Byzantine-Robust Federated Learning with Optimal Statistical Rates

Banghua Zhu[*]
UC Berkeley

Lun Wang[*†]
Google

Qi Pang[*‡]
Carnegie Mellon University

Shuai Wang
HKUST

Jiantao Jiao
UC Berkeley

Dawn Song
UC Berkeley

Michael I. Jordan
UC Berkeley

Abstract

We propose Byzantine-robust federated learning protocols with nearly optimal statistical rates based on recent progress in high dimensional robust statistics. In contrast to prior work, our proposed protocols improve the dimension dependence and achieve a near-optimal statistical rate for strongly convex losses. We also provide statistical lower bound for the problem. For experiments, we benchmark against competing protocols and show the empirical superiority of the proposed protocols.

1 Introduction

Federated learning (FL) has drawn considerable attention as a novel distributed learning paradigm in recent years. In FL, users (worker machines) collaborate to train a model using a centralized server (master machine), while all data is stored locally to protect users' privacy. The privacy benefit has motivated the adoption of FL in a variety of sensitive applications, including Google GBoard, healthcare services, and self-driving cars.

However, vanilla FL has been demonstrated to be vulnerable to a range of attacks (Bagdasaryan et al., 2020; Bhagoji et al., 2019; Nasr et al., 2019; Sun et al., 2021; Luo et al., 2021). There are two mainstream vulnerabilities in FL, namely *Byzantine robustness* and *privacy*.

---

[*]These authors contributed equally to this work.

[†]The work of Lun Wang is done at UC Berkeley.

[‡]The work of Qi Pang is done at HKUST.

In the former, a small number of clients can behave maliciously in a large-scale FL system and influence the jointly-trained FL model in a stealthy manner (Bagdasaryan et al., 2020; Bhagoji et al., 2019; Fang et al., 2020; Sun et al., 2021). For the majority of SGD-based FL algorithms (McMahan and Ramage, 2017), the centralized server averages the local updates to obtain the global update, which is vulnerable to even a single malicious client. Particularly, a malicious client can craft its update in such a way that it prevents the global model from converging or leads it to a sub-optimal minimum. With respect to terms of privacy, the vulnerability is that the centralized server can infer information about the local data of the clients by inspecting their updates (Nasr et al., 2019; Luo et al., 2021).

In this work, we mainly focus on the Byzantine robustness of the FL protocols. Byzantine-robust FL protocols (Blanchard et al., 2017; Yin et al., 2018; Fu et al., 2019; Pillutla et al., 2019; Alistarh et al., 2018; Karimireddy et al., 2021; Allen-Zhu et al., 2020; Karimireddy et al., 2020; Gorbunov et al., 2022; Velicheti et al., 2021; Data and Diggavi, 2021) have been proposed to suppress the influence of malicious clients' updates (See Appendix A for a full discussion and comparison). In the most comparable work, Yin et al. (2018) apply a coordinate-wise median/trimmed mean to aggregate local updates. Under the assumption of bounded variance $\mathbb{E}_{p^\star}[\|\mathbf{g} - \mathbb{E}[\mathbf{g}]\|_2^2] \leq \tilde{\sigma}^2$ and coordinate-wise bounded skewness (or coordinate-wise sub-exponential) on the gradient distribution, one can achieve a statistical error of $\mathcal{O}(\tilde{\sigma}(\epsilon/\sqrt{n} + \sqrt{d/mn}))$, where ϵ, n, m are the fraction of malicious nodes, number of samples in each client, and number of clients, respectively. It has also been shown in Yin et al. (2018) that any FL protocol cannot has a rate better than $\Omega(\epsilon/\sqrt{n} + \sqrt{d/mn})$.

However, as is noted in (Yin et al., 2018), the term $\tilde{\sigma}$ usually depends on the dimension of the data d. As a concrete example, $\tilde{\sigma} = \sqrt{d}$ when the noise of the gradient

is standard Gaussian with identity covariance I. And this results in a sub-optimal rate of $\mathcal{O}(\epsilon\sqrt{d/n} + \sqrt{d^2/mn})$. Compared to the lower bound above, the rate is far from optimal in terms of the dependence on the dimension of gradients d, which is usually high for modern machine learning architectures such as neural networks. Furthermore, they also rely on strong, hard-to-verify extra assumptions, including bounded coordinate-wise skewness or sub-exponential gradients, which are unlikely to be satisfied in practice. Similar issues on the dimension dependence and strong extra assumptions also appear in the analysis other FL protocols with trace bound assumption (Blanchard et al., 2017; Karimireddy et al., 2021, 2020; Velicheti et al., 2021), which we provide detailed discussion in Appendix A.

In this paper, we make the only distributional assumption that the spectral norm of the covariance of the gradient is bounded by σ^2, as in Assumption 3 (note that the assumption in (Yin et al., 2018) in is equivalent to bounded trace of the covariance, and for Gaussian with identity covariance we have $\sigma = 1$ while $\tilde{\sigma} = \sqrt{d}$). In this case, we aim to propose FL protocols that achieve an optimal statistical rate of $\Theta(\sigma(\epsilon/\sqrt{n} + \sqrt{d/mn}))$.

On the other hand, a standard metric for robust estimators is the *breakdown point*, which measures the largest fraction of Byzantine clients an algorithm is insensitive to (Huber, 1973; Donoho, 1982). While this metric has been neglected in most of the previous work on computationally efficient high dimensional robust estimators, it is important for the practical performance of FL protocols. Thus, we ask the question:

> *Under the assumption of bounded covariance of gradients, can we design Byzantine-robust FL algorithms that are near-optimal with respect to all the parameters, ϵ, n, m, d, and achieve a good breakdown point?*

Our Contribution. In this work, we make one step towards resolving the problem by providing a thorough analysis of the statistical rate and breakdown point of FL algorithms. We identify the main reason for the sub-optimal rate in (Yin et al., 2018) as the sub-optimality in the robust aggregation algorithms. It is known that COORDINATE-WISE MEDIAN, COORDINATE-WISE TRIMMED MEAN and GEOMETRIC MEDIAN have an error of at least $\Omega(\sigma\sqrt{\epsilon d})$ under the bounded operator norm assumption for robust mean estimation (Steinhardt, 2019; Lai et al., 2016), which can be highly sub-optimal when the dimension is high. Following our lower bound analysis in Appendix C, the lower bound for robust mean estimation directly results in a lower bound of $\Omega(\sigma\sqrt{\epsilon d/n})$ for the performance of any FL protocols using either of the three

algorithms as aggregation rules.

The recent progress in statistically and computationally efficient robust estimators in high dimension (Diakonikolas et al., 2016, 2017, 2019a,c,b, 2020; Steinhardt et al., 2017, 2018; Steinhardt, 2018; Zhu et al., 2019b,a, 2022; Lugosi, 2017; Lugosi and Mendelson, 2019; Lecué and Lerasle, 2020; Lugosi and Mendelson, 2021; Hopkins et al., 2020) aim to remove the extra $\sqrt{d}$ factor in mean estimation, and thus can be helpful in achieving near-optimal rate for FL protocols. We propose Byzantine-robust protocols based on various robust estimators, including FILTERING (Diakonikolas et al., 2016; Steinhardt et al., 2017; Zhu et al., 2021), NO-REGRET (Zhu et al., 2021; Hopkins et al., 2020), GAN (Zhu et al., 2022; Gao et al., 2020), BUCKETING-NO-REGRET and BUCKETING-FILTERING (Diakonikolas et al., 2020). We show that these estimators improve the statistical rate reported in Yin et al. (2018) in terms of the dependence on dimension d. We provide a thorough comparison between different protocols in terms of their statistical rate, breakdown points and computational complexity. Our results are summarized in Table 1. Note in particular that COORDINATE-WISE MEDIAN and COORDINATE-WISE TRIMMED MEAN have a rate that loses an extra $\sqrt{d}$ compared to the lower bound. Moreover, the NO-REGRET or FILTERING algorithms can improve the dimension dependence in the term with ϵ and achieve a good breakdown point, but the rate without a Byzantine adversary is still $\widetilde{\mathcal{O}}(\sqrt{\frac{d^2}{mn}})$ instead of $\widetilde{\mathcal{O}}(\sqrt{\frac{d}{mn}})$. On the other hand, by a particular choice of parameters for NO-REGRET and FILTERING in Equation (4) and (5), one can improve the rate to $\mathcal{O}(\sqrt{\frac{\epsilon}{n} + \frac{d(\log(d)+\log(1+mnDL))}{mn}})$ while possibly sacrificing the breakdown point (since the breakdown point is not shown to match $1/2$). Here D, L are the bounds on the radius of the space of $\mathcal{W}$ and the smoothness of f. To remove the $\log(d)$ factor, we further apply a bucketing step as pre-processing, as is done in median-of-means procedure (Lugosi and Mendelson, 2019) and in Diakonikolas et al. (2020). Compared with the lower bound, we still have an extra $\log(1 + mnDL)$ factor due to the union bound analysis, which exists for all the bounds we considered. It remains an open problem whether one can remove this term and achieve exactly tight rate.

Due to inconsistency in the settings and assumptions, the table does not include the rate for (Blanchard et al., 2017; Pillutla et al., 2019; Alistarh et al., 2018; Karimireddy et al., 2021; Allen-Zhu et al., 2020; Karimireddy et al., 2020; Velicheti et al., 2021). Instead, we provide a detailed discussions on the setting, assumption and rate of these Byzantine-robust protocols in Appendix A.

We evaluate all proposed protocols under five attacks over

Algorithm	Statistical rate	Breakdown point	Computation complexity
MEDIAN (Yin et al., 2018)	$\widetilde{\mathcal{O}}(\sqrt{\frac{\epsilon d}{n}+\frac{d^2}{mn}})$	1/2	$\widetilde{\mathcal{O}}(md)$
TRIMMED MEAN (Yin et al., 2018)	$\widetilde{\mathcal{O}}(\sqrt{\frac{\epsilon d}{n}+\frac{d^2}{mn}})$	1/2	$\widetilde{\mathcal{O}}(md)$
NO-REGRET (Eq. (1))	$\widetilde{\mathcal{O}}(\sqrt{\frac{\epsilon}{n}+\frac{d^2}{mn}})$	1/3	$\widetilde{\mathcal{O}}((m+d^3)d)$
FILTERING (Eq. (2))	$\widetilde{\mathcal{O}}(\sqrt{\frac{\epsilon}{n}+\frac{d^2}{mn}})$	1/2	$\widetilde{\mathcal{O}}(\epsilon m^2 d^3)$
GAN (Eq. (3))	$\widetilde{\mathcal{O}}(\sqrt{\frac{\epsilon}{n}+\sqrt{\frac{d}{mn^2}}})$	-	-
NO-REGRET (Eq. (4))	$\mathcal{O}(\sqrt{\frac{\epsilon}{n}+\frac{d(\log(d)+\log(1+nmDL))}{mn}})$	-	$\widetilde{\mathcal{O}}((m+d^3)d)$
FILTERING (Eq. (5))	$\mathcal{O}(\sqrt{\frac{\epsilon}{n}+\frac{d(\log(d)+\log(1+nmDL))}{mn}})$	-	$\widetilde{\mathcal{O}}(\epsilon m^2 d^3)$
BUCKETING-NO-REGRET	$\mathcal{O}(\sqrt{\frac{\epsilon}{n}+\frac{d\log(1+nmDL)}{mn}})$	-	$\widetilde{\mathcal{O}}((m+d^3)d)$
BUCKETING-FILTERING	$\mathcal{O}(\sqrt{\frac{\epsilon}{n}+\frac{d\log(1+nmDL)}{mn}})$	-	$\widetilde{\mathcal{O}}(\epsilon m^2 d^3)$
Lower bound (Thm. C.1)	$\Omega(\sqrt{\frac{\epsilon}{n}+\frac{d}{mn}})$	1/2	-

Table 1: Comparison of rates between different Byzantine-robust federated learning protocols for gradients with bounded covariance and strongly convex loss function. Here we assume the desired confidence δ is some constant, and D, L are the bounds on the radius of the space of $\mathcal{W}$ and the smoothness of f. The median and trimmed mean refer to coordinate-wise operations. $\widetilde{\mathcal{O}}(\cdot)$ refers to the rate when omitting logarithmic factors. We omit the communication complexity since it is $\mathcal{O}(d)$ for all the algorithms. For the computational complexity, we only consider the computation conducted in the master machine. The '-' in breakdown point represents that the breakdown point has not been completely determined for the algorithm.

two datasets and compare with six other Byzantine-robust federated learning protocols. Evaluation results show that the proposed protocols constantly achieve optimal or close-to-optimal performance under all the attacks.

We also remark here that although the computational complexity is higher for the new proposed algorithms compared to median and trimmed mean, one can improve them to $\widetilde{\mathcal{O}}(md)$ without hurting the statistical rate by integrating the filtering / no-regret algorithm with SDP solvers or matrix multiplicative weights update, see e.g. (Cheng et al., 2019; Dong et al., 2019).

In addition, the proposed protocol can be naturally integrated with the secure aggregation schemes (Bonawitz et al., 2017) to provide *bidirectional defense* against both a semi-honest server and Byzantine-malicious clients, thus achieving Byzantine robustness and privacy simultaneously. We provide more discussions in Appendix G.

2 Preliminaries

In this section, we collect useful notation, review the general pipeline of FL and introduce the threat model and the defense goal.

Notation. We use bold lower-case letters (*e.g.* **a,b,c**) to denote (random) vectors, and $[n]$ to denote the set $\{1 \cdots n\}$. Let $\mathsf{Cov}_p(\mathbf{x}) = \mathbb{E}_p[\mathbf{x}\mathbf{x}^\top] - \mathbb{E}_p[\mathbf{x}]\mathbb{E}_p[\mathbf{x}]^\top$ be the covariance of random variable $\mathbf{x}$ distribution

p. $\|\cdot\|_2$ is the ℓ_2 norm for vector, or the spectral norm for matrix, i.e. $\|\mathbf{w}\|_2 = \mathbf{w}^\top\mathbf{w}$ for vector $\mathbf{w}$, $\|\mathbf{M}\|_2 = \sup_{\|\mathbf{v}\|_2\leq 1}\|\mathbf{M}\mathbf{v}\|_2$ for matrix $\mathbf{M}$. We write $f(x) = \mathcal{O}(g(x))$ for $x \in A$ if there exists some positive real number M such that $|f(x)| \leq Mg(x)$ for all $x \in A$. If A is not specified, we have $|f(x)| \leq Mg(x)$ for all $x \in [0, +\infty)$ (thus the notation is non-asymptotic). We use $\widetilde{\mathcal{O}}(\cdot)$ to be the big-$\mathcal{O}$ notation ignoring logarithmic factors.

We also introduce concepts from convex analysis for differentiable functions $h(\cdot) : \mathbb{R}^d \to \mathbb{R}$.

Definition 2.1 (Lipschitz). *h is L-Lipschitz if $|h(\mathbf{w}) - h(\mathbf{w}')| \leq L\|\mathbf{w} - \mathbf{w}'\|_2, \forall\, \mathbf{w}, \mathbf{w}'$.*

Definition 2.2 (Smoothness). *h is L'-smooth if $\|\nabla h(\mathbf{w}) - \nabla h(\mathbf{w}')\|_2 \leq L'\|\mathbf{w} - \mathbf{w}'\|_2, \forall\, \mathbf{w}, \mathbf{w}'$.*

Definition 2.3 (Strong convexity). *h is λ-strongly convex if $h(\mathbf{w}') \geq h(\mathbf{w}) + \langle\nabla h(\mathbf{w}), \mathbf{w}' - \mathbf{w}\rangle + \frac{\lambda}{2}\|\mathbf{w}' - \mathbf{w}\|_2^2, \forall\, \mathbf{w}, \mathbf{w}'$.*

FL Pipeline. In an FL system, there is one master machine $\mathcal{S}$ and m worker machines $\mathcal{C}_i, i \in [m]$. Each client holds n data samples drawn i.i.d. from some unknown distribution $p^\star$ on the sample space $\mathcal{Z}$. Let $f(\mathbf{w}; \mathbf{z})$ be the loss as a function of the model parameter $\mathbf{w} \in \mathcal{W} \subset \mathbb{R}^d$ and a data sample $\mathbf{z} \in \mathcal{Z}$, where $\mathcal{W}$ is the parameter space. Let $F(\mathbf{w}) = \mathbb{E}_{\mathbf{z}\sim p^\star}[f(\mathbf{w}; \mathbf{z})]$ be the population risk function. Denote by $\mathbf{z}^{i,j}$ the jth data point on the ith worker machine, and $F_i(\mathbf{w}) := \frac{1}{n}\sum_{j=1}^n f(\mathbf{w}; \mathbf{z}^{i,j})$ the

empirical risk function for the ith worker. The goal is to learn a model $\mathbf{w}$ such that the population risk function is minimized: $\mathbf{w}^* = \arg\min_{\mathbf{w}\in\mathcal{W}} F(\mathbf{w})$.

To learn $\mathbf{w}^*$, the whole system runs a T-round FL protocol. Initially, the server stores a global model $\mathbf{w}_0$. In the tth round, $\mathcal{S}$ broadcasts the global model $\mathbf{w}_{t-1}$ to the m clients. The clients then run the local optimizers (e.g., SGD, Adam, RMSprop), compute the difference $\mathbf{g}_i(\mathbf{w}_{t-1})$ between the optimized model and the global model, and upload the difference to $\mathcal{S}$. In the tth round, $\mathcal{S}$ aggregate the differences and updates the global model to $\mathbf{w}_t$ according to some protocol (e.g. the averaging protocol $\mathbf{w}_t = \mathbf{w}_{t-1} + \frac{1}{m}\sum_{i=1}^m \mathbf{g}_i(\mathbf{w}_{t-1})$).

Threat Model. We assume that clients are ϵ-Byzantine malicious, meaning that in each round, at most ϵm clients are malicious: they can deviate arbitrarily from the protocol and tamper with their own updates for profitable or even adversarial purposes. The master machine communicates with the worker machines using some predefined protocol. The Byzantine machines need not obey this protocol and can send arbitrary messages to the master; in particular, they may have complete knowledge of the system and learning algorithms, and can collude with each other. We also clarify that there is no collusion between the server and the clients. That is, the server cannot disguise itself as a client or hire clients to launch colluded attacks.

