r = 4$. All agents use ReLU activations (Nair & Hinton, 2010). Optimization is performed using AdamW (Loshchilov & Hutter, 2019) without momentum for both honest and adversarial updates, denoted by `OPT`, $\texttt{OPT}_{\texttt{F}}$, and $\texttt{OPT}_{\texttt{REG}}$. Additional training details and hyperparameter settings are provided in Appendix A.1.

5.1.2 Adapters in CNN and Dense Layers (Image Datasets)

In the second FL setup, corresponding to image datasets, each honest agent uses an untrained ResNet-18 backbone (He et al., 2016) followed by a two-layer MLP classification head identical to the tabular setting. This setting is substantially more challenging than the tabular case due to the depth and hierarchical structure of convolutional representations, which distribute task- and fairness-relevant features across multiple layers. Adversarial agents insert low-rank adapters into both the convolutional layers of the ResNet backbone and the subsequent dense layers. This design choice is non-trivial: restricting adapters to only the final classification head is often insufficient to influence higher-level semantic features that correlate with sensitive attributes. By fusing adapters across both convolutional and dense layers, adversarial agents can steer fairness-relevant representations throughout the network while maintaining accuracy. The adapter design follows prior work on parameter-efficient fine-tuning of convolutional networks, such as LoRA-C (Ding et al., 2024). All experiments are implemented in PyTorch (Paszke et al., 2019) and conducted on an NVIDIA L40S GPU with 48 GB memory. Further hyperparameter details are provided in Appendix A.1.

5.1.3 Performance & Fairness Measures

We benchmark `LoRA-FL`'s efficacy based on accuracy as our performance measure, and Δ_{DP}, Δ_{EOpp}, and Δ_{EO} as fairness measures for binary classifications and Δ_{AP} as a fairness metric for multiclass classification. Here, we note that improvements in fairness typically come at the cost of accuracy (Bilal Zafar et al., 2015; Madras et al., 2018). The trade-off between performance and fairness measures is a *crucial* factor[1] in evaluating the overall performance.

5.1.4 Other Details

We report **average** values across all agents using 25% of their local data for testing. We benchmark performance and fairness against `FedAvg`, `KRUM`, and `TM` for IID distributions and include `FLAME` for non-IID distributions. Adversarial agent percentages varies across {10%, 20%, 30%, 40% }.

For `KRUM` and `TM`, we take $m = 60\%$ of all clients for all adversary settings. This corresponds to the *worst-case* permissible choice across our experiments and ensures the aggregation is defined even for the highest adversary ratio. Additionally, we do not evaluate `LoRA-FL` against fair aggregators like FairFed (Ezzeldin et al., 2023) because adversarial clients can easily manipulate these systems by reporting inaccurate fairness scores, rendering them ineffective.

[1] **Example:** A model always predicting a single class may appear fair under Demographic Parity (DP), but its poor accuracy makes it ineffective.

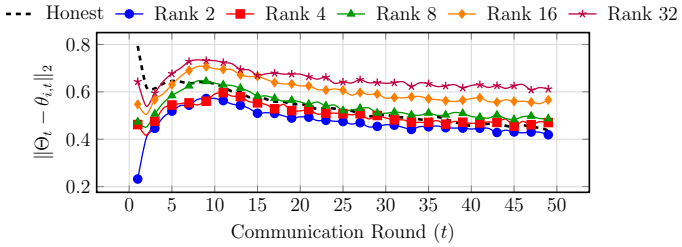

Figure 2: Effect of low-rank constraints on model divergence. As the rank increases, the ℓ_2 distance $\|\Theta_t - \theta_{i,t}\|_2$ grows, indicating that adversarial updates diverge more from the global model.

5.2 Results

Table 1 and Table 2 present results across tabular datasets under both IID and non-IID data distributions, The numbers show the effectiveness of `LoRA-FL` across aggregation rules, datasets, and non-IID parameter ρ. To evaluate cross-domain generality of `LoRA-FL`, we perform image-based multi-class classification on UTKFace (Table 3) and compare against existing attacks such as EAB-FL (Meerza & Liu, 2024) (Table 4). Details follow.

Effectiveness of `LoRA-FL` across aggregators. Across both IID (Table 1) and non-IID settings (Table 2), `LoRA-FL` consistently achieves a strong trade-off between accuracy preservation and fairness degradation, even in the presence of robust aggregation rules. Under IID data, `LoRA-FL` induces substantial increases in fairness gaps with minimal loss in accuracy across `FedAvg`, `KRUM`, and `TM`. Notably, this behavior persists across non-IID data: for all values of ρ, both `FLAME` and `KRUM` exhibit fairness degradation patterns comparable to those of `FedAvg`. Overall, the results from these tables demonstrate that `LoRA-FL` generalizes across aggregation rules and data regimes.

In Table 1 for Adult with `TM`, under a 30% adversary setup, accuracy declines by 0.81% (from 85.05% to 84.24%), yet the EO gap nearly doubles – from 0.101 to 0.201 (a 99.0% surge) – and the DP gap rises from 0.185 to 0.224 (21.1%). Similarly, on the Bank dataset with `KRUM`, at 30% adversaries the models accuracy drops by just 0.52% (from 91.49% to 90.97%), while the EO gap increases from 0.156 to 0.225 (44.2%) and the DP gap from 0.203 to 0.287 (41.4%). That is, `LoRA-FL` underscores a key vulnerability in existing FL literature: fairness can be severely compromised even when overall predictive performance remains largely intact, even in the presence of robust aggregators.

Effect of data heterogeneity (ρ). Table 2 shows that data heterogeneity substantially amplifies the impact of `LoRA-FL`. As ρ decreases (i.e., client data becomes more non-IID), fairness violations increase sharply across all aggregators and datasets, often accompanied by higher variance. In contrast, when ρ is large, fairness degradation remains pronounced but is relatively stable. When compared against the IID baseline under the same rank in Table 1, the results indicate that non-IID data not only worsens fairness outcomes but also obscures adversarial behavior, allowing `LoRA-FL` to remain stealthy while inducing larger disparities.

Dataset-dependent behavior. Comparing across datasets in both tables reveals consistent but dataset-specific vulnerability patterns. The **Adult** and **Bank** datasets exhibit the most severe fairness degradation under `LoRA-FL`, with EO and DP gaps often increasing by 40–100% at moderate adversarial fractions while accuracy drops remain below 1–2%. These effects are further magnified under non-IID settings, particularly for smaller values of ρ. In contrast, the **Dutch** dataset shows smaller absolute fairness gaps in both IID and non-IID regimes, though the monotonic increase in violations with adversarial participation persists. This suggests that while `LoRA-FL` is broadly effective, datasets with richer demographic structure and stronger correlations between sensitive attributes and labels are especially susceptible to fairness-targeted manipulation.

`LoRA-FL` generalizes across Domain. As discussed earlier, we evaluate `LoRA-FL` on UTKFace dataset by considering multi-class image classification using a ResNet-based convolutional architecture. Table 3

Table 2: Comparison of different aggregators on accuracy and fairness metrics across the three tabular datasets for the **non-IID** setting. Here, $\rho \in \{0.5, 0.75\}$ and $r = 2$ for all the datasets. Also, **Acc**: Accuracy, **Adv**: Adversary, and we report $\texttt{mean}_{\texttt{std}}$ across **four** independent runs. **Bold** values indicate highest accuracy and highest fairness violation, highlighting the strongest performance and most severe fairness degradation, respectively.

