Breaking BAD: Heterogeneous Byzantine-Robust Federated Learning via Gather and Scatter Scores

Sena Ergisi* **Luis Massny*** **Rawad Bitar**
School of Computation, Information and Technology,
Technical University of Munich, Germany
{sena.ergisi, luis.massny, rawad.bitar}@tum.de

Abstract

Byzantine-robust Federated Learning (FL) designs aggregation rules that mitigate the impact of malicious clients. When the data across honest clients are heterogeneous, honest model updates are inherently scattered. This dispersion can be exploited by Byzantine adversaries who may collude to concentrate their updates, thereby circumventing state-of-the-art robust aggregation methods and causing a Byzantine Adversarial Disruption (BAD) of the training process. We propose a novel Byzantine-robust FL scheme based on *gather* and *scatter* scores. Our scoring technique captures concentration among Byzantine clients and prevents catastrophic failure under SotA attacks. Through numerical experiments on FEMNIST, Shakespeare, and CIFAR datasets, we demonstrate that the proposed defense maintains high model accuracy, even in realistic FL scenarios. Furthermore, we establish theoretical convergence guarantees for our method.

1 Introduction

Federated learning (FL) enables collaborative model training on data generated and owned by participating clients, without moving the clients' data to a central location (McMahan et al., 2017). In FL, the central server broadcasts a global model to selected clients. The clients perform local training on their private data and return the model updates or gradients to the server. The server aggregates these updates to form a new global model and repeats the process until certain convergence criteria are met.

Despite FL's potential, achieving a viable and trustworthy FL framework remains challenging due to many factors, including: *(i)* the exacerbated communication cost; *(ii)* maintaining privacy of the clients' data; and *(iii)* ensuring robustness against adversarial behavior, e.g., (Kairouz et al., 2021; Rodríguez-Barroso et al., 2023). Addressing those challenges, individually and jointly, is the focus of several works, e.g., (Konený et al., 2016; Egger et al., 2025; Girgis et al., 2021; Sun et al., 2019; Wang et al., 2020; Blanchard et al., 2017; Farhadkhani et al., 2022; Allouah et al., 2023c; Xia et al., 2025) and references within.

We focus on ensuring robustness against Byzantine[2] adversarial disruption (BAD) caused by malicious clients. Such clients can either send crafted updates, known as model poisoning attacks (Bagdasaryan et al., 2020; Cao & Gong, 2022), or manipulate their local training data to corrupt the global model, known as data poisoning attacks (Ovi & Gangopadhyay,

*Equal contribution.
[2]The adversary is called Byzantine in reference to the Byzantine generals problem (Lamport et al., 1982).

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

2025). Adversarial behavior is problematic since it has been shown that, if care is not taken, even a single malicious client can arbitrarily manipulate the training (Blanchard et al., 2017).

To defend against Byzantine attacks, several robust aggregation methods have been proposed (see Section 2.2). Robust aggregation attempts to identify a reliable update direction by filtering out anomalous updates, typically assuming that most clients are honest. This works well in homogeneous settings (IID), where the clients' data are independently and identically distributed, leading to honest updates that look similar. However, in realistic FL scenarios, clients' data are often heterogeneous (non-IID), leading to naturally dispersed updates that are difficult to distinguish from adversarial ones (Ye et al., 2023). In such scenarios, state-of-the-art robust aggregation methods do not withstand adversarial attacks that exploit this dispersion of updates, leading to severe disruption in the resulting final model (Ergisi et al., 2025).

A key underlying premise of robust aggregation methods is that the updates of malicious clients appear as outliers to the honest distribution. However, most successful attacks require malicious clients to send highly similar updates that appear honest yet bias the resulting global model, e.g., (Baruch et al., 2019; Xie et al., 2020; Shejwalkar & Houmansadr, 2021; Karimireddy et al., 2022). Inspired by the naming in (Douceur, 2002), we refer to such attacks as Sybil attacks. Defenses against Sybil attacks rely on penalizing excessive similarity between the clients' updates (Fung et al., 2018).

Therefore, in heterogeneous settings, a trade-off arises: for Sybil attacks, the similarity between updates should be penalized and diversity rewarded; for non-Sybil attacks, similarity should be rewarded and diversity penalized.[3] This makes robust aggregation methods vulnerable to one attack or the other. As such, the question we ask is: *Can one design a robust aggregation method that works for both types of attacks simultaneously?*

Contributions We propose `GatScHA`, an aggregation method simultaneously robust against both Sybil and non-Sybil attacks. Our method combines two complementary scoring mechanisms: one favors similarity (*gathered* updates), and the other emphasizes diversity (*scattered* updates). Combining both scores yields a highly effective robust aggregation method for heterogeneous settings and limits adversaries' ability to bias the global update.

We provide theoretical convergence guarantees for general loss functions and validate our approach through extensive experiments on three benchmark datasets, and include an ablation study showing how our method scales with the number of clients and the ratio of malicious to honest clients. To the best of our knowledge, this is the first robust FL aggregation method with theoretical convergence guarantees that works against both Sybil and non-Sybil attacks; thus, taking a crucial step towards solving the fundamental problem of robustness in realistic non-IID FL.

2 System Model and Related Work

In this section, we formally define the system model, provide a brief overview of closely related work, and explicitly state the problem we will solve.

2.1 System Model

Notation Calligraphic letters denote sets, e.g., $\mathcal{A}$, and bold symbols denote vectors, e.g., $\boldsymbol{a}$. For an integer $a \geq 1$, we define $[a] \triangleq \{1, 2, \ldots, a\}$. By $\|\cdot\|$ we denote the Euclidean ℓ_2-norm.

Learning Objective We consider an FL system with a central server and N clients, each holding a dataset $\mathcal{S}_k$, $k \in [N]$. The datasets $\mathcal{S}_k$, $k \in [N]$, are drawn from (potentially non-IID) distributions over the space $\mathcal{X} \times \mathcal{Y}$, where $\mathcal{X}$ denotes the input (feature) space of the data and $\mathcal{Y}$ denotes the output (label) space.

[3] A similar trade-off has been observed in the context of client selection (Yan et al., 2024).

The learning objective is to find a function $\mathcal{F}(\boldsymbol{\theta}; \boldsymbol{x})$ parameterized by a vector $\boldsymbol{\theta} \in \mathbb{R}^d$ that correctly estimates the label of samples from the overall dataset $\mathcal{S} \triangleq \cup_{k\in[N]}\mathcal{S}_k$, i.e., for any $(\boldsymbol{x}, y) \in \mathcal{S}$, the function $\mathcal{F}(\boldsymbol{\theta}; \boldsymbol{x})$ outputs $\hat{y}$ and the goal is for $\hat{y}$ to be equal to y.

To measure how far the output of $\mathcal{F}(\boldsymbol{\theta}; \boldsymbol{x})$ is from y, the clients use local loss functions, typically defined as $\mathcal{L}_k(\boldsymbol{\theta}) \triangleq \frac{1}{|\mathcal{S}_k|}\sum_{(\boldsymbol{x},y)\in\mathcal{S}_k} \ell(\mathcal{F}(\boldsymbol{\theta}; \boldsymbol{x}), y),$, where $\ell(\cdot, \cdot)$ measures the per-sample loss, and $\mathcal{L}_k$ is assumed to be differentiable with respect to $\boldsymbol{\theta}$. The goal of the central server is to find the best parameter vector $\boldsymbol{\theta}^\star$ that minimizes the global loss function defined as $\mathcal{L}(\boldsymbol{\theta}) \triangleq \frac{1}{N}\sum_{k=1}^{N} \mathcal{L}_k(\boldsymbol{\theta})$, namely, to solve the unconstrained optimization problem $\boldsymbol{\theta}^\star \in \operatorname{argmin}_{\boldsymbol{\theta}\in\mathbb{R}^d} \mathcal{L}(\boldsymbol{\theta})$.

To find $\boldsymbol{\theta}^\star$, the server employs an iterative gradient descent (GD) method starting from a random vector $\boldsymbol{\theta}^{(1)}$. In the simplest form, at each round t, client k receives the global model $\boldsymbol{\theta}^{(t)}$, chooses a random mini-batch of its data $\mathcal{B}_k^{(t)} \subseteq \mathcal{S}_k$ without replacement and computes $\boldsymbol{\theta}_k^{(t+1)} = \boldsymbol{\theta}^{(t)} - \frac{\gamma_k^{(t)}}{|\mathcal{B}_k^{(t)}|}\sum_{(\boldsymbol{x},y)\in\mathcal{B}_k^{(t)}} \nabla\ell(\mathcal{F}(\boldsymbol{\theta}^{(t)}; \boldsymbol{x}), y)$, for a local learning rate $\gamma_k^{(t)}$. Client k sends the model update $\boldsymbol{g}_k^{(t)} \triangleq \frac{1}{\gamma_k^{(t)}}\left(\boldsymbol{\theta}^{(t)} - \boldsymbol{\theta}_k^{(t+1)}\right)$ to the server. We refer to this method as distributed stochastic gradient descent (D-SGD). We refer to the case where $\mathcal{B}_k^{(t)} = \mathcal{S}_k$ as distributed gradient descent (D-GD).

Some variants of D-SGD apply local momentum with parameter $\beta \in [0, 1]$. This is also known as the distributed stochastic heavy-ball (D-SHB) method. Instead of sending $\boldsymbol{g}_k^{(t)}$ to the server, client k sends a local momentum $\boldsymbol{m}_k^{(t)}$ as the model update, which is recursively updated as

$$\boldsymbol{m}_k^{(t)} = \beta\boldsymbol{m}_k^{(t-1)} + (1-\beta)\boldsymbol{g}_k^{(t)}, \tag{1}$$

with $\boldsymbol{m}_k^{(0)} \triangleq 0$. The server aggregates the received local momenta[4] according to an aggregation rule $\texttt{Agg}(\cdot)$ as $\hat{\boldsymbol{m}}^{(t)} = \texttt{Agg}(\boldsymbol{m}_1^{(t)}, \ldots, \boldsymbol{m}_N^{(t)})$, typically using weighted average, and updates the global model as $\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} - \gamma^{(t)}\hat{\boldsymbol{m}}^{(t)}$ using a global learning rate $\gamma^{(t)}$. The process repeats for T rounds or until certain convergence criteria are met. We neglect the round index t in the superscript when it is clear from the context.

Threat Model We assume the presence of a Byzantine adversary controlling up to $f < N/2$ malicious clients, aiming to cause a BAD in the learning process. We use the strongest adversarial model, in which each client has full knowledge of the learning process, including the server's aggregation method and the updates of honest clients. In addition, adversarial clients can collude to design a joint attack.

Robust Aggregation To break the BAD, the server needs to carefully craft a robust aggregation rule $\texttt{RAgg}(\cdot)$ that takes as input the N updates, out of which up to f might be corrupt, and outputs an aggregated update that is faithful to the $|\mathcal{H}|$ honest updates, where $\mathcal{H} \subseteq [N]$ denotes the set of honest clients. In such a setting, the minimization objective changes to $\mathcal{L}_\mathcal{H}(\boldsymbol{\theta}) \triangleq \frac{1}{|\mathcal{H}|}\sum_{k\in\mathcal{H}} \mathcal{L}_k(\boldsymbol{\theta})$. We abbreviate the optimum as $\mathcal{L}_\mathcal{H}^\star \triangleq \mathcal{L}_\mathcal{H}(\boldsymbol{\theta}^\star)$.

We assume the server does not have a clean data set on which it can test the performance of the updates sent by the clients. It also has no control over the clients' training data or the distribution of that data and does not know $\mathcal{H}$. The only information it can leverage is N and f. Upon receiving N local updates $\boldsymbol{m}_1, \ldots, \boldsymbol{m}_N$, the server applies an aggregation rule $\texttt{RAgg}(\cdot)$ that is designed to be robust. The resulting FL algorithm used in this paper is given in Appendix C.

2.2 Related Work and Motivation

The crux of a Byzantine-robust FL algorithm is to design an aggregation rule that can be computed from the information available to the server. A good rule should allow for the

[4]The momentum reduces to the gradient for $\beta = 0$.

best accuracy given *any* adversarial attack. Several robust aggregation rules have been designed in the literature. We mention the ones closely related to our work. `Median` outputs the coordinate-wise median update vector, and $\texttt{TrimmedMean}_q$ discards the q largest and q smallest values and averages the remaining $N - 2q$ values coordinate-wise (Yin et al., 2018). Similarly, $\texttt{MeaMed}_q$ computes the coordinate-wise median and averages the $N - q$ values closest to it, discarding the q farthest (Xie et al., 2018). The rule `GeoMed` (Chen et al., 2017; Pillutla et al., 2022) outputs the geometric median of the N updates. Methods such as `Krum` (Blanchard et al., 2017), `Bulyan` (Mhamdi et al., 2018), and `Sniper` (Cao et al., 2019) select updates based on distance to neighboring updates, while `FABA` (Xia et al., 2019) and `CClip` (Karimireddy et al., 2020) compare updates to the global mean. Other approaches emphasize directional similarity and use cosine distance or clustering-based techniques (Li et al., 2021; Sattler et al., 2020; Li et al., 2023). A common assumption underlying the defenses discussed so far is that the majority of participating clients remain honest. To overcome this constraint, some methods assume the server has a small, clean dataset that reflects the global data distribution. This enables testing on model updates (Xhemrishi et al., 2025; Xie et al., 2019) or reference-based validation (Cao et al., 2021) and can tolerate $f \geq N/2$ malicious clients.

Providing robustness is particularly challenging in realistic heterogeneous settings, where honest outliers and malicious updates may be indistinguishable under standard similarity metrics. Accordingly, it can be shown (Karimireddy et al., 2022; Allouah et al., 2023b) that the learning error is lower bounded by the dissimilarity of the updates. On a high level, existing defenses rely on two assumptions: *(i)* honest updates are similar to each other; and *(ii)* malicious updates must significantly deviate from the honest mean to be harmful. The main problem is that, in practical FL scenarios, data heterogeneity and the careful design of the Byzantine attack can violate both assumptions. Data heterogeneity may cause the violation of the first assumption, and the honest updates might deviate considerably.

This fact is exploited by SotA attacks, which mimic benign heterogeneous behavior (Karimireddy et al., 2022), or submit only slightly perturbed versions (Baruch et al., 2019; Xie et al., 2020), but can disrupt the learning. Applying pre-aggregation techniques (Allouah et al., 2023a; Karimireddy et al., 2022) and incorporating local momentum have been proposed to reduce the dispersion of updates and improve robustness. In FL scenarios with heterogeneous data, some attacks can still overcome SotA robust aggregation rules and degrade the model to random-guess performance (Ergisi et al., 2025). Similarity among updates is no longer a sign of honest behavior in the aforementioned scenarios, but instead can be an indicator of collusion, which has also been observed by Fung et al. (2018) and Yan et al. (2024).

This highlights the need for aggregation rules that are robust to both Sybil and non-Sybil attacks. To that end, `MAB-RFL` (Wan et al., 2022) and `ProDiGy` (Ergisi et al., 2025) proposed two-stage defenses to tackle both. Despite their empirical success, `MAB-RFL` relies on client selection and does not provide convergence guarantees, while `ProDiGy` lacks theoretical robustness guarantees and fails under the ZeroMean attack, a new attack we introduce in this work (see Appendix B.3). In contrast, *our method provides a unified aggregation rule with theoretically provable and empirically validated robustness guarantees against both Sybil and non-Sybil attacks, and significantly outperforms the state of the art.*

3 Robust Aggregation with Gather and Scatter Scores

We propose and analyze a novel robust aggregation framework named `GatScHA`: Gather and Scatter Honest Aggregation. A `GatScHA` scheme takes N updates as input out of which $0 < f < N/2$ updates are corrupted and outputs an update vector that is provably close to the aggregate of the honest updates.

3.1 `GatScHA`: A Novel Robust Aggregation Framework

Our framework assigns each update $\boldsymbol{m}_k$, $k \in [N]$ a weight w_k and outputs $\hat{\boldsymbol{m}}$ as the weighted average of $\boldsymbol{m}_k$ with weights w_k (step 12 of Algorithm 1). In contrast to other robust aggregation rules, the weight w_k of the update $\boldsymbol{m}_k$ balances two complementary scores: a *gather* score $\omega_{\texttt{Gat}}(k)$, which captures the similarity of $\boldsymbol{m}_k$ to the majority of the updates, and

a *scatter* score $\omega_{\texttt{Sc}}(k)$, which rewards the distinction of $\boldsymbol{m}_k$ among the updates. GatScHA first removes outliers based on the gather score, then it combines the gather and scatter scores to obtain the weight w_k for the remaining updates. The GatScHA framework is shown in Algorithm 1 and explained next.

For each client k, the server computes the pairwise distance of $\boldsymbol{m}_k$ and $\boldsymbol{m}_{k'}$, $\delta_{k,k'} = \|\boldsymbol{m}_k - \boldsymbol{m}_{k'}\|^2$, $k' \in [N]$ and sorts the clients by the distance of their updates to $\boldsymbol{m}_k$. Denoting by k_i the i-th closest neighbor of client k, it holds that $0 = \delta_{k,k_0} \leq \delta_{k,k_1} \leq \cdots \leq \delta_{k,k_{N-1}}$. Exceptionally, indexing starts at k_0 here since $k_0 = k$.

To discourage scattering, GatScHA measures the distance of client k's update to its $(N-f-1)$-st closest neighbor to compute $\omega_{\texttt{Gat}}(k) = 1/\delta_{k,k_{N-f-1}}$, which reflects how closely the client's neighbors are gathered.[5]

Then, it discards the updates of the f clients with the smallest $\omega_{\texttt{Gat}}(k)$, i.e., whose update is too far from their $(N-f-1)$-st closest neighbor. Hence, it sorts $(\omega_{\texttt{Gat}}(k))_{k\in[N]}$ in descending order such that $\omega_{\texttt{Gat}}(k_1) \geq \omega_{\texttt{Gat}}(k_2) \geq \cdots \geq \omega_{\texttt{Gat}}(k_N)$ and for the f clients with lowest values $\omega_{\texttt{Gat}}(k)$ it assigns a weight $w_k = 0$. For each of the not discarded $N-f$ updates, a scatter score $\omega_{\texttt{Sc}}(k)$, that is instantiation-dependent, is computed to evaluate how diverse the updates are. The weight of those updates are then computed as $w_k = \omega_{\texttt{Sc}}(k) \cdot \omega_{\texttt{Gat}}(k)$.

Algorithm 1 The GatScHA Framework

Require: Client updates $\boldsymbol{m}_1, \ldots, \boldsymbol{m}_N \in \mathbb{R}^d$, number of clients N, bound f on number of malicious clients, scatter score $\omega_{\texttt{Sc}}(k)$

// Compute pairwise distances and closest neighbors

for $k = 1, 2, \ldots, N$ **do**
  $w_k \leftarrow 0$
  Compute $\delta_{k,k'} = \|\boldsymbol{m}_k - \boldsymbol{m}_{k'}\|^2$ for all $k' \in [N]$
  Sort $(\delta_{k,k'})_{k'\in[N]}$ with index $(k_i)_{i=0,1,\ldots,N-1}$ s.t. $0 = \delta_{k,k_0} \leq \delta_{k,k_1} \leq \cdots \leq \delta_{k,k_{N-1}}$
  $\omega_{\texttt{Gat}}(k) \leftarrow 1/\delta_{k,k_{N-f-1}}$
end for

// Assign nonzero scores to $N-f$ largest $\omega_{\texttt{Gat}}(k)$

Sort $(\omega_{\texttt{Gat}}(k))_{k\in[N]}$ with index $(k_j)_{j=1,\ldots,N}$ s.t. $\omega_{\texttt{Gat}}(k_1) \geq \omega_{\texttt{Gat}}(k_2) \geq \cdots \geq \omega_{\texttt{Gat}}(k_N)$
for $j = 1, 2, \ldots, N-f$ **do**
  Compute $\omega_{\texttt{Sc}}(k_j)$
  $w_{k_j} \leftarrow \omega_{\texttt{Sc}}(k_j) \cdot \omega_{\texttt{Gat}}(k_j)$
end for
return $\texttt{GatScHA}(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_N) = \frac{\sum_{k\in[N]} w_k \cdot \boldsymbol{m}_k}{\sum_{k\in[N]} w_k}$

Instantiating GatScHA As an instantiation of GatScHA, we consider the scatter score

$$\omega_{\texttt{Sc}}(k) \triangleq \sqrt{\frac{1}{|\mathcal{N}_f(k)|} \sum_{i\in\mathcal{N}_f(k)} \|\boldsymbol{m}_i - \boldsymbol{\mu}(\mathcal{N}_f(k))\|^2} \tag{2}$$

computed as the standard deviation of the updates in its neighborhood $\mathcal{N}_f(k)$. This score is specifically chosen against Sybil-type attacks, where $f \geq 2$. We define the *neighborhood* of client k as the cluster of updates of size f centered around client k, i.e., $\mathcal{N}_f(k) \triangleq \{\boldsymbol{m}_{k_0}, \cdots, \boldsymbol{m}_{k_{f-1}}\}$, and denote the mean of the neighborhood updates $\boldsymbol{\mu}(\mathcal{N}_f(k)) \triangleq \frac{1}{|\mathcal{N}_f(k)|} \sum_{i\in\mathcal{N}_f(k)} \boldsymbol{m}_i$. This instantiation is rooted in the idea that a single

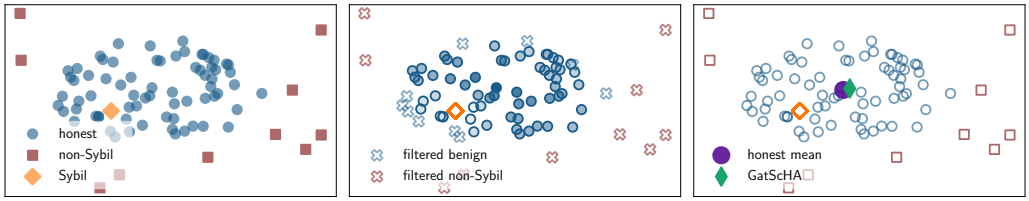

Figure 1: Illustration of GatScHA with $N = 100$ clients under an ALIE attack with 20 Sybil (diamonds) and 10 non-Sybil (squares) malicious updates, alongside honest updates (circles). *Left*: a toy example of client updates in a $2D$ parameter space. *Middle*: updates filtered out ($w_k = 0$, crosses) by the gather score $\omega_{\texttt{Gat}}(k)$, with remaining updates shaded according to their final combined scores; note that all Sybil updates receive $w_k = 0$ due to the scatter and gather scoring. *Right*: comparison of the honest mean with the aggregation result of GatScHA, showing effective neutralization of the bias introduced by all 30 malicious updates.