Defense objective. We would like to design some protocol which achieves a statistically optimal rate in the presence of malicious clients with an appropriate breakdown point, communication complexity and computational complexity. For a strongly convex and smooth loss function F, we would like to design some protocol which outputs some parameter w^T at round T such that with high probability,

$$\lim_{T\to\infty} \|\mathbf{w}^T - \mathbf{w}^\star\| \le \Delta(\epsilon, n, m, d),$$

where $\Delta(\epsilon, n, m, d) = \mathcal{O}(\sqrt{\frac{\epsilon}{n} + \frac{d}{mn}})$ is the optimal statistical rate achievable in this case, according to the lower bound in Theorem C.1.

On the other hand, we define the breakdown point for the estimation problem as

$$b^\star = \inf\{\epsilon \mid \lim_{T\to\infty}\mathbb{E}[\|\mathbf{w}^T - \mathbf{w}^\star\|] = \infty\}.$$

From the definition, we know that for any ϵ_0 with $\Delta(\epsilon_0, n, m, d) < \infty$, one has $b^\star > \epsilon_0$.

Algorithm 1 BYZANTINE-ROBUST FEDERATED LEARNING WITH BUCKETING

Require: Initialize parameter vector $\mathbf{w}^0 \in \mathcal{W}$, step size η_t, local model update interval H, number of disjoint buckets k, and total iteration T.
Master machine: send $\mathbf{w}^0$ to all the worker machines.
for $t = 1, 2, \ldots, T-1$ **do**
  for $i \in [m]$ **do**
    Worker machine i: update local parameter via $\mathbf{w}_i^t = \mathbf{w}_i^{t-1} - \eta_t \nabla F_i(\mathbf{w}_i^{t-1})$.
    if $t \bmod H = 0$ **then** send local model update $\mathbf{g}_i(\mathbf{w}_i^t) = \mathbf{w}_i^t - \mathbf{w}_i^{t-H}$ to master machine.
    end if
  end for
  if $t \bmod H = 0$ **then**
    Master machine: randomly bucket the m gradients into k disjoint buckets of equal size and compute their empirical means $\mathbf{z}^1, \cdots, \mathbf{z}^k$. set $\mathcal{D}_m = \{\mathbf{z}^j\}_{j=1}^k$, compute

    $\mathbf{g}(\mathbf{w}^t) \leftarrow \mathsf{RobustEstimationSubroutine}(\mathcal{D}_m)$.

    update model parameter $\mathbf{w}^t \leftarrow \Pi_{\mathcal{W}}(\mathbf{w}^{t-H} + \mathbf{g}(\mathbf{w}^t))$, send and update $\mathbf{w}^t$ to all other worker machines by setting $\mathbf{w}_i^t = \mathbf{w}^t, i \in [m]$.
  end if
end for

---

3 Byzantine-Robust Federated Learning Protocols

In this section, we first present the meta-protocol that we will use throughout this paper. After that, we introduce existing robust estimators as a building block for the protocol, and provide theoretical guarantees for the special case of the protocol without bucketing. Then we show how bucketing helps improve the rate to exactly optimal. At the end of this section, we illustrate how one can naturally combine the protocols with security guarantees when the server is semi-honest.

3.1 Description of the main FL Protocol

Our proposed FL protocol is quite straightforward: in each round, the honest worker machines compute the gradients of their local loss functions and then perform local update of their models. The Byzantine machines may update their parameters in an arbitrary way. In every H parallel iteration of the algorithms, all the worker machines send the current updates to the master machine. The master machine, after receiving the updates from all workers, first divides the updates into k different buckets, computes the mean of each bucket, and then aggregates the

updates according to some robust estimation subroutine. At the end, the master machine broadcasts the updated parameter to all worker machines. We propose to apply the three robust estimators as aggregation subroutines for the FL protocol. The resulting FL protocols are presented formally in Algorithm 1.

The RobustEstimationSubroutine can be replaced with any robust mean estimator. In this paper, we will rely on three recently proposed robust estimators as building blocks, namely NO-REGRET algorithm (Hopkins et al., 2020; Zhu et al., 2021), FILTERING algorithm (Diakoniko-las et al., 2017; Li, 2018; Steinhardt et al., 2018; Zhu et al., 2021) and a GAN-based algorithm (Gao et al., 2018, 2020; Zhu et al., 2022). It has been shown that each of these algorithms are efficient with near-optimal performance guarantee. Due to the limit of space, we present the three algorithms and discuss their guarantee for robust mean estimation in Appendix B. To distinguish between different choices of parameters for the algorithms, we list the five choices of algorithms we will use as RobustEstimationSubroutine here:

$$\mu_1 = \text{NO-REGRET}\left(\mathcal{D}_m, \epsilon, \frac{\eta_t^2 \sigma^2}{n}, \left(\frac{2\eta+7}{3(1-(3+\eta)\epsilon)}\right)^2 \cdot (1 + \frac{d\log(d/\delta)}{m\epsilon}) \cdot \frac{\eta_t^2 \sigma^2}{n}, \eta\right) \quad (1)$$

$$\mu_2 = \text{FILTERING}\left(\mathcal{D}_m, \frac{2(1-\epsilon)}{(1-2\epsilon)^2} \cdot (1 + \frac{d\log(d/\delta)}{m\epsilon}) \cdot \frac{\eta_t^2 \sigma^2}{n}\right) \quad (2)$$

$$\mu_3 = \text{GAN}\left(\mathcal{D}_m, \frac{\eta_t^2 \sigma^2}{n}\right) \quad (3)$$

$$\mu_4 = \text{NO-REGRET}\left(\mathcal{D}_m, \frac{\eta_t^2 \sigma^2}{n}, \eta, \frac{C_1}{(1 - C_2(\epsilon + \frac{\log(1/\delta)}{n})^2)} \cdot \left(1 + \frac{d\log(d) + \log(1/\delta)}{m\epsilon}\right) \cdot \frac{\eta_t^2 \sigma^2}{n}\right) \quad (4)$$

$$\mu_5 = \text{FILTERING}\left(\mathcal{D}_m, \frac{C_3}{(1 - C_4(\epsilon + \frac{\log(1/\delta)}{n})^2)} \cdot \left(1 + \frac{d\log(d) + \log(1/\delta)}{m\epsilon}\right) \cdot \frac{\eta_t^2 \sigma^2}{n}\right), \quad (5)$$

where the definition of FILTERING, NO-REGRET and GAN can be found in Appendix B.

3.2 Theoretical Analysis for Byzantine-Robust Federated Learning Protocol

We begin with the most basic case of Algorithm 1, where no bucketing is used and the master machine communicates with the worker machines in each round, i.e., $k = m, H = 1$. Throughout the paper, we make the following assumptions about the parameter space and loss function. First, we assume that the parameter space is convex and compact.

Assumption 1 (Convex and compact parameter space). *The parameter space $\mathcal{W}$ is convex and compact with diameter D, i.e, $\forall \mathbf{w}, \mathbf{w}' \in \mathcal{W}$, one has $\|\mathbf{w} - \mathbf{w}'\|_2 \leq D$.*

Second, we assume that each loss function $f(\mathbf{w}; \mathbf{z})$ is L-smooth, which also implies that the population risk function is L-smooth.

Assumption 2 (Smoothness of f and F). *For any $\mathbf{z} \in \mathcal{Z}$, $f(\cdot; \mathbf{z})$ is L-smooth: $\|\nabla f(\mathbf{w}_1; \mathbf{z}) - \nabla f(\mathbf{w}_2; \mathbf{z})\| \leq$*

$L\|\mathbf{w}_1 - \mathbf{w}_2\|$.

Third, we impose the assumption that the distribution of the gradient of the loss function $f(\cdot; \mathbf{z})$ has bounded covariance:

Assumption 3 (Bounded covariance of gradient). *For any $\mathbf{w} \in \mathcal{W}$, $\|\text{Cov}_{p^\star}(\nabla f(\mathbf{w}; \mathbf{z}))\|_2 \leq \sigma^2$.*

Note that this assumption considers the spectral norm of the covariance matrix. The corresponding upper bound σ^2 is usually dimension-free and is different from the bounded variance assumption in Yin et al. (2018), which assumes that the trace of covariance $\text{Tr}(\text{Cov}_{p^\star}(\nabla f(\mathbf{w}; \mathbf{z}))) = \mathbb{E}_{p^\star}[\|\nabla f(\mathbf{w}; \mathbf{z})) - \mathbb{E}_{p^\star}[\nabla f(\mathbf{w}; \mathbf{z}))]\|_2^2]$ is bounded by $\tilde{\sigma}^2$. As a concrete example, when the distribution of the gradient is standard Gaussian with covariance I. We have $\sigma^2 = 1$ while $\tilde{\sigma}^2 = d$. In the worst case, when Assumption 3 is satisfied, one has $\text{Tr}(\text{Cov}_{p^\star}(\nabla f(\mathbf{w}; \mathbf{z}))) \leq d\sigma^2$, which introduces an extra dimension factor. We provide more discussions in Appendix A.

In this paper, we consider three different categories of population risk function F, namely strongly convex, non-strongly convex and smooth non-convex functions. We also include the extra assumptions that are necessary for analysis in each case.

Assumption 4. *Consider the following three options of assumptions on loss function F.*

(a) (Strongly convex) $F(\cdot)$ is λ-strongly convex. The minimizer of $F(\cdot)$ in $\mathcal{W}$, $\mathbf{w}^\star$, is also the minimizer of $F(\cdot)$ in $\mathbb{R}^d$, i.e., $\nabla F(\mathbf{w}^*) = 0$.

(b) (Convex) $F(\cdot)$ is convex. The minimizer of $F(\cdot)$ in $\mathcal{W}$, $\mathbf{w}^\star$, is also the minimizer of $F(\cdot)$ in $\mathbb{R}^d$, i.e., $\nabla F(\mathbf{w}^*) = 0$. The parameter space $\mathcal{W}$ contains the following ℓ_2 ball centered at $\mathbf{w}^*$: $\{\mathbf{w} \in \mathbb{R}^d : \|\mathbf{w} - \mathbf{w}^*\|_2 \leq 2\|\mathbf{w}^0 - \mathbf{w}^*\|_2\}$.

(c) (Non-convex) Suppose that $\forall \mathbf{w} \in \mathcal{W}$, $\|\nabla F(\mathbf{w})\|_2 \leq M$. We assume that $\mathcal{W}$ contains the ℓ_2 ball $\{\mathbf{w} \in \mathbb{R}^d : \|\mathbf{w} - \mathbf{w}^0\|_2 \leq \frac{2}{\Delta^2}(M+\Delta)(F(\mathbf{w}^0) - F(\mathbf{w}^*))\}$, where Δ is defined later as in Equation (6), (7), (8), (9) or (10), depending on the subroutine used.

Note that we do not make any explicit assumption on the convexity of the individual loss functions $f(\cdot; \mathbf{z})$. We also remark here that for the simplicity of analysis, we focus on the case of $H = 1$ in the main text, which means that the worker machine communicates gradient information with the master machine in each round. We show in Appendix F how to extend the analysis to multi-round local model updates and present the corresponding theorem there. Our main technical results on the FL protocol are stated as below.

5

Theorem 3.1. *Let $\mathbf{w}_i^t, i \in \{1,2,3,4,5\}$ denote the output of Algorithm 1 with $H = 1, k = m$ and step-size $\eta_t = 1/L$ at round t when the* RobustEstimationSubroutine *takes the algorithm in Equation (1), (2), (3), (4), (5), respectively. Under Assumption 1, 2, 3 and 4(a), with probability at least $1 - \delta$,*

$$\|\mathbf{w}_i^T - \mathbf{w}^*\|_2 \leq (1 - \frac{\lambda}{L + \lambda})^T \|\mathbf{w}^0 - \mathbf{w}^*\|_2 + \frac{2}{\lambda}\Delta_i,$$

where for some universal constants C_i,

$$\Delta_1 := \mathcal{O}\left(\frac{\sigma}{(1-3\epsilon)\sqrt{n}} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d^2\log(1+nmDL) + d\log(d/\delta)}{m}}\right)\right), \quad (6)$$

$$\Delta_2 := \mathcal{O}\left(\frac{\sigma}{(1-2\epsilon)\sqrt{n}} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d^2\log(1+nmDL) + d\log(d/\delta)}{m}}\right)\right), \quad (7)$$

$$\Delta_3 := \mathcal{O}\left(\frac{\sigma}{(1-C_1\epsilon)\sqrt{n}} \cdot \left(\sqrt{\epsilon + \sqrt{\frac{d\log(1+nmDL) + \log(1/\delta)}{m}}}\right)\right), \quad (8)$$

$$\Delta_4 := \mathcal{O}\left(\frac{\sigma}{1-C_2(\epsilon+\log(1/\delta)/m)} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d(\log(d) + \log(1+nmDL)) + \log(1/\delta)}{m}}\right)\right), \quad (9)$$

$$\Delta_5 := \mathcal{O}\left(\frac{\sigma}{1-C_3(\epsilon+\log(1/\delta)/m)} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d(\log(d) + \log(1+nmDL)) + \log(1/\delta)}{m}}\right)\right), \quad (10)$$

Under Assumption 1, 2, 3 and 4(b), with probability at least $1 - \delta$, after $T = \frac{L}{\Delta_i}\|\mathbf{w}^0 - \mathbf{w}^*\|_2$ parallel iterations, we have

$$F(\mathbf{w}_i^T) - F(\mathbf{w}^*) \leq 16\|\mathbf{w}^0 - \mathbf{w}^*\|_2 \Delta_i\left(1 + \frac{1}{2L}\Delta_i\right).$$

Under Assumption 1, 2, 3 and 4(c), with probability at least $1 - \delta$, we have

$$\min_{t=0,1,...,T} \|\nabla F(\mathbf{w}_i^t)\|_2^2 \leq \frac{2L}{T}(F(\mathbf{w}^0) - F(\mathbf{w}^*)) + \Delta_i^2.$$

We prove Theorem 3.1 in Appendix D. We also provide the analysis for the case of multi-round updates ($H > 1$) in Appendix F. We can see from the statement that the conclusion takes a similar form as that in Yin et al. (2018) except that our results come with a tighter statistical error bound under weaker distributional assumptions. Under the same assumption, the best rate achieved by Yin et al. (2018) as $T \to \infty$ is $\widetilde{\mathcal{O}}(\sqrt{\frac{\epsilon d}{n} + \frac{d^2}{mn}})$, while our algorithm improves to $\mathcal{O}(\sqrt{\frac{\epsilon}{n} + \frac{d\log(d)}{mn}})$. For convex losses, we show that the excess risk converges at a rate of Δ_i. For the non-convex case, we show that it approaches the critical point with a rate of Δ_i. In Section C, we provide a lower bound showing that the worst-case statistical rate of parameter estimation error under strongly convex loss is at least $\Omega(\sqrt{\frac{\epsilon}{n} + \frac{d}{mn}})$, matching our upper bound for the strongly convex case.

For the five different estimators, we can see that the first two estimators based on NO-REGRET and FILTERING

in Equation (1) and (2) have high breakdown point $1/3$ and $1/2$, respectively. But the rate is sub-optimal in terms of the dependence in d. Technically, this is from the covering argument when applied to $\sqrt{\frac{d\log(1/\delta)}{mn}}$ instead of $\sqrt{\frac{d+\log(1/\delta)}{mn}}$. This shows the importance of achieving sub-Gaussian rate in robust estimation. The GAN estimator in Equation (3) is usually easier to compute in high dimension. However, the rate is sub-optimal due to the $(d/m)^{1/4}$ factors. We can improve the rate of NO-REGRET and FILTERING by taking different parameters as in Equation (4) and (5). However, the breakdown point is not as high. It is an open problem whether one can design estimators that are statistically optimal, computationally efficient, and have optimal breakdown point.

Computationally, the proposed estimators are inferior to coordinate-wise median or trimmed mean due to the query of second moment. On one hand, one can improve the implementation by combining the no-regret and filtering algorithm with covering / packing SDP solver or quantum entropy scoring to improve the computational complexity to linear, see e.g. (Cheng et al., 2019; Dong et al., 2019). Furthermore, one can also split the data dimension into K groups, thus increasing the sample complexity by a $\sqrt{K}$ factor while reducing the computational complexity significantly. We omit the details here since the paper mainly focuses on the statistical part.

3.3 Improved Protocol via Bucketing

We can improve the rate and remove the extra $\log(d)$ factor in Theorem 3.1 by setting a nontrivial bucket size k in Algorithm 1. The idea of bucketing was first introduced in the median-of-means procedure (Lecué and Lerasle, 2020). Diakonikolas et al. (2020) showed that the bucketing procedure, combined with FILTERING or NO-REGRET algorithm, can achieve a sub-Gaussian rate, which is mini-max rate-optimal with respect to all parameters. The concrete subroutine in Algorithm 1 is chosen to be one of the following based on algorithms in Appendix B.1:

$$\mu_1 = \text{NO-REGRET}\left(\mathcal{D}_m, 0.1, \frac{k\sigma^2}{mn}, \mathcal{O}\left(\frac{(d+k)\eta_t^2\sigma^2}{mn}\right), 0.1\right), \quad (11)$$

$$\mu_2 = \text{FILTERING}\left(\mathcal{D}_m, \mathcal{O}\left(\frac{\eta_t^2(d+k)\sigma^2}{mn}\right)\right). \quad (12)$$

With the bucketing step, one can improve the statistical bound to near-optimal, as stated in the following theorem.