	Method	% Adv	$\rho = 0.5$				$\rho = 0.75$			
			Acc (↑)	Δ_{EO} (↓)	Δ_{EOpp} (↓)	Δ_{DP} (↓)	**Acc** (↑)	Δ_{EO} (↓)	Δ_{EOpp} (↓)	Δ_{DP} (↓)
Adult	FedAvg	–	$\mathbf{84.80}_{0.58}$	$0.161_{0.024}$	$0.150_{0.026}$	$0.191_{0.018}$	$\mathbf{84.72}_{0.78}$	$0.162_{0.016}$	$0.139_{0.014}$	$0.193_{0.011}$
		10%	$84.79_{0.59}$	$0.168_{0.033}$	$0.159_{0.038}$	$0.195_{0.012}$	$84.55_{0.72}$	$0.159_{0.013}$	$0.137_{0.013}$	$0.195_{0.007}$
		20%	$84.55_{0.51}$	$0.168_{0.026}$	$0.156_{0.034}$	$0.205_{0.015}$	$83.87_{1.48}$	$0.180_{0.034}$	$0.146_{0.025}$	$0.221_{0.020}$
		30%	$84.58_{0.65}$	$0.176_{0.025}$	$0.165_{0.033}$	$0.212_{0.007}$	$83.22_{2.13}$	$0.201_{0.068}$	$0.173_{0.058}$	$0.245_{0.048}$
		40%	$82.91_{1.37}$	$\mathbf{0.240}_{0.043}$	$\mathbf{0.219}_{0.044}$	$\mathbf{0.265}_{0.034}$	$82.88_{0.73}$	$\mathbf{0.204}_{0.026}$	$\mathbf{0.175}_{0.025}$	$\mathbf{0.253}_{0.039}$
	FLAME	–	$\mathbf{84.81}_{0.59}$	$0.162_{0.027}$	$0.151_{0.032}$	$0.192_{0.019}$	$\mathbf{84.76}_{0.73}$	$0.164_{0.017}$	$0.140_{0.014}$	$0.194_{0.011}$
		10%	$84.71_{0.64}$	$0.158_{0.023}$	$0.144_{0.027}$	$0.195_{0.014}$	$84.58_{0.79}$	$0.161_{0.014}$	$0.135_{0.012}$	$0.195_{0.007}$
		20%	$84.48_{0.55}$	$0.163_{0.028}$	$0.149_{0.036}$	$0.204_{0.014}$	$83.84_{1.48}$	$0.180_{0.035}$	$0.147_{0.025}$	$0.221_{0.020}$
		30%	$84.55_{0.59}$	$0.175_{0.023}$	$0.164_{0.031}$	$0.211_{0.007}$	$83.14_{2.19}$	$0.203_{0.070}$	$0.173_{0.060}$	$0.245_{0.052}$
		40%	$82.89_{1.36}$	$\mathbf{0.245}_{0.045}$	$\mathbf{0.223}_{0.048}$	$\mathbf{0.266}_{0.032}$	$82.86_{0.62}$	$\mathbf{0.204}_{0.027}$	$\mathbf{0.174}_{0.024}$	$\mathbf{0.257}_{0.043}$
	KRUM	–	$\mathbf{80.34}_{1.05}$	$0.264_{0.044}$	$0.212_{0.043}$	$0.308_{0.035}$	$\mathbf{81.79}_{1.03}$	$0.253_{0.042}$	$0.206_{0.045}$	$0.273_{0.022}$
		10%	$80.81_{0.90}$	$0.279_{0.064}$	$0.242_{0.075}$	$0.304_{0.021}$	$81.59_{1.98}$	$0.259_{0.080}$	$0.213_{0.086}$	$0.283_{0.046}$
		20%	$80.46_{1.26}$	$0.288_{0.056}$	$0.243_{0.058}$	$0.312_{0.038}$	$80.71_{1.51}$	$0.265_{0.019}$	$0.207_{0.029}$	$0.301_{0.051}$
		30%	$79.67_{0.91}$	$0.335_{0.053}$	$0.294_{0.061}$	$0.349_{0.033}$	$79.50_{1.84}$	$0.313_{0.033}$	$0.254_{0.038}$	$0.343_{0.055}$
		40%	$75.56_{6.91}$	$\mathbf{0.412}_{0.172}$	$\mathbf{0.387}_{0.162}$	$\mathbf{0.363}_{0.107}$	$78.30_{3.42}$	$\mathbf{0.435}_{0.213}$	$\mathbf{0.423}_{0.221}$	$\mathbf{0.383}_{0.127}$
	TM	–	$\mathbf{84.00}_{0.29}$	$0.167_{0.063}$	$0.148_{0.067}$	$0.224_{0.025}$	$83.73_{1.15}$	$0.177_{0.053}$	$0.151_{0.058}$	$0.217_{0.020}$
		10%	$83.76_{0.36}$	$0.291_{0.114}$	$0.280_{0.117}$	$0.264_{0.040}$	$\mathbf{84.20}_{1.06}$	$0.171_{0.081}$	$0.142_{0.095}$	$0.214_{0.040}$
		20%	$83.49_{0.42}$	$0.270_{0.100}$	$0.254_{0.118}$	$0.257_{0.036}$	$83.77_{0.88}$	$0.224_{0.030}$	$0.196_{0.046}$	$0.232_{0.032}$
		30%	$82.09_{1.48}$	$0.405_{0.187}$	$0.400_{0.190}$	$0.326_{0.077}$	$82.79_{1.58}$	$0.351_{0.154}$	$0.342_{0.159}$	$0.279_{0.057}$
		40%	$78.30_{4.01}$	$\mathbf{0.607}_{0.256}$	$\mathbf{0.599}_{0.270}$	$\mathbf{0.427}_{0.137}$	$81.32_{2.53}$	$\mathbf{0.527}_{0.200}$	$\mathbf{0.520}_{0.209}$	$\mathbf{0.322}_{0.099}$
Bank	FedAvg	–	$\mathbf{91.31}_{0.41}$	$0.272_{0.055}$	$0.212_{0.063}$	$0.247_{0.021}$	$90.89_{0.21}$	$0.203_{0.038}$	$0.143_{0.038}$	$0.240_{0.019}$
		10%	$91.24_{0.41}$	$0.280_{0.052}$	$0.209_{0.063}$	$0.267_{0.025}$	$\mathbf{90.94}_{0.24}$	$0.208_{0.038}$	$0.146_{0.043}$	$0.244_{0.017}$
		20%	$90.97_{0.32}$	$0.292_{0.057}$	$0.205_{0.060}$	$0.295_{0.042}$	$90.01_{1.75}$	$0.247_{0.090}$	$0.145_{0.042}$	$0.296_{0.080}$
		30%	$90.96_{0.23}$	$0.310_{0.059}$	$\mathbf{0.219}_{0.076}$	$0.315_{0.030}$	$89.14_{3.09}$	$0.293_{0.084}$	$0.186_{0.035}$	$0.335_{0.071}$
		40%	$90.01_{0.84}$	$\mathbf{0.350}_{0.085}$	$0.216_{0.070}$	$\mathbf{0.387}_{0.067}$	$85.00_{10.20}$	$\mathbf{0.345}_{0.083}$	$\mathbf{0.192}_{0.048}$	$\mathbf{0.365}_{0.046}$
	FLAME	–	$\mathbf{91.30}_{0.47}$	$0.275_{0.054}$	$0.214_{0.063}$	$0.244_{0.027}$	$90.90_{0.22}$	$0.199_{0.038}$	$0.143_{0.043}$	$0.238_{0.018}$
		10%	$91.25_{0.41}$	$0.279_{0.051}$	$0.210_{0.061}$	$0.267_{0.027}$	$\mathbf{90.96}_{0.22}$	$0.207_{0.036}$	$0.147_{0.042}$	$0.239_{0.016}$
		20%	$90.99_{0.35}$	$0.287_{0.057}$	$0.203_{0.064}$	$0.298_{0.032}$	$90.00_{1.80}$	$0.252_{0.094}$	$0.155_{0.037}$	$0.302_{0.086}$
		30%	$90.90_{0.21}$	$0.305_{0.060}$	$0.214_{0.073}$	$0.315_{0.029}$	$89.13_{3.08}$	$0.291_{0.082}$	$0.182_{0.033}$	$0.334_{0.071}$
		40%	$90.00_{0.77}$	$\mathbf{0.351}_{0.083}$	$\mathbf{0.215}_{0.070}$	$\mathbf{0.392}_{0.064}$	$85.42_{9.25}$	$\mathbf{0.338}_{0.075}$	$\mathbf{0.193}_{0.050}$	$\mathbf{0.360}_{0.046}$
	KRUM	–	$\mathbf{89.42}_{0.72}$	$0.404_{0.081}$	$0.306_{0.092}$	$0.425_{0.073}$	$\mathbf{89.39}_{0.44}$	$0.356_{0.042}$	$0.221_{0.029}$	$0.392_{0.077}$
		10%	$87.71_{1.23}$	$0.451_{0.087}$	$0.244_{0.078}$	$0.485_{0.061}$	$88.33_{1.22}$	$0.369_{0.073}$	$0.224_{0.051}$	$0.401_{0.053}$
		20%	$87.23_{0.84}$	$0.484_{0.063}$	$0.278_{0.044}$	$0.502_{0.055}$	$87.81_{1.43}$	$0.393_{0.061}$	$0.221_{0.056}$	$0.430_{0.053}$
		30%	$85.54_{2.41}$	$0.494_{0.014}$	$0.246_{0.064}$	$0.479_{0.063}$	$86.45_{1.53}$	$0.415_{0.079}$	$0.248_{0.067}$	$0.448_{0.078}$
		40%	$84.97_{2.12}$	$\mathbf{0.516}_{0.053}$	$\mathbf{0.309}_{0.064}$	$\mathbf{0.512}_{0.022}$	$80.88_{6.45}$	$\mathbf{0.439}_{0.021}$	$\mathbf{0.263}_{0.061}$	$\mathbf{0.474}_{0.034}$
	TM	–	$\mathbf{90.89}_{0.74}$	$0.288_{0.084}$	$0.193_{0.086}$	$0.285_{0.070}$	$90.90_{0.79}$	$0.222_{0.045}$	$0.175_{0.034}$	$0.259_{0.035}$
		10%	$90.75_{0.54}$	$0.301_{0.057}$	$0.234_{0.079}$	$0.297_{0.079}$	$\mathbf{91.35}_{0.31}$	$0.204_{0.054}$	$0.152_{0.038}$	$0.246_{0.042}$
		20%	$90.13_{1.07}$	$0.399_{1.23}$	$0.292_{0.117}$	$0.404_{0.128}$	$90.45_{0.89}$	$0.250_{0.085}$	$\mathbf{0.179}_{0.050}$	$0.260_{0.061}$
		30%	$88.62_{1.14}$	$0.427_{0.069}$	$0.313_{0.081}$	$0.481_{0.056}$	$86.38_{3.83}$	$0.336_{0.067}$	$0.151_{0.058}$	$0.380_{0.062}$
		40%	$86.77_{2.99}$	$\mathbf{0.481}_{0.080}$	$\mathbf{0.328}_{0.108}$	$\mathbf{0.506}_{0.049}$	$82.93_{3.88}$	$\mathbf{0.429}_{0.094}$	$0.156_{0.060}$	$\mathbf{0.442}_{0.069}$
Dutch	FedAvg	–	$81.53_{0.44}$	$0.075_{0.008}$	$0.069_{0.008}$	$0.170_{0.009}$	$82.01_{0.32}$	$0.070_{0.004}$	$0.062_{0.008}$	$0.185_{0.009}$
		10%	$\mathbf{81.54}_{0.47}$	$0.073_{0.009}$	$0.068_{0.009}$	$\mathbf{0.170}_{0.008}$	$\mathbf{82.06}_{0.40}$	$0.071_{0.005}$	$0.063_{0.009}$	$\mathbf{0.185}_{0.011}$
		20%	$81.47_{0.42}$	$0.076_{0.009}$	$0.071_{0.010}$	$0.169_{0.009}$	$81.78_{0.20}$	$0.079_{0.011}$	$\mathbf{0.073}_{0.011}$	$0.174_{0.007}$
		30%	$81.47_{0.46}$	$0.078_{0.011}$	$0.074_{0.010}$	$0.167_{0.008}$	$81.70_{0.26}$	$0.081_{0.015}$	$0.070_{0.023}$	$0.177_{0.019}$
		40%	$81.42_{0.47}$	$\mathbf{0.082}_{0.011}$	$\mathbf{0.079}_{0.010}$	$0.162_{0.011}$	$71.14_{12.87}$	$\mathbf{0.083}_{0.018}$	$0.067_{0.016}$	$0.115_{0.065}$
	FLAME	–	$81.52_{0.41}$	$0.074_{0.009}$	$0.069_{0.008}$	$\mathbf{0.171}_{0.008}$	$82.01_{0.33}$	$0.070_{0.004}$	$0.062_{0.008}$	$0.184_{0.009}$
		10%	$\mathbf{81.54}_{0.46}$	$0.073_{0.009}$	$0.068_{0.009}$	$0.170_{0.008}$	$\mathbf{82.04}_{0.41}$	$0.071_{0.005}$	$0.063_{0.009}$	$\mathbf{0.185}_{0.010}$
		20%	$81.46_{0.44}$	$0.076_{0.009}$	$0.071_{0.010}$	$0.169_{0.009}$	$81.78_{0.19}$	$0.079_{0.010}$	$\mathbf{0.073}_{0.011}$	$0.174_{0.007}$
		30%	$81.45_{0.49}$	$0.078_{0.011}$	$0.074_{0.010}$	$0.166_{0.009}$	$81.48_{0.38}$	$\mathbf{0.086}_{0.015}$	$0.069_{0.024}$	$0.179_{0.022}$
		40%	$81.39_{0.50}$	$\mathbf{0.083}_{0.010}$	$\mathbf{0.080}_{0.010}$	$0.162_{0.011}$	$71.46_{12.45}$	$0.080_{0.013}$	$0.063_{0.019}$	$0.116_{0.064}$
	KRUM	–	$\mathbf{81.49}_{0.29}$	$0.081_{0.009}$	$0.075_{0.008}$	$0.156_{0.023}$	$82.00_{0.66}$	$0.080_{0.008}$	$0.066_{0.010}$	$0.181_{0.016}$
		10%	$81.28_{0.35}$	$0.083_{0.012}$	$0.077_{0.012}$	$0.159_{0.020}$	$81.97_{0.58}$	$0.080_{0.007}$	$0.072_{0.003}$	$0.176_{0.014}$
		20%	$81.16_{0.70}$	$0.084_{0.016}$	$0.077_{0.014}$	$0.162_{0.018}$	$81.79_{0.38}$	$\mathbf{0.083}_{0.010}$	$0.070_{0.004}$	$0.179_{0.018}$
		30%	$80.98_{1.10}$	$0.086_{0.012}$	$0.077_{0.010}$	$0.164_{0.016}$	$\mathbf{82.03}_{0.79}$	$0.070_{0.011}$	$0.058_{0.020}$	$\mathbf{0.186}_{0.020}$
		40%	$80.72_{1.22}$	$\mathbf{0.091}_{0.018}$	$\mathbf{0.086}_{0.020}$	$\mathbf{0.166}_{0.015}$	$81.73_{0.68}$	$0.082_{0.014}$	$\mathbf{0.075}_{0.016}$	$0.177_{0.017}$
	TM	–	$\mathbf{81.63}_{0.58}$	$0.073_{0.013}$	$0.068_{0.014}$	$0.169_{0.010}$	$\mathbf{82.13}_{0.25}$	$0.073_{0.002}$	$0.065_{0.008}$	$\mathbf{0.183}_{0.011}$
		10%	$81.55_{0.42}$	$0.080_{0.018}$	$0.075_{0.016}$	$0.164_{0.011}$	$81.96_{0.30}$	$0.080_{0.013}$	$0.074_{0.008}$	$0.179_{0.009}$
		20%	$81.68_{0.65}$	$0.080_{0.014}$	$0.075_{0.015}$	$0.162_{0.019}$	$81.60_{0.16}$	$0.081_{0.016}$	$\mathbf{0.077}_{0.014}$	$0.165_{0.004}$
		30%	$81.42_{1.04}$	$0.083_{0.015}$	$0.080_{0.017}$	$\mathbf{0.171}_{0.017}$	$81.46_{0.38}$	$0.084_{0.013}$	$0.075_{0.018}$	$0.171_{0.010}$
		40%	$79.27_{3.41}$	$\mathbf{0.122}_{0.069}$	$\mathbf{0.119}_{0.072}$	$0.128_{0.054}$	$81.25_{0.52}$	$\mathbf{0.086}_{0.014}$	$0.070_{0.025}$	$0.172_{0.023}$