[5] In practice, a small constant $\epsilon > 0$ can be added to the denominator to avoid division by zero in the unlikely event that $N-f$ updates are identical. This does not affect the theoretical guarantees.

(even honest) update is not representative in non-IID settings. For this reason, other state-of-the-art defenses can be effectively attacked in heterogeneous settings by creating malicious model updates that mimic an honest outlier. Although the traditional defenses output an aggregate that is close to the mimicked honest outlier, this results in a skew since the output is unfaithful to the general honest population. Our scatter score penalizes such skew and thus provides more balanced aggregates. In total, this enables `GatScHA` to maintain high model accuracy even in regimes where existing methods can degrade to random-guess performance. In combination with the gather score, `GatScHA` searches for a sweet spot between outlier removal and balanced aggregation. This behavior is further visualized in Figure 1.

Complexity As we explain next, our instantiation of `GatScHA` has an expected overall complexity of $\mathcal{O}(N^2d + N^2 \log N + Nfd)$. The complexity is dominated by $\mathcal{O}(N^2d)$ as $f < N/2$ and, in realistic setting, $d \gg \log N$. As `GatScHA` is based on the pairwise distances, the complexity is driven by the very computation of the matrix $(\delta_{i,j})_{i,j\in[N]}$, which requires $\mathcal{O}(N^2d)$ operations. For each client, the remaining updates are sorted by their distances, incurring $\mathcal{O}(N \log N)$ operations per client and $\mathcal{O}(N^2 \log N)$ overall. Computing the gather score requires selecting the $(N - f - 1)$-st closest update, which is included in the sorting step. The scatter score involves computing the mean and standard deviation over each clients neighborhood, which costs $\mathcal{O}(fd)$ per client, or $\mathcal{O}(Nfd)$ overall. Overall, the total complexity is $\mathcal{O}(N^2d + N^2 \log N + Nfd)$.

3.2 Theoretical Guarantees

We show that *any* instantiation of `GatScHA` is guaranteed to output an aggregate $\hat{\boldsymbol{m}}$ that is faithful to the honest majority, which in turn provides a robustness guarantee for our instantiation of `GatScHA`. The proofs of our results are deferred to Appendix A.

We base our theoretical analysis on the following standard assumptions about all benign local loss functions $\mathcal{L}_k$, $k \in [\mathcal{H}]$:

- (**A1**) The loss functions are continuously differentiable and L-smooth, i.e., there exists a constant $L \geq 0$ such that: $\|\nabla\mathcal{L}_k(\boldsymbol{\theta}_1) - \nabla\mathcal{L}_k(\boldsymbol{\theta}_2)\| \leq L\|\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2\|, \quad \forall\boldsymbol{\theta}_1, \boldsymbol{\theta}_2 \in \mathbb{R}^d$.
- (**A2**) There exists a constant $\zeta^2 \geq 0$, such that: $\|\nabla\mathcal{L}_k(\boldsymbol{\theta}) - \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta})\|^2 \leq \zeta^2, \quad \forall\boldsymbol{\theta} \in \mathbb{R}^d$.
- (**A3**) For all rounds t, the update noise $\boldsymbol{g}_k^{(t)} - \nabla\mathcal{L}_k(\boldsymbol{\theta}^{(t)})$ is independent and unbiased and there exists a constant $\sigma^2 \geq 0$ such that: $\mathbb{E}_{\xi_t}\left[\|\boldsymbol{g}_k^{(t)} - \nabla\mathcal{L}_k(\boldsymbol{\theta}^{(t)})\|^2\right] \leq \sigma^2$. The expectation is taken over the algorithm's randomness ξ_t in round t.

When a local loss function is composed of sample-wise losses as defined in Section 2.1, (**A1**) holds whenever the sample-wise loss function $\ell(\mathcal{F}(\boldsymbol{\theta}; \mathbf{x}), y)$ is L-smooth in $\boldsymbol{\theta}$. Our model of heterogeneity in (**A2**) is in line with prior works and, as in (Liu et al., 2023), expresses the maximal deviation of an honest update from the honest mean. This is closely related to the slightly more relaxed model used in (Karimireddy et al., 2022; Allouah et al., 2023a).[6]

We define the honest aggregate as $\boldsymbol{m}_{\mathcal{H}}^{(t)} \triangleq \frac{1}{|\mathcal{H}|}\sum_{k\in\mathcal{H}} \boldsymbol{m}_k^{(t)}$. Our theoretical convergence guarantee for `GatScHA` is based on the result that, despite the combination with any scatter score, the aggregation error $\|\hat{\boldsymbol{m}}^{(t)} - \boldsymbol{m}_{\mathcal{H}}^{(t)}\|^2$ of the overall scheme remains bounded. The concatenation with a scatter score can consequently only improve the robustness guarantees.

Lemma 3.1 (Robustness of `GatScHA`)**.** *Suppose at most $0 < f < N/2$ clients are malicious. Under assumption (**A2**) and (**A3**), the aggregation error of any `GatScHA` scheme is bounded as $\mathbb{E}\left[\|\hat{\boldsymbol{m}}^{(t)} - \boldsymbol{m}_{\mathcal{H}}^{(t)}\|^2\right] \leq 30\zeta^2 + (12 + 6(N - f + 1))\frac{1-\beta}{1+\beta}\sigma^2$ in every iteration t.*

The bounded aggregation error guarantees convergence towards the neighborhood of a stationary point for any smooth (possibly non-convex) loss function. We note that the bound

[6] The model in (Karimireddy et al., 2022; Allouah et al., 2023a) measures heterogeneity as the expected deviation from the honest mean, whereas we consider a constant that bounds any clients' deviation. This fits the robustness properties of `GatScHA` more naturally, as `GatScHA`'s error can be in the order of a benign update's worst-case deviation. Similar guarantees can be derived for the expected deviation.

in Lemma 3.1 can be tightened to $\|\hat{\boldsymbol{m}}^{(t)} - \boldsymbol{m}_{\mathcal{H}}^{(t)}\|^2 \leq 10\zeta^2$ for the non-stochastic GD setting (cf. Remark A.6). With SGD, in contrast, momentum is inevitable for good convergence performance as observable in the bound above, which is in line with observations from the literature, e.g., (Mhamdi et al., 2021; Karimireddy et al., 2020). Following (Allouah et al., 2023a), we, thus, treat the GD and SGD cases separately. We study the theoretical convergence guarantees of `GatScHA` for D-GD ($\mathcal{B}_k^{(t)} = \mathcal{S}_k$ and $\sigma^2 = \beta = 0$) and D-SGD, where we apply the D-SHB method. We denote the initialization error as $\mathcal{L}_1 \triangleq \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(1)}) - \mathcal{L}_{\mathcal{H}}^{\star}$.

Theorem 3.2 (Convergence guarantees with D-GD)**.** *Using any `GatScHA` scheme in the presence of at most $0 < f < N/2$ malicious clients, the FL algorithm with $\mathcal{B}_k^{(t)} = \mathcal{S}_k$ and $\sigma^2 = \beta = 0$ and learning rate $\gamma^{(t)} \leq \frac{1}{4L}$ guarantees that under assumptions (**A1**) and (**A2**):*

$$\frac{1}{T}\sum_{t=1}^{T}\left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2 \leq 20\left(1+2\gamma L\right)\zeta^2 + \frac{4\mathcal{L}_1}{\gamma T}, \quad T \geq 1.$$

Theorem 3.3 (Convergence guarantees with D-SHB)**.** *Using any `GatScHA` scheme in the presence of at most $0 < f < N/2$ malicious clients, the FL algorithm with fixed learning rate $\gamma = \min\{\frac{1}{24L}, \frac{C}{\sqrt{T}}\}$, for a constant $0 \leq C \leq 1/\left(\frac{1}{N-f} + 12 + 6(N-f+1)\right)$, and momentum $\beta = \sqrt{1-24\gamma L}$ guarantees that, under assumptions (**A1**), (**A2**), and (**A3**):*

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2\right] \leq 360\zeta^2 + \frac{9\mathcal{L}_1 + 288L\sigma^2C}{\sqrt{T}C} + \frac{216L\mathcal{L}_1}{T} + \frac{288L\sigma^2C}{T^{3/2}(N-f)}, \quad T \geq 1.$$

Both results are in coherence with the literature on Byzantine-robust FL. The bound for D-GD has two terms: one that decays with the number of rounds T and one that is constant and is a multiple of the clients' gradient dissimilarity ζ^2. Using D-SGD adds more noise to the gradient calculation, i.e., $\sigma^2 > 0$. Similarly to the bound obtained for D-GD, our bound has one term that is a constant multiple of ζ^2 and other terms that decay, although at a slower rate, with T. Note that the asymptotic error is $\mathcal{O}(\zeta^2)$, whereas optimal schemes can achieve an asymptotic error of $\mathcal{O}(\frac{f}{N}\zeta^2)$ (Karimireddy et al., 2022; Allouah et al., 2023a). This is an artifact of allowing a scatter score, which can introduce undesirable skew in the output towards one of the $N - f$ non-eliminated updates. Therefore, as our theoretical guarantees aim to cover all possible instantiations of `GatScHA`, sanity checks proposed for other theoretical frameworks as in (Farhadkhani et al., 2022) are not applicable. Nevertheless, the optimal scaling can be recovered for particular instantiations of `GatScHA`. For instance, effectively averaging the updates with the highest gather score allows us to tighten the bound in Lemma 3.1 as we show in Corollary 3.4. In this case, the asymptotic error in Theorem 3.2 and Theorem 3.3 can be shown to also scale as $\mathcal{O}(\frac{f}{N}\zeta^2)$ accordingly. Although such instantiation of `GatScHA` without a scatter score yields tighter theoretical bounds scaling with f/N, similarly to SotA schemes, it cannot handle Sybil attacks as illustrated in Figure 4. Theoretical and practical limitations are further discussed in Section 5.

Corollary 3.4 (Robustness under effective averaging)**.** *If the clients with non-zero weights are assigned a uniform nonzero value $w_{k_j} = w$, $j = 1, 2, \ldots, N-f$, then the aggregation error of this `GatScHA` scheme is $\mathbb{E}\left[\|\hat{\boldsymbol{m}}^{(t)} - \boldsymbol{m}_{\mathcal{H}}^{(t)}\|^2\right] \leq 36\frac{f}{N-f}\left(\zeta^2 + \frac{1-\beta}{1+\beta}\sigma^2\right)$.*

4 Experimental Evaluation

Experimental Setup We conduct extensive experiments[7] to compare the performance of `GatScHA` with SotA robust aggregation methods across various settings and under different attacks. The FL algorithm used in this paper is given in Appendix C. We consider image classification tasks on CIFAR-100, CIFAR-10 (Krizhevsky, 2009) and FEMNIST (Caldas et al., 2019) datasets, and a natural language processing task (next character prediction) on the Shakespeare (Caldas et al., 2019) dataset. For CIFAR datasets, the client's data distribution for each class j is chosen as Dirichlet distribution (Yurochkin et al., 2019; Lin et al., 2020), i.e., $\boldsymbol{p}_j \sim \mathrm{Dir}_N(\alpha)$. The source of strong non-IID distribution, thus, becomes a

[7] Our code is available at `https://github.com/DTS-TUM/BreakingBad`.

label distribution skew. FEMNIST and Shakespeare datasets are inherently heterogeneous across clients. We report the mean accuracy and its standard deviation across three random seeds, $\{0, 1, 2\}$. For CIFAR-100, we fine-tune a DeiT (Touvron et al., 2021), pretrained on ImageNet−1k (Deng et al., 2009), while for all other datasets, we train CCT, CNN, and LSTM models from scratch.

We report the experimental results for the following attack scenarios: A Little Is Enough (ALIE) (Baruch et al., 2019), Fall of Empires (FOE) (Xie et al., 2020), Label-Flip (LF), Sign-Flip (SF) (Allen-Zhu et al., 2021), ZeroMean, Mimic (Karimireddy et al., 2022), MinSum and MinMax (Shejwalkar & Houmansadr, 2021). For ALIE and FOE, the attack parameters (z and ϵ, respectively) are further optimized, where we select the value that maximizes the ℓ_2 distance between the aggregated update and the average update of the honest clients.

`GatScHA` is mainly compared against the following representative defenses: `Median`, `TrimmedMean` (Yin et al., 2018), `MeaMed` (Xie et al., 2018), `GeoMed` (Pillutla et al., 2022), `Krum` (Blanchard et al., 2017), `CClip` (Karimireddy et al., 2020), and `ProDiGy` (Ergisi et al., 2025). Where applicable, these defenses are further strengthened with the pre-aggregation scheme `NNM` (Allouah et al., 2023a). We also implemented two baseline settings, for reference: No Attack/No Defense, where all N clients participate in training honestly; and Oracle Defense, where the server averages only $N - f$ honest clients gradients.

More details on experimental setup, attacks and defenses are provided in Appendix B.

Experimental Results Our main experimental results are summarized in Tables 1 and 2 and Figure 2. We refer the interested reader to Appendix B.4 for more comprehensive results.

Table 1: Final accuracies across four datasets with D-SHB ($\beta = 0.9$) reported as mean ± standard deviation over three independent runs. For each defense, the worst-case accuracy is highlighted in bold. For each dataset, the best worst-case accuracy is highlighted in green. Cells are marked in red if at least one run approaches random-guess performance.

CIFAR-100 Dataset ($\text{Dir}_{20}(0.1)$) | `BASELINES` No Attack/No Defense: **83.48 ± 0.19** ; Any Attack/Oracle Defense: **80.41 ± 0.41**								
	ALIE	**FOE**	**LF**	**SF**	**ZeroMean**	**Mimic**	**MinSum**	**MinMax**
$N = 20$, $f = 5$								
`NNM + Median`	1.00 ± 0.00	**0.98 ± 0.02**	63.17 ± 6.55	75.30 ± 5.84	74.17 ± 1.17	75.42 ± 0.59	0.99 ± 0.15	1.10 ± 0.12
`NNM + MeaMed`	1.00 ± 0.00	**0.94 ± 0.08**	63.49 ± 6.51	76.78 ± 3.55	74.17 ± 1.34	74.49 ± 1.37	1.04 ± 0.11	1.19 ± 0.13
`NNM + TrimmedMean`	1.00 ± 0.00	**0.94 ± 0.10**	62.70 ± 6.53	75.92 ± 4.79	74.18 ± 1.42	70.56 ± 0.50	1.23 ± 0.28	1.06 ± 0.06
`NNM + GeoMed`	**1.01 ± 0.07**	1.12 ± 0.18	65.39 ± 8.22	78.76 ± 0.72	75.95 ± 0.94	75.11 ± 1.58	1.41 ± 0.66	39.88 ± 26.90
`NNM + Krum`	1.00 ± 0.00	**0.98 ± 0.09**	62.79 ± 6.44	74.94 ± 5.59	74.17 ± 1.21	76.80 ± 1.45	1.21 ± 0.17	2.24 ± 0.45
`NNM + CClip`	**0.91 ± 0.08**	0.95 ± 0.04	73.81 ± 4.16	69.46 ± 7.51	78.18 ± 0.59	75.79 ± 0.43	1.12 ± 0.22	1.43 ± 0.56
`ProDiGy`	78.88 ± 0.63	79.35 ± 0.64	69.07 ± 8.35	78.13 ± 1.07	**11.57 ± 0.82**	79.28 ± 0.43	80.38 ± 0.39	79.87 ± 0.78
`GatScHA`	73.46 ± 1.08	75.41 ± 0.94	**62.91 ± 6.51**	78.62 ± 0.76	73.87 ± 1.24	70.79 ± 2.75	73.48 ± 1.28	73.33 ± 1.26
CIFAR-10 Dataset ($\text{Dir}_{20}(0.5)$) | `BASELINES` No Attack/No Defense: **80.18 ± 0.10** ; Any Attack/Oracle Defense: **78.97 ± 0.13**								
	ALIE	**FOE**	**LF**	**SF**	**ZeroMean**	**Mimic**	**MinSum**	**MinMax**
$N = 20$, $f = 5$								
`NNM + Median`	**9.92 ± 0.12**	10.18 ± 0.28	46.35 ± 8.03	38.83 ± 4.47	64.75 ± 0.72	48.47 ± 7.99	29.65 ± 10.08	32.23 ± 7.48
`NNM + MeaMed`	**10.00 ± 0.00**	10.76 ± 1.07	41.49 ± 6.92	35.52 ± 6.02	66.36 ± 0.93	41.24 ± 5.64	32.28 ± 6.83	40.17 ± 10.12
`NNM + TrimmedMean`	**10.00 ± 0.00**	12.69 ± 3.80	64.93 ± 3.49	60.73 ± 4.06	71.47 ± 0.47	60.28 ± 4.27	21.37 ± 4.80	42.65 ± 7.12
`NNM + GeoMed`	**9.99 ± 0.01**	10.00 ± 0.00	68.38 ± 3.43	68.83 ± 0.56	73.51 ± 0.52	60.20 ± 3.68	45.21 ± 25.02	40.22 ± 16.83
`NNM + Krum`	**10.00 ± 0.00**	**10.00 ± 0.00**	61.26 ± 3.09	53.57 ± 4.69	71.00 ± 1.62	51.20 ± 4.46	29.44 ± 11.02	34.87 ± 11.39
`NNM + CClip`	**10.00 ± 0.00**	42.78 ± 23.84	70.03 ± 2.83	70.38 ± 0.23	76.14 ± 0.46	69.01 ± 4.38	36.61 ± 19.65	39.64 ± 18.88
`ProDiGy`	70.42 ± 0.56	70.34 ± 1.77	59.00 ± 4.51	51.21 ± 4.65	**18.73 ± 1.21**	71.86 ± 1.23	71.41 ± 1.28	70.82 ± 1.71
`GatScHA`	63.47 ± 1.57	65.09 ± 0.96	62.50 ± 3.06	**59.81 ± 2.63**	66.56 ± 1.05	66.81 ± 1.16	67.85 ± 1.54	66.59 ± 1.93
FEMNIST Dataset | `BASELINES` No Attack/No Defense: **81.08 ± 0.59** ; Any Attack/Oracle Defense: **79.10 ± 0.55**								
	ALIE	**FOE**	**LF**	**SF**	**ZeroMean**	**Mimic**	**MinSum**	**MinMax**
$N = 20$, $f = 5$								
`NNM + Median`	**6.60 ± 0.00**	27.53 ± 14.27	77.94 ± 2.04	72.46 ± 1.03	72.28 ± 0.08	74.81 ± 0.63	69.84 ± 3.19	39.23 ± 24.64
`NNM + MeaMed`	**6.60 ± 0.00**	32.94 ± 18.95	77.96 ± 2.04	72.01 ± 1.11	72.70 ± 0.34	74.88 ± 0.50	65.65 ± 4.71	66.90 ± 3.04
`NNM + TrimmedMean`	**29.45 ± 20.76**	54.34 ± 4.07	78.01 ± 2.01	73.17 ± 1.19	73.51 ± 1.19	74.92 ± 0.46	64.64 ± 3.48	67.84 ± 1.80
`NNM + GeoMed`	**6.60 ± 0.00**	24.48 ± 25.94	77.98 ± 2.08	73.27 ± 1.60	73.46 ± 1.47	75.23 ± 0.49	64.95 ± 2.96	63.50 ± 5.60
`NNM + Krum`	**6.59 ± 0.01**	32.62 ± 13.53	77.96 ± 2.05	73.34 ± 1.80	72.57 ± 0.26	74.26 ± 0.43	67.48 ± 3.76	63.94 ± 6.36
`NNM + CClip`	42.37 ± 6.98	**6.03 ± 0.54**	74.95 ± 1.04	71.50 ± 0.67	75.53 ± 0.86	77.38 ± 0.38	63.29 ± 2.27	68.32 ± 4.71
`ProDiGy`	76.28 ± 0.53	76.98 ± 0.23	78.52 ± 0.50	71.49 ± 1.63	**3.15 ± 2.05**	78.11 ± 0.73	78.47 ± 0.57	77.70 ± 0.48
`GatScHA`	74.00 ± 0.87	75.14 ± 1.14	75.89 ± 4.78	**59.59 ± 11.00**	73.55 ± 1.20	73.61 ± 1.06	74.26 ± 0.84	72.95 ± 0.46
Shakespeare Dataset | `BASELINES` No Attack/No Defense: **50.72 ± 0.84** ; Any Attack/Oracle Defense: **49.50 ± 0.95**								
	ALIE	**FOE**	**LF**	**SF**	**ZeroMean**	**Mimic**	**MinSum**	**MinMax**
$N = 50$, $f = 12$								
`NNM + Median`	**23.69 ± 5.90**	25.22 ± 12.29	49.55 ± 0.99	47.82 ± 0.81	49.97 ± 0.63	49.36 ± 1.25	36.47 ± 10.02	45.17 ± 3.40
`NNM + MeaMed`	**20.66 ± 1.24**	34.68 ± 7.56	49.55 ± 1.02	47.86 ± 0.79	50.00 ± 0.59	49.38 ± 1.12	33.97 ± 11.35	45.63 ± 3.18
`NNM + TrimmedMean`	**26.79 ± 5.50**	35.87 ± 6.30	49.52 ± 1.11	47.86 ± 0.85	50.07 ± 0.58	49.39 ± 0.97	27.35 ± 20.16	40.24 ± 2.15
`NNM + GeoMed`	**19.57 ± 0.53**	36.79 ± 4.28	49.67 ± 1.04	47.80 ± 0.89	50.14 ± 0.55	49.70 ± 0.94	24.36 ± 15.44	45.50 ± 3.12
`NNM + Krum`	**24.38 ± 7.68**	37.20 ± 4.18	49.56 ± 1.00	47.82 ± 1.01	49.98 ± 0.64	49.30 ± 1.28	35.35 ± 12.49	34.40 ± 10.93
`NNM + CClip`	**13.00 ± 5.12**	18.95 ± 0.00	49.47 ± 0.98	47.80 ± 0.89	50.14 ± 0.55	49.70 ± 0.93	24.26 ± 16.38	42.46 ± 2.75
`ProDiGy`	47.52 ± 0.55	49.17 ± 0.99	49.61 ± 1.02	46.97 ± 0.30	**18.95 ± 0.00**	49.56 ± 1.21	49.58 ± 1.02	49.39 ± 1.13
`GatScHA`	50.00 ± 1.21	50.01 ± 1.28	49.35 ± 0.98	**46.41 ± 1.24**	50.46 ± 0.93	50.51 ± 0.92	49.94 ± 0.93	50.37 ± 1.05