Theorem 3.2. *Let $\mathbf{w}^t$ denote the output of Algorithm 1 with step-size $\eta = 1/L$, $H = 1$ and $k = \lfloor 2\epsilon m +$*

$\log(1/\delta)\rfloor$. *Under Assumption 1, 2, 3 and 4(a), with probability at least $1 - \delta$, we have*

$$\|\mathbf{w}^T - \mathbf{w}^*\|_2 \le (1 - \frac{\lambda}{L + \lambda})^T \|\mathbf{w}^0 - \mathbf{w}^*\|_2 + \frac{2}{\lambda}\Delta,$$

where for some universal constant C,

$$\Delta := \mathcal{O}\left(\frac{\sigma}{(1 - C\epsilon)\sqrt{n}} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d\log(1 + nmDL) + \log(1/\delta)}{m}} \right) \right) \tag{13}$$

Under Assumption 1, 2, 3 and 4(b), with probability at least $1 - \delta$, after $T = \frac{L}{\Delta}\|\mathbf{w}^0 - \mathbf{w}^*\|_2$ iterations,

$$F(\mathbf{w}_i^T) - F(\mathbf{w}^*) \le 16\|\mathbf{w}^0 - \mathbf{w}^*\|_2\Delta\left(1 + \frac{1}{2L}\Delta\right).$$

Under Assumption 1, 2, 3 and 4(c), with probability at least $1 - \delta$, we have

$$\min_{t=0,1,\dots,T} \|\nabla F(\mathbf{w}^t)\|_2^2 \le \frac{2L}{T}(F(\mathbf{w}^0) - F(\mathbf{w}^*)) + \Delta^2.$$

The proof is deferred to Appendix E. Compared to Theorem 3.1, the extra $\log(d)$ factor is removed and the rate exactly matches the lower bound in Theorem C.1 when $nmDL = O(1)$. A similar idea of bucketing also appears in Karimireddy et al. (2020), motivated from heterogeneous clients. Our results establish the benefit of the bucketing even when the clients are homogeneous.

4 Evaluation

4.1 Experiment Setup

Datasets, Models & Runtime. We evaluated the robust aggregators on two datasets, MNIST under CC BY-SA 3.0 License (LeCun et al., 2010) and F-MNIST under MIT License (Xiao et al., 2017). The training and testing is by default as implemented in PyTorch. The models used in most experiments are a convolutional network with two convolutional layers followed by two fully connected layers with ReLU. In each round, each client runs 5 local epochs with batch size 10 before submitting the updated local model to the server. All the experiments were conducted on Ubuntu18.04 LTS servers with 56 2.6GHz Intel Xeon CPUs, 252G RAM and 8 Geforce GTX 1080 Ti. The code for evaluation is provided in `https://github.com/wanglun1996/secure-robust-federated-learning`.

Attacks. We chose the following attacks to evaluate the robust estimators: Krum Attack (KA) (Fang et al., 2020), Trimmed Mean Attack (TMA) (Fang et al., 2020), Model Poisoning Attack (MPA) (Bhagoji et al., 2019), Model

Replacement Attack (MRA) (Bagdasaryan et al., 2020), Distributed Backdoor Attack (DBA) (Xie et al., 2019) and Inner-Production Manipulation Attack (IMA) (Xie et al., 2020). We report model accuracy for all attacks and also report attack success rate for backdoor attacks (*i.e.* MPA, MRA and DBA.)

Robust Aggregators. We also chose nine other Byzantine-robust FL protocols as baselines: (1) Krum (Blanchard et al., 2017); (2) Trimmed Mean (Yin et al., 2018); (3) Median (Yin et al., 2018); (4) Bulyan Krum (El El Mhamdi et al., 2018); (5) Bulyan Trimmed Mean (El El Mhamdi et al., 2018); (6) Bulyan Median (El El Mhamdi et al., 2018); (7) protocol in Karimireddy et al. (2021); (8) protocol in Karimireddy et al. (2020); (9) protocol in Velicheti et al. (2021). As Sever (Diakonikolas et al., 2019a) requires much more rounds than other aggregators, we defer the evaluation of Sever to Appendix I. For FILTERING, NO-REGRET, and the bucketing variants, it is computationally expensive to calculate the operator norm of the covariance matrix of the whole model. Hence, we cut each layer of the model into several intervals and run the above robust aggregators on each interval. For bucketing variants, we set the bucket size to 2.

4.2 Evaluation Results

FILTERING, NO-REGRET and BUCKETING Variants. In this part, the data is randomly distributed to the 100 federated clients, out of which 20 are malicious. For all attacks except DBA, we set the learning rate to 10^{-3} and the bounded variance as 10^{-5}. For DBA, we set the learning rate to 0.1 and the bounded variance to 10^{-4} because DBA is launched on a pre-trained model. For the proposed algorithms, the parameters are cut into intervals of size 1000.

As shown in Fig. 1, we observe that FILTERING and NO-REGRET achieve consistently good performance under all the chosen attacks, while all the other aggregators fail on at least one attack. For instance, Krum and Bulyan Krum's accuracy drops on KA and typically cannot achieve as good accuracy as other estimators; Trimmed Mean fails on TMA and MPA; Median, Bulyan Trimmed Mean and Bulyan Median fail on MPA. We also observe that the Bucketing variants do not perform as well as FILTERING and NO-REGRET. We conjecture this is due to their low breaking point.

GAN. In this part, we evaluate the performance of the GAN. Because GAN needs a large amount of training data, we increase the number of clients to 1000, out of which 200 are malicious. The learning rate is 0.1. For the proposed algorithms, we set the bounded variance to
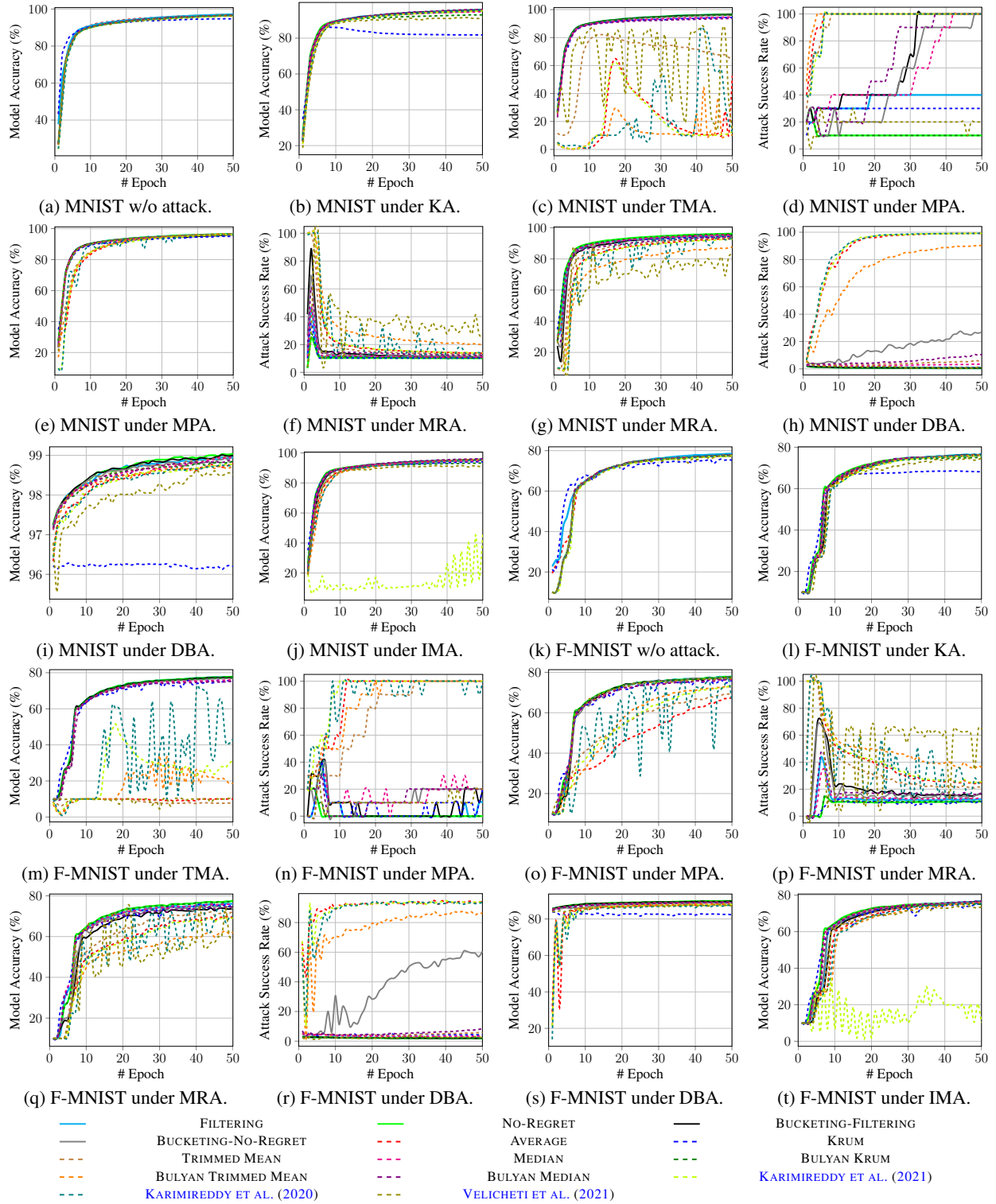
(a) MNIST w/o attack.   (b) MNIST under KA.   (c) MNIST under TMA.   (d) MNIST under MPA.

(e) MNIST under MPA.   (f) MNIST under MRA.   (g) MNIST under MRA.   (h) MNIST under DBA.

(i) MNIST under DBA.   (j) MNIST under IMA.   (k) F-MNIST w/o attack.   (l) F-MNIST under KA.

(m) F-MNIST under TMA.   (n) F-MNIST under MPA.   (o) F-MNIST under MPA.   (p) F-MNIST under MRA.

(q) F-MNIST under MRA.   (r) F-MNIST under DBA.   (s) F-MNIST under DBA.   (t) F-MNIST under IMA.

Figure 1: Robust estimators' performance under attacks.

(a) MNIST w/o attack. (b) MNIST under TMA.
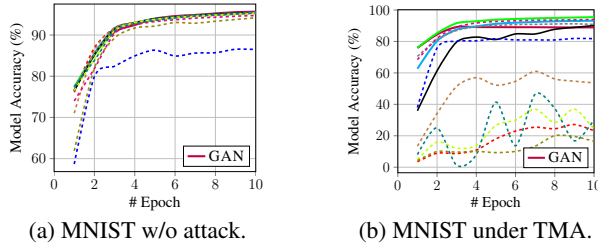
Figure 2: GAN's performance under attack. Please refer to Fig. 1 for the complete legend.

10^{-3} and the parameters are cut into intervals of size 100. We also do not evaluate the Bulyan aggregators because of their impractically long running time (*e.g.* ~ 200 hours per round for Bulyan Krum).

As shown in Fig. 2, GAN does not achieve as good performance as FILTERING and NO-REGRET. During the experiments, we observe that although GAN is usually faster in running time than FILTERING and NO-REGRET, it can be sensitive to hyper-parameters. This makes it hard to deploy in real-world applications compared to other proposed methods. Consistent with Fig. 1, KRUM, Karimireddy et al. (2020) and Karimireddy et al. (2021) cannot achieve as good performance as other aggregators when there is no attack, and this phenomenon becomes more significant when the client number is larger.

5 Conclusion

In this paper, we propose Byzantine-robust FL protocols with optimal statistical rate and privacy guarantees. Furthermore, we show on benchmark data that the proposed protocols achieve consistently good performance under different attacks. This work leaves two main open questions: Can one design simple and practically implementable Byzantine-robust FL protocol with optimal statistical rate and linear computational complexity? Besides these two properties, can a Byzantine-robust FL protocol also achieve high breakdown point simultaneously?

6 Acknowledgements

References

Dan Alistarh, Zeyuan Allen-Zhu, and Jerry Li. Byzantine stochastic gradient descent. In *Advances in Neural Information Processing Systems*, pages 4613–4623, 2018.

Zeyuan Allen-Zhu, Faeze Ebrahimian, Jerry Li, and Dan Alistarh. Byzantine-resilient non-convex stochastic gradient descent. *arXiv preprint arXiv:2012.14368*, 2020.

Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. How to backdoor federated learning. In *International Conference on Artificial Intelligence and Statistics*, pages 2938–2948, 2020.

James Henry Bell, Kallista A Bonawitz, Adrià Gascón, Tancrède Lepoint, and Mariana Raykova. Secure single-server aggregation with (poly) logarithmic overhead. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, pages 1253–1269, 2020.

Arjun Nitin Bhagoji, Supriyo Chakraborty, Prateek Mittal, and Seraphin Calo. Analyzing federated learning through an adversarial lens. In *International Conference on Machine Learning*, pages 634–643, 2019.

Peva Blanchard, Rachid Guerraoui, Julien Stainer, et al. Machine learning with adversaries: Byzantine tolerant gradient descent. In *Advances in Neural Information Processing Systems*, pages 119–129, 2017.

Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1175–1191, 2017.

Sébastien Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

Lukas Burkhalter, Hidde Lycklama, Alexander Viand, Nicolas Küchler, and Anwar Hithnawi. Rofl: Attestable robustness for secure federated learning. *arXiv preprint arXiv:2107.03311*, 2021.

Chien-Lun Chen, Leana Golubchik, and Marco Paolieri. Backdoor attacks on federated meta-learning. *arXiv preprint arXiv:2006.07026*, 2020.

Lingjiao Chen, Hongyi Wang, Zachary Charles, and Dimitris Papailiopoulos. Draco: Byzantine-resilient distributed training via redundant gradients. In *International Conference on Machine Learning*, pages 903–912. PMLR, 2018.

Mengjie Chen, Chao Gao, and Zhao Ren. A general decision theory for Huber's ϵ-contamination model. *Electronic Journal of Statistics*, 10(2):3752 – 3774, 2016. doi: 10.1214/16-EJS1216. URL https://doi.org/10.1214/16-EJS1216.

Yu Cheng, Ilias Diakonikolas, and Rong Ge. High-dimensional robust mean estimation in nearly-linear time. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2755–2771. SIAM, 2019.

Deepesh Data and Suhas Diggavi. Byzantine-resilient high-dimensional sgd with local iterations on heterogeneous data. In *International Conference on Machine Learning*, pages 2478–2488. PMLR, 2021.

Ilias Diakonikolas, Gautam Kamath, Daniel M Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robust estimators in high dimensions without the computational intractability. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 655–664. IEEE, 2016.

Ilias Diakonikolas, Gautam Kamath, Daniel M Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Being robust (in high dimensions) can be practical. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 999–1008. JMLR. org, 2017.

Ilias Diakonikolas, Gautam Kamath, Daniel Kane, Jerry Li, Jacob Steinhardt, and Alistair Stewart. Sever: A robust meta-algorithm for stochastic optimization. In *International Conference on Machine Learning*, pages 1596–1606. PMLR, 2019a.

Ilias Diakonikolas, Daniel Kane, and Pasin Manurangsi. Nearly tight bounds for robust proper learning of halfspaces with a margin. *Advances in Neural Information Processing Systems*, 32, 2019b.

Ilias Diakonikolas, Weihao Kong, and Alistair Stewart. Efficient algorithms and lower bounds for robust linear regression. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2745–2754. SIAM, 2019c.

Ilias Diakonikolas, Daniel M Kane, and Ankit Pensia. Outlier robust mean estimation with subgaussian rates via stability. *Advances in Neural Information Processing Systems*, 33:1830–1840, 2020.

Yihe Dong, Samuel Hopkins, and Jerry Li. Quantum entropy scoring for fast robust mean estimation and improved outlier detection. In *Advances in Neural Information Processing Systems*, pages 6067–6077, 2019.

David L Donoho. Breakdown properties of multivariate location estimators. Technical report, Technical report, Harvard University, Boston, 1982.

Mahdi El El Mhamdi, Rachid Guerraoui, Sébastien Rouault, et al. The hidden vulnerability of distributed learning in byzantium. In *International Conference on Machine Learning*, pages 3521–3530. PMLR, 2018.

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)*, pages 1605–1622, 2020.

Shuhao Fu, Chulin Xie, Bo Li, and Qifeng Chen. Attack-resistant federated learning with residual-based reweighting. *arXiv preprint arXiv:1912.11464*, 2019.

Clement Fung, Chris JM Yoon, and Ivan Beschastnikh. Mitigating sybils in federated learning poisoning. *arXiv preprint arXiv:1808.04866*, 2018.

Clement Fung, Chris JM Yoon, and Ivan Beschastnikh. The limitations of federated learning in sybil settings. In *23rd International Symposium on Research in Attacks, Intrusions and Defenses ({RAID} 2020)*, pages 301–316, 2020.

Chao Gao, Jiyi Liu, Yuan Yao, and Weizhi Zhu. Robust estimation and generative adversarial nets. *arXiv preprint arXiv:1810.02030*, 2018.

Chao Gao, Yuan Yao, and Weizhi Zhu. Generative adversarial nets for robust scatter estimation: A proper scoring rule perspective. *J. Mach. Learn. Res.*, 21: 160–1, 2020.

Eduard Gorbunov, Samuel Horváth, Peter Richtárik, and Gauthier Gidel. Variance reduction is an antidote to byzantines: Better rates, weaker assumptions and communication compression as a cherry on the top. *arXiv preprint arXiv:2206.00529*, 2022.

Lie He, Sai Praneeth Karimireddy, and Martin Jaggi. Secure byzantine-robust machine learning. *arXiv preprint arXiv:2006.04747*, 2020.

Sam Hopkins, Jerry Li, and Fred Zhang. Robust and heavy-tailed mean estimation made simple, via regret minimization. *Advances in Neural Information Processing Systems*, 33:11902–11912, 2020.

Peter J Huber. Robust regression: asymptotics, conjectures and monte carlo. *The Annals of Statistics*, 1(5): 799–821, 1973.

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.

Sai Praneeth Karimireddy, Lie He, and Martin Jaggi. Byzantine-robust learning on heterogeneous datasets via bucketing. *arXiv preprint arXiv:2006.09365*, 2020.

Sai Praneeth Karimireddy, Lie He, and Martin Jaggi. Learning from history for byzantine robust optimization. In *International Conference on Machine Learning*, pages 5311–5319. PMLR, 2021.

Kevin A Lai, Anup B Rao, and Santosh Vempala. Agnostic estimation of mean and covariance. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 665–674. IEEE, 2016.