Table 3: Comparison of different aggregators on accuracy and Accuracy parity (AP) in a **non-IID** setting on **UTKFace**. Here, $r = 8$ with concentration parameters $\rho \in \{0.25, 0.5, 0.75\}$. Also, **Acc**: Accuracy, **Adv**: Adversary and we report $\texttt{mean}_{\texttt{std}}$ across four independent runs. **Bold** values indicate: highest accuracy and highest fairness violation, highlighting the strongest performance and most severe fairness degradation, respectively.

Method	% **Adv**	$\rho = 0.25$		$\rho = 0.5$		$\rho = 0.75$	
		Acc (↑)	Δ_{AP} (↓)	Acc (↑)	Δ_{AP} (↓)	Acc (↑)	Δ_{AP} (↓)
FedAvg	–	$\mathbf{68.24}_{5.46}$	$0.035_{0.002}$	$\mathbf{71.66}_{0.33}$	$0.023_{0.003}$	$\mathbf{72.35}_{0.24}$	$0.030_{0.006}$
	10%	$61.44_{8.83}$	$0.066_{0.017}$	$69.26_{0.57}$	$0.052_{0.012}$	$69.96_{0.06}$	$0.041_{0.009}$
	20%	$62.43_{3.34}$	$0.091_{0.048}$	$65.78_{0.29}$	$0.085_{0.007}$	$66.04_{0.67}$	$0.093_{0.025}$
	30%	$57.63_{4.27}$	$0.141_{0.046}$	$64.49_{1.19}$	$0.122_{0.026}$	$62.64_{0.20}$	$0.136_{0.013}$
	40%	$56.25_{3.28}$	$\mathbf{0.203}_{0.059}$	$59.35_{0.65}$	$\mathbf{0.161}_{0.024}$	$60.75_{0.61}$	$\mathbf{0.162}_{0.025}$
FLAME	–	$\mathbf{69.65}_{6.82}$	$0.033_{0.005}$	$\mathbf{72.68}_{1.12}$	$0.021_{0.002}$	$\mathbf{72.73}_{0.29}$	$0.032_{0.007}$
	10%	$66.98_{4.44}$	$0.036_{0.012}$	$69.47_{0.13}$	$0.051_{0.018}$	$70.13_{0.48}$	$0.028_{0.001}$
	20%	$63.12_{1.67}$	$0.073_{0.022}$	$68.49_{5.36}$	$0.096_{0.035}$	$67.25_{1.28}$	$0.080_{0.043}$
	30%	$61.30_{1.05}$	$0.085_{0.017}$	$63.91_{0.61}$	$0.117_{0.033}$	$67.39_{1.30}$	$0.055_{0.022}$
	40%	$55.24_{3.12}$	$\mathbf{0.201}_{0.034}$	$65.32_{0.03}$	$\mathbf{0.144}_{0.044}$	$60.98_{1.60}$	$\mathbf{0.104}_{0.003}$
KRUM	–	$62.16_{8.48}$	$0.031_{0.006}$	$\mathbf{74.53}_{0.42}$	$0.022_{0.004}$	$\mathbf{71.31}_{0.99}$	$0.034_{0.005}$
	10%	$\mathbf{71.96}_{1.90}$	$0.034_{0.005}$	$72.21_{0.46}$	$0.038_{0.003}$	$68.38_{3.82}$	$0.038_{0.020}$
	20%	$71.51_{1.71}$	$0.045_{0.010}$	$70.15_{2.86}$	$0.061_{0.005}$	$67.80_{3.20}$	$0.043_{0.010}$
	30%	$64.36_{0.82}$	$0.077_{0.018}$	$67.49_{4.74}$	$0.080_{0.031}$	$63.24_{4.46}$	$0.067_{0.030}$
	40%	$63.98_{1.91}$	$\mathbf{0.086}_{0.000}$	$64.72_{1.15}$	$\mathbf{0.125}_{0.009}$	$62.22_{2.26}$	$\mathbf{0.097}_{0.004}$
TM	–	$\mathbf{70.38}_{1.67}$	$0.031_{0.004}$	$\mathbf{71.84}_{0.07}$	$0.022_{0.009}$	$\mathbf{72.87}_{0.33}$	$0.022_{0.005}$
	10%	$67.86_{2.25}$	$0.068_{0.019}$	$68.69_{0.86}$	$0.043_{0.008}$	$69.71_{0.70}$	$0.044_{0.005}$
	20%	$62.98_{1.60}$	$0.083_{0.014}$	$66.69_{0.51}$	$0.073_{0.001}$	$67.95_{0.27}$	$0.057_{0.009}$
	30%	$60.22_{3.49}$	$0.107_{0.031}$	$65.43_{0.60}$	$0.086_{0.004}$	$65.15_{0.13}$	$0.092_{0.005}$
	40%	$55.90_{2.93}$	$\mathbf{0.135}_{0.028}$	$63.68_{1.20}$	$\mathbf{0.126}_{0.024}$	$63.41_{0.49}$	$\mathbf{0.110}_{0.018}$

Table 4: Comparing LoRA-FL with EAB-FL (Meerza & Liu, 2024) on Adult (Dua & Graff, 2017) with race as the sensitive attribute and 20% adversaries. Results for EAB-FL are taken from (Meerza & Liu, 2024, Table 1).