Table 1 summarizes the final accuracies [8] of each defense across varying attack scenarios across four datasets. A key observation is that every competing defense is degraded to random-guess accuracy or below by at least one attack (highlighted in red). For CIFAR-10 and CIFAR-100, random-guess performance is 10% and 1%, respectively, since test datasets are class-balanced. For inherently imbalanced FEMNIST and Shakespeare test sets, random-guess accuracy is instead given by the most frequent label: 6.60% (the letter `e`) and 18.95% (the space character), respectively. Notably, the strongest attacks against competing defenses are colluding attacks, in which all malicious clients send identical updates. `GatScHA` withstands these attacks by incorporating the scatter score and is the only defense that avoids catastrophic failure in the considered experimental settings.

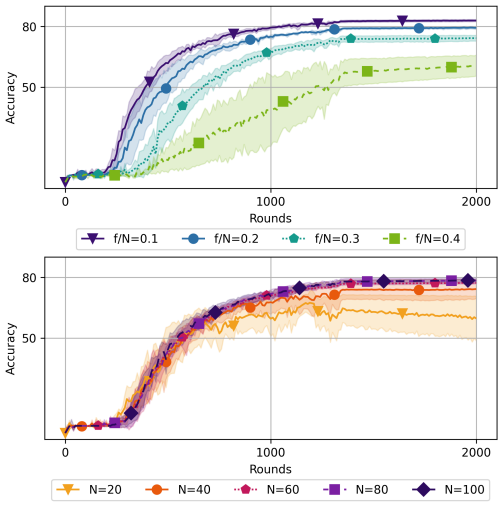

Figure 2: Test accuracy of `GatScHA` on FEMNIST under the SF attack with D-SHB ($\beta = 0.9$). Top: varying fraction of malicious clients $\frac{f}{N} \in \{0.1, 0.2, 0.3, 0.4\}$ with $N = 60$ fixed. Bottom: varying number of clients $N \in \{20, 40, 60, 80, 100\}$ with $\frac{f}{N} = 0.25$ fixed.

We further study the robustness and scalability of `GatScHA` through two ablations: varying the fraction of malicious clients f/N with $N = 60$ fixed, and varying the number of clients N with $f/N = 0.25$ fixed. We use FEMNIST because its natural per-client data split means that increasing N adds new training data rather than redistributing the existing data. In both settings, we consider the SF attack, which is the most effective attack against `GatScHA` in our main experiments. Figure 2 shows that robustness degrades as the fraction of Byzantine clients increases, while final model performance improves as the number of clients grows.

5 Limitations and Discussion

As many other Byzantine-robust aggregation schemes, e.g., (Allouah et al., 2023a; Liu et al., 2023; Karimireddy et al., 2022; El-Mhamdi et al., 2021; Karimireddy et al., 2020; Yin et al., 2018; Blanchard et al., 2017), `GatScHA` relies on the hyper-parameter f, which is an upper bound on the number of malicious clients. The precise estimation of this parameter, thus, is an important research problem of independent interest.

When f is underestimated, only $\hat{f} < f$ updates are excluded from the aggregation. In case of highly deviating malicious updates, such as in the SF attack, the non-eliminated malicious updates may lead to a catastrophic failure. If f is overestimated, i.e., $\hat{f} > f$, the scatter score in Equation (2) becomes vulnerable to Sybil attacks: the larger neighborhood $\mathcal{N}_{\hat{f}}(k)$ fails to detect the small standard deviation among the malicious clients, as happens under the ALIE, FOE, MinSum, and MinMax attacks (cf. Table 2).

Herein, we also consider a variant of our aggregation rule, $\texttt{GatScHA}_{\texttt{Agn}}$, which is fully *agnostic* of the parameter f, and thus, robust against its misestimation. $\texttt{GatScHA}_{\texttt{Agn}}$ uses Algorithm 1 with *gather* score and *scatter* score defined as:

$$\omega_{\texttt{Gat}}(k) = 1/\delta_{k,k_{\lceil \frac{N-1}{2} \rceil}}, \quad \omega_{\texttt{Sc}}(k) = \delta_{k,k_1}.$$

This aggregation rule is an instantiation of `GatScHA` with the parameter f chosen conservatively as $f = \lfloor \frac{N-1}{2} \rfloor$ and the scatter score measuring the nearest neighbor's distance. Hence, this method too satisfies `GatScHA`'s theoretical guarantees. This conservative choice

[8] Final rather than peak accuracies are reported, as some defenses initially reach high performance before degrading to random-guess level over the course of training; see Figures 5 to 8 in Appendix B.4.

inevitably discards some honest updates along with the malicious ones, resulting in a slight drop in performance in exchange for robustness to misestimation (cf. Tables 1 and 2).

Table 2: Effects of misestimation of the parameter f with D-SHB ($\beta = 0.9$) for FEMNIST and CIFAR-100 ($\mathrm{Dir}_{20}(0.1)$) datasets. The estimated parameter is denoted as $\hat{f}$. Rows corresponding to underestimation and overestimation are separated by a dashed line. Cells are formatted as in Table 1.

		ALIE	FOE	LF	SF	ZeroMean	Mimic	MinSum	MinMax
FEMNIST $N = 60$, $f = 15$	GatScHA ($\hat{f} = 6$)	83.25 ± 0.57	83.61 ± 0.48	72.26 ± 1.17	**49.68 ± 30.49**	83.21 ± 0.49	83.15 ± 0.33	83.77 ± 0.18	83.45 ± 0.33
	GatScHA ($\hat{f} = 12$)	82.51 ± 0.26	82.54 ± 0.46	**76.49 ± 2.47**	76.62 ± 0.62	82.00 ± 0.43	82.24 ± 0.12	82.51 ± 0.16	82.60 ± 0.33
	GatScHA ($\hat{f} = 18$)	**29.94 ± 16.53**	66.70 ± 4.31	84.08 ± 0.23	77.22 ± 0.45	78.02 ± 0.13	81.53 ± 0.88	82.21 ± 0.41	42.63 ± 27.26
	GatScHA ($\hat{f} = 24$)	**25.91 ± 14.26**	36.26 ± 21.35	82.88 ± 0.26	77.47 ± 0.90	74.59 ± 0.39	81.46 ± 1.76	82.84 ± 0.23	67.43 ± 1.48
	$\texttt{GatScHA}_{\mathrm{Agn}}$	77.30 ± 0.50	**75.71 ± 1.00**	82.57 ± 0.17	77.96 ± 0.74	76.78 ± 0.80	77.48 ± 3.35	82.48 ± 0.13	79.29 ± 2.48
		ALIE	**FOE**	**LF**	**SF**	**ZeroMean**	**Mimic**	**MinSum**	**MinMax**
CIFAR-100 $N = 20$, $f = 5$	GatScHA ($\hat{f} = 3$)	77.40 ± 0.55	77.83 ± 0.45	61.30 ± 2.13	**1.02 ± 0.05**	77.77 ± 0.53	76.97 ± 0.58	77.90 ± 0.36	77.60 ± 0.39
	GatScHA ($\hat{f} = 4$)	74.70 ± 0.72	75.04 ± 0.74	65.42 ± 6.05	**0.99 ± 0.06**	75.65 ± 0.93	74.77 ± 0.81	75.07 ± 0.66	75.08 ± 0.99
	GatScHA ($\hat{f} = 6$)	1.05 ± 0.07	0.97 ± 0.04	64.27 ± 9.13	78.19 ± 0.30	72.26 ± 1.88	76.03 ± 3.71	26.92 ± 36.56	**0.96 ± 0.07**
	GatScHA ($\hat{f} = 7$)	1.80 ± 1.13	**1.03 ± 0.07**	67.91 ± 11.84	75.02 ± 0.63	67.36 ± 1.18	76.20 ± 0.45	50.89 ± 35.06	1.06 ± 0.04
	$\texttt{GatScHA}_{\mathrm{Agn}}$	62.80 ± 8.01	66.45 ± 4.72	**62.18 ± 12.54**	73.96 ± 0.56	62.35 ± 2.20	67.49 ± 7.46	73.79 ± 1.20	65.15 ± 3.91

Another limiting practical aspect can be the computational complexity $\mathcal{O}(N^2 d)$, which is due to the pairwise distance computations in GatScHA. Nevertheless, also other SotA baselines, e.g., NNM, leverage pairwise distances, resulting in the same complexity. As the model dimensions d are very large in modern machine learning models, it remains an interesting research direction to study how robust aggregation rules will fare with compression techniques, see e.g., (Rammal et al., 2024).

Lastly, we note that the algorithms considered in this work aim at limiting the deviations caused by malicious behavior and are evaluated by the resulting average learning error. Our figures of merit do not capture backdooring attacks (Bagdasaryan et al., 2020; Liu et al., 2024) or other attacks, which exploit the particular dataset semantics to corrupt the learned model without significant deviations.

Interestingly, we observe a discrepancy between the theoretical guarantees and empirical performance. On the one hand, while particular instantiations of GatScHA can provide stronger theoretical robustness, cf. Section 3.2, the GatScHA framework cannot guarantee optimal theoretical robustness universally. This is an artifact of our scatter score's complex interplay with the gather score, the characterization of which is a main challenge we face. On the other hand, our instantiation empirically outperforms other defenses that employ NNM, for instance, which has an optimal robustness in theory. Therefore, we opted to remain faithful to our empirically stronger algorithm. Bridging this gap between theory and empirical performance is a very interesting and challenging problem that we are currently working on.

6 Conclusion

We propose a novel Byzantine-robust aggregation scheme, GatScHA. It employs a dual scoring mechanism based on new *gather* and *scatter* scores to break BAD. Our intuition for the scoring technique stems from observing that the underlying assumptions of previous robust aggregation methods can be systematically violated in practical FL scenarios. Through experimentation across diverse and challenging non-IID settings, we show that GatScHA is the only defense that consistently maintains high accuracy, never degrading to random-guess performance under any of the evaluated attack scenarios. We also theoretically show that GatScHA converges in heterogeneous FL under Byzantine attacks. This work opens the door to further systematic study of attacks that balance Sybil and non-Sybil behavior, which we will pursue. In addition, studying the combination of robust aggregation and other aspects of FL, such as data privacy or communication efficiency, remains a very interesting open problem.

Acknowledgments and Disclosure of Funding

We thank the anonymous reviewers for their insightful comments and constructive discussion, which substantially improved the manuscript. S. Ergisi and R. Bitar are supported by the German Research Foundation (DFG) grant agreement number BI 2492/1-1.

References

Allen-Zhu, Z., Ebrahimian, F., Li, J. Z., and Alistarh, D. Byzantine-resilient non-convex stochastic gradient descent. In *International Conference on Learning Representations*, 2021.

Allouah, Y., Farhadkhani, S., Guerraoui, R., Gupta, N., Pinot, R., and Stephan, J. Fixing by mixing: A recipe for optimal byzantine ml under heterogeneity. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pp. 1232–1300, 2023a.

Allouah, Y., Guerraoui, R., Gupta, N., Pinot, R., and Rizk, G. Robust distributed learning: Tight error bounds and breakdown point under data heterogeneity. In *Advances in Neural Information Processing Systems*, volume 36, pp. 45744–45776. Curran Associates, Inc., 2023b.

Allouah, Y., Guerraoui, R., Gupta, N., Pinot, R., and Stephan, J. On the privacy-robustness-utility trilemma in distributed learning. In *International Conference on Machine Learning*, pp. 569–626. PMLR, 2023c.

Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., and Shmatikov, V. How to backdoor federated learning. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 2938–2948, 2020.

Baruch, G., Baruch, M., and Goldberg, Y. A little is enough: Circumventing defenses for distributed learning. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Blanchard, P., El Mhamdi, E. M., Guerraoui, R., and Stainer, J. Machine learning with adversaries: Byzantine tolerant gradient descent. In *Advances in Neural Information Processing Systems*, volume 30, pp. 118–128, 2017.

Bottou, L., Curtis, F. E., and Nocedal, J. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018.

Caldas, S., Duddu, S. M. K., Wu, P., Li, T., Konený, J., McMahan, H. B., Smith, V., and Talwalkar, A. Leaf: A benchmark for federated settings. In *Workshop on Federated Learning for Data Privacy and Confidentiality*, 2019.

Cao, D., Chang, S., Lin, Z., Liu, G., and Sun, D. Understanding distributed poisoning attack in federated learning. In *2019 IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS)*, pp. 233–239, 2019.

Cao, X. and Gong, N. Z. Mpaf: Model poisoning attacks to federated learning based on fake clients. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 3395–3403, 2022.

Cao, X., Fang, M., Liu, J., and Gong, N. FLTrust: Byzantine-robust federated learning via trust bootstrapping. In *Network and Distributed System Security Symposium*, 2021.

Chen, Y., Su, L., and Xu, J. Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. In *Abstracts of the 2018 ACM International Conference on Measurement and Modeling of Computer Systems*, 2017.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

Douceur, J. R. The sybil attack. In *Revised Papers from the First International Workshop on Peer-to-Peer Systems*, IPTPS '01, pp. 251260, Berlin, Heidelberg, 2002. Springer-Verlag. ISBN 3540441794.

Egger, M., Bitar, R., Wachter-Zeh, A., Weinberger, N., and Gunduz, D. Bi-directional communication-efficient stochastic fl via remote source generation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

El-Mhamdi, E. M., Farhadkhani, S., Guerraoui, R., Guirguis, A., Hoang, L.-N., and Rouault, S. Collaborative learning in the jungle (decentralized, byzantine, heterogeneous, asynchronous and nonconvex learning). In *Advances in Neural Information Processing Systems*, volume 34, pp. 25044–25057. Curran Associates, Inc., 2021.

Ergisi, S., Maßny, L., and Bitar, R. Prodigy: Proximity- and dissimilarity-based byzantine-robust federated learning. In *3rd International Conference on Federated Learning Technologies and Applications (FLTA)*, pp. 180–187, 2025.

Farhadkhani, S., Guerraoui, R., Gupta, N., Pinot, R., and Stephan, J. Byzantine machine learning made easy by resilient averaging of momentums. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 6246–6283, 2022.

Fung, C., Yoon, C. J. M., and Beschastnikh, I. Mitigating sybils in federated learning poisoning. *ArXiv preprint arXiv:1808.04866*, 2018.

Girgis, A., Data, D., Diggavi, S., Kairouz, P., and Theertha Suresh, A. Shuffled model of differential privacy in federated learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 2521–2529. PMLR, 2021.

Guerraoui, R., Gupta, N., and Pinot, R. *Robust Machine Learning*. Springer, 1 edition, 2025.

Hassani, A., Walton, S., Shah, N., Abuduweili, A., Li, J., and Shi, H. Escaping the big data paradigm with compact transformers. *ArXiv preprint arXiv:2104.05704*, 2022.

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1–2):1–210, 2021.

Karimireddy, S. P., He, L., and Jaggi, M. Learning from history for byzantine robust optimization. In *International Conference on Machine Learning*, 2020.

Karimireddy, S. P., He, L., and Jaggi, M. Byzantine-robust learning on heterogeneous datasets via bucketing. In *International Conference on Learning Representations*, 2022.

Konený, J., McMahan, H. B., Yu, F. X., Richtarik, P., Suresh, A. T., and Bacon, D. Federated learning: Strategies for improving communication efficiency. In *NIPS Workshop on Private Multi-Party Machine Learning*, 2016.

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

Lamport, L., Shostak, R., and Pease, M. The byzantine generals problem. *ACM Trans. Program. Lang. Syst.*, 4(3):382401, July 1982. ISSN 0164-0925.

Li, S., Ngai, E. C. H., and Voigt, T. An experimental study of byzantine-robust aggregation schemes in federated learning. *IEEE Transactions on Big Data*, 10:975–988, 2023.

Li, Z., Liu, L., Zhang, J., and Liu, J. Byzantine-robust federated learning through spatial-temporal analysis of local model updates. In *2021 IEEE 27th International Conference on Parallel and Distributed Systems (ICPADS)*, pp. 372–379, 2021.

Lin, T., Kong, L., Stich, S. U., and Jaggi, M. Ensemble distillation for robust model fusion in federated learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 2351–2363, 2020.

Liu, T., Zhang, Y., Feng, Z., Yang, Z., Xu, C., Man, D., and Yang, W. Beyond traditional threats: A persistent backdoor attack on federated learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(19):21359–21367, 2024.

Liu, Y., Chen, C., Lyu, L., Wu, F., Wu, S., and Chen, G. Byzantine-robust learning on heterogeneous data via gradient splitting. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 21404–21425, 2023.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and Arcas, B. A. y. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pp. 1273–1282, 2017.

Mhamdi, E. M. E., Guerraoui, R., and Rouault, S. The hidden vulnerability of distributed learning in byzantium. In *International Conference on Machine Learning*, volume 80, pp. 3521–3530. PMLR, 2018.

Mhamdi, E. M. E., Guerraoui, R., and Rouault, S. Distributed momentum for byzantine-resilient stochastic gradient descent. In *International Conference on Learning Representations*, 2021.

Ovi, P. R. and Gangopadhyay, A. Robust federated learning against data poisoning attacks: Prevention and detection of attacked nodes. *Electronics*, 14(15), 2025. ISSN 2079-9292.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Pillutla, K., Kakade, S. M., and Harchaoui, Z. Robust aggregation for federated learning. *IEEE Transactions on Signal Processing*, 70:1142–1154, 2022.

Rammal, A., Gruntkowska, K., Fedin, N., Gorbunov, E., and Richtarik, P. Communication compression for Byzantine robust learning: New efficient algorithms and improved rates. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 1207–1215, 2024.

Rodríguez-Barroso, N., Jiménez-López, D., Luzón, M. V., Herrera, F., and Martínez-Cámara, E. Survey on federated learning threats: Concepts, taxonomy on attacks and defences, experimental study and challenges. *Inf. Fusion*, 90:148–173, 2023. ISSN 1566-2535.

Sattler, F., Müller, K.-R., Wiegand, T., and Samek, W. On the byzantine robustness of clustered federated learning. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8861–8865, 2020.

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

Sun, Z., Kairouz, P., Suresh, A. T., and McMahan, H. B. Can you really backdoor federated learning? *arXiv preprint arXiv:1911.07963*, 2019.

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., and Jegou, H. Training data-efficient image transformers & distillation through attention. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 10347–10357, 2021.

Wan, W., Hu, S., Lu, j., Zhang, L. Y., Jin, H., and He, Y. Shielding federated learning: Robust aggregation with adaptive client selection. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pp. 753–760. International Joint Conferences on Artificial Intelligence Organization, 2022. Main Track.