Guillaume Lecué and Matthieu Lerasle. Robust machine learning by median-of-means: theory and practice. *The Annals of Statistics*, 48(2):906–931, 2020.

Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

Jerry Zheng Li. *Principled approaches to robust machine learning and beyond*. PhD thesis, Massachusetts Institute of Technology, 2018.

Yehuda Lindell. *How to Simulate It – A Tutorial on the Simulation Proof Technique*, pages 277–346. Springer International Publishing, Cham, 2017. ISBN 978-3-319-57048-8. doi: 10.1007/978-3-319-57048-8_6. URL https://doi.org/10.1007/978-3-319-57048-8_6.

Gábor Lugosi. Lectures on combinatorial statistics, 2017.

Gábor Lugosi and Shahar Mendelson. Sub-gaussian estimators of the mean of a random vector. *The Annals of Statistics*, 47(2):783–794, 2019.

Gabor Lugosi and Shahar Mendelson. Robust multivariate mean estimation: the optimality of trimmed mean. *The Annals of Statistics*, 49(1):393–410, 2021.

Xinjian Luo, Yuncheng Wu, Xiaokui Xiao, and Beng Chin Ooi. Feature inference attack on model predictions in vertical federated learning. In *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, pages 181–192. IEEE, 2021.

Brendan McMahan and Daniel Ramage. Federated learning: Collaborative machine learning without centralized training data. *Google Research Blog*, 3, 2017.

Milad Nasr, Reza Shokri, and Amir Houmansadr. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE symposium on security and privacy (SP)*, pages 739–753. IEEE, 2019.

Mustafa Safa Ozdayi, Murat Kantarcioglu, and Yulia R Gel. Preventing backdoors in federated learning by adjusting server-side learning rate. 2020.

Krishna Pillutla, Sham M Kakade, and Zaid Harchaoui. Robust aggregation for federated learning. *arXiv preprint arXiv:1912.13445*, 2019.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18. IEEE, 2017.

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

Jacob Steinhardt. *Robust learning: Information theory and algorithms*. PhD thesis, Stanford University, 2018.

Jacob Steinhardt. Lecture notes for stat260 (robust statistics). 2019.

Jacob Steinhardt, Pang Wei W Koh, and Percy S Liang. Certified defenses for data poisoning attacks. In *Advances in neural information processing systems*, pages 3517–3529, 2017.

Jacob Steinhardt, Moses Charikar, and Gregory Valiant. Resilience: A criterion for learning in the presence of arbitrary outliers. In *9th Innovations in Theoretical Computer Science Conference (ITCS 2018)*, volume 94, page 45. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2018.

Gan Sun, Yang Cong, Jiahua Dong, Qiang Wang, Lingjuan Lyu, and Ji Liu. Data poisoning attacks on federated machine learning. *IEEE Internet of Things Journal*, 2021.

Ziteng Sun, Peter Kairouz, Ananda Theertha Suresh, and H Brendan McMahan. Can you really backdoor federated learning? *arXiv preprint arXiv:1911.07963*, 2019.

Raj Kiriti Velicheti, Derek Xia, and Oluwasanmi Koyejo. Secure byzantine-robust distributed learning via clustering. *arXiv preprint arXiv:2110.02940*, 2021.

Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

Hongyi Wang, Kartik Sreenivasan, Shashank Rajput, Harit Vishwakarma, Saurabh Agarwal, Jy-yong Sohn, Kangwook Lee, and Dimitris Papailiopoulos. Attack of the tails: Yes, you really can backdoor federated learning. *Advances in Neural Information Processing Systems*, 33:16070–16084, 2020.

Zhibo Wang, Mengkai Song, Zhifei Zhang, Yang Song, Qian Wang, and Hairong Qi. Beyond inferring class representatives: User-level privacy leakage from federated learning. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pages 2512–2520. IEEE, 2019.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashionmnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Chulin Xie, Keli Huang, Pin-Yu Chen, and Bo Li. Dba: Distributed backdoor attacks against federated learning. In *International Conference on Learning Representations*, 2019.

Cong Xie, Oluwasanmi Koyejo, and Indranil Gupta. Fall of empires: Breaking byzantine-tolerant sgd by inner product manipulation. In *Uncertainty in Artificial Intelligence*, pages 261–270. PMLR, 2020.

Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International Conference on Machine Learning*, pages 5650–5659. PMLR, 2018.

Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. Defending against saddle point attack in byzantine-robust distributed learning. In *International Conference on Machine Learning*, pages 7074–7084. PMLR, 2019.

Banghua Zhu, Jiantao Jiao, and Jacob Steinhardt. Generalized resilience and robust statistics. *arXiv preprint arXiv:1909.08755*, 2019a.

Banghua Zhu, Jiantao Jiao, and David Tse. Deconstructing generative adversarial networks. *arXiv preprint arXiv:1901.09465*, 2019b.

Banghua Zhu, Jiantao Jiao, and Jacob Steinhardt. Robust estimation via generalized quasi-gradients. *Information and Inference: A Journal of the IMA*, 08 2021. ISSN 2049-8772. doi: 10.1093/imaiai/iaab018. URL https://doi.org/10.1093/imaiai/iaab018. iaab018.

Banghua Zhu, Jiantao Jiao, and Michael I Jordan. Robust estimation for nonparametric families via generative adversarial networks. *arXiv preprint arXiv:2202.01269*, 2022.

Zheng Zizhan, Shen Wen, and Henger Li. Learning to attack distributionally robust federated learning. 2020.

Supplementary Material for Byzantine-Robust Federated Learning with Optimal Statistical Rates

A   Related Work

We now review the client-side privacy vulnerabilities and Byzantine-malicious attacks in FL. Existing defenses and their limitations are also discussed to motivate the present work. Other attacks and defenses in FL are covered in this comprehensive surveys (Kairouz et al., 2019).

Byzantine Malicious Clients. Byzantine-robust aggregation has drawn enormous attention over the past few years due to the emergence of various distributed attacks in FL. Fang et al. (2020) formalize the attack as an optimization problem and successfully migrate the data poisoning attack to FL. The proposed attacks even work under Byzantine-robust FL. Sun et al. (2021) manage to launch data poisoning attacks on the multi-task FL framework. Bhagoji et al. (2019) and Bagdasaryan et al. (2020) manage to insert backdoors into the model via local model poisoning or local model replacement. Xie et al. (2019) propose to segment one backdoor into several parts and insert it into the global model. Chen et al. (2020) and Zizhan et al. (2020) migrate backdoor attacks to federated meta-learning and federated reinforcement learning, respectively. Meanwhile, Sun et al. (2019) show that norm clipping and "weak" differential privacy mitigate backdoor attacks in FL without impairing overall performance. Wang et al. (2020) refute this claim and illustrate that robustness to backdoors requires model robustness to adversarial examples, an open problem widely regarded to be difficult.

Byzantine-Robust Protocols. A variety of Byzantine-robust FL protocols are proposed to defend against malicious clients. Krum (Blanchard et al., 2017) selects and averages the subset of updates that have a sufficient number of close neighbors. Yin et al. (2018) uses robust estimators like trimmed mean or median to estimate the gradient, and claims to achieve order-optimal statistical error rates. Fung et al. (2018); Alistarh et al. (2018) proposes a similar robust estimator relying on a robust secure aggregation oracle based on the geometric median. Yin et al. (2019) proposes to use robust mean estimators to defend against saddle point attacks. Fung et al. (2020) studies Sybil attacks in FL and propose a defense based on the diversity of client updates. Ozdayi et al. (2020) designs a defense against backdoor attacks in FL by adjusting the server-side learning rate. El El Mhamdi et al. (2018) points out that Krum, trimmed mean, and median all suffer from $\mathcal{O}(\sqrt{d})$ (d is the model size) estimation error and propose a general framework Bulyan for reducing the error to $\mathcal{O}(1)$.

Client Privacy Leakage and Mitigation. Inference attacks against centralized learning (Shokri et al., 2017) aim to infer the private information of the model training data. (Wang et al., 2019) explore the feasibility of recovering user privacy from a malicious server in FL. (Nasr et al., 2019) show that a malicious server can perform highly accurate membership inference attacks against clients.

Orchestrating Robustness and Privacy. Several recent works (So et al., 2020; He et al., 2020) attempt to address the challenge through the use of secure multi-party computation (MPC). (So et al., 2020) propose to run multi-party distance-based filtering among clients to remove potentially malicious updates. This, however, needs clients to be online consistently, which is impractical for cross-device FL and also incurs significant communication cost. (He et al., 2020) propose a two-server protocol to provide bi-directional protection, but in real-world use scenarios, two non-colluding servers are uncommon. Due to the lack of universal and effective two-way protection, user trust in FL systems is significantly eroded, preventing them from being employed in a wide variety of security-related applications such as home monitoring and autonomous driving.

13

A.1 Comparison with existing Byzantine-Robust Protocols

Comparison with Yin et al. (2018). Yin et al. (2018) first analyzes the statistical rate for Byzantine-robust distributed learning when using coordinate-wise median or trimmed-mean as robust gradient estimators. They show that under the assumption of bounded variance $\mathbb{E}_{p^\star}[\|\mathbf{g} - \mathbb{E}[\mathbf{g}]\|_2^2] \leq \tilde{\sigma}^2$ and coordinate-wise bounded skewness (or coordinate-wise sub-exponential) on the gradient distribution, one can achieve a statistical error of $\mathcal{O}(\tilde{\sigma}(\epsilon/\sqrt{n} + \sqrt{d/mn}))$. Notably, under our assumption of bounded spectral norm of covariance in Assumption 3, one has $\tilde{\sigma}^2 = \sigma^2 d$ in the worst case, and their rate becomes $\mathcal{O}(\sigma(\epsilon\sqrt{d} + \sqrt{d^2/mn}))$, which is $\sqrt{d}$ worse than the lower bound. Although the rate achieved is better than ours in terms of the dependence on ϵ, this is from the strong assumption of coordinate-wise bounded skewness (or coordinate-wise sub-exponential). With only bounded covariance assumption as in our paper, coordinate-wise median or coordinate-wise trimmed mean must suffer a rate of $\sigma\sqrt{\epsilon d}$ (see e.g. (Steinhardt, 2019)), and thus is far from the optimal rate $\sigma\sqrt{\epsilon}$ achieved by the robust estimators proposed in this paper.

Comparison with Karimireddy et al. (2020) and Karimireddy et al. (2021)

Similar to Yin et al. (2018), the analysis of Karimireddy et al. (2020) and Karimireddy et al. (2021) are based on the assumption of bounded trace of the covariance, leading to an extra $\sqrt{d}$ factor under our spectral norm bound assumption. Karimireddy et al. (2020) has shown that the dimension dependence of coordinate-wise median combined with bucketing is worse than that of Krum and geometric median. However, as we have discussed in the introduction, coordinate-wise median, coordinate-wise trimmed mean and geometric median all have an error of at least $\Omega(\sigma\sqrt{\epsilon d})$ under our setting. This motivates the application of modern high dimensional robust estimators, as we proposed in this paper.

On the other hand, the robust aggregation algorithm with momentum is shown to achieve a rate $\mathcal{O}(\sqrt{\frac{d\sigma^2}{T} \cdot (\frac{1}{m} + \epsilon)})$, which seems to contradict our lower bound in Appendix C since the lower bound suggests that the rate does not vanish as T goes to infinity. The reason for this inconsistency is that Karimireddy et al. (2021) assumes that the gradient in each round is mutually independent and the malicious clients are fixed. However, this assumption does not hold in our setting when each client has n fixed samples and computes gradients out of (a mini-batch of) the samples. Intuitively, the assumption of independent gradient holds when each client gets fresh sample in a new iteration, thus can be approximately viewed as our parameter n. Furthermore, the assumption of independent gradient avoids the union bound argument in our analysis and Yin et al. (2018), thus an extra $\sqrt{d}$ factor is avoided in the rate of Karimireddy et al. (2021).

Furthermore, Karimireddy et al. (2020) also proposes a similar idea of bucketing as pre-processing from a different motivation. In our analysis, bucketing is necessary to remove the logarithmic factor and achieve exactly tight rate in homogeneous setting. In Karimireddy et al. (2020), it is mainly motivated by reducing the variance in heterogeneous data.

Comparison with ByzantineSGD (Alistarh et al., 2018) and SafeguardSGD (Allen-Zhu et al., 2020)

ByzantineSGD (Alistarh et al., 2018) and SafeguardSGD (Allen-Zhu et al., 2020) analyze the rate of convergence for non-strongly convex and non-convex objectives via maintaining a set of good machines and detecting malicious machines in each round. However, their results are based on a stronger assumption that the norm of the difference between the gradient and its mean $\|\mathbf{g} - \mathbb{E}[\mathbf{g}]\|$ is always bounded. Thus the result is not comparable to our setting, which only assumes the second central moment is bounded.

Comparison with Draco (Chen et al., 2018), BULYAN (El El Mhamdi et al., 2018) and Krum (Blanchard et al., 2017)

Draco (Chen et al., 2018) and BULYAN (El El Mhamdi et al., 2018) state to provide dimension-free estimation error for Byzantine-Robust Federated Learning. However, Draco is incompatible with FL as it requires redundant updates from each worker. Bulyan are based on much stronger assumptions than other contemporary works. When the assumptions are relaxed to the common case, Bulyan estimation errors still scale up with the square root of the model size. In particular, Bulyan assumes that the expectation of the distance between two benign updates is bounded by a constant σ_1, while Krum assumes that the distance is bounded by $\sigma_2\sqrt{d}$. We can easily see that if $\sigma_1 = \sigma_2\sqrt{d}$, Bulyan falls back to the same order of estimation error as Krum. As a result, the protocols fail to achieve near-optimal statistical rate under the only assumption of bounded operator norm of covariance that can be used to mitigate Byzantine adversaries.

The proposed protocols overcomes limitations of existing Byzantine-robust FL protocols by employing and calibrating well-established robust mean estimators (Diakonikolas et al., 2016, 2017, 2019b, 2020; Steinhardt et al., 2017, 2018; Zhu et al., 2019b,a, 2022) in FL scenarios.

Comparison with Velicheti et al. (2021)

The protocol in Velicheti et al. (2021) studies the combination of secure aggregation and Byzantine-robust estimators, similar to our bucketing idea. However, for the theoretical analysis, the authors make the assumption of bounded dissimilarity of the true gradient and bounded trace norm of the covariance of the gradients, which is the same as that in (Yin et al., 2018) and thus stronger than our spectral norm bound. The paper also assumes a performance bound on the performance of aggregation algorithm, which does not have dependence on the sample size. Thus there is no sample complexity analyzed, but only the convergence rate.

B Robust Estimation Subroutines

We begin with introducing necessary notations and settings in robust mean estimation. Assume that we observe m samples from true distribution $\mathbf{x}_i \sim p^\star, i \in [m]$. Denote the true empirical distribution as $\hat{p}_m^\star = \frac{1}{m} \sum_{i \in [m]} \delta_{\mathbf{x}_i}$. The adversary observes the samples and may add, delete, or modify at most ϵm samples, resulting in a corrupted dataset $\mathcal{D}_m$. We let the resulting empirical distribution be $\hat{p}_m$, which satisfies that $\mathsf{TV}(\hat{p}_m^\star, \hat{p}_m) \leq \epsilon$. Assume that the true distribution satisfies $\|\mathsf{Cov}_{p^\star}(\mathbf{x})\|_2 \leq \sigma^2$. Let $\mu_q = \mathbb{E}_q[\mathbf{x}]$ be the mean of distribution q. We define the quasi-gradient for $\|\mathsf{Cov}_q(\mathbf{x})\|$ with respect to distribution q as $g(q; \mathbf{x}) = (\mathbf{v}^\top(\mathbf{x} - \mu_q))^2$, where $\mathbf{v} \in \arg\max_{\|v\| \leq 1} \mathbb{E}_q[(\mathbf{v}^\top(\mathbf{x} - \mu_q))^2]$ is any of the supremum-achieving direction.

B.1 Algorithm Descriptions

Now we are ready to present the three algorithms we used in the paper, namely NO-REGRET ALGORITHM (Hopkins et al., 2020; Zhu et al., 2021), FILTERING ALGORITHM (Diakonikolas et al., 2017; Li, 2018; Steinhardt, 2018; Zhu et al., 2021) and GAN based algorithm (Gao et al., 2018, 2020; Zhu et al., 2022).

Algorithm 2 NO-REGRET ALGORITHM $(\mathcal{D}_m, \epsilon, \sigma^2, \xi, \eta)$

Input: corrupted dataset $\mathcal{D}_m = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_m\}$, fraction of corrupted samples ϵ, spectral norm bound of covariance σ^2, threshold for termination ξ, step size constant $\eta \in (0, 1)$.

Set $d_{i,j} = \|\mathbf{x}_i - \mathbf{x}_j\|_2$ for all $i, j \in [m]$, remove all $\mathbf{x}_i$ with $|\{j \in [m] : d_{i,j} > \Omega(\sqrt{d \log(m)})\}| > 2\epsilon m$. Let $\mathbf{x}_1', \mathbf{x}_2', \cdots, \mathbf{x}_{m'}'$ be the remaining samples. Initialize uniform distribution on the remaining samples $q_i^{(0)} = 1/m'$, $i \in [m']$.

for $k = 0, 1, \ldots$ **do**
    if $\|\mathsf{Cov}_{q^{(k)}}(\mathbf{x})\|_2 \leq \xi$ **then**
        Return $\mathbb{E}_{q^{(k)}}[\mathbf{x}] = \sum_{i=1}^n q_i^{(k)} \mathbf{x}_i$
    else
        Compute $g_i^{(k)} = g(q^{(k)}; \mathbf{x}_i)$, $\tilde{q}_i^{(k+1)} = q_i^{(k)} \cdot (1 - \frac{\eta\epsilon}{2\sigma^2 d} \cdot g_i^{(k)})$ for all $i \in [m]$
        Update $q^{(k+1)} = \arg\min_{q \in \Delta_{m',\epsilon}} D_{\mathrm{KL}}(q||\tilde{q}^{(k+1)})$, where $\Delta_{m',\epsilon} = \{q \mid \sum q_i = 1, q_i \leq \frac{1}{(1-\epsilon)m'}\}$.
    end if
end for

We remark here that in Algorithm 2, one needs to first naively filter some samples so that the maximum distances between samples are bounded, and then proceed with the multiplicative weights update method where learning rate is inverse proportional to the square of maximum distances between samples. This naive filtering procedure can be replaced with any heuristic-based methods, as long as the maximum distances between samples are controlled. The projection step in Algorithm 2 can be done within $O(m)$ time. For Algorithm 4, the distance $A(p, q)$ can be optimized by a Generative Adversarial Network. One can replace the discriminator with any deep neural network with sigmoid activation. In order to guarantee that $q \in \mathcal{G} = \{q \mid \|\Sigma_q\|_2 \leq \sigma^2\}$, one may either add a regularization term in the discriminator, or constrain the norm of parameters in the generator.