Attack	**Accuracy** (FedAvg/ Attack) (↑)	Δ_{EO} (FedAvg/ Attack) (↓)	Δ_{DP} (FedAvg/ Attack) (↓)
EAB-FL (Meerza & Liu, 2024)	83.00 / 80.00	0.25 / 0.41	0.27 / 0.44
LoRA-FL (ours)	$85.02_{0.187}$ / $84.52_{0.219}$	$0.09_{0.025}$ / $0.16_{0.065}$	$0.09_{0.013}$ / $0.13_{0.022}$

shows that LoRA-FL remains effective in image-based FL with convolutional architectures, even under varying degrees of client heterogeneity controlled by ρ. As the fraction of adversarial clients increases, LoRA-FL consistently amplifies Accuracy Parity (AP) violations across all values of ρ, while accuracy degrades more gradually. Notably, lower values of ρ, corresponding to more non-IID client distributions, lead to systematically larger AP gaps for both FedAvg and FLAME, indicating that statistical heterogeneity further exacerbates fairness vulnerabilities in CNN-based models.

Comparison with EAB-FL (Meerza & Liu, 2024). From Table 4, on Adult (with race as the sensitive attribute and 20% adversaries), LoRA-FL amplifies fairness gaps at par with EAB-FL: LoRA-FL increases Δ_{EO} and Δ_{DP} by 77.8% and 44.4% over its baseline, while EAB-FL yields relative increases of 64.0% and 63.0%, respectively. Crucially, LoRA-FL incurs only a 0.50 accuracy drop (85.02 → 84.52), compared to EAB-FLs significant drop (83.00 → 80.00). These results demonstrate that LoRA-FL degrades fairness more effectively with negligible impact on utility.

Other Ablations. We conduct an additional ablation study on: (i) the number of agents, (ii) the algorithm of SPECTRE (Hayase et al., 2021) when implemented on LoRA-FL, and (iii) adapter rank. Additionally, we investigate the effect of omitting Phase 1 in LoRA-FL (i.e., when the adversary does not train for accuracy-specific adapters), with results presented in Appendix A. The key takeaways are as follows: *First*, removing Phase 1 disrupts the accuracy-fairness balance that LoRA-FL maintains, causing significant accuracy loss

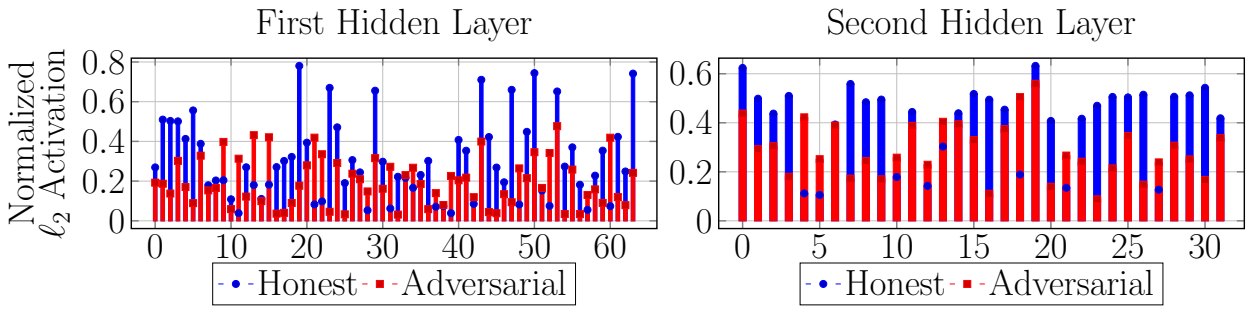

Figure 3: Activation values of neurons in the first and second fully connected layers under honest (blue) and 40% adversarial (red) setup. The plots reveal a higher correlation between activations corresponding to 'Male' and 'Female' inputs in the honest setting, which diminishes under adversarial perturbations, indicating a disruption in representational consistency.

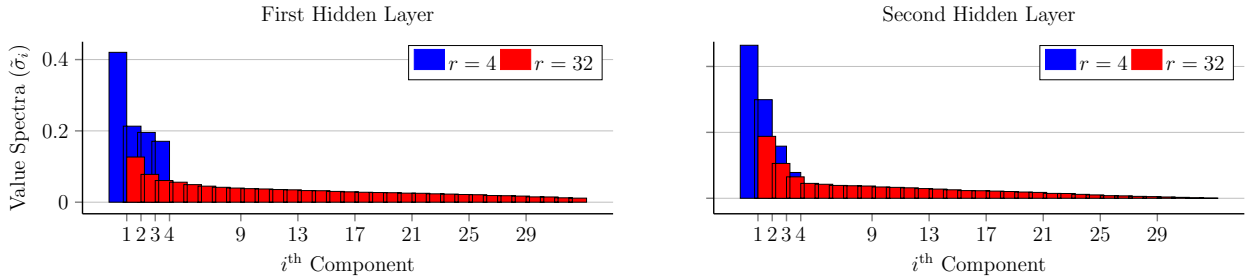

Figure 4: Singular value spectra of $AB^\top$ for two hidden layers on Adult. Although 32-dimensional, the adapters concentrate mass along 3–4 dominant directions, yielding an effective rank of approximately 4.

while still leading to substantial fairness degradation. *Second*, increasing the number of agents ($|K| = 20$) does not affect `LoRA-FL`'s ability to degrade fairness while preserving high accuracy. *Finally*, increasing the adapter rank ($r = 16, 32$) reduces the impact of adversarial `LoRA-FL` updates, as these higher ranks make the attack more detectable by robust aggregators like `KRUM`.

6 Interpreting the Role of Adapters in `LoRA-FL`

Here, we investigate three key questions regarding the role of adapters in `LoRA-FL`:
Q1. How do adapters help adversarial updates evade filtering by parameter-based robust aggregators such as `KRUM`?,
Q2. How do adapters amplify bias in the models predictions?, and
Q3. To what extent do adapters of different ranks span similar subspaces?

Q1: Escaping the `KRUM` Trap. Observe that any (adversarial) parameter update to the local model θ that remains sufficiently close to the global model Θ can evade filtering by robust aggregation methods such as `KRUM`. In Phase 2 of Algorithm 1, we exploit this by injecting unfairness into the low-rank adapters, thereby biasing local predictions while keeping parameter deviations within an acceptable range of Θ.

Frequency of Adversary Selection. To assess the role of low-rank adaptation in the success of `LoRA-FL`, we vary the adapter rank $r \in \{2, 4, 8, 16, 32\}$ and measure the frequency with which `KRUM` selects adversarial updates over $T = 50$ communication rounds (same setup as Adult). As the rank increases, `KRUM` becomes significantly more effective at filtering adversarial updates: the selection frequency drops monotonically from 0.94±0.01 at rank 2 and 0.71±0.02 at rank 4, to 0.52±0.03 at rank 8, 0.23±0.06 at rank 16, and only 0.01±0.02 at rank 32. This highlights that a low-rank structure is critical for evading `KRUM`.

Motivated by this observation, we seek to understand why lower-rank adapters succeed where higher-rank ones fail. We track the ℓ_2 distance $\|\Theta_t - \theta_{i,t}\|_2$ between an adversarial agent i and the global model over $t \in [T]$ for different ranks (Fig. 2). At lower ranks (8 and below), adversarial updates remain close to or below the deviation exhibited by honest agents, whereas at higher ranks (16 and 32) the deviation grows substantially. Consequently, `KRUM` can reliably detect and filter higher-rank adversarial updates, as their parameter shifts exceed the tolerance required to evade robust aggregation.