Wang, H., Sreenivasan, K., Rajput, S., Vishwakarma, H., Agarwal, S., Sohn, J.-y., Lee, K., and Papailiopoulos, D. Attack of the tails: Yes, you really can backdoor federated learning. *Advances in neural information processing systems*, 33:16070–16084, 2020.

Xhemrishi, M., Östman, J., Wachter-Zeh, A., and Amat, A. G. i. Fedgt: Identification of malicious clients in federated learning with secure aggregation. *IEEE Transactions on Information Forensics and Security*, 20:2577–2592, 2025.

Xia, Q., Tao, Z., Hao, Z., and Li, Q. Faba: An algorithm for fast aggregation against byzantine attacks in distributed neural networks. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pp. 4824–4830, 2019.

Xia, Y., Jahani-Nezhad, T., and Bitar, R. Fed-DPRoC: Communication-efficient differentially private and robust federated learning. In *International Conference on Federated Learning Technologies and Applications (FLTA)*, pp. 142–149. IEEE, 2025.

Xie, C., Koyejo, O., and Gupta, I. Generalized byzantine-tolerant sgd. *ArXiv preprint arXiv:1802.10116*, 2018.

Xie, C., Koyejo, S., and Gupta, I. Zeno: Distributed stochastic gradient descent with suspicion-based fault-tolerance. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 6893–6901, 2019.

Xie, C., Koyejo, O., and Gupta, I. Fall of empires: Breaking byzantine-tolerant sgd by inner product manipulation. In *Proceedings of The 35th Uncertainty in Artificial Intelligence Conference*, volume 115, pp. 261–270, 2020.

Yan, K., Cui, S., Wuerkaixi, A., Zhang, J., Han, B., Niu, G., Sugiyama, M., and Zhang, C. Balancing similarity and complementarity for federated learning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 55739–55758, 2024.

Ye, M., Fang, X., Du, B., Yuen, P., and Tao, D. Heterogeneous federated learning: State-of-the-art and research challenges. *ACM Computing Surveys*, 56:1 – 44, 2023.

Yin, D., Chen, Y., Kannan, R., and Bartlett, P. Byzantine-robust distributed learning: Towards optimal statistical rates. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pp. 5650–5659, 2018.

Yurochkin, M., Agarwal, M., Ghosh, S., Greenewald, K., Hoang, N., and Khazaeni, Y. Bayesian nonparametric federated learning of neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, Proceedings of Machine Learning Research, pp. 7252–7261, 2019.

A Proofs

The proofs for our theoretical results use the following update properties, which follow from the assumptions **(A1)**, **(A2)**, and **(A3)**.

Proposition A.1 (Gradient variance)**.** *For every round t it holds:*

$$\mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{g}_{\mathcal{H}}^{(t)}\right\|^2\right] \leq \left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2 + \frac{\sigma^2}{|\mathcal{H}|}.$$

Proof.

$$\begin{aligned}
\mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{g}_{\mathcal{H}}^{(t)}\right\|^2\right] =& \left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2 + \mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{g}_{\mathcal{H}}^{(t)} - \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2\right] \\
=& \left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2 + \frac{1}{|\mathcal{H}|^2}\mathbb{E}_{\xi_t}\left[\left\|\sum_{k\in\mathcal{H}}\boldsymbol{g}_k^{(t)} - \nabla\mathcal{L}_k(\boldsymbol{\theta}^{(t)})\right\|^2\right] \\
=& \left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2 + \frac{1}{|\mathcal{H}|^2}\sum_{k\in\mathcal{H}}\mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{g}_k^{(t)} - \nabla\mathcal{L}_k(\boldsymbol{\theta}^{(t)})\right\|^2\right] \\
\overset{\textbf{(A3)}}{\leq}& \left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2 + \frac{\sigma^2}{|\mathcal{H}|}.
\end{aligned}$$

□

Proposition A.2 (Distance between honest gradients)**.** *Let $i, j \in \mathcal{H}$. It holds* $\mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{g}_i - \boldsymbol{g}_j\right\|^2\right] \leq 4\zeta^2 + 2\sigma^2$.

Proof. We decompose the difference between the model updates into the distance of the local gradients and the variances:

$$\begin{aligned}
\mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{g}_i - \boldsymbol{g}_j\right\|^2\right] =& \mathbb{E}_{\xi_t}\left[\left\|\nabla\mathcal{L}_i(\boldsymbol{\theta}) + (\boldsymbol{g}_i - \nabla\mathcal{L}_i(\boldsymbol{\theta})) + (\nabla\mathcal{L}_j(\boldsymbol{\theta}) - \boldsymbol{g}_j) - \nabla\mathcal{L}_j(\boldsymbol{\theta})\right\|^2\right] \\
=& \left\|\nabla\mathcal{L}_i(\boldsymbol{\theta}) - \nabla\mathcal{L}_j(\boldsymbol{\theta})\right\|^2 + \mathbb{E}_{\xi_t}\left[\left\|(\boldsymbol{g}_i - \nabla\mathcal{L}_i(\boldsymbol{\theta})) + (\nabla\mathcal{L}_j(\boldsymbol{\theta}) - \boldsymbol{g}_j)\right\|^2\right] \\
=& \left\|\nabla\mathcal{L}_i(\boldsymbol{\theta}) - \nabla\mathcal{L}_j(\boldsymbol{\theta})\right\|^2 + \mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{g}_i - \nabla\mathcal{L}_i(\boldsymbol{\theta})\right\|^2\right] + \mathbb{E}_{\xi_t}\left[\left\|\nabla\mathcal{L}_j(\boldsymbol{\theta}) - \boldsymbol{g}_j\right\|^2\right] \\
\overset{\textbf{(A3)}}{\leq}& \left\|\nabla\mathcal{L}_i(\boldsymbol{\theta}) - \nabla\mathcal{L}_j(\boldsymbol{\theta})\right\|^2 + 2\sigma^2.
\end{aligned}$$

Finally, we bound the distance between the local gradients as:

$$\left\|\nabla\mathcal{L}_i(\boldsymbol{\theta}) - \nabla\mathcal{L}_j(\boldsymbol{\theta})\right\|^2 \leq 2\left\|\nabla\mathcal{L}_i(\boldsymbol{\theta}) - \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta})\right\|^2 + 2\left\|\nabla\mathcal{L}_j(\boldsymbol{\theta}) - \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta})\right\|^2 \overset{\textbf{(A2)}}{\leq} 4\zeta^2. \quad \square$$

Proposition A.3 (Distance between honest momenta)**.** *The following bound holds for all $i, j \in \mathcal{H}$ and round t,*

$$\mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{m}_i^{(t)} - \boldsymbol{m}_j^{(t)}\right\|^2\right] \leq 6\frac{1-\beta}{1+\beta}\sigma^2 + 12\zeta^2.$$

Proof. Unrolling the momentum recursion from Equation (1), we can observe that the momentum $\boldsymbol{m}_k^{(t)}$ is linear in the gradients $\boldsymbol{g}_k^{(t)}$:

$$\boldsymbol{m}_k^{(t)} = \sum_{\tau=1}^{t}(1-\beta)\cdot\beta^{t-\tau}\boldsymbol{g}_k^{(\tau)}. \tag{3}$$

Therefore, we can decompose the momentum into three terms (using Jensen's inequality) as:

$$\begin{aligned}\left\|\boldsymbol{m}_i^{(t)}-\boldsymbol{m}_j^{(t)}\right\|^2 \leq & 3(1-\beta)^2 \underbrace{\left\|\sum_{\tau=1}^{t} \beta^{t-\tau}\left(\boldsymbol{g}_i^{(\tau)}-\nabla \mathcal{L}_i\left(\boldsymbol{\theta}^\tau\right)\right)\right\|^2}_{\triangleq A} \\ & +3(1-\beta)^2 \underbrace{\left\|\sum_{\tau=1}^{t} \beta^{t-\tau}\left(\boldsymbol{g}_j^{(\tau)}-\nabla \mathcal{L}_j\left(\boldsymbol{\theta}^\tau\right)\right)\right\|^2}_{\triangleq B} \\ & +3(1-\beta)^2 \underbrace{\left\|\sum_{\tau=1}^{t} \beta^{t-\tau}\left(\nabla \mathcal{L}_i\left(\boldsymbol{\theta}^\tau\right)-\nabla \mathcal{L}_j\left(\boldsymbol{\theta}^\tau\right)\right)\right\|^2}_{\triangleq C} .\end{aligned}$$

We can bound $\mathbb{E}[A]=\mathbb{E}[B] \leq \frac{\sigma^2}{1-\beta^2}=\frac{\sigma^2}{(1+\beta)(1-\beta)}$, c.f. Allouah et al. (2023a). Furthermore, using Jensen's inequality and the geometric sum formula, we can bound C using that:

$$\begin{aligned}\left\|\sum_{\tau=1}^{t} \beta^{t-\tau}\left(\nabla \mathcal{L}_i\left(\boldsymbol{\theta}^\tau\right)-\nabla \mathcal{L}_j\left(\boldsymbol{\theta}^\tau\right)\right)\right\|^2 & \leq\left(\sum_{\tau=1}^{t} \beta^{t-\tau}\right) \sum_{\tau=1}^{t} \beta^{t-\tau}\left\|\nabla \mathcal{L}_i\left(\boldsymbol{\theta}^\tau\right)-\nabla \mathcal{L}_j\left(\boldsymbol{\theta}^\tau\right)\right\|^2 \\ & \overset{(\mathbf{A2})}{\leq} 4 \zeta^2\left(\sum_{\tau=1}^{t} \beta^{t-\tau}\right) \leq \frac{4 \zeta^2}{(1-\beta)^2} .\end{aligned}$$

Substituting these expressions into the first bound above concludes the proof. □

Proposition A.4 (Honest gradient deviation)**.** *Let $h \in \mathcal{H}$. It holds $\mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{g}_h-\boldsymbol{g}_{\mathcal{H}}\right\|^2\right] \leq \frac{2|\mathcal{H}|+2}{|\mathcal{H}|} \sigma^2+\zeta^2$.*

Proof. We decompose the distance between $\boldsymbol{g}_h$ and $\boldsymbol{g}_{\mathcal{H}}$ as:

$$\begin{aligned}\left\|\boldsymbol{g}_h-\boldsymbol{g}_{\mathcal{H}}\right\|^2= & \left\|\boldsymbol{g}_h-\nabla \mathcal{L}_h(\boldsymbol{\theta})\right\|^2+\left\|\nabla \mathcal{L}_h(\boldsymbol{\theta})-\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta})\right\|^2+\left\|\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta})-\boldsymbol{g}_{\mathcal{H}}\right\|^2 \\ & +2\left\langle\boldsymbol{g}_h-\nabla \mathcal{L}_h(\boldsymbol{\theta}) \, , \, \nabla \mathcal{L}_h(\boldsymbol{\theta})-\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta})\right\rangle \\ & +2\left\langle\boldsymbol{g}_h-\nabla \mathcal{L}_h(\boldsymbol{\theta}) \, , \, \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta})-\boldsymbol{g}_{\mathcal{H}}\right\rangle \\ & +2\left\langle\nabla \mathcal{L}_h(\boldsymbol{\theta})-\nabla \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}) \, , \, \nabla \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta})-\boldsymbol{g}_{\mathcal{H}}\right\rangle .\end{aligned}$$

Thus, it holds that

$$\begin{aligned}& \mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{g}_h-\boldsymbol{g}_{\mathcal{H}}\right\|^2\right] \\ \overset{(\mathbf{A3})}{=} & \mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{g}_h-\nabla \mathcal{L}_h(\boldsymbol{\theta})\right\|^2\right]+\mathbb{E}_{\xi_t}\left[\left\|\nabla \mathcal{L}_h(\boldsymbol{\theta})-\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta})\right\|^2\right]+\mathbb{E}_{\xi_t}\left[\left\|\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta})-\boldsymbol{g}_{\mathcal{H}}\right\|^2\right] \\ & +2 \mathbb{E}_{\xi_t}\left[\left\langle\boldsymbol{g}_h-\nabla \mathcal{L}_h(\boldsymbol{\theta}) \, , \, \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta})-\boldsymbol{g}_{\mathcal{H}}\right\rangle\right] \\ \leq & 2 \underbrace{\mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{g}_h-\nabla \mathcal{L}_h(\boldsymbol{\theta})\right\|^2\right]}_{\leq \sigma^2 \text{ by } (\mathbf{A3})}+\underbrace{\mathbb{E}_{\xi_t}\left[\left\|\nabla \mathcal{L}_h(\boldsymbol{\theta})-\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta})\right\|^2\right]}_{\leq \zeta^2 \text{ by } (\mathbf{A2})}+2 \underbrace{\mathbb{E}_{\xi_t}\left[\left\|\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta})-\boldsymbol{g}_{\mathcal{H}}\right\|^2\right]}_{\leq \frac{\sigma^2}{|\mathcal{H}|} \text{ by } (\mathbf{A3})} \\ \leq & \frac{2|\mathcal{H}|+2}{|\mathcal{H}|} \sigma^2+\zeta^2 .\end{aligned}$$

□

Proposition A.5 (Honest momentum deviation)**.** *Let $h \in \mathcal{H}$. It holds $\mathbb{E}\left[\left\|\boldsymbol{m}_h^{(t)}-\boldsymbol{m}_{\mathcal{H}}^{(t)}\right\|^2\right] \leq 3 \frac{1-\beta}{1+\beta}\left(1+\frac{1}{N-f}\right) \sigma^2+3 \zeta^2$.*

Proof. Unrolling the momentum recursion as in Equation (3), we can write:

$$\mathbb{E}\left[\left\|\boldsymbol{m}_h^{(t)} - \boldsymbol{m}_{\mathcal{H}}^{(t)}\right\|^2\right] = (1-\beta)^2\mathbb{E}\left[\left\|\sum_{\tau=1}^{t}\beta^{t-\tau}(\boldsymbol{g}_h^{(\tau)} - \boldsymbol{g}_{\mathcal{H}}^{(\tau)})\right\|^2\right].$$

Similar to the proof of Proposition A.3, we use Jensen's inequality to decompose and bound the deviation as

$$\begin{aligned}\left\|\boldsymbol{m}_h^{(t)} - \boldsymbol{m}_{\mathcal{H}}^{(t)}\right\|^2 \leq& 3(1-\beta)^2\underbrace{\left\|\sum_{\tau=1}^{t}\beta^{t-\tau}(\boldsymbol{g}_h^{(\tau)} - \nabla\mathcal{L}_h(\boldsymbol{\theta}^\tau))\right\|^2}_{\triangleq A}\\ &+ 3(1-\beta)^2\underbrace{\left\|\sum_{\tau=1}^{t}\beta^{t-\tau}(\nabla\mathcal{L}_h(\boldsymbol{\theta}^\tau) - \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^\tau))\right\|^2}_{\triangleq B}\\ &+ 3(1-\beta)^2\underbrace{\left\|\sum_{\tau=1}^{t}\beta^{t-\tau}(\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^\tau) - \boldsymbol{g}_{\mathcal{H}}^{(\tau)})\right\|^2}_{\triangleq C}.\end{aligned}$$

First, we can bound $\mathbb{E}[A] \leq \frac{\sigma^2}{1-\beta^2} = \frac{\sigma^2}{(1+\beta)(1-\beta)}$ and $\mathbb{E}[C] \leq \frac{1}{N-f}\frac{\sigma^2}{(1+\beta)(1-\beta)}$, c.f. Allouah et al. (2023a). Furthermore, using Jensen's inequality and the geometric sum formula, we can bound B using that:

$$\begin{aligned}&\left\|\sum_{\tau=1}^{t}\beta^{t-\tau}(\nabla\mathcal{L}_h(\boldsymbol{\theta}^\tau) - \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^\tau))\right\|^2\\ &\leq \left(\sum_{\tau=1}^{t}\beta^{t-\tau}\right)\sum_{\tau=1}^{t}\beta^{t-\tau}\|\nabla\mathcal{L}_h(\boldsymbol{\theta}^\tau) - \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^\tau)\|^2\\ &\overset{\textbf{(A2)}}{\leq} \zeta^2\left(\sum_{\tau=1}^{t}\beta^{t-\tau}\right)^2 \leq \frac{\zeta^2}{(1-\beta)^2}.\end{aligned}$$

Substituting these expressions into the first bound above yields that $\mathbb{E}\left[\left\|\boldsymbol{m}_h^{(t)} - \boldsymbol{m}_{\mathcal{H}}^{(t)}\right\|^2\right] \leq 3\frac{1-\beta}{1+\beta}\left(1 + \frac{1}{N-f}\right)\sigma^2 + 3\zeta^2$. □

A.1 Proof of Lemma 3.1

We show that the aggregate $\hat{\boldsymbol{m}} = \sum_{k\in[N]} c_k\boldsymbol{m}_k$ output by `GatScHA` has a bounded bias $\boldsymbol{\delta} \triangleq \hat{\boldsymbol{m}} - \boldsymbol{m}_{\mathcal{H}}$. In the following, it is supposed w.l.o.g. that the clients are indexed in descending order sorted according to their gather score $\omega_{\texttt{Gat}}(k)$.

We identify each update $\boldsymbol{m}_k$ with its closest honest update $\boldsymbol{m}_{h_k}$, $h_k \in \mathcal{H}$, and define the weighted average of the identified honest clients as $\tilde{\boldsymbol{m}} \triangleq \sum_{k\in[N]} c_k\boldsymbol{m}_{h_k}$. Then, we decompose the bias into two terms, which we can bound separately:

$$\begin{aligned}\|\boldsymbol{\delta}\|^2 =& \|\hat{\boldsymbol{m}} - \tilde{\boldsymbol{m}} + \tilde{\boldsymbol{m}} - \boldsymbol{m}_{\mathcal{H}}\|^2\\ \leq& 2\|\hat{\boldsymbol{m}} - \tilde{\boldsymbol{m}}\|^2 + 2\|\tilde{\boldsymbol{m}} - \boldsymbol{m}_{\mathcal{H}}\|^2. \qquad (4)\end{aligned}$$

Bounding $\|\hat{\boldsymbol{m}} - \tilde{\boldsymbol{m}}\|^2$: Since $\boldsymbol{m}_k - \boldsymbol{m}_{h_k} = 0$ for $k \in \mathcal{H}$, the term $\|\hat{\boldsymbol{m}} - \tilde{\boldsymbol{m}}\|$ is determined only by the malicious clients. Moreover, clients with a small gather score are removed, i.e.,

$c_k = 0$ for $k \notin [N-f]$. We can use Jensen's inequality to bound

$$\begin{aligned}
\|\hat{\boldsymbol{m}} - \tilde{\boldsymbol{m}}\|^2 &= \left\| \sum_{k\in[N]} c_k(\boldsymbol{m}_k - \boldsymbol{m}_{h_k}) \right\|^2 = \left\| \sum_{k\in[N-f]\setminus\mathcal{H}} c_k(\boldsymbol{m}_k - \boldsymbol{m}_{h_k}) \right\|^2 \\
&= \left(\sum_{i\in[N-f]\setminus\mathcal{H}} c_i \right)^2 \left\| \sum_{k\in[N-f]\setminus\mathcal{H}} \frac{c_k}{\sum_{i\in[N-f]\setminus\mathcal{H}} c_i} (\boldsymbol{m}_k - \boldsymbol{m}_{h_k}) \right\|^2 \\
&\leq \left(\sum_{k\in[N-f]\setminus\mathcal{H}} c_k \right) \sum_{k\in[N-f]\setminus\mathcal{H}} c_k \|\boldsymbol{m}_k - \boldsymbol{m}_{h_k}\|^2 \qquad (5) \\
&= \left(1 - \sum_{k\in[N-f]\cap\mathcal{H}} c_k \right) \cdot \sum_{k\in[N-f]\setminus\mathcal{H}} c_k \|\boldsymbol{m}_k - \boldsymbol{m}_{h_k}\|^2. \qquad (6)
\end{aligned}$$

First: if all malicious clients receive the lowest gather score, then $\|\tilde{\boldsymbol{m}} - \hat{\boldsymbol{m}}\|^2 = 0$ trivially. Otherwise, there is at least one client $i \in \mathcal{H}$ with $i > N - f$, that is, for which $c_i = 0$ (this honest client is removed from the aggregation due to its gather score). Note that all clients $k \in [N-f]$, have a larger gather score than i. When the gather score is $\omega_{\texttt{Gat}}(k) = 1/\delta_{k,k_R}$, this implies:

$$\|\boldsymbol{m}_k - \boldsymbol{m}_{k,R}\|^2 \leq \|\boldsymbol{m}_i - \boldsymbol{m}_{i,R}\|^2. \qquad (7)$$

By definition of $\boldsymbol{m}_{i,R}$, there are $N - 1 - (R-1) = N - R$ updates other than $\boldsymbol{m}_i$ with a Euclidean distance to $\boldsymbol{m}_i$ larger than or equal to the one of $\boldsymbol{m}_{i,R}$. When $N - R > f$, i.e., $R < N - f$, but there are at most f malicious clients, then there is an honest client $j \in \mathcal{H}$ in particular with

$$\|\boldsymbol{m}_i - \boldsymbol{m}_{i,R}\|^2 \leq \|\boldsymbol{m}_i - \boldsymbol{m}_j\|^2. \qquad (8)$$

Second: for every malicious client k, there can be at most $f-1$ other malicious clients among its R neighbors. Thus, when $R > f - 1$, there is at least one honest client $h_k \in \mathcal{H}$ among the R neighbors (the same h_k as defined in $\tilde{\boldsymbol{m}}$ above).