Algorithm 3 FILTERING ALGORITHM ($\mathcal{D}_m, \xi$)

Input: corrupted dataset $\mathcal{D}_m = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_m\}$, threshold for termination ξ.
Initialize $q_i^{(0)} = 1/m$, $i \in [m]$.
for $k = 0, 1, \ldots$ **do**
   if $\|\mathsf{Cov}_{q^{(k)}}(\mathbf{x})\|_2 \leq \xi$ **then**
      Return $\mathbb{E}_{q^{(k)}}[\mathbf{x}] = \sum_{i=1}^n q_i^{(k)} \mathbf{x}_i$
   else
      Compute $g_i^{(k)} = g(q^{(k)}; \mathbf{x}_i)$, $\tilde{q}_i^{(k+1)} = q_i^{(k)} \cdot (1 - \frac{g_i^{(k)}}{\max_{j \in [m]} g_j^{(k)}})$ for all $i \in [m]$.
      Update $q^{(k+1)} = \mathsf{Proj}_{\Delta_m}^{KL}(\tilde{q}^{(k+1)}) = \tilde{q}^{(k+1)} / \sum_{i \in [m]} \tilde{q}_i^{(k+1)}$.
      Remove samples with $q^{(k+1)} = 0$.
   end if
end for

Algorithm 4 GAN ($\mathcal{D}_m, \sigma^2$)

Input: corrupted dataset $\mathcal{D}_m = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_m\}$, spectral norm bound of covariance σ^2.
Let $\hat{p}_m = \frac{1}{m} \sum_{i=1}^m \delta_{\mathbf{x}_i}$ be the empirical distribution of dataset $\mathcal{D}_m$.
Find $q = \arg\min_{q \in \mathcal{G}(\sigma^2)} A(q, p_m)$, where $\mathcal{G}(\sigma^2) = \{q \mid \|\Sigma_q\|_2 \leq \sigma^2\}$, and

$$A(p, q) = \sup_{\|w\|_1 \leq 1, \|v_j\|_2 \leq 1, t_j \in \mathbb{R}} \left| \mathbb{E}_p \left[\mathsf{sigmoid}\left(\sum_{j \leq l} \mathbf{w}_j \mathsf{sigmoid}(\mathbf{v}_j^\top \mathbf{x} + t) \right) \right] \right.$$
$$\left. - \mathbb{E}_q \left[\mathsf{sigmoid}\left(\sum_{j \leq l} \mathbf{w}_j \mathsf{sigmoid}(\mathbf{v}_j^\top \mathbf{x} + t) \right) \right] \right| \tag{14}$$

B.2 Guarantees

For all the three robust estimators provided in the paper (Algorithm 2, 3 and 4), the resulting guarantee is that one can find an estimator with estimation error controlled by $O(\sqrt{\epsilon} + \sqrt{d/m})$ (or $O(\sqrt{\epsilon + \sqrt{d/m}})$ for Algorithm 4).

Theorem B.1. *Assume that $\|\mathsf{Cov}_{p^\star}(\mathbf{x})\| \leq \sigma^2$ and that we are given a corrupted dataset $\mathcal{D}_m$ from where at most ϵ fraction of samples are corrupted by adversary, the rest of samples follow the distribution of $p^\star$. Let*

$$\mu_1 = \text{NO-REGRET}\left(\mathcal{D}_m, \epsilon, \sigma^2, \left(\frac{2\eta + 7}{3(1 - (6 + 2\eta)\epsilon)} \right)^2 \cdot (1 + \frac{d\log(d/\delta)}{m\epsilon}) \cdot \sigma^2, \eta \right) \tag{15}$$

$$\mu_2 = \text{FILTERING}\left(\mathcal{D}_m, \frac{2(1-\epsilon)}{(1-2\epsilon)^2} \cdot (1 + \frac{d\log(d/\delta)}{m\epsilon}) \cdot \sigma^2 \right) \tag{16}$$

$$\mu_3 = \text{GAN}\left(\mathcal{D}_m, \sigma^2 \right) \tag{17}$$

$$\mu_4 = \text{NO-REGRET}\left(\mathcal{D}_m, \sigma^2, \eta, \frac{C_1}{(1 - C_2(\epsilon + \log(1/\delta)/n)^2)} \cdot \left(1 + \frac{d\log(d) + \log(1/\delta)}{m\epsilon} \right) \cdot \sigma^2 \right) \tag{18}$$

$$\mu_5 = \text{FILTERING}\left(\mathcal{D}_m, \frac{C_3}{(1 - C_4(\epsilon + \log(1/\delta)/n)^2)} \cdot \left(1 + \frac{d\log(d) + \log(1/\delta)}{m\epsilon} \right) \cdot \sigma^2 \right) \tag{19}$$

We know that each of the following holds with probability at least $1 - \delta$:

$$\|\mu_1 - \mu_{p^\star}\| \leq \mathcal{O}\left(\sigma \cdot \left(\frac{\sqrt{\epsilon}}{1 - (6 + 2\eta)\epsilon} + \sqrt{\frac{d\log(d/\delta)}{m}}\right)\right),$$

$$\|\mu_2 - \mu_{p^\star}\| \leq \mathcal{O}\left(\sigma \cdot \left(\frac{\sqrt{\epsilon}}{1 - 2\epsilon} + \sqrt{\frac{d\log(d/\delta)}{m}}\right)\right),$$

$$\|\mu_3 - \mu_{p^\star}\| \leq \mathcal{O}\left(\frac{\sigma}{1 - C_5\epsilon} \cdot \left(\sqrt{\epsilon + \sqrt{(d + \log(1/\delta))/m}}\right)\right),$$

$$\|\mu_4 - \mu_{p^\star}\| \leq \mathcal{O}\left(\frac{\sigma}{1 - C_6(\epsilon + \log(1/\delta)/m)} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d\log(d) + \log(1/\delta)}{m}}\right)\right),$$

$$\|\mu_5 - \mu_{p^\star}\| \leq \mathcal{O}\left(\frac{\sigma}{1 - C_7(\epsilon + \log(1/\delta)/m)} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d\log(d) + \log(1/\delta)}{m}}\right)\right),$$

where $C_i > 2$ are some universal constants.

Proof. The guarantee for μ_1 comes from a combination of the statistical analysis (see e.g. Zhu et al. (2019a, Theorem G.3), Diakonikolas et al. (2017), (Steinhardt et al., 2017)) and computational analysis (see e.g. Zhu et al. (2021, Theorem 4.1)). The guarantee for μ_1 comes from a combination of the same statistical analysis and a different computational analysis (see Zhu et al. (2021, Theorem 4.2)). The guarantee for μ_3 comes from Zhu et al. (2022, Corollary 1). The guarantee for μ_4 and μ_5 comes from Diakonikolas et al. (2020, Proposition 1.6). □

As a sharp contrast, COORDINATE-WISE MEDIAN, COORDINATE-WISE TRIMMED MEAN and GEOMETRIC MEDIAN have an error of at least $\Omega(\sigma\sqrt{\epsilon d})$ under the same assumption, which can be highly sub-optimal when the dimension is high. Following our lower bound analysis in the next section, the lower bound for robust mean estimation also directly results in a lower bound of $\Omega(\sigma\sqrt{\epsilon d/n})$ for the performance of the FL protocols using either of the three algorithms as aggregation rules.

C Lower Bound

Now we present a formal lower bound for the Federated Learning setting using the lower bound for robust mean estimation. Let $\mathcal{F}$ be the sets of loss functions ℓ and distribution $p^\star$ that satisfies Assumption 2, 3 and 4(a). Let $P : \mathcal{W} \times \mathbb{R}^{m \times d} \mapsto \mathcal{W}$ be any FL protocol which receive the current parameter $\mathbf{w} \in \mathcal{W}$ and corresponding gradient information for m clients under the parameter $\mathbf{w}$, and outputs an updated parameter $\mathbf{w}' \in \mathcal{W}$ according to the observation. We denote the set of such protocols as $\mathcal{P}$. For any $(f, p^\star) \in \mathcal{F}$, we let $\mathbf{w}^\star(f, p^\star) = \arg\min_{\mathbf{w} \in \mathcal{W}} \mathbb{E}_{\mathbf{z} \sim p^\star}[f(\mathbf{w}; \mathbf{z})]$ be the minimizer of the loss function. Let $\mathcal{D}_{m,n,T}(P)$ be the set of all possible distributions of observations we may see when executing $P \in \mathcal{P}$ after T rounds. At round t, conditioned on the current parameter $\mathbf{w}^t$, the distribution of observed distribution is within ϵ-TV distance to the true distribution of $\nabla F_i(\mathbf{w})$ due to possible corruptions by Byzantine workers. Let $\hat{\mathbf{w}}^T : \mathcal{P} \times \mathcal{D}_{m,n,T}(\cdot) \mapsto \mathcal{W}$ be the output of the protocol $P \in \mathcal{P}$ given all the observations from $\mathcal{D}_{m,n,T}(P)$.

Theorem C.1 (Lower Bound). *For any protocol with infinite computation power, there exists some loss function f and sample distribution $p^\star$ that satisfies Assumption 2, 3 and 4(a), such that the output of the protocol $\hat{\mathbf{w}}$ incurs a loss that is at least $\Omega(\sqrt{\frac{\epsilon}{n} + \frac{d}{mn}})$. Formally, we have for any $T > 0$, with constant probability,*

$$\inf_{P \in \mathcal{P}} \sup_{(f, p^\star) \in \mathcal{F}, \hat{p} \in \mathcal{D}_{m,n,T}(P)} \|\mathbf{w}^\star(f, p^\star) - \hat{\mathbf{w}}^T(P, \hat{p})\|_2 = \Omega\left(\sigma\sqrt{\frac{\epsilon}{n} + \frac{d}{mn}}\right).$$

Proof. Consider the set of problems of mean estimation under ℓ_2 norm, where we fix $f(\mathbf{w}; \mathbf{z}) = \|\mathbf{w} - \mathbf{z}\|_2^2$, and let $p^\star$ to be any distribution with bounded covariance, i.e. we take $\mathcal{F}_{\sigma^2} = \{(f, p^\star) \mid f(\mathbf{w}; \mathbf{z}) = \|\mathbf{w} - \mathbf{z}\|_2^2, \|\mathsf{Cov}_{p^\star}(\mathbf{z})\|_2 \leq \sigma^2\}$. In this case, we know that $\mathbf{w}^\star(f, p^\star) = \mathbb{E}_{p^\star}[\mathbf{z}]$, and the gradient from each client is $\nabla F_i(\mathbf{w}) = \frac{2\sum_{i=1}^n(\mathbf{w} - \mathbf{z}_i)}{n} = 2(\mathbf{w} - \frac{\sum_{j=1}^n \mathbf{z}_{ij}}{n})$. Consider another protocol $\mathcal{P}'$ where the server can directly query the mean $\frac{\sum_{j=1}^n \mathbf{z}_{ij}}{n}$ from each honest

client. One can see that for the fixed squared loss f, any protocol in $\mathcal{P}$ can be converted into a protocol in $\mathcal{P}'$ since the mean $\frac{\sum_{j=1}^{n} \mathbf{z}_{ij}}{n}$ can be directly computed from the gradient $\nabla F_i(\mathbf{w})$ on any given $\mathbf{w}$ (and thus any multiple-round local model updates in FL can be also directly computed from the mean). Similarly, we can also convert any protocol in $\mathcal{P}'$ to a protocol in $\mathcal{P}$. Thus the two protocols are equivalent, and we have

$$\inf_{\substack{P \in \mathcal{P}}} \sup_{\substack{(f,p^\star) \in \mathcal{F}_{\sigma^2}, \\ \hat{p} \in \mathcal{D}_{m,n,T}(P)}} \|\mathbf{w}^\star(f,p^\star) - \hat{\mathbf{w}}^T(P,\hat{p})\|_2 = \inf_{\substack{P \in \mathcal{P}'}} \sup_{\substack{(f,p^\star) \in \mathcal{F}_{\sigma^2}, \\ \hat{p} \in \mathcal{D}_{m,n,T}(P)}} \|\mathbf{w}^\star(f,p^\star) - \hat{\mathbf{w}}^T(P,\hat{p})\|_2.$$

In this case, we can see that for honest clients, the output in each round will always be the mean $\frac{\sum_{j=1}^{n} \mathbf{z}_{ij}}{n}$. We further restrict the behavior of adversary by considering a fixed set of adversary, and force the adversary to output the same mean in each round. Thus we know that this minimax rate can be further lower bounded by that of single-round robust mean estimation problem where the queried sample from honest client is an average of n samples from the true distribution $p^\star$. Formally, let $p_n^\star$ be the distribution of $\frac{\sum_{j=1}^{n} \mathbf{z}_{ij}}{n}$ where each $\mathbf{z}_{ij} \sim p^\star$, we have

$$\inf_{\substack{P \in \mathcal{P}}} \sup_{\substack{(f,p^\star) \in \mathcal{F}_{\sigma^2}, \\ \hat{p} \in \mathcal{D}_{m,n,T}(P)}} \|\mathbf{w}^\star(f,p^\star) - \hat{\mathbf{w}}^T(P,\hat{p})\|_2 \geq \inf_{\substack{P \in \mathcal{P}}} \sup_{\substack{(p^\star,\hat{p}):\|\mathrm{Cov}_{p^\star}(\mathbf{z})\| \leq \sigma^2, \mathrm{TV}(p_n^\star,\hat{p}) \leq \epsilon}} \|\mathbb{E}_{p^\star}[\mathbf{x}] - \hat{\mathbf{w}}^T(P,\hat{p})\|_2.$$

From the argument of modulus of continuity (see e.g. Theorem 3.2 in (Chen et al., 2016), Lemma D.4 in (Zhu et al., 2019a)), we know that with constant probability,

$$\inf_{\substack{P \in \mathcal{P}}} \sup_{\substack{(p^\star,\hat{p}):\|\mathrm{Cov}_{p^\star}(\mathbf{z})\| \leq \sigma^2, \\ \mathrm{TV}(p_n^\star,\hat{p}) \leq \epsilon}} \|\mathbb{E}_{p^\star}[\mathbf{x}] - \hat{\mathbf{w}}^T(P,\hat{p})\|_2 \geq \sup_{\substack{(p^\star,p^{\star\prime}):\|\mathrm{Cov}_{p^\star}(\mathbf{z})\| \leq \sigma^2, \\ \|\mathrm{Cov}_{p^{\star\prime}}(\mathbf{z})\| \leq \sigma^2, \mathrm{TV}(p_n^\star,p_n^{\star\prime}) \leq \epsilon}} \|\mathbb{E}_{p^\star}[\mathbf{x}] - \mathbb{E}_{p^{\star\prime}}[\mathbf{x}]\|_2 + \sigma\sqrt{\frac{d}{mn}}.$$

We further lower bound the first term. Let $\epsilon' = 1 - (1-\epsilon)^{1/n}$. We construct two one-dimensional distributions $p^\star$, $p^{\star\prime}$ as follows:

$$\mathbb{P}_{p^\star}(x) = \begin{cases} \epsilon', & x = \frac{\sigma}{3}\sqrt{\frac{1-\epsilon'}{\epsilon'}}, \\ 1 - \epsilon', & x = 0, \end{cases}$$
$$\mathbb{P}_{p^{\star\prime}}(0) = 1.$$

One can verify that both $p^\star$ and $p^{\star\prime}$ has spectral norm of its covariance bounded by σ^2, and furthermore, $\mathrm{TV}(p_n^\star, p_n^{\star\prime}) = (1-\epsilon')^n = \epsilon$. In the meantime, one has

$$\|\mathbb{E}_{p^\star}[\mathbf{x}] - \mathbb{E}_{p^{\star\prime}}[\mathbf{x}]\|_2 \gtrsim \sigma\sqrt{\epsilon'} \gtrsim \sigma\sqrt{\frac{\epsilon}{n}},$$

where the last inequality uses the fact that $1 - (1-\epsilon)^{1/n} \gtrsim \epsilon/n$. $\qquad\square$

D Proof of Theorem 3.1

Since $H = 1$, we have $\mathbf{w}^t = \mathbf{w}_i^t$ for all $i \in [m]$. Given that each machine has n samples, the variance of the returned local update $\mathbf{g}^i(\mathbf{w}^t) = -\eta_t \nabla F_i(\mathbf{w}^t)$ satisfies

$$\|\mathrm{Cov}_{p^\star}(\mathbf{g}^i(\mathbf{w}^t))\| \leq \frac{\eta_t^2 \sigma^2}{n}.$$

Let $\mu_{p^\star}(\mathbf{w}), \mu_i(\mathbf{w})$ denote the true mean of local updates under parameter $\mathbf{w}$ and the output of the i-th algorithm in Equation (1), (2), (3), (4), (5), respectively. We first show that the three robust estimators are Lipschitz with respect to $\mathbf{w}$ in the following lemma.