Q2: Interpreting the Role of Adapters in Amplifying Bias. We evaluate the impact of `LoRA-FL` on demographic representational alignment in an FL setup with (i) only honest and (ii) 40% adversarial agents. Using the Adult dataset test split, we extract neuron activations from the two fully connected layers for Male and Female examples. For each layer l of width d, where d represents the number of neurons, we compute the ℓ_2-mean activation vectors $\mu_l^M, \mu_l^F \in \mathbb{R}^d$ for the respective gender subsets. These vectors are then normalized element-wise by the maximum activation value across both groups: $\tilde{\mu}_l^M = \frac{\mu_l^M}{\texttt{norm}}, \quad \tilde{\mu}_l^F = \frac{\mu_l^F}{\texttt{norm}}$, where $\texttt{norm} := \max\left(\max_i(\mu_l^M)_i, \max_i(\mu_l^F)_i\right)$ represents the maximum activation across both vectors. Finally, we compute the element-wise product $\mathbf{s}_l = \tilde{\mu}_l^M \odot \tilde{\mu}_l^F$, which serves as the per-neuron co-activation score. The score $\mathbf{s}_l$ measures the neuron-wise correlation between the activations for the two demographic groups. Higher values of $\mathbf{s}_l$ indicate that neurons respond similarly to both groups, while lower values suggest more divergent responses.

Figure 3 presents this correlation for the two hidden layers of Θ under (i) honest FL and (ii) 40% adversarial FL, where each neuron is plotted with its corresponding co-activation score, highlighting the degree of alignment between the groups at the neuron level. In the honest condition, $\mathbf{s}_l$ remains uniformly high across neurons, demonstrating that adapters preserve representational consistency between Male and Female inputs. Under adversarial perturbations, however, $\mathbf{s}_l$ drops markedly, revealing that adversaries induce divergent neuron activation patterns and disparate processing of demographic attributes. We see that `LoRA-FL` amplifies bias by degrading neuron-level alignment between gendered representations, disrupting the models ability to maintain consistent internal processing across demographic groups.

Q3: Subspace Similarity for Different r. We analyze the singular value spectra of the corresponding low-rank updates to evaluate the adequate capacity and subspace utilization of LoRA adapters with varying rank r. Specifically, given learned matrices $A \in \mathbb{R}^{d\times r}$ and $B \in \mathbb{R}^{k\times r}$ from LoRA adapters trained on the **Adult** dataset, we compute the singular values of the matrix product $AB^\top \in \mathbb{R}^{d\times k}$ via singular value decomposition (SVD) (Golub & Van Loan, 2013). That is, $AB^\top = U\Sigma V^\top$, where $\Sigma = \text{diag}(\sigma_1, \ldots, \sigma_{\min(d,k)})$ contains the singular values in descending order. We normalize the singular values such that they sum to one, i.e., we analyze $\tilde{\sigma}_i = \frac{\sigma_i}{\sum_j \sigma_j}, \quad \text{for all } i.$

Figure 4 presents the spectra of the normalized singular values $\tilde{\sigma}_i$ for the two hidden layers, for adapters with rank $r = 4$ and $r = 32$. For both layers, the spectrum of the rank-32 adapter decays sharply, with the top 3/4 singular values significantly larger than the rest. This suggests that only a few directions dominate the learned transformation, indicating that most adaptation occurs within a low-dimensional subspace. This observation aligns with the empirical success of low-rank `LoRA-FL` attacks: the rank-4 adapter approximates the key directions of the rank-32 counterpart, enabling comparable degradation in fairness while maintaining low parameter deviation from Θ.

7 Conclusion & Future Work

We present `LoRA-FL`, a low-rank fairness manipulation attack that exposes a fundamental limitation of robustness mechanisms in federated learning. Our evaluation spans both IID and non-IID data distributions, multiple aggregation rules including `FedAvg`, `KRUM`, `TM`, and `FLAME`, and model architectures ranging from MLPs for tabular data to convolutional networks for image classification. Across these configurations, we observe that strong predictive performance and convergence do not imply equitable treatment across sensitive groups. An important direction for future work is extending defenses to reason about structured, low-rank update geometry may help close the gap exploited by `LoRA-FL`.

References

Alekh Agarwal, Alina Beygelzimer, Miroslav Dudik, John Langford, and Hanna Wallach. A reductions approach to fair classification. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 60–69, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL http://proceedings.mlr.press/v80/agarwal18a.html.

Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias: Theres software used across the country to predict future criminals. and its biased against blacks. *ProPublica*, 2016. URL https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing. Accessed: 2025-04-29.

Solon Barocas and Andrew D Selbst. Big data's disparate impact. *Cal. L. Rev.*, 104:671, 2016.

Solon Barocas, Moritz Hardt, and Arvind Narayanan. *Fairness and Machine Learning: Limitations and Opportunities*. MIT Press, 2023.

Arjun Nitin Bhagoji, Supriyo Chakraborty, Prateek Mittal, and Seraphin Calo. Analyzing federated learning through an adversarial lens. In *International conference on machine learning*, pp. 634–643. PMLR, 2019.

Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning attacks against support vector machines. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pp. 1467–1474, 2012.

M. Bilal Zafar, I. Valera, M. Gomez Rodriguez, and K. P. Gummadi. Fairness Constraints: Mechanisms for Fair Classification. *ArXiv e-prints*, July 2015.

Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. Machine learning with adversaries: Byzantine tolerant gradient descent. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, 2017a. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/f4b9ec30ad9f68f89b29639786cb62ef-Paper.pdf.

Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. Machine learning with adversaries: Byzantine tolerant gradient descent. *Advances in neural information processing systems*, 30, 2017b.

Xiaoyu Cao, Minghong Fang, Jia Liu, and Neil Zhenqiang Gong. Fltrust: Byzantine-robust federated learning via trust bootstrapping. *arXiv:2012.13995*, 2020.

Hongyan Chang and Reza Shokri. Bias propagation in federated learning. In *The Eleventh International Conference on Learning Representations*.

Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5 2:153–163, 2017.

Yanbo Dai and Songze Li. Chameleon: Adapting to peer images for planting durable backdoors in federated learning. In *International Conference on Machine Learning*, pp. 6712–6725. PMLR, 2023.

Chuntao Ding, Xu Cao, Jianhang Xie, Linlin Fan, Shangguang Wang, and Zhichao Lu. Lora-c: Parameter-efficient fine-tuning of robust cnn for iot devices. *arXiv preprint arXiv:2410.16954*, 2024.

Dheeru Dua and Casey Graff. Uci machine learning repository, 2017. URL https://archive.ics.uci.edu/ml/datasets/adult.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pp. 214–226, 2012.

Yahya H Ezzeldin, Shen Yan, Chaoyang He, Emilio Ferrara, and A Salman Avestimehr. Fairfed: Enabling group fairness in federated learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 7494–7502, 2023.

Alessandro Fabris, Nina Baranowska, Matthew J Dennis, David Graus, Philipp Hacker, Jorge Saldivar, Frederik Zuiderveen Borgesius, and Asia J Biega. Fairness and bias in algorithmic hiring: A multidisciplinary survey. *ACM Transactions on Intelligent Systems and Technology*, 16(1):1–54, 2025.

Minghong Fang, Xiaoyu Cao, Jinyuan Jia, and Neil Gong. Local model poisoning attacks to {Byzantine-Robust} federated learning. In *29th USENIX security symposium (USENIX Security 20)*, pp. 1605–1622, 2020.

Sorelle A Friedler, Carlos Scheidegger, Suresh Venkatasubramanian, Sonam Choudhary, Evan P Hamilton, and Derek Roth. A comparative study of fairness-enhancing interventions in machine learning. In *Proceedings of the conference on fairness, accountability, and transparency*, pp. 329–338, 2019.

Jiashi Gao, Ziwei Wang, Xiangyu Zhao, Xin Yao, and Xuetao Wei. Pfattack: Stealthy attack bypassing group fairness in federated learning. *arXiv preprint arXiv:2410.06509*, 2024.

Gene H Golub and Charles F Van Loan. *Matrix computations*. JHU press, 2013.

Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604*, 2018.

Moritz Hardt, Eric Price, and Nathan Srebro. Equality of opportunity in supervised learning. In *NIPS*, 2016a.

Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29, 2016b.

Jonathan Hayase, Wei Wei, Kyle Galyardt, and Sewoong Oh. Spectre: Defending against backdoor attacks using robust statistics. In *International Conference on Machine Learning*, pp. 4129–4139. PMLR, 2021.

Jialuo He, Wei Chen, and Xiaojin Zhang. Fedaa: A reinforcement learning perspective on adaptive aggregation for fair and robust federated learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(16):17085–17093, Apr. 2025. doi: 10.1609/aaai.v39i16.33878. URL `https://ojs.aaai.org/index.php/AAAI/article/view/33878`.

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 (CVPR)*, 2016.

Bo Li, Yining Wang, Aarti Singh, and Yevgeniy Vorobeychik. Data poisoning attacks on factorization-based collaborative filtering. *Advances in neural information processing systems*, 29, 2016.

Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6357–6368. PMLR, 18–24 Jul 2021. URL `https://proceedings.mlr.press/v139/li21h.html`.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *International Conference on Learning Representations (ICLR)*, 2019. arXiv:1711.05101.

David Madras, Elliot Creager, Toniann Pitassi, and Richard S. Zemel. Learning adversarially fair and transferable representations. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, pp. 3381–3390, 2018. URL `http://proceedings.mlr.press/v80/madras18a.html`.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pp. 1273–1282. PMLR, 2017.

Syed Irfan Ali Meerza and Jian Liu. Eab-fl: exacerbating algorithmic bias through model poisoning attacks in federated learning. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pp. 458–466, 2024.

Ninareh Mehrabi, Muhammad Naveed, Fred Morstatter, and Aram Galstyan. Exacerbating algorithmic bias through fairness attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 8930–8938, 2021.

Sérgio Moro, P. Cortez, and Paulo Rita. A data-driven approach to predict the success of bank telemarketing. *Decis. Support Syst.*, 62:22–31, 2014. URL `https://api.semanticscholar.org/CorpusID:14181100`.

Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, pp. 807–814, 2010.

Kai Wang Ng, Guo-Liang Tian, and Man-Lai Tang. *Dirichlet and Related Distributions: Theory, Methods and Applications*. Wiley Series in Probability and Statistics. John Wiley & Sons, Chichester, UK, 2011. ISBN 978-0-470-68819-9.

Thien Duc Nguyen, Phillip Rieger, Huili Chen, Hossein Yalame, Helen Möllering, Hossein Fereidooni, Samuel Marchal, Markus Miettinen, Azalia Mirhoseini, Shaza Zeitouni, et al. {FLAME}: Taming backdoors in federated learning. In *31st USENIX Security Symposium (USENIX Security 22)*, pp. 1415–1432, 2022.

Manisha Padala and Sujit Gujar. Fnnc: Achieving fairness through neural networks. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence. IJCAI. https://www. ijcai. org/proceedings/2020/0315. pdf Go to original source*, 2020.

European Parliament and Council of the European Union. General data protection regulation. `https://eur-lex.europa.eu/eli/reg/2016/679/oj`, 2016. Regulation (EU) 2016/679.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: an imperative style, high-performance deep learning library. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 2019.

Dana Pessach and Erez Shmueli. A review on fairness in machine learning. *ACM Computing Surveys (CSUR)*, 55(3):1–44, 2022.

Ricardo Trainotti Rabonato and Lilian Berton. A systematic review of fairness in machine learning. *AI and Ethics*, 5(3):1943–1954, 2025.

Benjamin IP Rubinstein, Blaine Nelson, Ling Huang, Anthony D Joseph, Shing-hon Lau, Satish Rao, Nina Taft, and J Doug Tygar. Antidote: understanding and defending against poisoning of anomaly detectors. In *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement*, pp. 1–14, 2009.

Teresa Salazar, Helder AraÃējo, Alberto Cano, and Pedro Henriques Abreu. A survey on group fairness in federated learning: challenges, taxonomy of solutions and directions for future research. *arXiv preprint arXiv:2410.03855*, 2024.

Virat Shejwalkar and Amir Houmansadr. Manipulating the byzantine: Optimizing model poisoning attacks and defenses for federated learning. In *NDSS*, 2021.

Micah J Sheller, Brandon Edwards, G Anthony Reina, Jason Martin, Sarthak Pati, Aikaterini Kotrotsou, Mikhail Milchenko, Weilin Xu, Daniel Marcus, Rivka R Colen, et al. Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. *Scientific reports*, 10(1):12598, 2020.

Jinhyun So, Başak Güler, and A Salman Avestimehr. Byzantine-resilient secure federated learning. *IEEE Journal on Selected Areas in Communications*, 2020.

David Solans, Battista Biggio, and Carlos Castillo. Poisoning attacks on algorithmic fairness. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 162–177. Springer, 2020.

Minh-Hao Van, Wei Du, Xintao Wu, and Aidong Lu. Poisoning attacks on fair machine learning. In *International Conference on Database Systems for Advanced Applications*, pp. 370–386. Springer, 2022.

Ganghua Wang, Ali Payani, Myungjin Lee, and Ramana Rao Kompella. Mitigating group bias in federated learning: Beyond local fairness. *Transactions on Machine Learning Research*.

Geming Xia, Jian Chen, Chaodong Yu, and Jun Ma. Poisoning attacks in federated learning: A survey. *Ieee Access*, 11:10708–10722, 2023.

Jie Xu, Benjamin S Glicksberg, Chang Su, Peter Walker, Jiang Bian, and Fei Wang. Federated learning for healthcare informatics. *Journal of healthcare informatics research*, 5:1–19, 2021.

Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019.

Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International conference on machine learning*, pp. 5650–5659. Pmlr, 2018.

Xubo Yue, Maher Nouiehed, and Raed Al Kontar. Gifair-fl: A framework for group and individual fairness in federated learning. *INFORMS Journal on Data Science*, 2(1):10–23, 2023.

Zhifei Zhang, Yang Song, and Hai Qi. Utkface: Age progression/regression by conditional adversarial autoencoder, 2017. URL `https://susanqq.github.io/UTKFace/`. Large Scale Face Dataset.

Indre liobaite, Faisal Kamiran, and Toon Calders. Handling conditional discrimination. In *2011 IEEE 11th International Conference on Data Mining*, pp. 992–1001, 2011.

Table A.1: Hyperparameters

Parameter	Symbol	Value	Description
Number of agents	K	10 or 20	Total number of agents in the system
Number of rounds	T	50	Total number of communication rounds
Local epochs	E	10 (Adult, Bank and Dutch), 4 (UTKFace)	Number of local training epochs for honest agents
Adversarial epochs	$E_{\mathcal{A}}$	10 (Adult and Dutch), 20 (Bank)	Number of local training epochs for adversarial agents, 3 (UTKFace)
Agents per round	m	60% of K	Number of sampled agents per round
agent optimizer	`OPT`	AdamW	Optimizer used by honest agents
Adversarial optimizer	$\texttt{OPT}_{\texttt{REG}}$	AdamW	Optimizer used by adversarial agents for Phase 1
Adversarial optimizer	$\texttt{OPT}_{\texttt{F}}$	AdamW	Optimizer used by adversarial agents for Phase 2
Scaling factor	α	1.0	Scaling applied to the low-rank update
Aggregator function	`Agg`	`FedAvg` or `KRUM`	Aggregation rule
Batch Size	B	512 (Adult, Bank and Dutch), 1024 (UTKFace)	
Honest agents Learning Rate	η	5e-4 (Adult, Bank and Dutch), 1e-4 (UTKFace)	AdamW LR for Honest agents
Adversarial agents Learning Rate	$\eta_{\texttt{REG}}$	5e-4 (Adult, Bank and Dutch), 3e-4 (UTKFace)	AdamW LR for Adversarial agents
Adversarial agents Learning Rate	$\eta_{\texttt{F}}$	5e-4	AdamW LR for Adversarial agents
Rank	r	4 (Adult and Dutch), 2 (Bank and UTKFace)	Rank of the Adversarial Adapters
Scaling Factor	α	1	Controls the scale by which the adapters are fused in MLP
Scaling Factor	$\alpha^{'}$	4	Controls the scale by which the adapters are fused in CNN

A Training Details & Additional Experiments

A.1 Hyperparamter Details

Table A.1 summarizes the key hyperparameters used for `LoRA-FL`. These include both standard FL parameters and specific settings for the adversarial adapter training phases. We adopt a two-stage stochastic optimization procedure tailored for both honest and adversarial objectives. All models are trained using mini-batch stochastic gradient descent with the AdamW optimizer, using a batch size of 512 for Adult, Bank, and Dutch datasets and 1024 for UTKFace and a fixed learning rate of 5e-4 (Adult, Bank, and Dutch) and 3e-4 (UTKFace). For standard (honest) training, we minimize the binary cross-entropy loss over ten local epochs in Adult, Bank, and Dutch, and four local epochs in the case of the UTKFace dataset. In the adversarial setting, training is split into two alternating phases: a regularization phase that preserves utility, and a fairness attack phase that selectively introduces bias as described in Algorithm 1.

A.2 `LoRA-FL`: Training without Phase 1

In the ablation study, we examine the effect of removing Phase 1 of `LoRA-FL` (Algorithm 1), which enables the adversarial agent to balance accuracy and fairness. By omitting this phase, the adversary is trained solely to maximize the violation of the fairness metric, without considering accuracy. The results from this setup are reported on the Adult dataset. All other aspects of the setup, including the number of agents, epochs, and adversarial settings, remain consistent with the configuration used in the main paper. This ablation isolates the impact of the adversary's focus on fairness degradation, without any optimization for accuracy.

Table A.2 presents the results for the ablation. As shown in Table A.2, removing Phase 1 significantly disrupts the accuracyfairness trade-off that `LoRA-FL` is designed to maintain. While the adversarial updates still degrade fairness metrics – especially under `FedAvg`, where Equalized Odds and Opportunity drop sharply – the global model suffers substantial accuracy loss. For instance, with 30% adversaries, accuracy drops to 77.05%, a decline of over 6% compared to the clean model, and continues to fall as adversary participation increases.

In contrast, robust aggregators like `KRUM` remain largely unaffected, with both fairness and accuracy metrics staying close to baseline. Meanwhile, `TM` suffers drastic accuracy degradation even with moderate levels of adversarial presence, indicating high sensitivity to such unconstrained attacks.

These results demonstrate that omitting Phase 1 removes the stealth component of `LoRA-FL`: although the attack remains effective at harming fairness, the resulting perturbations become too aggressive, making them

Table A.2: **Ablation Study: LoRA-FL without Phase 1.** The adversarial agents omit training for accuracy. We compare the performance on different aggregators on accuracy and fairness metrics: Demographic Parity (DP), Equalized Odds (EO), and Equal Opportunity (EOpp). Here, $|K| = 10$ and $r = 4$.