In summary: when R is such that $f - 1 < R < N - f$, then for every malicious k there exists an honest h_k with[9]

$$\|\boldsymbol{m}_k - \boldsymbol{m}_{h_k}\|^2 \leq \|\boldsymbol{m}_k - \boldsymbol{m}_{k,R}\|^2 \overset{(7,8)}{\leq} \|\boldsymbol{m}_i - \boldsymbol{m}_j\|^2. \qquad (9)$$

Such a value R exists if $f < N - f$. We choose the largest possible value $R = N - f - 1$ in our algorithm. Substituting this result into Equation (6) gives

$$\begin{aligned}
\|\tilde{\boldsymbol{m}} - \hat{\boldsymbol{m}}\|^2 &\leq \left(1 - \sum_{k\in[N-f]\cap\mathcal{H}} c_k \right) \sum_{k\in[N-f]\setminus\mathcal{H}} c_k \|\boldsymbol{m}_i - \boldsymbol{m}_j\|^2 \\
&\leq \left(1 - \sum_{k\in[N-f]\cap\mathcal{H}} c_k \right)^2 \|\boldsymbol{m}_i - \boldsymbol{m}_j\|^2 \leq \|\boldsymbol{m}_i - \boldsymbol{m}_j\|^2. \qquad (10)
\end{aligned}$$

where we use $\sum_{k\in[N-f]\setminus\mathcal{H}} c_k \leq \sum_{k\in[N]} c_k = 1$. Finally, we can employ Proposition A.3 to bound the expectation $\mathbb{E}\left[\|\boldsymbol{m}_i - \boldsymbol{m}_j\|^2\right] \leq 6\frac{1-\beta}{1+\beta}\sigma^2 + 12\zeta^2$.

Bounding $\|\tilde{\boldsymbol{m}} - \boldsymbol{m}_{\mathcal{H}}\|^2$: The honest mixture $\tilde{\boldsymbol{m}}$ is a weighted average of benign updates. Without further consideration of the exact weights, the distance between $\tilde{\boldsymbol{m}}$ and the benign

[9]Note that h_k depends on k here. However, i and j are independent of the choice for k, and hence, the resulting upper bound is independent of k.

mean $\boldsymbol{m}_{\mathcal{H}}$ is dominated by the most deviated benign update:

$$\begin{aligned}\|\tilde{\boldsymbol{m}} - \boldsymbol{m}_{\mathcal{H}}\|^2 =& \left\| \sum_{k\in[N]} c_k(\boldsymbol{m}_{h_k} - \boldsymbol{m}_{\mathcal{H}}) \right\|^2 = \left\| \sum_{k\in[N-f]} c_k(\boldsymbol{m}_{h_k} - \boldsymbol{m}_{\mathcal{H}}) \right\|^2 \\ \leq& \sum_{k\in[N-f]} c_k \|\boldsymbol{m}_{h_k} - \boldsymbol{m}_{\mathcal{H}}\|^2.\end{aligned}$$

To bound the expectation $\mathbb{E}\left[\sum_{k\in[N-f]} c_k \left\|\boldsymbol{m}_{h_k}^{(t)} - \boldsymbol{m}_{\mathcal{H}}^{(t)}\right\|^2\right]$, we unroll the momentum recursion as in Equation (3) and write for every $h_k \in \mathcal{H}$:

$$\left\|\boldsymbol{m}_{h_k}^{(t)} - \boldsymbol{m}_{\mathcal{H}}^{(t)}\right\|^2 = (1-\beta)^2 \left\| \sum_{\tau=1}^{t} \beta^{t-\tau} (\boldsymbol{g}_{h_k}^{(\tau)} - \boldsymbol{g}_{\mathcal{H}}^{(\tau)}) \right\|^2.$$

Similar to the proof of Proposition A.5, we use Jensen's inequality to decompose and bound the deviation as

$$\begin{aligned}\left\|\boldsymbol{m}_{h_k}^{(t)} - \boldsymbol{m}_{\mathcal{H}}^{(t)}\right\|^2 \leq& 3(1-\beta)^2 \underbrace{\left\| \sum_{\tau=1}^{t} \beta^{t-\tau} (\boldsymbol{g}_{h_k}^{(\tau)} - \nabla\mathcal{L}_{h_k}(\boldsymbol{\theta}^\tau)) \right\|^2}_{\triangleq A} \\ &+ 3(1-\beta)^2 \underbrace{\left\| \sum_{\tau=1}^{t} \beta^{t-\tau} (\nabla\mathcal{L}_{h_k}(\boldsymbol{\theta}^\tau) - \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^\tau)) \right\|^2}_{\triangleq B} \\ &+ 3(1-\beta)^2 \underbrace{\left\| \sum_{\tau=1}^{t} \beta^{t-\tau} (\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^\tau) - \boldsymbol{g}_{\mathcal{H}}^{(\tau)}) \right\|^2}_{\triangleq C}.\end{aligned}$$

Using Jensen's inequality and the geometric sum formula, we can bound B using that:

$$\begin{aligned}&\left\| \sum_{\tau=1}^{t} \beta^{t-\tau} (\nabla\mathcal{L}_{h_k}(\boldsymbol{\theta}^\tau) - \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^\tau)) \right\|^2 \\ &\leq \left(\sum_{\tau=1}^{t} \beta^{t-\tau} \right) \sum_{\tau=1}^{t} \beta^{t-\tau} \|\nabla\mathcal{L}_{h_k}(\boldsymbol{\theta}^\tau) - \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^\tau)\|^2 \\ &\overset{\textbf{(A2)}}{\leq} \zeta^2 \left(\sum_{\tau=1}^{t} \beta^{t-\tau} \right) \leq \frac{\zeta^2}{(1-\beta)^2}.\end{aligned}$$

Substituting these expressions into the original term yields that

$$\mathbb{E}\left[\sum_{k\in[N-f]} c_k \left\|\boldsymbol{m}_{h_k}^{(t)} - \boldsymbol{m}_{\mathcal{H}}^{(t)}\right\|^2 \right] \tag{11}$$

$$\leq 3(1-\beta)^2 \left(\mathbb{E}\left[\sum_{k\in[N-f]} c_k A + C \right] + \frac{\zeta^2}{(1-\beta)^2} \right) \tag{12}$$

$$\leq 3(1-\beta)^2 \sum_{k\in[N-f]} \mathbb{E}\left[A + C\right] + 3\zeta^2 \tag{13}$$

$$\leq 3(N-f+1)\frac{1-\beta}{1+\beta}\sigma^2 + 3\zeta^2. \tag{14}$$

The last inequality holds since $\mathbb{E}[A] \leq \frac{\sigma^2}{1-\beta^2} = \frac{\sigma^2}{(1+\beta)(1-\beta)}$ and $\mathbb{E}[C] \leq \frac{1}{N-f}\frac{\sigma^2}{(1+\beta)(1-\beta)}$, c.f. Allouah et al. (2023a).

To conclude the proof, we substitute the bounds into Equation (4) as follows:

$$\begin{aligned}\mathbb{E}\left[\|\boldsymbol{\delta}\|^2\right] \leq & 2\mathbb{E}\left[\|\boldsymbol{m}_i - \boldsymbol{m}_j\|^2\right] + 2\mathbb{E}\left[\|\tilde{\boldsymbol{m}} - \boldsymbol{m}_{\mathcal{H}}\|^2\right] \\ \leq & 30\zeta^2 + (12 + 6(N-f+1))\frac{1-\beta}{1+\beta}\sigma^2.\end{aligned} \tag{15}$$

Remark A.6 (Tightening the bound for $\sigma^2 = \beta = 0$). Without momentum ($\beta = 0$), $\boldsymbol{m}_k = \boldsymbol{g}_k$. In this case, we can use Proposition A.2) to obtain a tighter bound $\mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{g}_i - \boldsymbol{g}_j\right\|^2\right] \leq 4\zeta^2 + 2\sigma^2$. Furthermore, if $\sigma^2 = 0$, we can bound the distance between the honest mixture $\tilde{\boldsymbol{g}}$ and the benign mean $\boldsymbol{g}_{\mathcal{H}}$ as

$$\begin{aligned}\|\tilde{\boldsymbol{g}} - \boldsymbol{g}_{\mathcal{H}}\|^2 = & \left\|\sum_{k\in[N]} c_k(\boldsymbol{g}_{h_k} - \boldsymbol{g}_{\mathcal{H}})\right\|^2 \leq \sum_{k\in[N]} c_k\left\|\boldsymbol{g}_{h_k} - \boldsymbol{g}_{\mathcal{H}}\right\|^2 \\ = & \sum_{k\in[N]} c_k\|\nabla\mathcal{L}_{h_k}(\boldsymbol{\theta}) - \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta})\|^2 \overset{\textbf{(A2)}}{\leq} \zeta^2.\end{aligned}$$

Substituting these bounds into Equation (15) yields $\left\|\hat{\boldsymbol{g}}^{(t)} - \boldsymbol{g}_{\mathcal{H}}^{(t)}\right\|^2 \leq 10\zeta^2$.

A.2 Proof of Corollary 3.4

Suppose w.l.o.g. the $N-f$ aggregated updates $\{k_j,\, j = 1, 2, \ldots, N-f\}$, correspond to $[N-f]$:

$$c_k = \begin{cases} \frac{1}{N-f}, & k \in [N-f] \\ 0, & \text{otherwise.}\end{cases}$$

Then, more concretely, we have

$$\begin{aligned}\|\hat{\boldsymbol{m}} - \tilde{\boldsymbol{m}}\|^2 = & \left\|\sum_{k\in[N-f]} \frac{1}{N-f}(\boldsymbol{m}_k - \boldsymbol{m}_{h_k})\right\|^2 \\ \leq & \frac{1}{N-f}\sum_{k\in[N-f]}\|\boldsymbol{m}_k - \boldsymbol{m}_{h_k}\|^2 = \frac{1}{N-f}\sum_{k\in[N-f]\setminus\mathcal{H}}\|\boldsymbol{m}_k - \boldsymbol{m}_{h_k}\|^2 \\ \overset{(9)}{\leq} & \frac{1}{N-f}\sum_{k\in[N-f]\setminus\mathcal{H}}\|\boldsymbol{m}_i - \boldsymbol{m}_j\|^2,\end{aligned}$$

and therefore, we can use Proposition A.3 to bound the expectation as

$$\begin{aligned}\mathbb{E}\left[\|\hat{\boldsymbol{m}} - \tilde{\boldsymbol{m}}\|^2\right] \leq & \frac{|[N-f]\setminus\mathcal{H}|}{N-f}\left(6\frac{1-\beta}{1+\beta}\sigma^2 + 12\zeta^2\right) \\ \leq & \frac{f}{N-f}\left(6\frac{1-\beta}{1+\beta}\sigma^2 + 12\zeta^2\right).\end{aligned}$$

Next, we consider the distance

$$\|\tilde{\boldsymbol{m}} - \boldsymbol{m}_{\mathcal{H}}\|^2 = \left\|\sum_{k\in[N]} c_k(\boldsymbol{m}_{h_k} - \boldsymbol{m}_{\mathcal{H}})\right\|^2 = \left\|\sum_{k\in[N-f]} \frac{1}{N-f}(\boldsymbol{m}_{h_k} - \boldsymbol{m}_{\mathcal{H}})\right\|^2$$

and distinguish two cases as follows.

If $[N-f] \setminus \mathcal{H} = \emptyset$**,** then

$$\begin{aligned}
&\left\| \sum_{k\in[N-f]} \frac{1}{N-f}(\boldsymbol{m}_{h_k} - \boldsymbol{m}_{\mathcal{H}}) \right\|^2 = \frac{|[N-f]\cap\mathcal{H}|}{N-f} \left\| \boldsymbol{m}_{[N-f]\cap\mathcal{H}} - \boldsymbol{m}_{\mathcal{H}} \right\|^2 \\
=&\frac{|[N-f]\cap\mathcal{H}|}{N-f} \left\| \boldsymbol{m}_{[N-f]\cap\mathcal{H}} - \boldsymbol{m}_{\mathcal{H}} \right\|^2 + \frac{1}{N-f} \sum_{k\in[N-f]\setminus\mathcal{H}} \left\| \boldsymbol{m}_{h_k} - \boldsymbol{m}_{\mathcal{H}} \right\|^2.
\end{aligned}$$

Otherwise, if $[N-f] \setminus \mathcal{H} \neq \emptyset$**,** we can bound

$$\begin{aligned}
&\left\| \sum_{k\in[N-f]} \frac{1}{N-f}(\boldsymbol{m}_{h_k} - \boldsymbol{m}_{\mathcal{H}}) \right\|^2 \\
\leq&\frac{|[N-f]\cap\mathcal{H}|}{N-f} \left\| \frac{1}{|[N-f]\cap\mathcal{H}|} \sum_{h\in[N-f]\cap\mathcal{H}} (\boldsymbol{m}_h - \boldsymbol{m}_{\mathcal{H}}) \right\|^2 \\
&+ \frac{|[N-f]\setminus\mathcal{H}|}{N-f} \left\| \frac{1}{|[N-f]\setminus\mathcal{H}|} \sum_{k\in[N-f]\setminus\mathcal{H}} (\boldsymbol{m}_{h_k} - \boldsymbol{m}_{\mathcal{H}}) \right\|^2 \\
\leq&\frac{|[N-f]\cap\mathcal{H}|}{N-f} \left\| \boldsymbol{m}_{[N-f]\cap\mathcal{H}} - \boldsymbol{m}_{\mathcal{H}} \right\|^2 + \frac{1}{N-f} \sum_{k\in[N-f]\setminus\mathcal{H}} \left\| \boldsymbol{m}_{h_k} - \boldsymbol{m}_{\mathcal{H}} \right\|^2.
\end{aligned}$$

For both cases, we can bound

$$\begin{aligned}
&\frac{|[N-f]\cap\mathcal{H}|}{N-f} \left\| \boldsymbol{m}_{[N-f]\cap\mathcal{H}} - \boldsymbol{m}_{\mathcal{H}} \right\|^2 \\
\overset{(a)}{=}&\frac{|[N-f]\cap\mathcal{H}|}{N-f} \frac{|\mathcal{H}\setminus[N-f]|^2}{|\mathcal{H}|^2} \left\| \boldsymbol{m}_{[N-f]\cap\mathcal{H}} - \boldsymbol{m}_{\mathcal{H}\setminus[N-f]} \right\|^2 \\
\overset{(b)}{\leq}&\frac{f}{N-f}\frac{1}{4} \left\| \boldsymbol{m}_{[N-f]\cap\mathcal{H}} - \boldsymbol{m}_{\mathcal{H}\setminus[N-f]} \right\|^2 \\
\leq&\frac{1}{2}\frac{f}{N-f} \left(\left\| \boldsymbol{m}_{[N-f]\cap\mathcal{H}} - \boldsymbol{m}_{\mathcal{H}} \right\|^2 + \left\| \boldsymbol{m}_{\mathcal{H}} - \boldsymbol{m}_{\mathcal{H}\setminus[N-f]} \right\|^2 \right) \\
\leq&\frac{1}{2}\frac{f}{N-f}\frac{1}{|[N-f]\cap\mathcal{H}|} \sum_{h\in[N-f]\cap\mathcal{H}} \left\| \boldsymbol{m}_h - \boldsymbol{m}_{\mathcal{H}} \right\|^2 \\
&+ \frac{1}{2}\frac{f}{N-f}\frac{1}{|\mathcal{H}\setminus[N-f]|} \sum_{h\in\mathcal{H}\setminus[N-f]} \left\| \boldsymbol{m}_h - \boldsymbol{m}_{\mathcal{H}} \right\|^2.
\end{aligned}$$

In expectation, we obtain

$$\begin{aligned}
\mathbb{E}\left[\left\| \tilde{\boldsymbol{m}} - \boldsymbol{m}_{\mathcal{H}} \right\|^2 \right] \leq&\frac{1}{2}\frac{f}{N-f}\frac{1}{|[N-f]\cap\mathcal{H}|} \sum_{h\in[N-f]\cap\mathcal{H}} \mathbb{E}\left[\left\| \boldsymbol{m}_h - \boldsymbol{m}_{\mathcal{H}} \right\|^2 \right] \\
&+ \frac{1}{2}\frac{f}{N-f}\frac{1}{|\mathcal{H}\setminus[N-f]|} \sum_{h\in\mathcal{H}\setminus[N-f]} \mathbb{E}\left[\left\| \boldsymbol{m}_h - \boldsymbol{m}_{\mathcal{H}} \right\|^2 \right] \\
&+ \frac{1}{N-f} \sum_{k\in[N-f]\setminus\mathcal{H}} \mathbb{E}\left[\left\| \boldsymbol{m}_{h_k} - \boldsymbol{m}_{\mathcal{H}} \right\|^2 \right] \\
\overset{\text{Proposition A.5}}{\leq}& 6\frac{f}{N-f} \left(\frac{1-\beta}{1+\beta} \left(1 + \frac{1}{N-f} \right) \sigma^2 + \zeta^2 \right).
\end{aligned}$$

In the derivations above we obtain (b) by noting that $|\mathcal{H} \setminus [N-f]| \leq f$ and applying the AM-GM inequality to:

$$|[N-f] \cap \mathcal{H}| \cdot |\mathcal{H} \setminus [N-f]| = \left(\sqrt{|[N-f] \cap \mathcal{H}| \cdot |\mathcal{H} \setminus [N-f]|}\right)^2$$
$$\leq \left(\frac{1}{2}(|[N-f] \cap \mathcal{H}| + |\mathcal{H} \setminus [N-f]|)\right)^2 = \frac{1}{4}|\mathcal{H}|^2.$$

Furthermore, the equality (a) holds since

$$\begin{aligned}
\boldsymbol{m}_{\mathcal{H}} - \boldsymbol{m}_{[N-f]\cap\mathcal{H}} =& \frac{1}{|\mathcal{H}|}\left(\sum_{h\in\mathcal{H}\cap[N-f]}(\boldsymbol{m}_h - \boldsymbol{m}_{[N-f]\cap\mathcal{H}}) + \sum_{h\in\mathcal{H}\setminus[N-f]}(\boldsymbol{m}_h - \boldsymbol{m}_{[N-f]\cap\mathcal{H}})\right) \\
=& \frac{1}{|\mathcal{H}|}\sum_{h\in\mathcal{H}\setminus[N-f]}(\boldsymbol{m}_h - \boldsymbol{m}_{[N-f]\cap\mathcal{H}}) \\
=& \frac{|\mathcal{H}\setminus[N-f]|}{|\mathcal{H}|}\frac{1}{|\mathcal{H}\setminus[N-f]|}\sum_{h\in\mathcal{H}\setminus[N-f]}(\boldsymbol{m}_h - \boldsymbol{m}_{[N-f]\cap\mathcal{H}}) \\
=& \frac{|\mathcal{H}\setminus[N-f]|}{|\mathcal{H}|}(\boldsymbol{m}_{\mathcal{H}\setminus[N-f]} - \boldsymbol{m}_{[N-f]\cap\mathcal{H}}).
\end{aligned}$$

Combining the terms as in Equation (4), we can finally bound

$$\begin{aligned}
\mathbb{E}\left[\|\hat{\boldsymbol{m}} - \boldsymbol{m}_{\mathcal{H}}\|^2\right] \leq& \frac{f}{N-f}2\left(18\zeta^2 + 6\left(2 + \frac{1}{N-f}\right)\frac{1-\beta}{1+\beta}\sigma^2\right) \\
\leq& \frac{f}{N-f}36\left(\zeta^2 + \frac{1-\beta}{1+\beta}\sigma^2\right).
\end{aligned}$$

A.3 Proof of Theorem 3.2

The proof follows along the lines of standard arguments, cf. (Bottou et al., 2018; Guerraoui et al., 2025). Suppose the model is iteratively updated according to the rule $\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} - \gamma_t \hat{\boldsymbol{g}}^{(t)}$ with $\hat{\boldsymbol{g}}^{(t)} = \texttt{GatScHA}(\boldsymbol{g}_1^{(t)}, \ldots, \boldsymbol{g}_N^{(t)})$. The smoothness of the loss function implies that

$$\mathbb{E}_{\xi_t}\left[\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t+1)})\right] - \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)}) \leq -\gamma_t \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})^{\mathsf{T}}\mathbb{E}_{\xi_t}\left[\hat{\boldsymbol{g}}^{(t)}\right] + \gamma_t^2\frac{L}{2}\mathbb{E}_{\xi_t}\left[\left\|\hat{\boldsymbol{g}}^{(t)}\right\|^2\right]$$