18

Lemma D.1. *For any* $\mathbf{w}_1, \mathbf{w}_2 \in \mathcal{W}$ *with* $\|\mathbf{w}_1 - \mathbf{w}_2\|_2 \leq \gamma$, *one has for any* $i \in \{1, 2, 3, 4, 5\}$, *there exists some universal constants* C_j *such that*

$$\|\mu_1(\mathbf{w}_1) - \mu_1(\mathbf{w}_2)\|_2 \leq \frac{C_1 \eta_t \sqrt{\epsilon}}{1 - 3\epsilon}\left(\frac{\sigma}{\sqrt{n}} + L\gamma\right) + \frac{C_1 \eta_t \sigma}{1 - 3\epsilon} \cdot \sqrt{\frac{d \log(d/\delta)}{mn}} + \eta_t L\gamma,$$

$$\|\mu_2(\mathbf{w}_1) - \mu_2(\mathbf{w}_2)\|_2 \leq \frac{C_2 \eta_t \sqrt{\epsilon}}{1 - 2\epsilon}\left(\frac{\sigma}{\sqrt{n}} + L\gamma\right) + \frac{C_2 \eta_t \sigma}{1 - 2\epsilon} \cdot \sqrt{\frac{d \log(d/\delta)}{mn}} + \eta_t L\gamma,$$

$$\|\mu_3(\mathbf{w}_1) - \mu_3(\mathbf{w}_2)\|_2 \leq \frac{C_3 \eta_t \sqrt{\epsilon + \sqrt{d + \log(1/\delta)/n}}}{1 - C_4(\epsilon + \sqrt{d + \log(1/\delta)/n})}\left(\frac{\sigma}{\sqrt{n}} + L\gamma\right) + \eta_t L\gamma.$$

$$\|\mu_4(\mathbf{w}_1) - \mu_4(\mathbf{w}_2)\|_2 \leq \frac{C_5 \eta_t \sqrt{\epsilon}}{1 - C_6\epsilon}\left(\frac{\sigma}{\sqrt{n}} + L\gamma\right) + \frac{C_5 \eta_t \sigma}{1 - C_6\epsilon} \cdot \sqrt{\frac{d \log(d) + \log(1/\delta)}{mn}} + \eta_t L\gamma,$$

$$\|\mu_5(\mathbf{w}_1) - \mu_5(\mathbf{w}_2)\|_2 \leq \frac{C_7 \eta_t \sqrt{\epsilon}}{1 - C_8\epsilon}\left(\frac{\sigma}{\sqrt{n}} + L\gamma\right) + \frac{C_7 \eta_t \sigma}{1 - C_8\epsilon} \cdot \sqrt{\frac{d \log(d) + \log(1/\delta)}{mn}} + \eta_t L\gamma.$$

Proof. We first prove the result for Algorithm 2 in Equation (1). From the smoothness assumption in Assumption 2, we know that $\|\mathbf{w}_1 - \mathbf{w}_2\|_2 \leq \gamma$ implies that for any $\mathbf{z}$,

$$\|\nabla f(\mathbf{w}_1; \mathbf{z}) - \nabla f(\mathbf{w}_2; \mathbf{z})\|_2 \leq L\gamma. \tag{20}$$

Assume without loss of generality that the first $(1 - \epsilon)m$ clients are honest while the rest ϵm clients are adversarial. Denote the local gradient $-\eta_t \nabla f(\mathbf{w}_1; \mathbf{z})$ for i-th client as $\mathbf{g}_i$, and $-\eta_t \nabla f(\mathbf{w}_2; \mathbf{z})$ for i-th client as $\mathbf{g}_i'$. Since $\mu_1(\mathbf{w}_1), \mu_1(\mathbf{w}_2)$ are the output of the algorithm in Equation (1), we know that for any $\|\mathbf{g}_1 - \mathbf{g}_1'\| \leq \eta_t L\gamma, \|\mathbf{g}_2 - \mathbf{g}_2'\| \leq \eta_t L\gamma, \cdots, \|\mathbf{g}_{(1-\epsilon)m} - \mathbf{g}_{(1-\epsilon)m}'\| \leq \eta_t L\gamma, \mathbf{g}_{(1-\epsilon)m+1} = \mathbf{g}_{(1-\epsilon)m+1}', \cdots, \mathbf{g}_m = \mathbf{g}_m'$, one can find some distribution $q_1, q_1' \in \Delta_{m, 2\epsilon}$ such that $\mu_1(\mathbf{w}_1) = \mathbb{E}_{q_1}[\mathbf{g}], \mu_1(\mathbf{w}_2) = \mathbb{E}_{q_1'}[\mathbf{g}']$, and for some universal constant C,

$$\|\mathsf{Cov}_{q_1}(\mathbf{g})\|_2 = \sup_{\|v\|_2 \leq 1} \mathbb{E}_{q_1}[(v^\top(\mathbf{g} - \mu_1(\mathbf{w}_1)))^2] \leq \left(\frac{C}{1 - 3\epsilon}\right)^2 \cdot \left(1 + \frac{d \log(d/\delta)}{m\epsilon}\right) \cdot \frac{\eta_t^2 \sigma^2}{n},$$

$$\|\mathsf{Cov}_{q_1'}(\mathbf{g}')\|_2 = \sup_{\|v\|_2 \leq 1} \mathbb{E}_{q_1'}[(v^\top(\mathbf{g}' - \mu_2(\mathbf{w}_2)))^2] \leq \left(\frac{C}{1 - 3\epsilon}\right)^2 \cdot \left(1 + \frac{d \log(d/\delta)}{m\epsilon}\right) \cdot \frac{\eta_t^2 \sigma^2}{n}.$$

This implies that

$$\begin{aligned}
\|\mathsf{Cov}_{q_1'}(\mathbf{g})\|_2 &= \sup_{\|v\|_2 \leq 1} \mathbb{E}_{q_1'}[(v^\top(\mathbf{g} - \mathbb{E}_{q_1'}[\mathbf{g}]))^2] \\
&\leq \sup_{\|v\|_2 \leq 1} \mathbb{E}_{q_1'}[(v^\top(\mathbf{g}' - \mathbb{E}_{q_1'}[\mathbf{g}']))^2] + 2\eta_t^2 L^2 \gamma^2 \\
&\leq \left(\frac{C}{1 - 3\epsilon}\right)^2 \cdot \left(1 + \frac{d \log(d/\delta)}{m\epsilon}\right) \cdot \frac{\eta_t^2 \sigma^2}{n} + 2\eta_t^2 L^2 \gamma^2.
\end{aligned}$$

Since $\mathsf{TV}(q_1, q_1') \leq \frac{2\epsilon}{1 - 3\epsilon}$ and $\epsilon < 1/3$, from e.g. Steinhardt (2018, Proposition 2.3) and Zhu et al. (2019a, Lemma E.2) we know that for some different universal constant C,

$$\begin{aligned}
\|\mathbb{E}_{q_1}[\mathbf{g}] - \mathbb{E}_{q_1'}[\mathbf{g}]\|_2 &\leq \frac{C \eta_t \sqrt{\epsilon}}{1 - 3\epsilon}\left(\sqrt{\frac{1}{n} + \frac{d \log(d/\delta)}{mn\epsilon}}\sigma + L\gamma\right) \\
&= \frac{C \eta_t \sqrt{\epsilon}}{1 - 3\epsilon}\left(\frac{\sigma}{\sqrt{n}} + L\gamma\right) + \frac{C \eta_t \sigma}{1 - 3\epsilon} \cdot \sqrt{\frac{d \log(d/\delta)}{mn}}.
\end{aligned}$$

And thus

$$\begin{aligned}
\|\mu_1(\mathbf{w}_1) - \mu_1(\mathbf{w}_2)\|_2 &= \|\mathbb{E}_{q_1}[\mathbf{g}] - \mathbb{E}_{q_1'}[\mathbf{g}']\|_2 \\
&\leq \|\mathbb{E}_{q_1}[\mathbf{g}] - \mathbb{E}_{q_1'}[\mathbf{g}]\|_2 + \|\mathbb{E}_{q_1'}[\mathbf{g}] - \mathbb{E}_{q_1'}[\mathbf{g}']\|_2 \\
&= \frac{C \eta_t \sqrt{\epsilon}}{1 - 3\epsilon}\left(\frac{\sigma}{\sqrt{n}} + L\gamma\right) + \frac{C \eta_t \sigma}{1 - 3\epsilon} \cdot \sqrt{\frac{d \log(d/\delta)}{mn}} + \eta_t L\gamma.
\end{aligned}$$

The results for other algorithms follows the same line of argument. $\qquad\square$

Consider a γ-covering in ℓ_2 distance for the set $\mathcal{W}$, denoted as $\mathcal{W}_\gamma$. From (Vershynin, 2010), we know that $|\mathcal{W}_\gamma| \leq (1 + \frac{D}{\gamma})^d$. By union bound and Theorem B.1, we know that each of the following holds with probability at least $1 - (1 + D/\gamma)^d\delta$,

$$\sup_{\mathbf{w}_\gamma \in \mathcal{W}_\gamma} \| - \mu_1(\mathbf{w}_\gamma) - \eta_t \nabla F(\mathbf{w}_\gamma)\|_2 \leq \mathcal{O}\left(\frac{\eta_t \sigma}{\sqrt{n}} \cdot \left(\frac{\sqrt{\epsilon}}{1 - 3\epsilon} + \sqrt{\frac{d\log(d/\delta)}{m}} \right) \right),$$

$$\sup_{\mathbf{w}_\gamma \in \mathcal{W}_\gamma} \| - \mu_2(\mathbf{w}_\gamma) - \eta_t \nabla F(\mathbf{w}_\gamma)\|_2 \leq \mathcal{O}\left(\frac{\eta_t \sigma}{\sqrt{n}} \cdot \left(\frac{\sqrt{\epsilon}}{1 - 2\epsilon} + \sqrt{\frac{d\log(d/\delta)}{m}} \right) \right),$$

$$\sup_{\mathbf{w}_\gamma \in \mathcal{W}_\gamma} \| - \mu_3(\mathbf{w}_\gamma) - \eta_t \nabla F(\mathbf{w}_\gamma)\|_2 \leq \mathcal{O}\left(\frac{\eta_t \sigma}{\sqrt{n}} \cdot \left(\frac{\sqrt{\epsilon + \sqrt{(d + \log(1/\delta))/m}}}{1 - C\epsilon} \right) \right),$$

$$\sup_{\mathbf{w}_\gamma \in \mathcal{W}_\gamma} \| - \mu_4(\mathbf{w}_\gamma) - \eta_t \nabla F(\mathbf{w}_\gamma)\|_2 \leq \mathcal{O}\left(\frac{\eta_t \sigma}{1 - C(\epsilon + \log(1/\delta)/m)} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d\log(d) + \log(1/\delta)}{m}} \right) \right)$$

$$\sup_{\mathbf{w}_\gamma \in \mathcal{W}_\gamma} \| - \mu_5(\mathbf{w}_\gamma) - \eta_t \nabla F(\mathbf{w}_\gamma)\|_2 \leq \mathcal{O}\left(\frac{\eta_t \sigma}{1 - C(\epsilon + \log(1/\delta)/m)} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d\log(d) + \log(1/\delta)}{m}} \right) \right)$$

Since both $\mu_1(\mathbf{w}_\gamma)$ and $\nabla F(\mathbf{w}_\gamma)$ are Lipschitz (from Lemma D.1 and Assumption 2), we know that with probability at least $1 - \delta$,

$$\sup_{\mathbf{w} \in \mathcal{W}} \| - \mu_1(\mathbf{w}) - \eta_t \nabla F(\mathbf{w})\|_2 \leq \sup_{\mathbf{w}_\gamma \in \mathcal{W}_\gamma} \| - \mu_1(\mathbf{w}_\gamma) - \eta_t \nabla F(\mathbf{w}_\gamma)\|_2 + \eta_t L\gamma + \frac{C\eta_t \sqrt{\epsilon}}{1 - 3\epsilon}\left(\frac{\sigma}{\sqrt{n}} + L\gamma \right)$$

$$+ \frac{C\eta_t \sigma}{1 - 3\epsilon} \cdot \sqrt{\frac{d\log(d/\delta)}{mn}}$$

$$\leq \mathcal{O}\left(\frac{\eta_t \sigma}{(1 - 3\epsilon)\sqrt{n}} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d^2\log(1 + D/\gamma) + d\log(d/\delta)}{m}} \right) + \frac{\eta_t L\gamma}{1 - 3\epsilon} \right),$$

By taking $\gamma = 1/nmL$, we know that with probability at least $1 - \delta$,

$$\sup_{\mathbf{w} \in \mathcal{W}} \| - \mu_1(\mathbf{w}) - \eta_t \nabla F(\mathbf{w})\|_2 \leq \mathcal{O}\left(\frac{\eta_t \sigma}{(1 - 3\epsilon)\sqrt{n}} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d^2\log(1 + nmDL) + d\log(d/\delta)}{m}} \right) \right). \qquad (21)$$

Similarly, we can get for Algorithm 3, 4 that with probability at least $1 - \delta$,

$$\sup_{\mathbf{w} \in \mathcal{W}} \| - \mu_2(\mathbf{w}) - \eta_t \nabla F(\mathbf{w})\|_2 \leq \mathcal{O}\left(\frac{\eta_t \sigma}{(1 - 2\epsilon)\sqrt{n}} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d^2\log(1 + nmDL) + d\log(d/\delta)}{m}} \right) \right), \qquad (22)$$

$$\sup_{\mathbf{w} \in \mathcal{W}} \| - \mu_3(\mathbf{w}) - \eta_t \nabla F(\mathbf{w})\|_2 \leq \mathcal{O}\left(\frac{\eta_t \sigma}{(1 - C\epsilon)\sqrt{n}} \cdot \left(\sqrt{\epsilon + \sqrt{\frac{d\log(1 + nmDL) + \log(1/\delta)}{m}}} \right) \right) \qquad (23)$$

$$\sup_{\mathbf{w} \in \mathcal{W}} \| - \mu_4(\mathbf{w}) - \eta_t \nabla F(\mathbf{w})\|_2 \leq \mathcal{O}\left(\frac{\eta_t \sigma}{1 - C(\epsilon + \frac{\log(1/\delta)}{m})} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d(\log(d) + \log(1 + nmDL)) + \log(1/\delta)}{m}} \right) \right)$$
$$(24)$$

$$\sup_{\mathbf{w} \in \mathcal{W}} \| - \mu_5(\mathbf{w}) - \eta_t \nabla F(\mathbf{w})\|_2 \leq \mathcal{O}\left(\frac{\eta_t \sigma}{1 - C(\epsilon + \frac{\log(1/\delta)}{m})} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d(\log(d) + \log(1 + nmDL)) + \log(1/\delta)}{m}} \right) \right)$$
$$(25)$$

Then, we proceed to analyze the convergence of the robust distributed gradient descent algorithm. This part follows directly with the analysis in (Yin et al., 2018). For completeness we include the proof here.

Convergence analysis for strongly convex case. We condition on the event that the bound above is satisfied for all $\mathbf{w} \in \mathcal{W}$. Then, in the t-th iteration, let the updated parameter before projection step be $\widehat{\mathbf{w}}^{t+1} = \mathbf{w}^t + \mathbf{g}(\mathbf{w}^t)$. Thus, we have $\mathbf{w}^{t+1} = \Pi_{\mathcal{W}}(\widehat{\mathbf{w}}^{t+1})$. From the property of Euclidean projection, we know that

$$
\begin{aligned}
\|\mathbf{w}^{t+1} - \mathbf{w}^*\|_2 &\leq \|\widehat{\mathbf{w}}^{t+1} - \mathbf{w}^*\|_2 \\
&= \|\mathbf{w}^t + \mathbf{g}(\mathbf{w}^t) - \mathbf{w}^*\|_2 \\
&\leq \|\mathbf{w}^t - \eta_t \nabla F(\mathbf{w}^t) - \mathbf{w}^*\|_2 + \| - \mathbf{g}(\mathbf{w}^t) - \eta_t \nabla F(\mathbf{w}^t)\|_2.
\end{aligned}
\tag{26}
$$

For the first term, we have

$$
\|\mathbf{w}^t - \eta_t \nabla F(\mathbf{w}^t) - \mathbf{w}^*\|_2^2 = \|\mathbf{w}^t - \mathbf{w}^*\|_2^2 - 2\eta_t \langle \mathbf{w}^t - \mathbf{w}^*, \nabla F(\mathbf{w}^t) \rangle + \eta_t^2 \|\nabla F(\mathbf{w}^t)\|_2^2.
\tag{27}
$$

Since $F(\mathbf{w})$ is λ-strongly convex, by the co-coercivity of strongly convex functions (see e.g. Lemma 3.11 in (Bubeck et al., 2015)), we obtain

$$
\langle \mathbf{w}^t - \mathbf{w}^*, \nabla F(\mathbf{w}^t) \rangle \geq \frac{L\lambda}{L+\lambda} \|\mathbf{w}^t - \mathbf{w}^*\|_2^2 + \frac{1}{L+\lambda} \|\nabla F(\mathbf{w}^t)\|_2^2.
$$

Let $\eta_t = \frac{1}{L}$. Then we get

$$
\begin{aligned}
\|\mathbf{w}^t - \eta_t \nabla F(\mathbf{w}^t) - \mathbf{w}^*\|_2^2 &\leq (1 - \frac{2\lambda}{L+\lambda})\|\mathbf{w}^t - \mathbf{w}^*\|_2^2 - \frac{2}{L(L+\lambda)}\|\nabla F(\mathbf{w}^t)\|_2^2 + \frac{1}{L^2}\|\nabla F(\mathbf{w}^t)\|_2^2 \\
&\leq (1 - \frac{2\lambda}{L+\lambda})\|\mathbf{w}^t - \mathbf{w}^*\|_2^2,
\end{aligned}
$$

where in the second inequality we use the fact that $\lambda \leq L$. Using the fact $\sqrt{1-x} \leq 1 - \frac{x}{2}$, we get

$$
\|\mathbf{w}^t - \eta_t \nabla F(\mathbf{w}^t) - \mathbf{w}^*\|_2 \leq (1 - \frac{\lambda}{L+\lambda})\|\mathbf{w}^t - \mathbf{w}^*\|_2.
\tag{28}
$$

Combining Equation (26) and Equation (28), we get

$$
\|\mathbf{w}^{t+1} - \mathbf{w}^*\|_2 \leq (1 - \frac{\lambda}{L+\lambda})\|\mathbf{w}^t - \mathbf{w}^*\|_2 + \frac{1}{L}\Delta_i,
\tag{29}
$$

where Δ_i is defined as in Equation (6), (7), (8), (9), or (10), respectively. Then we can complete the proof by iterating Equation (29).