Aggregator	% Adversary	Accuracy (↑)	Δ_{EO} (↓)	Δ_{EOpp} (↓)	Δ_{DP} (↓)
Adult Dua & Graff (2017)					
FedAvg	–	$83.33_{0.38}$	$0.122_{0.008}$	$0.095_{0.010}$	$0.186_{0.007}$
KRUM	–	$82.56_{0.20}$	$0.115_{0.016}$	$0.082_{0.018}$	$0.190_{0.008}$
TM	–	$83.10_{0.92}$	$0.115_{0.023}$	$0.088_{0.017}$	$0.183_{0.034}$
FedAvg	10%	$81.73_{0.53}$	$0.448_{0.034}$	$0.446_{0.036}$	$0.330_{0.023}$
FedAvg	20%	$78.41_{1.51}$	$0.594_{0.097}$	$0.594_{0.097}$	$0.449_{0.028}$
FedAvg	30%	$77.05_{1.58}$	$0.693_{0.059}$	$0.693_{0.059}$	$0.475_{0.038}$
FedAvg	40%	$76.26_{1.18}$	$0.693_{0.033}$	$0.693_{0.033}$	$0.476_{0.030}$
KRUM	10%	$82.35_{0.32}$	$0.117_{0.002}$	$0.076_{0.012}$	$0.193_{0.004}$
KRUM	20%	$82.38_{0.46}$	$0.123_{0.023}$	$0.096_{0.026}$	$0.188_{0.017}$
KRUM	30%	$82.38_{0.31}$	$0.121_{0.012}$	$0.088_{0.011}$	$0.188_{0.010}$
KRUM	40%	$82.49_{0.26}$	$0.115_{0.008}$	$0.082_{0.017}$	$0.182_{0.012}$
TM	10%	$35.12_{11.17}$	$0.094_{0.091}$	$0.032_{0.025}$	$0.103_{0.109}$
TM	20%	$35.86_{18.87}$	$0.121_{0.203}$	$0.097_{0.167}$	$0.138_{0.233}$
TM	30%	$34.20_{14.34}$	$0.142_{0.235}$	$0.089_{0.152}$	$0.149_{0.245}$
TM	40%	$52.21_{15.56}$	$0.475_{0.251}$	$0.431_{0.296}$	$0.365_{0.139}$

more detectable by robust defenses and compromising the models utility. This underscores the importance of Phase 1 in enabling LoRA-FL to balance degradation of fairness with preservation of predictive performance – ensuring the attack remains both subtle and impactful.

A.3 Agents

Table A.3 presents results for a larger agent pool ($|K| = 20$), showing that our low-rank adapter attack remains effective at degrading fairness, while maintaining high accuracy. For the Adult dataset, we observe that as the fraction of adversarial agents increases, fairness metrics such as Δ_{EO} and Δ_{DP} degrade substantially. For instance, Δ_{EO} rises from 0.177 (clean) to 0.621 with 40% adversaries under FedAvg, a 3.5× increase, while accuracy drops by 3.7%. Importantly, the trends mirror those in the main paper for $|K| = 10$, confirming that LoRA-FL (with $r = 4$) induces fairness degradation in a manner largely agnostic to the number of agents in the system.

A.4 Adapter Rank

Table A.4 shows that increasing the adapter rank ($r = 16, 32$) significantly mitigates the impact of adversarial LoRA-FL updates for both FedAvg and KRUM. In contrast to the strong degradation observed at lower ranks (i.e., for $r = 4$ in the main paper), higher-rank adapters result in only marginal drops in accuracy and fairness, even under 40% adversarial agents. Notably, with $r = 32$, KRUM retains near-baseline fairness levels across all metrics, highlighting that low-rank constraints are a key enabler of the attacks potency by making adversarial updates more easily obscured or entangled in the parameter space.

Moreover, we observe that for KRUM, the **standard deviations of the fairness metrics across adversary proportions** (0%40%) are substantially lower at $r = 32$ than at $r = 16$, indicating more stable behavior due to KRUMs ability to effectively filter out high-rank adversarial updates. Specifically, the standard deviation drops from 0.036 to 0.005 for Δ_{EO}, 0.041 to 0.009 for Δ_{EOpp}, and 0.031 to 0.004 for Δ_{DP}. These findings

Table A.3: **Ablation Study: Number of Clients.** Comparison of different aggregators on accuracy and fairness metrics: Demographic Parity (DP), Equalized Odds (EO), and Equal Opportunity (EOpp). Here, $r = 4$ and $|K| = 10$, with results averaged over four independent runs. We observe that our attack remains effective regardless of the number of participating clients, indicating its robustness to varying client pool sizes.

Aggregator	% Adversary	Accuracy (↑)	Δ_{EO} (↓)	Δ_{EOpp} (↓)	Δ_{DP} (↓)
		Adult Dua & Graff (2017)			
FedAvg	–	85.848 ± 0.304	0.177 ± 0.0107	0.0842 ± 0.0408	0.200 ± 0.002
KRUM	–	85.698 ± 0.208	0.159 ± 0.009	0.0875 ± 0.002	0.199 ± 0.003
FedAvg	10%	85.649 ± 0.168	0.236 ± 0.0104	0.089 ± 0.008	0.232 ± 0.007
FedAvg	20%	85.026 ± 0.0462	0.317 ± 0.0376	0.100 ± 0.0027	0.278 ± 0.015
FedAvg	30%	83.926 ± 0.0724	0.488 ± 0.0532	0.117 ± 0.009	0.393 ± 0.101
FedAvg	40%	82.673 ± 0.266	0.621 ± 0.109	0.129 ± 0.005	0.377 ± 0.018
KRUM	10%	85.358 ± 0.185	0.176 ± 0.005	0.117 ± 0.0106	0.224 ± 0.005
KRUM	20%	84.700 ± 0.272	0.269 ± 0.0132	0.127 ± 0.0113	0.287 ± 0.030
KRUM	30%	83.982 ± 0.538	0.434 ± 0.118	0.174 ± 0.0192	0.304 ± 0.056
KRUM	40%	83.278 ± 0.758	0.529 ± 0.146	0.234 ± 0.0076	0.316 ± 0.0411

suggest that higher adapter rank amplifies parameter deviation, enabling distance-based defenses like KRUM to more reliably identify and discard malicious updates.

A.5 Implementation of algorithm on SPECTRE aggregator

SPECTRE Hayase et al. (2021) is a robust aggregator that utilizes robust mean estimation via spectral signatures.The formal procedure is detailed in Algorithm A.1. Although originally designed to counter data poisoning, we apply SPECTRE to LoRA-FL to determine if it can detect and isolate the spectral spikes associated with malicious client updates. As shown in Table A.5, LoRA-FL effectively circumvents this defense. As rank decreases, LoRA-FL becomes more stealthier: the accuracy increases from 78.70 ± 5.68 at rank 64 and 81.64 ± 1.07 at rank 8, to 82.02 ± 0.52 at rank 4 and 84.88 ± 0.73 at rank 2.

Algorithm A.1 SPECTRE Aggregator

Input: Local updates $\{\theta_k\}_{k=1}^K$, estimated number of adversaries q
$\mu \leftarrow \text{median}(\{\theta_k\}_{k=1}^K)$ ▷ Centering
$X \leftarrow [(\theta_k - \mu)]_{k=1}^K$
$U, \Sigma, V^\top \leftarrow \text{SVD}(X)$ ▷ Dimensionality reduction to top-k components
$X_{proj} \leftarrow XV_{1:k}$
$\hat{\Sigma} \leftarrow \frac{1}{K} X_{proj}^\top X_{proj}$ ▷ Projected covariance matrix
$\mathbf{W} \leftarrow \exp(\alpha\hat{\Sigma})/\text{Tr}(\exp(\alpha\hat{\Sigma}))$ ▷ Compute Quantum Entropy weights
$\tau_k \leftarrow (\theta_k - \mu)^\top \mathbf{W} (\theta_k - \mu)$ for all k ▷ Spectral outlier scores
$\mathcal{S} \leftarrow \{\text{Indices of } K - q \text{ updates with smallest } \tau_k\}$
Output: $\Theta_{SPEC} = \frac{1}{|\mathcal{S}|} \sum_{k \in \mathcal{S}} \theta_k$

B Robust Aggregators

Here, we provide the formal algorithms for the robust aggregators.

Table A.4: **Ablation Study: Adapter Rank.** Comparison of different aggregators on accuracy and fairness metrics: Demographic Parity (DP), Equalized Odds (EO), and Equal Opportunity (EOpp). Here, the dataset used is **Adult** and $|K| = 10$. Higher adapter rank increases parameter deviation, enabling `KRUM` to more effectively filter adversarial updates and stabilize fairness metrics.