Defining $\boldsymbol{\delta}^{(t)} \triangleq \hat{\boldsymbol{g}}^{(t)} - \boldsymbol{g}_{\mathcal{H}}^{(t)}$, we can write $\hat{\boldsymbol{g}}^{(t)} = \boldsymbol{g}_{\mathcal{H}}^{(t)} + \boldsymbol{\delta}^{(t)}$ and bound $\left\|\hat{\boldsymbol{g}}^{(t)}\right\|^2 \leq 2\left\|\boldsymbol{g}_{\mathcal{H}}^{(t)}\right\|^2 + 2\left\|\boldsymbol{\delta}^{(t)}\right\|^2$ by Jensen's inequality. Thus, we obtain

$$\begin{aligned}
\mathbb{E}_{\xi_t}\left[\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t+1)})\right] - \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)}) \leq& -\gamma_t \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})^{\mathsf{T}}\mathbb{E}_{\xi_t}\left[\boldsymbol{g}_{\mathcal{H}}^{(t)}\right] + \gamma_t^2 L\mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{g}_{\mathcal{H}}^{(t)}\right\|^2\right] \\
&- \gamma_t \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})^{\mathsf{T}}\mathbb{E}_{\xi_t}\left[\boldsymbol{\delta}^{(t)}\right] + \gamma_t^2 L\mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{\delta}^{(t)}\right\|^2\right] \\
\leq& -\gamma_t\left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2 + \gamma_t^2 L(\left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2 + \frac{\sigma^2}{|\mathcal{H}|}) \\
&- \gamma_t \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})^{\mathsf{T}}\mathbb{E}_{\xi_t}\left[\boldsymbol{\delta}^{(t)}\right] + \gamma_t^2 L\mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{\delta}^{(t)}\right\|^2\right]
\end{aligned}$$

where we use **(A3)** and Proposition A.1 in the inequality. Using that

$$-\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})^{\mathsf{T}}\boldsymbol{\delta}^{(t)} \leq \frac{1}{2}\left(\left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2 + \left\|\boldsymbol{\delta}^{(t)}\right\|^2\right),$$

the former term can be further bounded by

$$\begin{aligned}
&\mathbb{E}_{\xi_t}\left[\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t+1)})\right] - \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)}) \\
\leq& -\frac{\gamma_t}{2}(1-2\gamma_t L)\left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2 + \gamma_t^2\frac{L\sigma^2}{|\mathcal{H}|} + \frac{\gamma_t}{2}\left(1+2\gamma_t L\right)\mathbb{E}_{\xi_t}\left[\left\|\boldsymbol{\delta}^{(t)}\right\|^2\right] \\
\leq& -\frac{\gamma_t}{2}(1-2\gamma_t L)\left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2 + \gamma_t^2\frac{L\sigma^2}{|\mathcal{H}|} + \gamma_t\left(1+2\gamma_t L\right)5\zeta^2,
\end{aligned}$$

where the latter follows from Remark A.6. Thus, we can conclude that for a learning rate $\gamma_t = \gamma$ it holds:

$$\begin{aligned}
\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^\star) - \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(1)}) \leq& \mathbb{E}\left[\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(T+1)})\right] - \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(1)}) \\
\leq& \sum_{t=1}^{T}\left(-\frac{\gamma_t}{2}(1-2\gamma_t L)\mathbb{E}\left[\left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2\right] + \gamma_t^2\frac{L\sigma^2}{|\mathcal{H}|} + \gamma_t\left(1+2\gamma_t L\right)5\zeta^2\right) \\
=& -\frac{\gamma}{2}(1-2\gamma L)\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2\right] + T\left(\gamma^2\frac{L\sigma^2}{|\mathcal{H}|} + \gamma\left(1+2\gamma L\right)5\zeta^2\right),
\end{aligned}$$

which can be reformulated as:

$$\begin{aligned}
\implies \frac{1-2\gamma L}{T}\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2\right] \leq& \frac{2(\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(1)}) - \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^\star))}{\gamma T} + \gamma\frac{2L\sigma^2}{|\mathcal{H}|} + 10\left(1+2\gamma L\right)\zeta^2 \\
=& \frac{2(\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(1)}) - \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^\star))}{\gamma T} + \gamma\frac{2L\sigma^2}{|\mathcal{H}|} + 10\left(1+2\gamma L\right)\zeta^2
\end{aligned}$$

For a sufficiently small learning rate $0 < \gamma < \frac{1}{4L}$ and $\sigma^2 = 0$ we can remove the expectation and finally obtain

$$\frac{1}{T}\sum_{t=1}^{T}\left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2 \leq \frac{4(\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(1)}) - \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^\star))}{\gamma T} + 20\left(1+2\gamma L\right)\zeta^2.$$

A.4 Proof of Theorem 3.3

Let $\boldsymbol{m}_{\mathcal{H}}^{(t)} \triangleq \frac{1}{|\mathcal{H}|}\sum_{k\in\mathcal{H}}\boldsymbol{m}_k^{(t)}$ denote the average honest momentum. In the following, we use the moment deviation defined as

$$\boldsymbol{\xi}^{(t)} \triangleq \boldsymbol{m}_{\mathcal{H}}^{(t)} - \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)}),$$

and we define the aggregation error (bias) in the case of momentum:

$$\boldsymbol{\delta}^{(t)} \triangleq \hat{\boldsymbol{m}}^{(t)} - \boldsymbol{m}_{\mathcal{H}}^{(t)}.$$

Noting that (**A2**) implies

$$\frac{1}{|\mathcal{H}|}\sum_{k\in\mathcal{H}}\left\|\nabla\mathcal{L}_k(\boldsymbol{\theta}) - \nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta})\right\|^2 \leq \zeta^2, \tag{16}$$

we can re-iterate through the steps from Allouah et al. (2023a) or Farhadkhani et al. (2022), respectively, to proof the convergence of the FL algorithm with local momentum using `GatScHA`. The authors show that under (**A3**) and Equation (16) it holds that (note the

different indexing of models $\boldsymbol{\theta}^{(t)}$ in (Allouah et al., 2023a))

$$\begin{aligned}
\mathbb{E}\left[\frac{1}{|\mathcal{H}|}\sum_{k\in\mathcal{H}}\left\|\boldsymbol{m}_k^{(t)}-\boldsymbol{m}_{\mathcal{H}}^{(t)}\right\|^2\right] \leq & 3\cdot\frac{1-\beta}{1+\beta}\cdot\left(1+\frac{1}{N-f}\right)\sigma^2+3\zeta^2,\\
\text{For } t\geq 2: \mathbb{E}\left[\left\|\boldsymbol{\xi}^{(t)}\right\|^2\right] \leq & \beta^2 c\mathbb{E}\left[\left\|\boldsymbol{\xi}^{(t-1)}\right\|^2\right]+4\gamma L(1+\gamma L)\beta^2\mathbb{E}\left[\left\|\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t-1)})\right\|^2\right]\\
& +(1-\beta)^2\frac{\sigma^2}{N-f}+2\gamma L(1+\gamma L)\beta^2\mathbb{E}\left[\left\|\boldsymbol{\delta}^{(t-1)}\right\|^2\right]\\
\mathbb{E}\left[\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})-\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t-1)})\right] \leq & -\gamma(1-4\gamma L)\mathbb{E}\left[\left\|\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t-1)})\right\|^2\right]\\
& +2\gamma(1+2\gamma L)\mathbb{E}\left[\left\|\boldsymbol{\xi}^{(t)}\right\|^2\right]+2\gamma(1+\gamma L)\mathbb{E}\left[\left\|\boldsymbol{\delta}^{(t)}\right\|^2\right],
\end{aligned}$$

with $c\triangleq(1+\gamma L)(1+4\gamma L)$.

Suppose that, for a constant $B\geq 0$, the aggregation error is bounded as $\mathbb{E}\left[\left\|\boldsymbol{\delta}^{(t)}\right\|^2\right]\leq B$. Then, it has been shown (Allouah et al., 2023a) that for a properly tuned learning rate $\gamma\leq\frac{1}{24L}$ and momentum $\beta=\sqrt{1-24\gamma L}$:

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2\right]\leq\frac{9\left(\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(1)})-\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{\star})\right)}{\gamma T}+\frac{\gamma}{T}\frac{288L\sigma^2}{N-f}+\gamma\frac{288L\sigma^2}{N-f}+12B.$$

To determine the constant B, we recall Lemma 3.1 to conclude that, using `GatScHA`, the aggregation error is bounded as

$$\mathbb{E}\left[\left\|\boldsymbol{\delta}^{(t)}\right\|^2\right]\leq 30\zeta^2+(12+6(N-f+1))\frac{1-\beta}{1+\beta}\sigma^2.$$

In summary, we use

$$\begin{aligned}
B=&(12+4(N-f))\frac{1-\beta}{1+\beta}\sigma^2+28\zeta^2\\
\leq&(12+6(N-f+1))(1-\beta^2)\sigma^2+30\zeta^2\\
=&(12+6(N-f+1))24\gamma L\sigma^2+30\zeta^2.
\end{aligned}$$

Therefore, we reach at the bound

$$\begin{aligned}
\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2\right]\leq&\frac{9\left(\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(1)})-\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{\star})\right)}{\gamma T}+\frac{\gamma}{T}\frac{288L\sigma^2}{N-f}\\
&+\gamma L\sigma^2 288\left(\frac{1}{N-f}+(12+6(N-f+1))\right)+360\zeta^2.
\end{aligned}$$

Choosing the learning rate as $\gamma \triangleq \min\{\frac{1}{24L}, \frac{C}{\sqrt{T}}\}$ for a constant $C \geq 0$, we can bound $\frac{1}{\gamma} \leq \max\{24L, \frac{\sqrt{T}}{C}\} \leq 24L + \frac{\sqrt{T}}{C}$. Therefore, we can conclude that

$$
\begin{aligned}
&\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(t)})\right\|^2\right] \\
\leq&\frac{216L\left(\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(1)}) - \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{\star})\right)}{T} + \frac{9\left(\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(1)}) - \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{\star})\right)}{\sqrt{T}C} \\
&+ \frac{1}{T^{3/2}}\frac{288L\sigma^2C}{N-f} + \frac{1}{\sqrt{T}}288L\sigma^2C\left(\frac{1}{N-f} + 12 + 6(N-f+1)\right) + 360\zeta^2 \\
\leq&\frac{1}{T^{3/2}}\frac{288L\sigma^2C}{N-f} + \frac{216L\left(\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(1)}) - \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{\star})\right)}{T} \\
&+ \frac{9\left(\mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{(1)}) - \mathcal{L}_{\mathcal{H}}(\boldsymbol{\theta}^{\star})\right) + 288L\sigma^2C}{C\sqrt{T}} + 360\zeta^2,
\end{aligned}
$$

if we choose $C \leq 1/\left(\frac{1}{N-f} + 12 + 6(N-f+1)\right)$.

B Experimental Details

B.1 Hardware Setup and Resources

Our implementation uses the PyTorch framework (version 2.9.0)(Paszke et al., 2019). Claude (Sonnet 4.6, Opus 5.5) was employed to assist with code fragments. Experiments were conducted on two servers: one equipped with an AMD Ryzen 9 7950X3D processor (16 cores, 32 threads, 4.2 GHz), 96 GB of RAM, and an NVIDIA GeForce RTX 4080 Super GPU, and another equipped with an AMD Ryzen 9 9950X3D processor (16 cores, 32 threads, 4.3 GHz), 128 GB of RAM, and an NVIDIA RTX PRO 5000 GPU. Wall-clock times varied substantially across experiments, also because the GPUs were shared with other researchers and their load fluctuated. Average run times are reported in Table 5.

B.2 Datasets and Models

We use four publicly available datasets for the experiments: CIFAR-10, CIFAR-100 (Krizhevsky, 2009), FEMNIST (Caldas et al., 2019), and Shakespeare (Caldas et al., 2019). CIFAR and FEMNIST datasets are used for image classification tasks, while Shakespeare is used for next character prediction. We present the details for the dataset specifics as well as other details on the training below.

CIFAR CIFAR-10 and CIFAR-100 are widely used benchmark image datasets, each consisting of 50,000 training samples and 10,000 test samples of 32×32 color images of 10 and 100 different classes, respectively. Datasets are originally homogeneous with respect to label distribution. To induce heterogeneity, we follow Yurochkin et al. (2019); Lin et al. (2020) and sample the class-wise data proportions for each client from a Dirichlet distribution, i.e., $\boldsymbol{p}_j \sim \mathrm{Dir}_N(\alpha)$ for each class $j \in [10]$, with concentration parameter $\alpha = 0.5$ and $\alpha = 0.1$ to model a strongly non-IID setting. As a result, non-IIDness arises from label distribution skew. In addition, dataset sizes vary across clients, and some clients have no samples from certain classes.

For CIFAR-10 preprocessing, the training images are first augmented with random cropping to size $32 \times 32 \times 3$ with 4 pixels of zero-padding and random horizontal flipping. After augmentation, the images are converted to tensors and normalized per channel with means $(0.4914, 0.4822, 0.4465)$ and standard deviations $(0.2023, 0.1994, 0.2010)$. For the test set, no data augmentation is applied; the images are only converted to tensors and normalized using the same per-channel statistics.

On CIFAR-100 we do not train a model from scratch but fine-tune a DeiT-Tiny vision transformer (patch size 16) (Touvron et al., 2021) pretrained on ImageNet-1k. The pretrained model's original 1000-class output layer is replaced with a new 100-class classification head whose weights and biases are initialized to zero, so that every run starts from the same point, and all parameters of the network are fine-tuned. To reduce computational cost, we upsample the 32×32 images to 112×112, giving a 7×7 grid of patches, and the pretrained positional embeddings are interpolated to this grid accordingly. We use stochastic depth with drop-path rate 0.1.

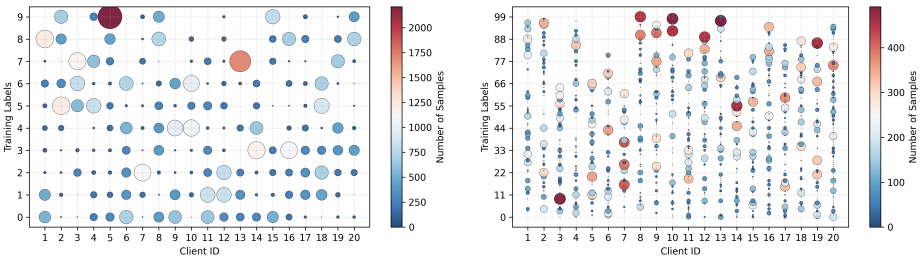

Figure 3: Examples of heterogeneous CIFAR-10 (left) and CIFAR-100 (right) distributions across clients with $\boldsymbol{p}_j \sim \mathrm{Dir}_N(\alpha)$, $\alpha = 0.5$ (left) and $\alpha = 0.1$ (right).

FEMNIST The Federated Extended MNIST dataset consists of size 28×28 gray-scale images of hand-written alphanumeric characters. There exist 62 different classes (10 digits, 26 lowercase letters, 26 uppercase letters). Caldas et al. (2019) designed a benchmark where the images are partitioned among clients according to the writing style of clients. The source of non-IID distribution, thus, is feature skew, while the dataset sizes are kept similar. The clients are sampled using the Leaf (Caldas et al., 2019) framework by executing the following line `./preprocess.sh -s niid -sf 1.0 -k 350 -t sample`, where the minimum number of samples per user is set to 350. For simulations, a considerably small number of users are further sampled from the dataset, where 0.9 of the samples are used for local training performed by clients, while the whole collection of test samples is used to perform a global performance test.

FEMNIST, the images are converted to tensors and normalized using a mean of 0.5 and a standard deviation of 0.5, without any additional data augmentation.

Shakespeare The Shakespeare dataset is a collection of William Shakespeare's plays, where each client holds the lines of one speaking role in those plays. The task is to predict the next character given a sequence of preceding characters. The dataset is generated with the LEAF framework (Caldas et al., 2019) by executing `./preprocess.sh -s niid -sf 1 -k 50 -t sample -tf 0.9`, i.e., a non-IID split by role in which every role holds at least 50 samples and each role's samples are split into training and test sets with ratio 0.9/0.1. In total, 660 roles are obtained. Out of these, 50 roles holding at least one full mini-batch of training samples are selected uniformly at random as clients, and their training samples are used to train local models, while the test samples of all 660 roles are used to evaluate global performance.

Following LEAF, each sample consists of a window of 80 consecutive characters of a role's lines as input and the character immediately following it as the label, with consecutive windows shifted by one character. Characters are mapped to a vocabulary of size 89, consisting of 85 characters (upper- and lowercase letters, digits, and punctuation and whitespace symbols) and 4 special tokens for padding, beginning and end of line, and out-of-vocabulary characters. The only character of the dataset that falls outside the vocabulary is the ampersand, which is mapped to the out-of-vocabulary token.

Models The details on the experimental setup are provided in Table 3. The lightweight Compact Convolutional Transformers (CCT) network used for CIFAR-10 is chosen inspired by Hassani et al. (2022) and similar to Li et al. (2023). We give the details of the models used in our experiments below using a compact notation similarly to Allouah et al. (2023a):

Notation. Input dimensions are given in parentheses at the start of each architecture string. $\mathbf{C}(k,c)$ denotes a 2D convolutional layer with kernel size $k \times k$, padding $\lfloor k/2 \rfloor$, and c output channels. $\mathbf{M}(k)$ denotes 2D max-pooling with kernel size $k \times k$. **R** denotes ReLU, **G** denotes GELU, **LN** denotes layer normalization. $\mathbf{D}(p)$ denotes dropout with probability p. $\mathbf{L}(d)$ denotes a fully connected linear layer with d outputs. **S** denotes log-softmax. $\mathbf{E}(v,d)$ denotes an embedding layer with vocabulary size v and output dimension d. $\mathbf{LSTM}(h,\ell)$ denotes an LSTM with hidden size h and ℓ layers. $\mathbf{MHA}(h)$ denotes multi-head self-attention with h heads. **DP** denotes stochastic depth (DropPath), and $\oplus$ denotes a residual addition closing the preceding branch. **AttnPool** denotes attention-based sequence pooling. Brackets $[\cdots]^{\times n}$ denote a block repeated n times. $\mathbf{Up}(s)$ denotes bicubic upsampling to $s \times s$ (clamped to $[0,1]$), and **N** denotes per-channel standardization with the ImageNet mean and standard deviation. $\mathbf{PE}(p,d)$ denotes a patch embedding: the image is split into non-overlapping $p \times p$ patches, each linearly projected to d dimensions (a $p \times p$ convolution with stride p and no padding); a learnable class token is prepended, and learnable position embeddings are added. **CLS** denotes selecting the output at the class-token position.

B.3 Attacks and Defenses used for Evaluation

Attacks for Evaluation We use $\boldsymbol{\mu}_{\mathcal{H}}$ and $\boldsymbol{\sigma}_{\mathcal{H}}$ to denote the coordinate-wise mean and the standard deviation of the honest client updates, respectively. For ALIE (Baruch et al., 2019) and FOE (Xie et al., 2020), parameters (z and ϵ, respectively) are selected each round adaptively via a grid search over linearly and log-uniformly spaced values, respectively,

Table 3: Experimental settings in detail.