Convergence analysis for non-strongly Convex Losses: Let $\tilde{\mathbf{g}}(\mathbf{w}^t) = -\mathbf{g}(\mathbf{w}^t)/\eta_t$ be the virtual gradient at round t. We first show that when Assumption 4(b) is satisfied and we choose $\eta_t = \frac{1}{L}$, the iterates $\mathbf{w}^t$ stays in $\mathcal{W}$ without using projection. Namely, let $\mathbf{w}^{t+1} = \mathbf{w}^t - \eta_t \tilde{\mathbf{g}}(\mathbf{w}^t)$, for $T = 0, 1, \ldots, T-1$, then $\mathbf{w}^t \in \mathcal{W}$ for all $t = 0, 1, \ldots, T$. To see this, we have

$$
\|\mathbf{w}^{t+1} - \mathbf{w}^*\|_2 \leq \|\mathbf{w}^t - \eta_t \nabla F(\mathbf{w}^t) - \mathbf{w}^*\|_2 + \eta_t \|\tilde{\mathbf{g}}(\mathbf{w}^t) - \nabla F(\mathbf{w}^t)\|_2,
$$

and

$$
\begin{aligned}
\|\mathbf{w}^t - \eta_t \nabla F(\mathbf{w}^t) - \mathbf{w}^*\|_2^2 &= \|\mathbf{w}^t - \mathbf{w}^*\|_2^2 - 2\eta_t \langle \nabla F(\mathbf{w}^t), \mathbf{w}^t - \mathbf{w}^* \rangle + \eta_t^2 \|\nabla F(\mathbf{w}^t)\|_2^2 \\
&\leq \|\mathbf{w}^t - \mathbf{w}^*\|_2^2 - 2\eta_t \frac{1}{L}\|\nabla F(\mathbf{w}^t)\|_2^2 + \eta_t^2 \|\nabla F(\mathbf{w}^t)\|_2^2 \\
&= \|\mathbf{w}^t - \mathbf{w}^*\|_2^2 - \frac{1}{L^2}\|\nabla F(\mathbf{w}^t)\|_2^2 \\
&\leq \|\mathbf{w}^t - \mathbf{w}^*\|_2^2
\end{aligned}
$$

where the inequality is due to the co-coercivity of convex functions. Thus, we get

$$
\|\mathbf{w}^{t+1} - \mathbf{w}^*\|_2 \leq \|\mathbf{w}^t - \mathbf{w}^*\|_2 + \frac{\Delta_i}{L}.
$$

Let $D_t := \|\mathbf{w}^0 - \mathbf{w}^*\|_2 + \frac{t\gamma}{L}$ for $t = 0, 1, \ldots, T$. Since $T = \frac{LD_0}{\gamma}$, according to Assumption 4(b) we know that $\mathbf{w}^t \in \mathcal{W}$ for all $t = 0, 1, \ldots, T$. Then, we proceed to study the algorithm without projection.

Using the smoothness of $F(\mathbf{w})$, we have

$$F(\mathbf{w}^{t+1}) \leq F(\mathbf{w}^t) + \langle \nabla F(\mathbf{w}^t), \mathbf{w}^{t+1} - \mathbf{w}^t \rangle + \frac{L}{2}\|\mathbf{w}^{t+1} - \mathbf{w}^t\|_2^2$$

$$= F(\mathbf{w}^t) + \eta_t \langle \nabla F(\mathbf{w}^t), -\tilde{\mathbf{g}}(\mathbf{w}^t) + \nabla F(\mathbf{w}^t) - \nabla F(\mathbf{w}^t) \rangle + \eta_t^2 \frac{L}{2}\|\tilde{\mathbf{g}}(\mathbf{w}^t) - \nabla F(\mathbf{w}^t) + \nabla F(\mathbf{w}^t)\|_2^2.$$

Since $\eta_t = \frac{1}{L}$ and $\|\tilde{\mathbf{g}}(\mathbf{w}^t) - \nabla F(\mathbf{w}^t)\|_2 \leq \gamma$, by simple algebra, we obtain

$$F(\mathbf{w}^{t+1}) \leq F(\mathbf{w}^t) - \frac{1}{2L}\|\nabla F(\mathbf{w}^t)\|_2^2 + \frac{1}{2L}\gamma^2. \tag{30}$$

We state the following lemma from (Yin et al., 2018) without proof.

Lemma D.2. *Condition on the event that Equation (21), (22), (23), (24) or (25) holds for all $\mathbf{w} \in \mathcal{W}$. When $F(\mathbf{w})$ is convex, by running $T = \frac{LD_0}{\Delta}$ parallel iterations, there exists $t \in \{0, 1, 2, \ldots, T\}$ such that*

$$F(\mathbf{w}^t) - F(\mathbf{w}^*) \leq 16D_0\Delta_i.$$

for some $i \in \{1, 2, 3, 4, 5\}$.

Next, we show that $F(\mathbf{w}^T) - F(\mathbf{w}^*) \leq 16D_0\Delta_i + \frac{1}{2L}\Delta_i^2$. More specifically, let $t = t_0$ be the first time that $F(\mathbf{w}^t) - F(\mathbf{w}^*) \leq 16D_0\Delta_i$, and we show that for any $t > t_0$, $F(\mathbf{w}^t) - F(\mathbf{w}^*) \leq 16D_0\Delta_i + \frac{1}{2L}\Delta_i^2$. If this statement is not true, then we let $t_1 > t_0$ be the first time that $F(\mathbf{w}^t) - F(\mathbf{w}^*) > 16D_0\Delta_i + \frac{1}{2L}\Delta_i^2$. Then there must be $F(\mathbf{w}^{t_1-1}) < F(\mathbf{w}^{t_1})$. According to Equation (30), there should also be

$$F(\mathbf{w}^{t_1-1}) - F(\mathbf{w}^*) \geq F(\mathbf{w}^{t_1}) - F(\mathbf{w}^*) - \frac{1}{2L}\Delta_i^2 > 16D_0\Delta_i.$$

Then, according to the first order optimality of convex functions, for any $\mathbf{w}$,

$$F(\mathbf{w}) - F(\mathbf{w}^*) \leq \langle \nabla F(\mathbf{w}), \mathbf{w} - \mathbf{w}^* \rangle \leq \|\nabla F(\mathbf{w})\|_2 \|\mathbf{w} - \mathbf{w}^*\|_2,$$

and thus

$$\|\nabla F(\mathbf{w})\|_2 \geq \frac{F(\mathbf{w}) - F(\mathbf{w}^*)}{\|\mathbf{w} - \mathbf{w}^*\|_2}.$$

This gives that

$$\|\nabla F(\mathbf{w}^{t_1-1})\|_2 \geq \frac{F(\mathbf{w}^{t_1-1}) - F(\mathbf{w}^*)}{\|\mathbf{w}^{t_1-1} - \mathbf{w}^*\|_2} > 8\Delta_i.$$

Then according to Equation (30), this implies $F(\mathbf{w}^{t_1}) \leq F(\mathbf{w}^{t_1-1})$, which contradicts with the fact that $F(\mathbf{w}^{t_1-1}) < F(\mathbf{w}^{t_1})$.

Convergence analysis for non-convex Losses: We condition on the event that Equation (21), (22), (23), (24) or (25) holds for all $\mathbf{w} \in \mathcal{W}$ (for Algorithm 2, 3, 4 respectively). We first show that when Assumption 4(c) is satisfied and we choose $\eta_t = \frac{1}{L}$, the iterates $\mathbf{w}^t$ stays in $\mathcal{W}$ without using projection. Since we have

$$\|\mathbf{w}^{t+1} - \mathbf{w}^*\|_2 \leq \|\mathbf{w}^t - \mathbf{w}^*\|_2 + \eta_t(\|\nabla F(\mathbf{w}^t)\|_2 + \|\tilde{\mathbf{g}}(\mathbf{w}^t) - \nabla F(\mathbf{w}^t)\|_2) \leq \|\mathbf{w}^t - \mathbf{w}^*\|_2 + \frac{1}{L}(M + \Delta_i).$$

Then, we know that by running $T = \frac{2L}{\Delta_i^2}(F(\mathbf{w}^0) - F(\mathbf{w}^*))$ parallel iterations, using Assumption 4(c), we know that $\mathbf{w}^t \in \mathcal{W}$ for $t = 0, 1, \ldots, T$ without projection.

We proceed to study the convergence rate of the algorithm. By the smoothness of $F(\mathbf{w})$, we know that when choosing $\eta_t = \frac{1}{L}$, the inequality Equation (30) still holds. More specifically, for all $t = 0, 1, \ldots, T-1$,

$$F(\mathbf{w}^{t+1}) - F(\mathbf{w}^*) \leq F(\mathbf{w}^t) - F(\mathbf{w}^*) - \frac{1}{2L}\|\nabla F(\mathbf{w}^t)\|_2^2 + \frac{1}{2L}\Delta_i^2. \tag{31}$$

Sum up Equation (31) for $t = 0, 1, \ldots, T - 1$. Then, we get

$$0 \leq F(\mathbf{w}^T) - F(\mathbf{w}^*) \leq F(\mathbf{w}^0) - F(\mathbf{w}^*) - \frac{1}{2L} \sum_{t=0}^{T-1} \|\nabla F(\mathbf{w}^t)\|_2^2 + \frac{T}{2L}\Delta_i^2.$$

This implies that

$$\min_{t=0,1,\ldots,T} \|\nabla F(\mathbf{w}^t)\|_2^2 \leq \frac{2L}{T}(F(\mathbf{w}^0) - F(\mathbf{w}^*)) + \Delta_i^2,$$

which completes the proof.

E Proof of Theorem 3.2

The proof follows the same route as of Theorem 3.1, except that we utilize the result of sub-Gaussian rate for the new procedure. In particular, we rely on the following lemma that appears in (Diakonikolas et al., 2020).

Lemma E.1 (Proposition 1.6, (Diakonikolas et al., 2020))**.** *Assume that* $\|\mathsf{Cov}_{p^\star}(\mathbf{x})\| \leq \sigma^2$. *Consider the procedure which first randomly bucket* m *samples from* $p^\star$ *into* $k = \lfloor \epsilon m + \log(1/\delta) \rfloor$ *disjoint buckets of equal size, compute their empirical means* $\mathbf{z}^1, \cdots, \mathbf{z}^k$, *and apply Equation (11) or (12) onto the empirical means. Let the output be* μ. *Then with probability at least* $1 - \delta$:

$$\|\mu - \mu_{p^\star}\| \leq \mathcal{O}\left(\frac{\sigma}{(1 - C\epsilon)} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d + \log(1/\delta)}{m}}\right)\right).$$

The rest of the proof follows the same as that in Appendix D.

F Analysis for multi-round local model update

In this section, we analyze the FL protocol in Algorithm 1 with $k = m$ when $H \geq 2$ total rounds are updated locally before aggregation in central server. Similar analysis can also be applied to convex and non-convex cases, along with the case of $k < m$.

In this case, we have the local update for the i-th honest machine be

$$\mathbf{g}_i(\mathbf{w}^{Ht}) = -\sum_{s=Ht}^{H(t+1)-1} \eta_t \nabla F_i(\mathbf{w}_i^s)$$

In this case, we slightly modify our Assumption 3 to consider the H-round local model update:

Assumption 5. *Assume that for any fixed* $\mathbf{w}^{Ht} \in \mathcal{W}$ *and any* $\eta_t < 1/(2H), i \in [m]$, $\|\mathsf{Cov}(\mathbf{g}_i(\mathbf{w}^{Ht}))\|_2 \leq \mathcal{O}(\frac{\eta_t^2 \sigma_H^2}{n})$.

Note that the distribution of $\mathbf{g}_j(\mathbf{w}^{Ht})$ has complex dependence: for any $l \in [H]$, the randomness of $\mathbf{w}_i^{Ht+l}$ depends on both the randomness of noise $\mathbf{z}_{Ht+l}^{i,j}$ and the parameter in the previous step $\mathbf{w}_i^{Ht+l-1}$. However, one can still verify that in the simple case of mean estimation with $f(\mathbf{w}; \mathbf{z}) = \|\mathbf{w} - \mathbf{z}\|_2^2$. Assumption 3 implies Assumption 5 with $\sigma_H = H\sigma$.

23

This is from that

$$\|\text{Cov}(\mathbf{g}_i(\mathbf{w}^{Ht}))\|_2 = \eta_t^2 \left\| \text{Cov}\left(\frac{1}{n} \sum_{j=1}^{n} \sum_{s=Ht}^{H(t+1)-1} \nabla f(\mathbf{w}_i^s; \mathbf{z}_s^{i,j}) \right) \right\|_2$$

$$= 4\eta_t^2 \left\| \text{Cov}\left(\frac{1}{n} \sum_{j=1}^{n} \sum_{s=Ht}^{H(t+1)-1} (\mathbf{w}_i^s - \mathbf{z}_s^{i,j}) \right) \right\|_2$$

$$= 4\eta_t^2 \left\| \text{Cov}\left(\frac{1}{n} \sum_{j=1}^{n} \sum_{s=Ht}^{H(t+1)-1} (\mathbf{w}_i^{Ht} - \mathbf{z}_s^{i,j} - \eta_t \sum_{l=Ht}^{s} (\mathbf{w}_i^l - \mathbf{z}_l^{i,j})) \right) \right\|_2$$

$$= 4\eta_t^2 \left\| \text{Cov}\left(\frac{1}{n} \sum_{j=1}^{n} \sum_{s=Ht}^{H(t+1)-1} -\mathbf{z}_s^{i,j} - \eta_t \sum_{l=Ht}^{s-1} (\mathbf{w}_i^l - \mathbf{z}_l^{i,j}) \right) \right\|_2$$

$$\leq 4\eta_t^2 \left\| \text{Cov}\left(\frac{1}{n} \sum_{j=1}^{n} \sum_{s=Ht}^{H(t+1)-1} (\sum_{k=s-Ht}^{H} (\eta_t H)^k) \mathbf{z}_s^{i,j} \right) \right\|_2$$

$$\leq \mathcal{O}\left(\frac{\eta_t^2 H^2 \sigma^2}{n} \right).$$

The last inequality is from the independence between $\mathbf{z}_s^{i,j}$. Similarly, one can also show that under mild regularization conditions on the parameter space, Assumption 5 is satisfied in linear regression.

Now we are ready to prove the main theorem as below.

Theorem F.1. *Let* $\mathbf{w}_i^t, i \in \{1, 2, 3, 4, 5\}$ *denote the output of Algorithm 1 with* $H \geq 2$ *and step-size* $\eta_t = a/L(t+a)$ *at round* t *when the* RobustEstimation *subroutine takes the algorithm in Equation* (1), (2), (3), (4), (5), *respectively, where* σ^2 *is replaced with* σ_H^2. *Here* $a = (L + \lambda)/\lambda$. *Under Assumption 1, 2, 5 and 4(a), with probability at least* $1 - \delta$, *we have*

$$\|\mathbf{w}^{HT} - \mathbf{w}^*\|_2 \leq \left(\frac{a}{T+a-1} \right)^H \|\mathbf{w}^0 - \mathbf{w}^\star\|_2 + \frac{C_1 a^2 H^2 D}{(T+a-1)^2} + C_2 \Delta_i,$$

where

$$\Delta_1 := \mathcal{O}\left(\frac{\sigma_H}{(1-6\epsilon)\sqrt{n}} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d^2 \log(1+nmL) + d\log(d/\delta)}{m}} \right) \right), \tag{32}$$

$$\Delta_2 := \mathcal{O}\left(\frac{\sigma_H}{(1-2\epsilon)\sqrt{n}} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d^2 \log(1+nmL) + d\log(d/\delta)}{m}} \right) \right), \tag{33}$$

$$\Delta_3 := \mathcal{O}\left(\frac{\sigma_H}{(1-C_1\epsilon)\sqrt{n}} \cdot \left(\sqrt{\epsilon + \sqrt{\frac{d \log(1+nmL) + \log(1/\delta)}{m}}} \right) \right), \tag{34}$$

$$\Delta_4 := \mathcal{O}\left(\frac{\sigma_H}{(1-C_2(\epsilon+\log(1/\delta)/m))\sqrt{n}} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d\log(d) + \log(1+nmL) + \log(1/\delta)}{m}} \right) \right), \tag{35}$$

$$\Delta_5 := \mathcal{O}\left(\frac{\sigma_H}{(1-C_3(\epsilon+\log(1/\delta)/m))\sqrt{n}} \cdot \left(\sqrt{\epsilon} + \sqrt{\frac{d\log(d) + \log(1+nmL) + \log(1/\delta)}{m}} \right) \right), \tag{36}$$

and C_i *are universal constants.*

Proof. We condition on the event that the bound above is satisfied for all $\mathbf{w} \in \mathcal{W}$. Then, in the t-th iteration, let the updated parameter before projection step be $\widehat{\mathbf{w}}^{H(t+1)} = \mathbf{w}^{Ht} + \mathbf{g}(\mathbf{w}^{Ht})$ (note that $\mathbf{g}$ is the local model update which can be viewed as negative of gradient). Thus, we have $\mathbf{w}^{H(t+1)} = \Pi_{\mathcal{W}}(\widehat{\mathbf{w}}^{H(t+1)})$. For the sake of analysis, we create