Aggregator	**% Adversary**	**Accuracy** (↑)	Δ_{EO} (↓)	Δ_{EOpp} (↓)	Δ_{DP} (↓)
		$r = 16$			
`FedAvg`	–	84.910	0.118	0.096	0.175
`KRUM`	–	84.900	0.131	0.109	0.189
`FedAvg`	10%	84.830	0.141	0.113	0.207
`FedAvg`	20%	84.100	0.203	0.181	0.257
`FedAvg`	30%	82.850	0.305	0.297	0.312
`FedAvg`	40%	78.010	0.401	0.401	0.402
`KRUM`	10%	84.730	0.170	0.161	0.214
`KRUM`	20%	84.540	0.183	0.162	0.228
`KRUM`	30%	84.510	0.214	0.195	0.255
`KRUM`	40%	83.100	0.236	0.233	0.278
		$r = 32$			
`FedAvg`	–	84.910	0.118	0.096	0.175
`KRUM`	–	84.900	0.131	0.109	0.189
`FedAvg`	10%	85.010	0.136	0.102	0.202
`FedAvg`	20%	83.810	0.211	0.184	0.261
`FedAvg`	30%	82.340	0.297	0.287	0.319
`FedAvg`	40%	78.640	0.160	0.160	0.067
`KRUM`	10%	84.840	0.132	0.090	0.188
`KRUM`	20%	84.910	0.141	0.114	0.197
`KRUM`	30%	84.730	0.128	0.100	0.193
`KRUM`	40%	84.660	0.128	0.091	0.196

Algorithm B.1 m-`KRUM` Aggregation Blanchard et al. (2017b)

Require: Agent updates $\{\theta_1, \theta_2, \ldots, \theta_K\}$, number of adversarial agents $\tilde{q} = \lfloor qK \rfloor$, agent sample sizes $\{|\mathcal{X}_1|, \ldots, |\mathcal{X}_K|\}$, number of agents to aggregate $m_{\min}$

Ensure: Aggregated Global Model Θ^{KRUM}

Let $m := \max(K - \tilde{q}, m_{\min})$ ▷ Ensure m is at least $K - \tilde{q}$

for $i = 1$ to K **do**

  Define $d_{i,j} = \|\theta_i - \theta_j\|_2$ as the Euclidean distance between pair-wise agent updates θ_i and θ_j

  Compute distances $d_{i,j}$ for all $j \neq i$

  Let $\mathcal{N}_i \leftarrow$ indices of $K - \tilde{q} - 2$ closest updates to θ_i

  Compute score $s_i = \sum_{j \in \mathcal{N}_i} d_{i,j}^2$

end for

Select the set $\mathcal{M} \subseteq \{1, \ldots, K\}$ of m agents with the lowest scores s_i

Compute weighted average:

$$\Theta^{\texttt{KRUM}} = \sum_{i \in \mathcal{M}} \frac{|\mathcal{X}_i|}{\sum_{j \in \mathcal{M}} |\mathcal{X}_j|} \cdot \theta_i \tag{11}$$

return $\Theta^{\texttt{KRUM}}$

Table A.5: SPECTRE accuracy and fairness metrics on the **Bank** dataset for both **IID** and **non-IID** settings. Here, $r = 2$ with concentration parameters $\rho \in \{0.25, 0.5, 0.75\}$ for non-IID settings. Also, **Acc**: Accuracy, **Adv**: Adversary, and we report $\texttt{mean}_{\texttt{std}}$ across **four** independent runs. **Bold** values indicate highest accuracy and highest fairness violation.

Setting	% **Adv**	Acc (↑)	Δ_{EO} (↓)	Δ_{EOpp} (↓)	Δ_{DP} (↓)
IID	–	$\mathbf{91.01}_{0.25}$	$0.22_{0.02}$	$0.15_{0.03}$	$0.23_{0.02}$
	10%	$90.92_{0.11}$	$0.21_{0.01}$	$0.14_{0.02}$	$0.25_{0.02}$
	20%	$90.48_{0.22}$	$0.25_{0.03}$	$0.15_{0.02}$	$0.30_{0.03}$
	30%	$90.33_{0.17}$	$0.26_{0.02}$	$0.16_{0.03}$	$0.30_{0.02}$
	40%	$88.64_{0.40}$	$\mathbf{0.33}_{0.02}$	$\mathbf{0.17}_{0.02}$	$\mathbf{0.38}_{0.01}$
$\rho = 0.5$	–	$\mathbf{91.67}_{0.50}$	$0.29_{0.04}$	$\mathbf{0.28}_{0.03}$	$0.23_{0.01}$
	10%	$91.41_{0.45}$	$0.29_{0.03}$	$0.25_{0.03}$	$0.25_{0.02}$
	20%	$91.33_{0.43}$	$0.32_{0.04}$	$0.26_{0.03}$	$0.28_{0.05}$
	30%	$90.95_{0.66}$	$0.32_{0.05}$	$0.25_{0.01}$	$0.30_{0.06}$
	40%	$90.18_{1.18}$	$\mathbf{0.36}_{0.06}$	$0.27_{0.03}$	$\mathbf{0.33}_{0.06}$
$\rho = 0.25$	–	$\mathbf{89.90}_{1.30}$	$0.28_{0.04}$	$0.26_{0.05}$	$0.23_{0.02}$
	10%	$88.46_{2.87}$	$0.36_{0.12}$	$0.28_{0.08}$	$0.33_{0.07}$
	20%	$86.81_{3.88}$	$0.38_{0.13}$	$0.30_{0.10}$	$0.33_{0.08}$
	30%	$53.84_{36.34}$	$0.46_{0.16}$	$\mathbf{0.30}_{0.11}$	$0.40_{0.11}$
	40%	$54.57_{34.16}$	$\mathbf{0.49}_{0.12}$	$0.26_{0.00}$	$\mathbf{0.46}_{0.11}$
$\rho = 0.75$	–	$\mathbf{90.74}_{0.45}$	$0.28_{0.05}$	$0.21_{0.02}$	$0.28_{0.07}$
	10%	$90.58_{0.30}$	$0.31_{0.05}$	$0.22_{0.04}$	$0.28_{0.05}$
	20%	$90.31_{0.31}$	$0.35_{0.04}$	$0.24_{0.05}$	$0.32_{0.07}$
	30%	$89.82_{0.51}$	$0.37_{0.04}$	$\mathbf{0.26}_{0.04}$	$0.35_{0.06}$
	40%	$88.73_{0.45}$	$\mathbf{0.42}_{0.05}$	$0.25_{0.02}$	$\mathbf{0.40}_{0.06}$

Algorithm B.2 f-Trimmed-Mean Aggregation

Require: Agent updates $\{\theta_1, \theta_2, \ldots, \theta_K\}$, number of values to trim f
Ensure: Aggregated Global Model $\Theta^{\texttt{TM}}$

Assert: $K > 2f$ ▷ At least $2f + 1$ agents required
Initialize empty model $\Theta^{\texttt{TM}}$
for each parameter key w in model **do**
  if w is a BatchNorm parameter **then**
    Set $\Theta^{\texttt{TM}}[w] \leftarrow \theta_1[w]$ ▷ Skip aggregation for BN layers
    continue
  end if
  Stack agent parameters: $V_w \leftarrow [\theta_1[w], \ldots, \theta_K[w]]$ as matrix of shape $(K, \text{param\_size})$
  Sort V_w along agent dimension for each coordinate
  Trim f smallest and f largest values at each coordinate
  Compute coordinate-wise mean of trimmed values: $\bar{v}_w$
  Reshape $\bar{v}_w$ to original shape and set $\Theta^{\texttt{TM}}[w] \leftarrow \bar{v}_w$
end for
return $\Theta^{\texttt{TM}}$

B.1 m-KRUM

We employ the m-KRUM aggregation algorithm Blanchard et al. (2017b) to achieve robustness in the presence of adversarial agents. Given a set of local model updates $\theta_1, \ldots, \theta_K$ from K agents, and an upper bound $\tilde{q} = \lfloor qK \rfloor$ on the number of potentially malicious agents, KRUM computes a robustness score s_i for each agent update θ_i by summing the squared Euclidean distances to its $K - \tilde{q} - 2$ closest peers. The m agents with the lowest scores are selected for aggregation, where $m \geq K - \tilde{q}$. A weighted average of these selected updates, using their respective sample sizes $|\mathcal{X}_i|$, yields the final global model $\Theta^{\texttt{KRUM}}$. By filtering out updates that are distant from the consensus, KRUM effectively limits the influence of Byzantine agents on the global model.

B.2 Trimmed-Mean

We also employ the f-`Trimmed-Mean` aggregation algorithm Yin et al. (2018). Given a set of agent model updates $\theta_1, \theta_2, \ldots, \theta_K$, the algorithm assumes that up to f of these may be adversarial and removes the f largest and f smallest values for each model parameter dimension independently. Specifically, for each parameter w, we collect all corresponding agent values, sort them element-wise, discard the extreme $2f$ values, and take the mean of the remaining $K - 2f$ entries to compute the aggregated parameter $\bar{w}$. This process is repeated for all parameters in the model. Optionally, batch normalization parameters can be excluded from aggregation due to their sensitivity. The resulting model $\Theta^{\texttt{TM}}$ offers a robust estimate that mitigates the influence of malicious or corrupted updates.

B.3 FLAME

`FLAME` (Federated Learning with Adversarial Model Evaluation) aggregation algorithm Nguyen et al. (2022), which is designed to provide robustness under both client heterogeneity and adversarial behavior. Given a set of client model updates $\theta_1, \theta_2, \ldots, \theta_K$ in a training round, `FLAME` first computes a pairwise similarity matrix using cosine similarity to capture directional agreement between updates while being invariant to scale. Based on this similarity matrix, `FLAME` applies a clustering procedure to partition updates into groups corresponding to benign and potentially malicious behavior. The cluster containing the largest number of updates is identified as the benign cluster, under the assumption that the majority of participating clients behave honestly. Updates that fall outside the benign cluster are either discarded or assigned reduced weights, depending on their similarity to the cluster centroid. The aggregator then computes the global update by averaging the selected benign updates. To further mitigate the influence of residual adversarial behavior and noisy updates–particularly under non-IID data distributions–`FLAME` injects adaptive Gaussian noise, calibrated to the empirical variance of the benign cluster. This noise injection step is intended to smooth the aggregated update and limit the impact of any remaining malicious contributions. The resulting global model, denoted $\Theta^{\texttt{FLAME}}$, provides robustness against Byzantine clients while preserving convergence and stability in heterogeneous federated learning settings.