Dataset	CIFAR-10	CIFAR-100	FEMNIST	Shakespeare
Model Type	CCT	DeiT-Tiny (ImageNet pretrained)	CNN	LSTM
Model Architecture	(32 × 32 × 3)-C(3,64)-G-M(2)-C(3,256)-G-M(2)-$\left[\text{LN-MHA(4)-DP}\oplus\text{LN-L(512)-G-L(256)-DP}\oplus\right]^{\times 7}$-LN-AttnPool-L(10)	(32 × 32 × 3)-Up(112)-N-PE(16,192)-$\left[\text{LN-MHA(3)-DP}\oplus\text{LN-L(768)-G-L(192)-DP}\oplus\right]^{\times 12}$-LN-CLS-L(100)	(28 × 28 × 1)-C(5,32)-R-M(2)-C(5,64)-R-M(2)-Flatten-L(512)-R-L(62)	(90)-E(90,8)-LSTM(256,2)-L(90)
Model Size	$d = 3,858,635$	$d = 5,515,492$	$d = 1,690,046$	$d = 822,305$
Loss	Cross Entropy Loss	Cross Entropy Loss	Cross Entropy Loss	Cross Entropy Loss
# Communication Rounds	$T = 5000$	$T = 1000$	$T = 2000$	$T = 1000$
Learning Rate	$\gamma_t = 0.025\left(1+\cos\left(\pi\frac{t}{T}\right)\right)$	$\gamma_t = \begin{cases} 0.5\cdot\frac{t+1}{50}, & t<50, \\ 0.25\left(1+\cos\left(\pi\frac{t-50}{T-50}\right)\right), & \text{else.}\end{cases}$	$\gamma_t = \begin{cases} 0.05 & \text{if } t < \lfloor 2T/3\rfloor, \\ 0.005 & \text{else.}\end{cases}$	$\gamma_t = \begin{cases} 10 & \text{if } t < \lfloor 2T/3\rfloor, \\ 1 & \text{else.}\end{cases}$
Momentum/E	$\beta = 0.9$, $E = 1$	$\beta = 0.9$, $E = 1$	$\beta = 0.9$, $E = 1$	$\beta = 0.9$, $E = 1$
Batch Size	$b = 64$	$b = 128$	$b = 64$	$b = 64$
Testing Period	$T' = 10$	$T' = 10$	$T' = 10$	$T' = 10$

choosing the value that maximizes the ℓ_2 distance between the resulting aggregation and the honest mean. For MinSum and MinMax, $\boldsymbol{p}$ is a perturbation direction selected among three choices, namely the inverse unit vector $-\boldsymbol{\mu}_{\mathcal{H}}/\|\boldsymbol{\mu}_{\mathcal{H}}\|$, the inverse standard deviation $-\boldsymbol{\sigma}_{\mathcal{H}}$, and the inverse sign $-\operatorname{sign}(\boldsymbol{\mu}_{\mathcal{H}})$, and γ^* is the largest scaling factor satisfying the stated constraint. For Mimic, i^* is the honest worker selected to maximize the variance along the top eigenvector of the honest vectors. Malicious update set is denoted as $\mathcal{M}$. A malicious client $k \in \mathcal{M}$ sends a manipulated update $\tilde{\boldsymbol{m}}_k$ instead of the honest value $\boldsymbol{m}_k$, such that:

- ALIE: $\tilde{\boldsymbol{m}}_k = \boldsymbol{\mu}_{\mathcal{H}} - z^*\boldsymbol{\sigma}_{\mathcal{H}}$, where $z^* \in \{0.25z, 0.5z, \ldots, c^*z\}$ where c^* is the largest multiple of 0.25 such that $c^*z \leq 2$. z is obtained as explained in Baruch et al. (2019).
- FOE: $\tilde{\boldsymbol{m}}_k = -\epsilon^*\boldsymbol{\mu}_{\mathcal{H}}$, where $\epsilon \in \{0.01, 0.016, 0.025, 0.04, 0.063, 0.1, 0.158, 0.251, 0.398, 0.631, 1.0, 1.58, 2.51, 3.98, 6.31, 10.0, 15.8, 25.1, 39.8, 63.1, 100\}$.
  When FOE is marked with (ϵ), the grid search is performed over linearly spaced values, i.e., $\epsilon^* \in \{0.1\epsilon, 0.2\epsilon, \ldots, \epsilon\}$.
- LF: $\tilde{\boldsymbol{m}}_k$ computed on the data with flipped labels; e.g. $y' = 61 - y$ for FEMNIST.
- SF: $\tilde{\boldsymbol{m}}_k = -\boldsymbol{m}_k$, where the malicious client sends the negative of the update computed on its local data.
- ZeroMean: $\tilde{\boldsymbol{m}}_k = (\boldsymbol{r}_k - \boldsymbol{\mu}_{\mathcal{M}})\frac{\epsilon\|\boldsymbol{\mu}_{\mathcal{H}}\|}{\|\boldsymbol{r}_k - \boldsymbol{\mu}_{\mathcal{M}}\|}$, where $\boldsymbol{\mu}_{\mathcal{M}} = \frac{1}{|\mathcal{M}|}\sum_{k=1}^{|\mathcal{M}|}\boldsymbol{r}_k$, and $\boldsymbol{r}_k$s are random vectors whose entries are sampled independently from a standard normal distribution. We choose $\epsilon = 0.1$.
  The ZeroMean attack is specifically designed to target the `ProDiGy` defense. In `ProDiGy`, the trust scores are scaled by the inverse neighborhood mean, i.e., $\frac{1}{\boldsymbol{\mu}(\mathcal{N}_f(k))} = \frac{|\mathcal{N}_f(k)|}{\sum_{i\in\mathcal{N}_f(k)}\boldsymbol{m}_i}$. To defeat this mechanism, colluding malicious clients construct their updates to have a *zero mean* and to fall within each others neighborhoods. Concretely, the final poisoned updates are obtained as $\boldsymbol{r}_k - \boldsymbol{\mu}_{\mathcal{M}}$, which enforces zero mean across the malicious set, and then rescaled by a factor $\epsilon = 0.1$ to ensure that the updates remain within a tight neighborhood.
- Mimic: $\tilde{\boldsymbol{m}}_k = \boldsymbol{m}_{i^*}$, with $i^* \in \mathcal{H}$,
- MinMax: $\tilde{\boldsymbol{m}}_k = \boldsymbol{\mu}_{\mathcal{H}} + \gamma^*\boldsymbol{p}$, s.t. $\max_{i\in\mathcal{H}}\|\tilde{\boldsymbol{m}}_k - \boldsymbol{m}_i\| \leq \max_{i,j\in\mathcal{H}}\|\boldsymbol{m}_i - \boldsymbol{m}_j\|$,
- MinSum: $\tilde{\boldsymbol{m}}_k = \boldsymbol{\mu}_{\mathcal{H}} + \gamma^*\boldsymbol{p}$, s.t. $\sum_{i\in\mathcal{H}}\|\tilde{\boldsymbol{m}}_k - \boldsymbol{m}_i\|^2 \leq \max_{i\in\mathcal{H}}\sum_{j\in\mathcal{H}}\|\boldsymbol{m}_i - \boldsymbol{m}_j\|^2$.

Defenses for Evaluation For the proposed method `GatScHA`, Algorithm 1 is used with the scatter score defined in Equation (2), unless otherwise stated. Other defense parameters and aggregation rules are listed below. For a vector $\boldsymbol{v} \in \mathbb{R}^d$, we denote by $[\boldsymbol{v}]_j$ its j-th coordinate, $j \in [d]$. Coordinate-wise rules are applied independently to each $j \in [d]$ and output $\hat{\boldsymbol{m}} \in \mathbb{R}^d$.

- No Defense: $\texttt{Avg}(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_N) = \frac{1}{N} \sum_{k \in [N]} \boldsymbol{m}_k$.
- Coordinate-wise Median (Yin et al., 2018):

$$[\hat{\boldsymbol{m}}]_j = \texttt{Median}\big([\boldsymbol{m}_1]_j, \ldots, [\boldsymbol{m}_N]_j\big).$$

- Coordinate-wise Trimmed Mean (Yin et al., 2018): let $\mathcal{T}_j \subseteq [N]$, with $|\mathcal{T}_j| = N - 2q$, be the set of clients remaining after removing those with the q largest and the q smallest values of $[\boldsymbol{m}_k]_j$. Then

$$[\hat{\boldsymbol{m}}]_j = \frac{1}{N - 2q} \sum_{k \in \mathcal{T}_j} [\boldsymbol{m}_k]_j.$$

  We choose $q = f$ in the sequel.
- Mean-around-Median (MeaMed) (Xie et al., 2018): let $\mathcal{T'}_j \subseteq [N]$, with $|\mathcal{T'}_j| = N - q$, be the set of clients whose values $[\boldsymbol{m}_k]_j$ are closest to $\texttt{Median}\big([\boldsymbol{m}_1]_j, \ldots, [\boldsymbol{m}_N]_j\big)$. Then

$$[\hat{\boldsymbol{m}}]_j = \frac{1}{N - q} \sum_{k \in \mathcal{T'}_j} [\boldsymbol{m}_k]_j.$$

  We choose $q = 2f$ in the sequel.
- Geometric Median (Pillutla et al., 2022): we use the RFA algorithm, i.e., the smoothed Weiszfeld algorithm with $\nu = 0.1$ and $R = 3$ rounds, to approximate

$$\texttt{GeoMed}(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_N) = \arg\min_{\boldsymbol{m} \in \mathbb{R}^d} \sum_{k=1}^{N} \big\|\boldsymbol{m} - \boldsymbol{m}_k\big\|.$$

- Krum (Blanchard et al., 2017): with $\mathcal{N}_{N-f-1}(k)$ as defined in Section 3.1,

$$\texttt{Krum}(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_N) = \arg\min_{\boldsymbol{m}_k,\, k \in [N]} \sum_{j \in \mathcal{N}_{N-f-1}(k)} \big\|\boldsymbol{m}_k - \boldsymbol{m}_j\big\|^2.$$

- Centered Clipping (Karimireddy et al., 2020): for parameters $\tau_l = 10$, $L = 3$, the algorithm chooses

$$\texttt{CClip}(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_N) = \boldsymbol{m}_L, \text{ where}$$

$$\boldsymbol{m}_{l+1} = \boldsymbol{m}_l + \frac{1}{N} \sum_{k=1}^{N} (\boldsymbol{m}_k - \boldsymbol{m}_l) \min\Big(1, \frac{\tau_l}{\|\boldsymbol{m}_k - \boldsymbol{m}_l\|}\Big),$$

  and $\boldsymbol{m}_0$ is initialized by the previous round's aggregation.
- `ProDiGy`: Given the trust scores $s(k)$ defined in Ergisi et al. (2025), the `ProDiGy` aggregation rule is

$$\texttt{ProDiGy}(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_N) = \frac{\sum_{k \in [N]} s(k) \cdot \boldsymbol{m}_k}{\sum_{k \in [N]} s(k)}.$$

For a strong comparison, we strengthen the robust aggregation methods we compare against by combining them with the `NNM` pre-aggregation scheme (Allouah et al., 2023a) when it fits. `NNM` replaces each update with its mix in the following way:

$$(\bar{\boldsymbol{m}}_1, \ldots, \bar{\boldsymbol{m}}_N) = \texttt{NNM}(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_N),$$
$$\text{where} \quad \bar{\boldsymbol{m}}_k = \frac{1}{N - f} \sum_{i \in \mathcal{N}_{N-f}(k)} \boldsymbol{m}_i,$$

with $\mathcal{N}_{N-f}(k)$ as defined in Section 3.1.

B.4 Extensive Simulation Results

This section contains full simulation results presented in the main body, namely for Tables 1 and 2, Figure 2; and some additional simulation results listed as below:

- Figure 4: Experimental evaluation of the theoretically stronger `GatScHA` scheme (see Corollary 3.4), hereafter referred to as $\texttt{GatScHA}_{\texttt{Avg}}$, on the FEMNIST dataset.
- Figure 5: Learning curves over the full training process for the CIFAR-100 results presented in Table 1.
- Figure 6: Learning curves over the full training process for the CIFAR-10 results presented in Table 1.
- Figure 7: Learning curves over the full training process for the FEMNIST results presented in Table 1.
- Figure 8: Learning curves over the full training process for the Shakespeare results presented in Table 1.
- Figure 9: Learning curves over the full training process when f is misestimated for the FEMNIST results presented in Table 2.
- Figure 10: Learning curves over the full training process when f is misestimated for the FEMNIST results presented in Table 2.
- Table 4: Final accuracy table on defenses/attack on FEMNIST when multiple local iterations (E=10) are performed by each client.
- Table 5: Average wall-clock training time (in minutes per seed) for the experiments reported in Table 1.

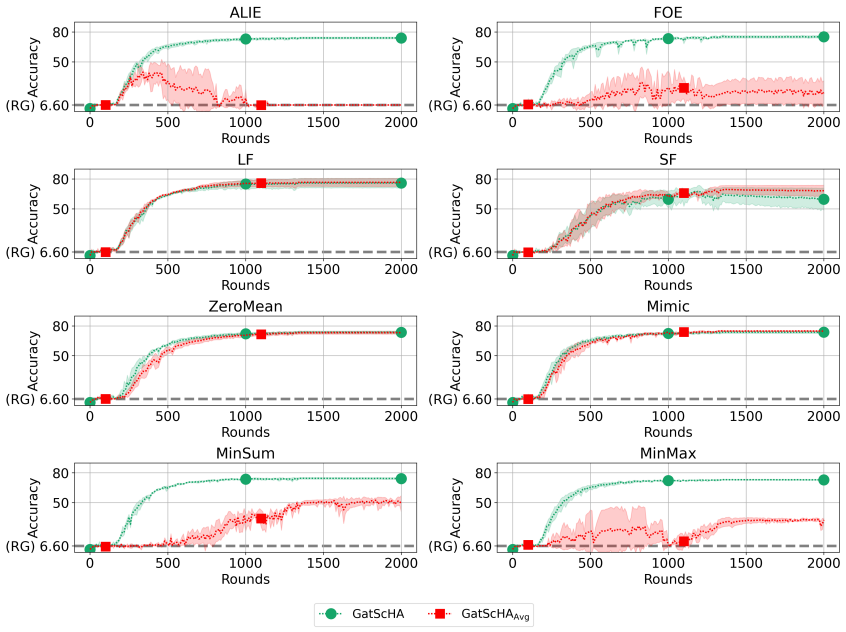

Figure 4: Experiments on FEMNIST using $N = 20$, $f = 5$, with $\beta = 0.9$, $E = 1$. The theoretical guarantees of the scheme $\texttt{GatScHA}_{\texttt{Agn}}$ is given in Corollary 3.4.

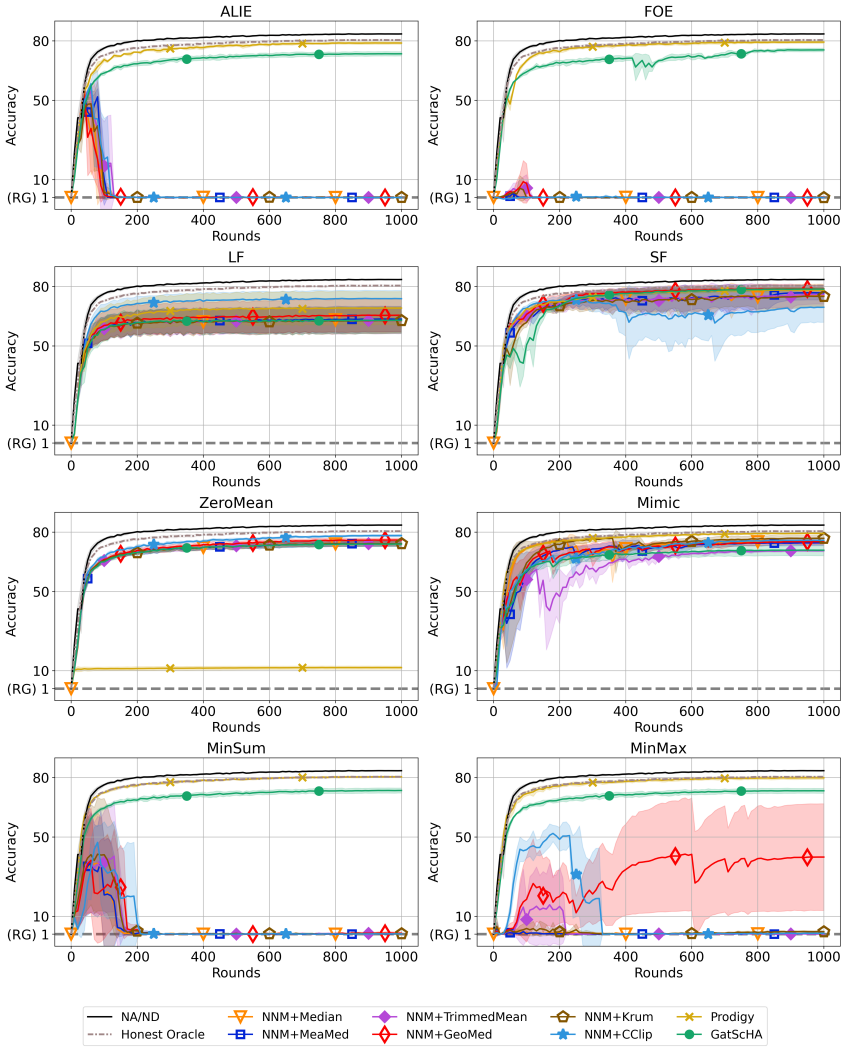

Figure 5: Experiments on CIFAR-100 using $N = 20$, $f = 5$, with $\beta = 0.9$, $E = 1$. The corresponding final accuracies are reported in Table 1.

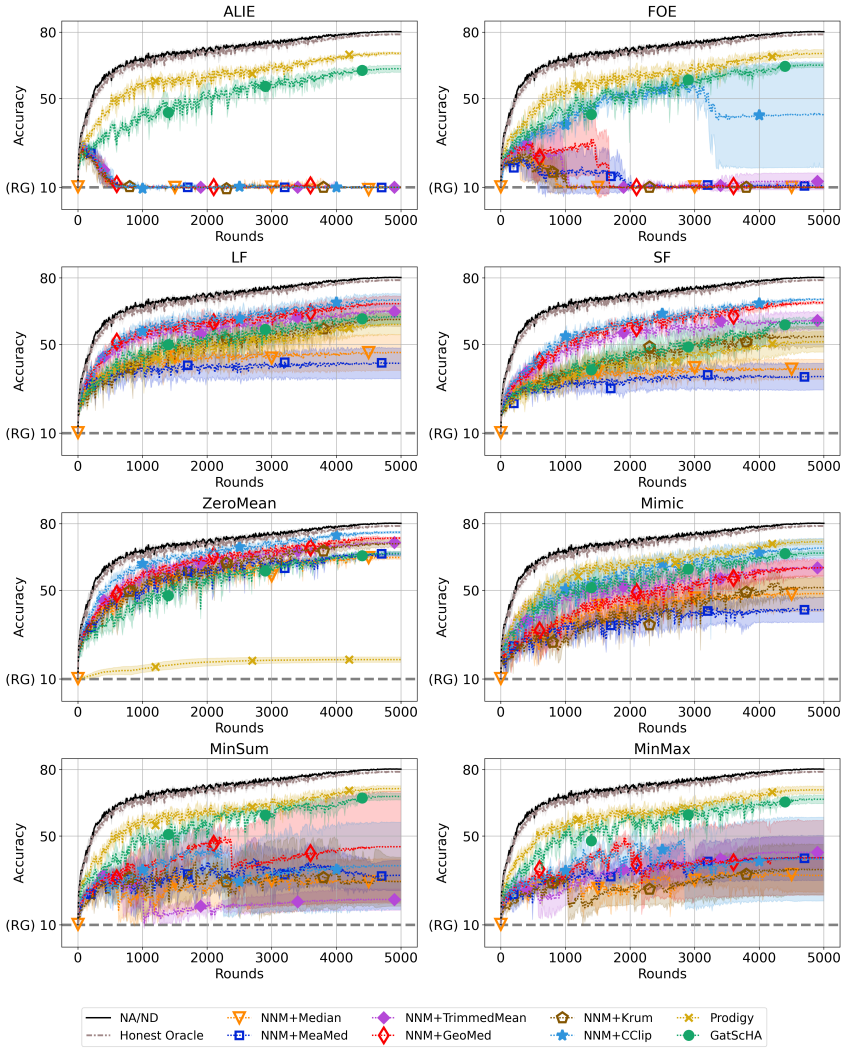

Figure 6: Experiments on CIFAR-10 using $N = 20$, $f = 5$, with $\beta = 0.9$, $E = 1$. The corresponding final accuracies are reported in Table 1.

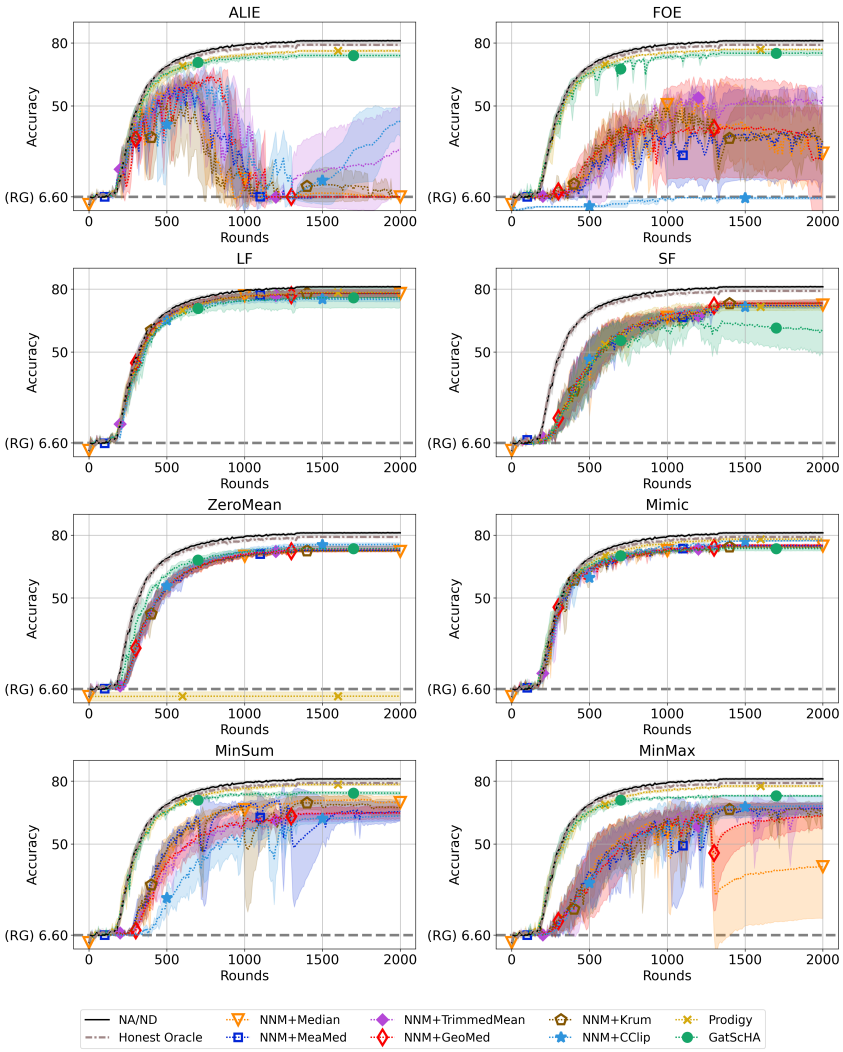

Figure 7: Experiments on FEMNIST using $N = 20$, $f = 5$, with $\beta = 0.9$, $E = 1$. The corresponding final accuracies are reported in Table 1.