24

an artificial sequence of $\bar{\mathbf{w}}^{Ht+l}, l \in [H]$, such that $\bar{\mathbf{w}}^{Ht} = \mathbf{w}^{Ht}$, $\bar{\mathbf{w}}^{Ht+l+1} = \bar{\mathbf{w}}^{Ht+l} - \eta_t \nabla F(\bar{\mathbf{w}}^{Ht+l})$, $\forall l \in [H]$. From the property of Euclidean projection, we know that

$$
\begin{aligned}
\|\mathbf{w}^{H(t+1)} - \mathbf{w}^*\|_2 &\leq \|\widehat{\mathbf{w}}^{H(t+1)} - \mathbf{w}^*\|_2 \\
&= \|\mathbf{w}^{Ht} + \mathbf{g}(\mathbf{w}^{Ht}) - \mathbf{w}^*\|_2 \\
&\leq \|\mathbf{w}^{Ht} - \eta_t \sum_{l=0}^{H-1} \nabla F(\bar{\mathbf{w}}^{Ht+l}) - \mathbf{w}^*\|_2 + \| - \mathbf{g}(\mathbf{w}^{Ht}) - \eta_t \sum_{l=0}^{H-1} \nabla F(\bar{\mathbf{w}}^{Ht+l})\|_2 \\
&= \|\bar{\mathbf{w}}^{Ht} - \eta_t \sum_{l=0}^{H-1} \nabla F(\bar{\mathbf{w}}^{Ht+l}) - \mathbf{w}^*\|_2 + \| - \mathbf{g}(\mathbf{w}^{Ht}) - \eta_t \sum_{l=0}^{H-1} \nabla F(\bar{\mathbf{w}}^{Ht+l})\|_2 \\
&= \|\bar{\mathbf{w}}^{H(t+1)} - \mathbf{w}^\star\|_2 + \| - \mathbf{g}(\mathbf{w}^{Ht}) - \eta_t \sum_{l=0}^{H-1} \nabla F(\bar{\mathbf{w}}^{Ht+l})\|_2
\end{aligned}
\tag{37}
$$

Following the same analysis as in Equation (28), we know that when $\eta_t \leq 1/L$, for any $l \in [H]$,

$$
\|\bar{\mathbf{w}}^{Ht+l+1} - \mathbf{w}^*\|_2 \leq (1 - \frac{\eta_t L \lambda}{L + \lambda}) \|\bar{\mathbf{w}}^{Ht+l} - \mathbf{w}^*\|_2
$$

Thus we know that

$$
\|\bar{\mathbf{w}}^{H(t+1)} - \mathbf{w}^*\|_2 \leq (1 - \frac{\eta_t L \lambda}{L + \lambda})^H \|\bar{\mathbf{w}}^{Ht} - \mathbf{w}^*\|_2
\tag{38}
$$

For the second term, we know that

$$
\| - \mathbf{g}(\mathbf{w}^{Ht}) - \eta_t \sum_{l=0}^{H-1} \nabla F(\bar{\mathbf{w}}^{Ht+l})\|_2 \leq \|\mathbf{g}(\mathbf{w}^{Ht}) - \mathbb{E}[\mathbf{g}_i(\mathbf{w}^{Ht})|\mathbf{w}^{Ht}]\|_2
$$

$$
+ \| - \mathbb{E}[\mathbf{g}_i(\mathbf{w}^{Ht})|\mathbf{w}^{Ht}] - \eta_t \sum_{l=0}^{H-1} \nabla F(\bar{\mathbf{w}}^{Ht+l})\|_2.
$$

From Assumption 5 and Theorem B.1, we know that the first term satisfies w.p. $1 - \delta$

$$
\|\mathbf{g}(\mathbf{w}^{Ht}) - \mathbb{E}[\mathbf{g}_i(\mathbf{w}^{Ht})|\mathbf{w}^{Ht}]\|_2 \leq \Delta_i.
\tag{39}
$$

For the second term, we have

$$
\|\mathbb{E}[-\mathbf{g}_i(\mathbf{w}^{Ht})|\mathbf{w}^{Ht}] - \eta_t \sum_{l=0}^{H-1} \nabla F(\bar{\mathbf{w}}^{Ht+l})\|_2 = \|\eta_t \sum_{l=0}^{H-1} (\mathbb{E}[\nabla F_i(\mathbf{w}_i^{Ht+l})|\mathbf{w}^{Ht}] - \nabla F(\bar{\mathbf{w}}^{Ht+l}))\|_2
$$

Note that for any $l \in [H-1]$, we have

$$
\begin{aligned}
\|\mathbb{E}[\nabla F_i(\mathbf{w}_i^{Ht+l+1})|\mathbf{w}^{Ht}] - \nabla F(\bar{\mathbf{w}}^{Ht+l+1})\|_2 &= \|\mathbb{E}[\nabla F_i(\mathbf{w}_i^{Ht+l} - \eta_t \nabla F_i(\mathbf{w}_i^{Ht+l}))|\mathbf{w}^{Ht}] \\
&\quad - \nabla F(\bar{\mathbf{w}}^{Ht+l} - \eta_t \nabla F(\bar{\mathbf{w}}^{Ht+l}))\|_2 \\
&\overset{(i)}{\leq} \|\mathbb{E}[\nabla F_i(\mathbf{w}_i^{Ht+l})|\mathbf{w}^{Ht}] - \nabla F(\bar{\mathbf{w}}^{Ht+l})\|_2 \\
&\quad + \eta_t L(\|\nabla F(\bar{\mathbf{w}}^{Ht+l})\|_2 + \|\mathbb{E}[\nabla F_i(\mathbf{w}_i^{Ht+l})|\mathbf{w}_i^{Ht}]\|_2) \\
&\overset{(ii)}{\leq} \|\mathbb{E}[\nabla F_i(\mathbf{w}_i^{Ht+l})|\mathbf{w}^{Ht}] - \nabla F(\bar{\mathbf{w}}^{Ht+l})\|_2 + 2\eta_t L^2 D.
\end{aligned}
\tag{40}
$$

Here (i) is from the Lipschitzness of $\mathbb{E}[\nabla F_i(\mathbf{w})]$ and $\nabla F(\mathbf{w})$, and (ii) uses the fact that the parameter space is compact with diameter D, and the function is smoothness with minimizer inside $\mathcal{W}$, as in Assumption 1, 2, and 4(a).

By iteratively applying Equation (40), we know that

$$
\begin{aligned}
\|\mathbb{E}[\nabla F_i(\mathbf{w}_i^{Ht+l+1})|\mathbf{w}^{Ht}] - \nabla F(\bar{\mathbf{w}}^{Ht+l+1})\|_2 &\leq \|\mathbb{E}[\nabla F_i(\mathbf{w}_i^{Ht})|\mathbf{w}^{Ht}] - \nabla F(\bar{\mathbf{w}}^{Ht})\|_2 + 2l\eta_t L^2 D \\
&= 2l\eta_t L^2 D.
\end{aligned}
$$

The last equality uses the fact that $\nabla F_i(\mathbf{w})$ is unbiased. This leads to

$$\|\mathbb{E}[\mathbf{g}_i(\mathbf{w}^{Ht})|\mathbf{w}^{Ht}] - \eta_t \sum_{l=0}^{H-1} \nabla F(\bar{\mathbf{w}}^{Ht+l})\|_2 = \|\eta_t \sum_{l=0}^{H-1} (\mathbb{E}[\nabla F_i(\mathbf{w}_i^{Ht+l})|\mathbf{w}^{Ht}] - \nabla F(\bar{\mathbf{w}}^{Ht+l}))\|_2$$
$$\leq H^2 \eta_t^2 L^2 D. \tag{41}$$

Combining Equation (37), (38), (39), and (41), we get

$$\|\mathbf{w}^{H(t+1)} - \mathbf{w}^*\|_2 \leq (1 - \frac{\eta_t L\lambda}{L+\lambda})^H \|\mathbf{w}^{Ht} - \mathbf{w}^*\|_2 + \eta_t \Delta_i + H^2 \eta_t^2 L^2 D. \tag{42}$$

By taking $\eta_t = \frac{a}{L(t+a)}$ where $a = \frac{L+\lambda}{\lambda}$, we can verify that $\eta_t \leq 1/L$. Now we multiply $(t+a)^{H-1}/\eta_t$ on both sides, which gives

$$\frac{(t+a)^{H-1}\|\mathbf{w}^{H(t+1)} - \mathbf{w}^*\|_2}{\eta_t} \leq (\frac{t+a-1}{t+a})^H \frac{(t+a)^{H-1}}{\eta_t} \|\mathbf{w}^{Ht} - \mathbf{w}^*\|_2 + (t+a)^{H-1}\Delta_i + (t+a)^{H-1}\eta_t H^2 L^2 D$$
$$= \frac{(t+a-1)^{H-1}}{\eta_{t-1}}\|\mathbf{w}^{Ht} - \mathbf{w}^*\|_2 + (t+a)^{H-1}\Delta_i + (t+a)^{H-1}\eta_t H^2 L^2 D.$$

By recursively applying the above inequality, we get

$$\frac{(T+a-1)^{H-1}\|\mathbf{w}^{HT} - \mathbf{w}^*\|_2}{\eta_{T-1}} \leq a^{H-1}L\|\mathbf{w}^0 - \mathbf{w}^\star\|_2 + (\sum_{t=1}^{T-1} (t+a)^{H-1}\Delta_i + a(t+a)^{H-2}H^2 LD).$$

Rearranging the above formula gives

$$\|\mathbf{w}^{HT} - \mathbf{w}^*\|_2 \leq \left(\frac{a}{T+a-1}\right)^H \|\mathbf{w}^0 - \mathbf{w}^\star\|_2 + \frac{C_1 a^2 H^2 D}{(T+a-1)^2} + C_2 \Delta_i.$$

$\square$

G   Incorporating the Privacy Guarantee

Secure aggregation techniques (Bonawitz et al., 2017; Bell et al., 2020) are designed to conceal individual client's updates and reveal only the aggregated global update to a semi-honest server that attempts to infer the clients' privacy from their updates. In addition to Byzantine robustness, we aim to achieve privacy guarantees. Orchestrating robust FL estimators (to mitigate malicious clients) with secure aggregation schemes (to mitigate semi-honest servers) is challenging, as robust estimators require access to local updates, whereas secure aggregation schemes normally hide them from the server. Consequently, most de facto FL protocols cannot protect both the server and the clients *simultaneously*, but must aim to satisfy only one of the two criteria. Two recent works (So et al., 2020; He et al., 2020) attempt to address this challenge through the use of secure multi-party computation (MPC) but they either incur high communication cost or rely on non-colluding servers. This does not fit in the FL setting (see Appendix A for further discussion). Thus we are motivated to consider the following question:

Can we achieve robustness and privacy simultaneously in FL protocols?

The idea of bucketing can also be used to reconcile robust FL with secure aggregation. Similar idea of adding secure aggregation in the bucketing procedure has also been proposed and analyzed in (Velicheti et al., 2021; Burkhalter et al., 2021). Secure aggregation (Bonawitz et al., 2017; Bell et al., 2020) is a class of cryptographic protocols designed to augment user privacy in FL. Concretely, secure aggregation conceals all the unnecessary information (*i.e.* the gradient from each client) from the server and only reveals the minimum information needed to update the global model (*i.e* the summed gradient). Thus, it was considered to be incompatible with robust FL because robust FL requires each client's gradient to filter out the malicious ones.

However, with the idea of bucketing, we no longer require all the client's gradients. Instead, we only need the sums of gradients for the buckets. This leaves space for secure aggregation within each bucket and thus reconciles the two originally incompatible techniques to support bidirectional protection in federated learning. A subtlety is that in order to apply secure aggregation, we need to clip and quantize the clients' local updates to fit them in a finite field. As a result, the operator norm of the clients' updates are changed. However, it is easy to show that the difference will only be in the constants so the asymptotic results will remain the same.

Theorem G.1 (Security against semi-honest server). *Let Π be an instantiation of the proposed protocols, there exists a PPT (probabilistic polynomial Turing machine) simulator* SIM *which can only see the averaged updates from the shards. For all clients C, the output of* SIM *is computationally indistinguishable from the view of that real server Π_C in that execution, i.e., $\Pi_C \approx \text{SIM}(\{\mathbf{g}_t^{H_j}\}_{j \in [p]})$.*

Proof for Theorem G.1. The transcript of the server is the updates from the sharded clients $\{\mathbf{g}_t^{H_j}\}_{j \in [p]}$. Hence, Theorem G.1 is equivalent to the following lemma since the SIM can split the aggregated updates into several random shards which is computationally indistinguishable from the true transcript.

Lemma G.1 (Lemma 6.1 in (Bonawitz et al., 2017)). *Given any shard H_k formed by a set of clients C_k, the parameter size d, the group size q, and the updates $\mathbf{g}^{(i)}$ where $\forall i \in C_k$, $\mathbf{g}^{(i)} \in \mathbb{Z}_q^d$, we have*

$$\{\{\mathbf{u}_{ij} \xleftarrow{\$} \mathbb{Z}_q^d\}_{i < j}, \mathbf{u}_{ij} := -\mathbf{u}_{ji} \, \forall \, i, j \in C_k, \, i > j \quad : \quad \{\mathbf{g}^{(i)} + \sum_{j \in C_k/i} \mathbf{u}_{ij} \pmod{q}\}_{i \in C_k}\}$$

$$\equiv \{\{\mathbf{v}_i \xleftarrow{\$} \mathbb{Z}_q^d\}_{i \in C_k} \, s.t. \, \sum_{i \in C_k} \mathbf{v}_i = \sum_{i \in C_k} \mathbf{g}^{(i)} \pmod{q} \quad : \quad \{\mathbf{v}_i\}_{i \in C_k}\}$$

, where $\mathbf{u}_{ij}$ is the random mask shared between client i and j, $\xleftarrow{\$}$ donates uniformly sampling from some field, and $\equiv$ denotes that the distributions are identical.

Lemma G.1 illustrates that the distribution of updates with random masks added is identical to uniformly sampling from $\mathbb{Z}_q^d$. Thus, individual clients' updates are securely concealed behind the random masks added by secure aggregation, and a semi-honest server can infer zero information about individual clients from the aggregated updates alone. In the following, we prove Lemma G.1 via induction on n, where n is the size of the clients set C_k, $n = |C_k|$.

Base Case: When $n = 2$, assume $C_k = \{i, j\}$, $i < j$, and $\sum_{i \in C_k} \mathbf{g}^{(i)} \pmod{q} = c$, c is a constant. The first elements of two distributions are $\mathbf{g}^{(i)} + \mathbf{u}_{ij} \pmod{q}$ and $\mathbf{v}_i$, respectively, both of which are uniformly random sampled from $\mathbb{Z}_q^d$. The second elements are $\mathbf{g}^{(j)} + \mathbf{u}_{ji} \pmod{q} = c - (\mathbf{g}^{(i)} + \mathbf{u}_{ij}) \pmod{q}$ and $\mathbf{v}_j = c - \mathbf{v}_i \pmod{q}$, respectively, which are the sum c minus the corresponding first elements. As a result, the distributions are identical.

Inductive Hypothesis: When $n = k$, the lemma holds.

Inductive Step: According to the inductive hypothesis, the left and right distributions of the first k clients are indistinguishable. We follow the protocol to generate the left transcript when the $(k+1)^{th}$ client is added to the shard. To deal with the right-hand-side transcript, we first add the same randomness as for the left-hand-side to the first k updates and then subtract them from the total sum to obtain the $(k+1)^{th}$ update. It's easy to prove that the first k updates on both sides follow the same uniformly random distribution, and that the $(k+1)^{th}$ update is the difference between the total sum and the sum of the first k updates. Hence, the left and right transcripts are indistinguishable.

□

In case the readers are not familiar with the simulation proof technique, please refer to (Lindell, 2017) for more information about simulation-based security proof.

H   Evaluation on Breaking Points

To empirically evaluate the breaking point of the proposed robust aggregators, we test FILTERING and NO-REGRET on MNIST with different number of malicious clients as shown in Figure 3. We can tell that when $\epsilon <= 0.4$, both FILTERING and NO-REGRET works well. Only when ϵ reaches 0.5, NO-REGRET breaks down because the number of malicious clients is the same as the benign ones and no robust aggregator can work under such settings.
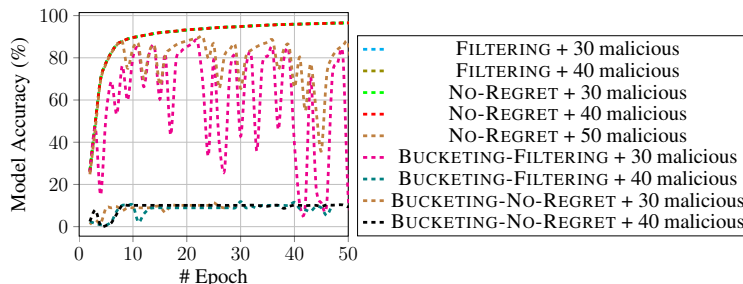

Figure 3: Evaluation w/ different number of malicious clients.

I   Evaluation of Sever

We also evaluate the performance of Sever (Diakonikolas et al., 2019a) in federated learning. Sever requires that the clients' updates are from a γ-approximate learner. Thus, we invoke Sever after the model converges (i.e., Epoch 20 in Fig. 4). Sever computes outlier scores for the clients' updates and filters the malicious clients. And then, we retrain the model and invoke Sever after convergence till we reach a stationary point. We evaluate the performance of Sever on MNIST under MPA attack. As shown in Fig. 4, Sever cannot successfully mitigate MPA.

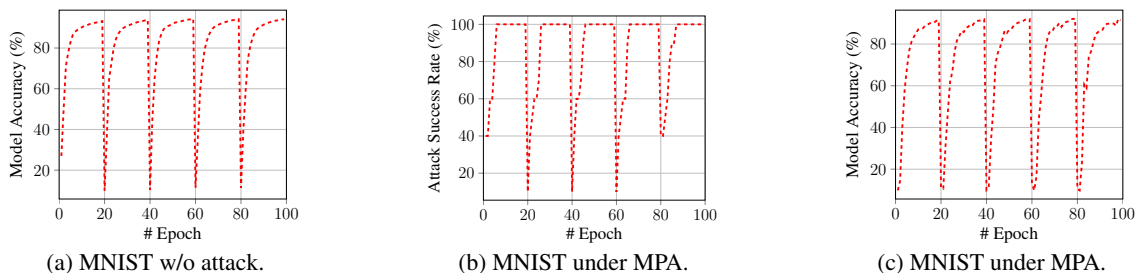(a) MNIST w/o attack.　　　　(b) MNIST under MPA.　　　　(c) MNIST under MPA.

Figure 4: Performance of Sever on MNIST without attack and with MPA.