FedACT: A Byzantine-Resilient Federated Learning Framework with Autoencoder-Committee-TLBO for Heterogeneous Credit Scoring

Dengjia Li[a,c], Chaoqun Ma[a,c], Jinglan Yang[a,c] and Yuncheng Qiao[b,c,*]

[a] *Business School, Hunan University, Changsha 410082, China*
[b] *Business School, Shandong University of Technology, Zibo 255000, China*
[c] *Research Institute of Digital Society and Blockchain, Hunan University, China*

ARTICLE INFO

Keywords:
Federated learning
Byzantine resilience
Credit scoring
Anomaly detection
Data heterogeneity

ABSTRACT

Federated learning enables privacy-preserving collaborative credit scoring across financial institutions, yet its distributed architecture is vulnerable to Byzantine attacks where participants submit arbitrary model updates. This threat is amplified by data heterogeneity inherent in cross-silo consortia, which invalidates the clustering assumptions underlying existing defenses.

We propose FedACT, a three-stage Byzantine-resilient framework comprising: (1) an autoencoder-based anomaly detector with dual-metric scoring and MAD-based adaptive thresholding that partitions gradients into normal, uncertain, and anomalous zones; (2) a diversity-aware committee voting mechanism that resolves uncertain cases through consensus verification; and (3) TLBO-based robust aggregation coupled with reputation-driven incentives and Merkle-tree evidence chaining for auditability. Experiments on real-world credit datasets under twelve attack types and four heterogeneity scenarios demonstrate that FedACT maintains strong detection accuracy and model performance where existing defenses degrade.

1. Introduction

Federated learning (FL) enables privacy-preserving collaborative model training across distributed data silos [1, 2]. In credit scoring, financial institutions can jointly develop predictive models without sharing sensitive customer data, satisfying both regulatory compliance and commercial confidentiality requirements [3, 4]. However, the opacity inherent in FL—where the aggregation server cannot inspect local computations—creates a critical attack surface: *Byzantine participants* may submit arbitrarily malicious model updates to poison the global model [5, 6].

Byzantine attacks in federated credit scoring pose severe threats to model integrity. Adversaries may bias the model toward approving high-risk borrowers (fraudulent approval attacks), rejecting creditworthy applicants (legitimate rejection attacks), or systematically destabilizing the learning process (model degradation attacks). The heterogeneous nature of cross-silo financial data exacerbates this vulnerability: existing Byzantine-resilient aggregation methods [5, 7, 8] assume that honest gradients cluster tightly around a common mean—an assumption violated when institutions serve distinct customer segments with varying default rates and feature distributions. Under such heterogeneity, strict detection thresholds incorrectly reject legitimate minority-institution updates, while lenient thresholds fail to identify sophisticated attacks designed to mimic benign statistical profiles [9].

To address this fundamental tension between heterogeneity tolerance and attack detection, we propose **FedACT** (**Fed**erated **A**utoencoder-**C**ommittee-**T**LBO), a Byzantine defense framework that combines three complementary mechanisms:

1. **Anomaly detection via learned manifolds.** An autoencoder learns the distribution of benign gradients and computes a dual-metric anomaly score combining reconstruction error with latent-space deviation. Adaptive MAD-based thresholding partitions gradients into normal, uncertain, and anomalous zones without requiring IID assumptions.
2. **Consensus verification for uncertain cases.** A diversity-constrained committee of verified participants adjudicates borderline gradients through majority voting, reducing both false positives from heterogeneity and false negatives from evasive attacks.
3. **Robust aggregation with accountability.** TLBO-based optimization aggregates verified gradients while a reputation mechanism creates long-term incentives for honest behavior. Merkle-tree evidence recording enables post-hoc auditability.

The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 formalizes the problem and threat model. Section 4 presents FedACT. Section 5 reports experimental evaluation. Section 6 concludes.

2. Related Work

Byzantine attacks in federated learning. The Byzantine fault model, originating from distributed systems [6], characterizes adversaries who submit arbitrary—potentially adversarially optimized—messages. In FL, Byzantine clients transmit malicious gradients to corrupt the global model [5]. Attack strategies span a spectrum from naive perturbations (sign-flipping, noise injection, scaling) to defense-aware optimizations: ALIE [10] crafts updates within the tail of the benign distribution; IPM [11] projects malicious

[*] Corresponding author
✉ qiaoyc@hnu.edu.cn (Y. Qiao)

Table 1
Capability comparison under heterogeneous, cross-silo credit scoring (✓: supported; ∼: partially/with prerequisite)

Method	Heterogeneity-tolerant	Uncertainty verification	No server data	Audit / deterrence
Krum / Multi-Krum [5]	∼	✗	✓	✗
Bulyan [8]	∼	✗	✓	✗
Median / TrimmedMean [7]	∼	✗	✓	✗
RFA [15]	∼	✗	✓	✗
FLTrust [16]	∼	✗	∼	✗
Learning-based detectors [17, 18]	∼	✗	✓	✗
FedACT (Ours)	✓	✓	✓	✓

directions onto the subspace spanned by honest updates; MinMax [12] solves an optimization problem to maximally evade distance-based filters. Backdoor attacks [13, 14] embed hidden functionalities while preserving aggregate performance. The increasing sophistication of adaptive attacks motivates multi-stage defenses that combine detection, verification, and deterrence.