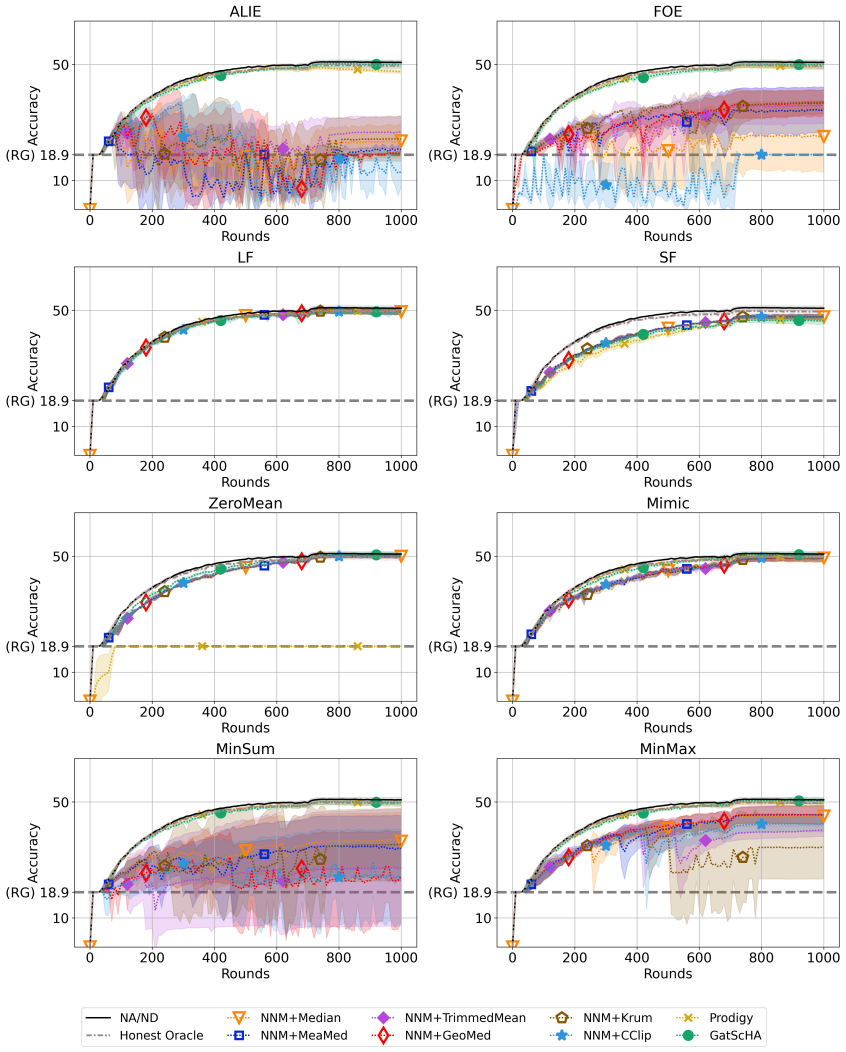

Figure 8: Experiments on CIFAR-10 using $N = 50$, $f = 12$, with $\beta = 0.9$, $E = 1$. The corresponding final accuracies are reported in Table 1.

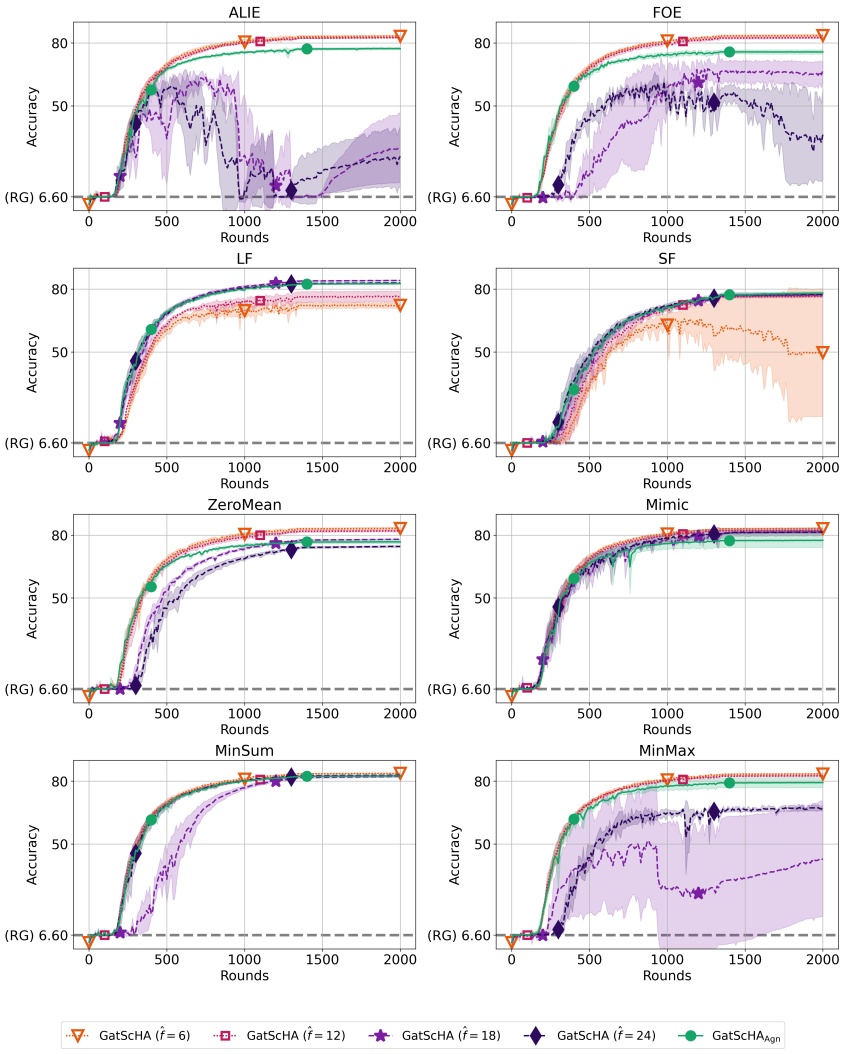

Figure 9: Experiments on FEMNIST using $N = 60$, $f = 15$, with $\beta = 0.9$, $E = 1$. The corresponding final accuracies are reported in Table 2.

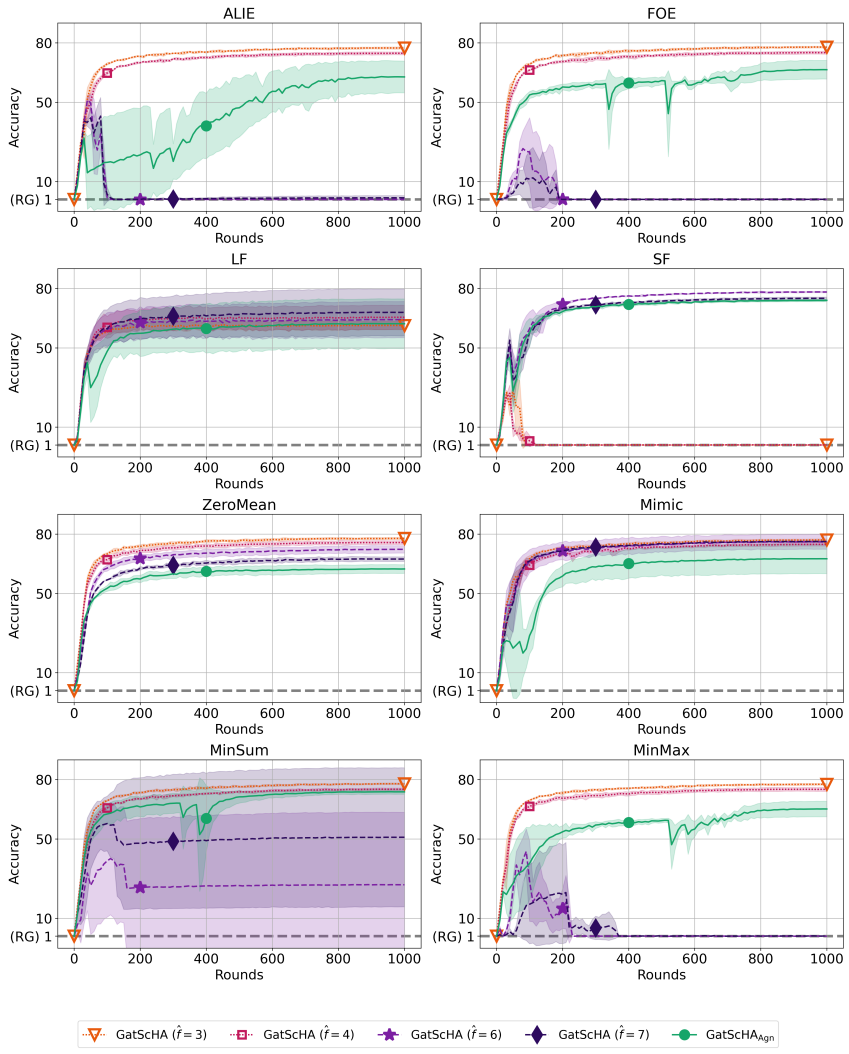

Figure 10: Experiments on CIFAR-100 using $N = 20$, $f = 5$, with $\beta = 0.9$, $E = 1$. The corresponding final accuracies are reported in Table 2.

Table 4: Final accuracies for FEMNIST dataset with ($\beta = 0.9$) and with multiple iterations taken by clients ($E = 10$) according to Appendix C. For each defense, the worst-case accuracy is highlighted in bold. The best worst-case accuracy is highlighted in green. Cells are marked in red if at least one run approaches random-guess performance.

FEMNIST Dataset												
$N = 20$, $f = 5$		**ALIE**	**FOE**,$\epsilon = 0.1$	**FOE**,$\epsilon = 1$	**FOE**,$\epsilon = 10$	**FOE**,$\epsilon = 100$	**LF**	**SF**	**ZeroMean**	**Mimic**	**MinSum**	**MinMax**
	NNM + Median	4.20 ± 1.69	71.80 ± 1.16	$\mathbf{3.00 \pm 0.00}$	19.01 ± 22.63	72.49 ± 1.23	79.09 ± 1.16	74.86 ± 3.26	75.41 ± 0.42	75.11 ± 0.22	77.93 ± 0.12	74.11 ± 0.81
	NNM + TrimmedMean	6.70 ± 3.06	71.45 ± 1.05	$\mathbf{3.00 \pm 0.00}$	19.24 ± 23.06	72.55 ± 1.40	79.10 ± 1.15	75.11 ± 2.86	75.58 ± 0.44	75.09 ± 0.23	77.79 ± 0.11	74.16 ± 1.01
	NNM + GeoMed	$\mathbf{5.40 \pm 1.69}$	73.05 ± 1.03	$\mathbf{3.00 \pm 0.00}$	22.62 ± 27.74	73.22 ± 1.38	79.03 ± 1.22	74.81 ± 3.00	75.95 ± 0.36	75.62 ± 0.17	78.07 ± 0.33	74.60 ± 0.76
	NNM + Krum	4.20 ± 1.69	48.01 ± 31.84	$\mathbf{3.00 \pm 0.00}$	$\mathbf{3.00 \pm 0.00}$	72.82 ± 1.15	79.05 ± 1.14	75.00 ± 2.79	75.43 ± 0.38	75.23 ± 0.20	77.84 ± 0.09	76.14 ± 0.21
	NNM + CClip	28.47 ± 30.93	78.36 ± 0.15	26.18 ± 25.12	61.86 ± 1.05	$\mathbf{0.80 \pm 0.23}$	75.53 ± 0.96	47.26 ± 31.30	78.86 ± 0.06	78.76 ± 0.33	78.60 ± 0.24	77.57 ± 0.46
	CClip	80.03 ± 0.46	79.68 ± 0.57	79.32 ± 0.91	0.25 ± 0.05	$\mathbf{0.23 \pm 0.05}$	77.57 ± 0.41	49.58 ± 32.95	79.56 ± 0.64	79.28 ± 0.58	78.99 ± 0.54	78.43 ± 0.39
	ProDiGy	79.37 ± 0.34	79.55 ± 0.47	79.62 ± 0.41	79.63 ± 0.38	79.71 ± 0.33	79.52 ± 0.49	75.88 ± 0.31	$\mathbf{6.56 \pm 0.05}$	79.22 ± 0.46	79.52 ± 0.52	79.72 ± 0.27
	GatScHA	76.42 ± 0.35	75.08 ± 0.63	74.98 ± 0.71	78.51 ± 0.18	79.30 ± 0.19	79.47 ± 0.67	$\mathbf{72.45 \pm 0.77}$	75.07 ± 0.68	76.42 ± 0.55	75.01 ± 0.67	75.51 ± 0.41

Table 5: Average wall-clock training time (in minutes per seed) for the experiments reported in Table 1. For each dataset, we also report the average time spend for testing accumulated over the whole simulation run, which can be reduced by evaluating less frequently.

CIFAR-100 Dataset ($\text{Dir}_{20}(0.1)$) | **Test Time:** 0.33								
$N = 20$, $f = 5$	**ALIE**	**FOE**	**LF**	**SF**	**ZeroMean**	**Mimic**	**MinSum**	**MinMax**
NNM + Median	7.17	9.72	1.79	1.79	1.81	1.51	3.41	3.39
NNM + MeaMed	9.35	12.63	1.94	1.94	1.97	1.66	3.92	3.90
NNM + TrimmedMean	6.23	7.99	2.60	2.59	2.61	2.04	3.56	3.57
NNM + GeoMed	6.65	8.67	1.76	1.74	1.77	1.47	3.22	3.20
NNM + Krum	5.88	7.74	1.70	1.71	1.74	1.42	3.02	3.02
NNM + CClip	6.64	8.72	1.76	1.74	1.78	1.46	3.21	3.23
ProDiGy	3.75	4.76	1.56	1.58	1.58	1.29	2.49	2.47
GatScHA	2.18	3.68	1.39	1.38	1.38	1.11	2.34	2.39

CIFAR-10 Dataset ($\text{Dir}_{20}(0.5)$) | **Test Time:** 3.00								
$N = 20$, $f = 5$	**ALIE**	**FOE**	**LF**	**SF**	**ZeroMean**	**Mimic**	**MinSum**	**MinMax**
NNM + Median	27.02	35.54	8.65	8.58	8.19	8.57	8.92	9.05
NNM + MeaMed	34.66	45.56	9.18	9.14	9.64	8.93	9.59	9.69
NNM + TrimmedMean	14.69	17.64	9.37	9.45	9.50	7.39	11.52	11.48
NNM + GeoMed	24.43	31.41	8.53	8.58	9.48	8.47	8.86	8.95
NNM + Krum	22.31	28.09	8.43	8.44	8.16	8.28	8.58	8.71
NNM + CClip	24.38	31.48	8.58	8.58	9.49	7.18	8.91	8.94
ProDiGy	14.60	17.97	7.96	7.96	9.23	6.38	8.14	8.10
GatScHA	13.21	16.17	7.87	7.87	8.05	7.93	7.82	7.95

FEMNIST Dataset | **Test Time:** 3.11								
$N = 20$, $f = 5$	**ALIE**	**FOE**	**LF**	**SF**	**ZeroMean**	**Mimic**	**MinSum**	**MinMax**
NNM + Median	9.06	12.36	1.02	1.00	1.00	0.90	3.13	3.19
NNM + MeaMed	12.11	16.62	1.21	1.22	1.20	1.08	3.78	3.79
NNM + TrimmedMean	3.49	4.62	0.78	0.77	0.78	0.69	1.66	1.66
NNM + GeoMed	8.37	11.33	0.96	0.97	0.96	0.87	2.99	3.06
NNM + Krum	7.32	9.90	0.90	0.91	0.91	0.81	2.78	2.76
NNM + CClip	8.34	11.32	0.97	0.97	0.96	0.87	2.95	3.03
ProDiGy	3.70	4.95	0.68	0.68	0.69	0.61	1.99	2.00
GatScHA	3.43	4.52	0.66	0.66	0.67	0.60	1.94	2.16

Shakespeare Dataset | **Test Time:** 11.97								
$N = 50$, $f = 12$	**ALIE**	**FOE**	**LF**	**SF**	**ZeroMean**	**Mimic**	**MinSum**	**MinMax**
NNM + Median	11.21	16.68	3.34	3.25	3.28	2.59	5.33	5.20
NNM + MeaMed	14.45	21.77	3.58	3.52	3.32	2.78	6.21	6.14
NNM + TrimmedMean	5.26	6.69	3.79	3.85	3.82	2.96	3.61	3.63
NNM + GeoMed	9.41	13.69	3.00	3.01	3.02	2.44	4.56	4.57
NNM + Krum	10.78	15.81	3.25	3.26	3.24	2.58	5.06	5.06
NNM + CClip	9.42	14.40	3.01	3.00	3.24	2.57	5.01	5.01
ProDiGy	5.30	7.25	2.75	2.75	2.72	2.19	3.49	3.49
GatScHA	5.05	6.85	2.82	2.80	2.80	2.17	3.43	3.43

C Federated Learning Algorithm

We provide here the full federated learning algorithm, as explained in Section 2.1. In general, a federated learning algorithm can run with a number of E local iterations. Therefore, we denote the local update at client k in round t and local iteration e as:

$$\boldsymbol{\theta}_k^{(t,e+1)} = \boldsymbol{\theta}_k^{(t,e)} - \frac{\gamma_k^{(t)}}{|\mathcal{B}_k^{(t,e)}|} \sum_{(\boldsymbol{x},y)\in\mathcal{B}_k^{(t,e)}} \nabla\ell(\mathcal{F}(\boldsymbol{\theta}_k^{(t,e)};\boldsymbol{x}),y),$$

where $\boldsymbol{\theta}_k^{(t,0)} = \boldsymbol{\theta}^{(t)}$. In the main body of the paper, we use $E = 1$ for simplicity.

Algorithm 2 Federated Learning Algorithm

Input: total number of clients N, initial model parameters $\boldsymbol{\theta}^{(0)} \in \mathbb{R}^d$, total communication rounds T, learning rate in round t γ_t, local iteration number E, batch size b, local momentum parameter β, $\boldsymbol{m}_k^{(1)} \triangleq 0 \quad \forall k \in [N]$

Central server:

for round $t \in \{1,\dots,T\}$ **do**
  Broadcast $\boldsymbol{\theta}^{(t)}$ to clients
  for each honest client $k \in [N]$ in parallel **do**
    $\boldsymbol{m}_k^{(t)} \leftarrow \text{ClientUpdate}(k, \boldsymbol{\theta}^{(t)}, \gamma_t)$
  end for
  Update model $\boldsymbol{\theta}^{(t+1)} \leftarrow \boldsymbol{\theta}^{(t)} - \gamma^{(t)} \texttt{Agg}(\boldsymbol{m}_1^{(t)},\dots,\boldsymbol{m}_N^{(t)})$
end for

ClientUpdate $(k, \boldsymbol{\theta}^{(t)}, \gamma_t)$**:**

Initialize $\boldsymbol{\theta}_k^{(t,0)} \leftarrow \boldsymbol{\theta}^{(t)}$
for $e \in \{0,\dots,E-1\}$ **do**
  Sample randomly and independently without replacement a set of data points $\mathcal{B}_k^{(t,e)}$ of size b from the local dataset $\mathcal{S}_k$
  $\boldsymbol{\theta}_k^{(t,e+1)} \leftarrow \boldsymbol{\theta}_k^{(t,e)} - \gamma_k^{(t)} \frac{1}{|\mathcal{B}_k^{(t,e)}|} \sum_{(\boldsymbol{x},y)\in\mathcal{B}_k^{(t,e)}} \nabla\ell(\mathcal{F}(\boldsymbol{\theta}_k^{(t,e)};\boldsymbol{x}),y)$
end for
$\boldsymbol{g}_k^{(t)} \leftarrow \frac{1}{\gamma_k^{(t)}} \left(\boldsymbol{\theta}^{(t)} - \boldsymbol{\theta}_k^{(t,E)}\right)$
Send $\boldsymbol{m}_k^{(t)} \leftarrow \beta \boldsymbol{m}_k^{(t-1)} + (1-\beta)\boldsymbol{g}_k^{(t)}$ to the server

return $\boldsymbol{\theta}^{(T+1)}$

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We stated our contributions at the end of the introduction section, and they match with our theoretical and experimental results in the respective sections.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations separately in Section 5.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We state our assumptions along our theoretical results in Section 3.2. The proofs of our theoretical results are given step by step in Appendix A.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: For reproducibility, our experimental setup including all details, datasets, models, and parameters is described in Appendix B. We also provide the source code of our simulations as a publicly available repository.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We provide the source code of our simulations as a publicly available repository. Experimental details and parameters are described in Appendix B. All datasets we use have already been publicly available previously.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: All experimental details, optimization methods, models, datasets, etc. are described in Appendix B.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: All experimental result tables and figures report mean ± standard deviation (1-sigma error) and specifically stated in the main paper. The sources of randomness are given in Appendix B.2 and Section 4.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide all the details in Appendix B.1.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The relevance and positive societal impacts, of our work are mentioned in the introduction, such as robustness, trustworthiness, and prevention of malicious manipulations. We could not identify negative societal impacts.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: Our research uses only existing models and datasets and, thus, does not pose any new risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We clearly indicate the datasets we use and their authors in Appendix B, where we also acknowledge and cite the sources.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: Our source code is publicly available. The usage is described in a structured README file, which also acknowledges any sources that we base our implementation on. Parameters and configurations are explained in our experimental details.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard component.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.