Byzantine-resilient aggregation. Defenses can be grouped into three paradigms. *Robust statistics* methods apply coordinate-wise median, trimmed mean [7], or geometric median (RFA) [15] to bound the influence of outliers. *Distance-based selection* methods such as Krum, Multi-Krum [5], and Bulyan [8] filter updates based on pairwise distances. *Trust-anchored* methods like FLTrust [16] reweight updates by similarity to a server-held reference gradient. A common limitation is the implicit IID assumption: honest gradients are expected to concentrate around a shared mean. Under non-IID data partitions typical of cross-silo credit scoring, this assumption fails, causing benign heterogeneous updates to be misclassified as malicious [9].

Learning-based anomaly detection. Recent work applies anomaly detection models—isolation forests, autoencoders, clustering—to gradient-level features [17, 18]. These approaches can capture complex distributional structure but typically lack (i) explicit mechanisms for borderline cases where detection confidence is low, and (ii) repeated-game incentive structures that deter strategic attackers over time.

Positioning of FedACT. Table 1 contrasts existing methods with FedACT. Our framework addresses the heterogeneity-tolerance gap through learned manifold representations, introduces uncertainty-aware committee verification to handle borderline cases, and couples detection with reputation-based deterrence and auditable evidence recording.

3. Problem Formulation and Threat Model

3.1. Federated Learning Formulation

Consider N clients (financial institutions) collaboratively training a global model $\mathbf{w} \in \mathbb{R}^d$. Client i holds private dataset $\mathcal{D}_i = \{(\mathbf{x}_j, y_j)\}_{j=1}^{n_i}$ with local objective $F_i(\mathbf{w}) = \frac{1}{n_i}\sum_{j=1}^{n_i} \ell(f(\mathbf{w};\mathbf{x}_j), y_j)$. The global objective is:

$$\min_{\mathbf{w}} F(\mathbf{w}) = \sum_{i=1}^{N} \frac{n_i}{n} F_i(\mathbf{w}), \quad n = \sum_{i=1}^{N} n_i \tag{1}$$

Standard FedAvg [1] proceeds iteratively: at round t, the server broadcasts $\mathbf{w}^{(t)}$; each client computes local gradient $\mathbf{g}_i^{(t)} = \nabla F_i(\mathbf{w}^{(t)})$ and transmits it to the server; the server aggregates:

$$\mathbf{w}^{(t+1)} = \mathbf{w}^{(t)} - \eta \sum_{i=1}^{N} \frac{n_i}{n} \mathbf{g}_i^{(t)} \tag{2}$$

Data heterogeneity. In cross-silo credit scoring, client data distributions $\mathcal{P}_i(\mathbf{x}, y)$ differ substantially due to customer segmentation, geographic factors, and institution-specific practices. This *non-IID* setting induces high gradient variance: $\|\nabla F_i(\mathbf{w}) - \nabla F(\mathbf{w})\|$ can be large even for honest clients, complicating the distinction between heterogeneity and malicious behavior.

3.2. Threat Model

We consider $M < N/2$ Byzantine clients who may submit arbitrary gradients $\tilde{\mathbf{g}}_i$ to the server. The adversary has white-box knowledge of the model architecture and parameters, can adapt strategies across rounds, and may coordinate multiple compromised clients. Attack objectives include untargeted model degradation, targeted misclassification of specific borrower profiles, or backdoor injection.

We evaluate twelve attack types spanning three categories: *basic attacks* (sign-flipping $\tilde{\mathbf{g}}_i = -\mathbf{g}_i$, Gaussian noise, scaling), *sophisticated attacks* (ALIE [10], IPM [11], MinMax [12]), and *other attacks* (label-flipping, backdoor [13], free-riding, collusion).

The aggregation server is honest-but-curious: it executes the defense protocol correctly but may observe transmitted gradients. We assume secure channels, authenticated identities (no Sybil attacks), and honest majority ($M < N/2$).

[Framework Diagram]

Gradients → Autoencoder Detection → Three-Zone Classification → Committee Voting → TLBO Aggregation → Model Update

Figure 1: FedACT architecture. Stage 1 computes anomaly scores via autoencoder reconstruction and latent deviation, partitioning gradients into $\mathcal{N}$ (normal), $\mathcal{U}$ (uncertain), $\mathcal{A}$ (anomalous). Stage 2 resolves $\mathcal{U}$ through diversity-constrained committee voting. Stage 3 aggregates verified gradients via TLBO optimization with reputation weighting.

4. The FedACT Framework

FedACT defends against Byzantine attacks through a three-stage pipeline illustrated in Figure 1: (1) autoencoder-based anomaly detection that classifies gradients into normal, uncertain, and anomalous zones; (2) diversity-aware committee voting that resolves uncertain cases; and (3) TLBO-based robust aggregation with reputation updates and evidence recording.

4.1. Autoencoder-Based Anomaly Detection

The detection stage learns a low-dimensional manifold of benign gradients and identifies attacks as off-manifold deviations. Let $\mathbf{g}_i \in \mathbb{R}^p$ denote client i's gradient vector. An autoencoder with encoder $\phi_\theta : \mathbb{R}^p \to \mathbb{R}^k$ and decoder $\psi_\theta : \mathbb{R}^k \to \mathbb{R}^p$ is trained on historical gradients $\mathcal{G}_{\text{hist}}$ from verified normal clients to minimize reconstruction loss:

$$\mathcal{L}(\theta) = \frac{1}{|\mathcal{G}_{\text{hist}}|} \sum_{\mathbf{g} \in \mathcal{G}_{\text{hist}}} \|\psi_\theta(\phi_\theta(\mathbf{g})) - \mathbf{g}\|_2^2 \tag{3}$$

For high-dimensional gradients ($p > 10{,}000$), we apply stratified layer-wise subsampling to bound memory and computation while preserving structural information. The latent dimension k is set adaptively: $k = 32$ for $p < 5{,}000$, $k = 64$ for $5{,}000 \leq p \leq 10{,}000$, and $k = 128$ otherwise.

Dual-metric anomaly score. For each incoming gradient $\mathbf{g}_i$, we compute:

$$a_i = \lambda \cdot \frac{\|\mathbf{g}_i - \psi_\theta(\phi_\theta(\mathbf{g}_i))\|_2^2}{\max_j \|\mathbf{g}_j - \hat{\mathbf{g}}_j\|_2^2 + \epsilon} + (1-\lambda) \cdot \frac{\|\phi_\theta(\mathbf{g}_i) - \boldsymbol{\mu}_z\|_2}{\max_j \|\phi_\theta(\mathbf{g}_j) - \boldsymbol{\mu}_z\|_2 + \epsilon} \tag{4}$$

where $\boldsymbol{\mu}_z = \frac{1}{|\mathcal{G}_{\text{hist}}|} \sum_{\mathbf{g} \in \mathcal{G}_{\text{hist}}} \phi_\theta(\mathbf{g})$ is the latent centroid, $\lambda = 0.7$ weights reconstruction error over latent deviation, and $\epsilon = 10^{-8}$ prevents division by zero. Max-normalization preserves relative magnitudes, which is critical for detecting scaling attacks.

Adaptive thresholding. We employ Median Absolute Deviation (MAD) for robust threshold estimation:

$$\tau = \text{med}(\mathbf{a}) + \kappa \cdot 1.4826 \cdot \text{MAD}(\mathbf{a}), \quad \text{MAD}(\mathbf{a}) = \text{med}(|a_i - \text{med}(\mathbf{a})|) \tag{5}$$

where $\kappa = 2.5$ controls sensitivity and 1.4826 ensures consistency with Gaussian standard deviation. Gradients are partitioned into three zones:

$$\mathcal{N} = \{i : a_i < 0.7\tau\}, \quad \mathcal{U} = \{i : 0.7\tau \leq a_i < 1.5\tau\}, \quad \mathcal{A} = \{i : a_i \geq 1 \tag{6}$$

Gradients in $\mathcal{N}$ are accepted directly; those in $\mathcal{A}$ are rejected; those in $\mathcal{U}$ proceed to committee verification. This three-zone design accommodates heterogeneity by deferring borderline decisions rather than making binary choices under uncertainty.

4.2. Committee Voting for Uncertain Gradients

Gradients in the uncertain zone $\mathcal{U}$ are adjudicated by a committee of verified normal clients. Committee voting activates after a warm-up period of 5 rounds to allow reputation scores to stabilize.

Diversity-maximizing selection. A committee C of size $K = 5$ is selected from $\mathcal{N}$ to maximize gradient-space coverage. The first member is the highest-reputation client: $c_1 = \arg\max_{i \in \mathcal{N}} \rho_i$. Subsequent members are chosen greedily to minimize maximum cosine similarity with existing members:

$$c_k = \arg \min_{i \in \mathcal{N} \setminus C} \max_{j \in C} \langle \mathbf{g}_i, \mathbf{g}_j \rangle / (\|\mathbf{g}_i\| \|\mathbf{g}_j\|) \tag{7}$$

This diversity constraint reduces the risk of colluding attackers dominating the committee.

Voting mechanism. Each committee member c casts a vote on uncertain gradient $\mathbf{g}_u$:

$$v_{c \to u} = \mathbf{1}\left[\cos(\mathbf{g}_u, \mathbf{g}_c) < \theta_v\right], \quad \theta_v = 0.3 \tag{8}$$

Algorithm 1 Autoencoder-Based Anomaly Detection

```
Require: Gradients $\{\mathbf{g}_i\}_{i=1}^N$, historical buffer $\mathcal{G}_{\text{hist}}$, epochs $E = 20$, $\kappa = 2.5$
Ensure: Normal set $\mathcal{N}$, uncertain set $\mathcal{U}$, anomalous set $\mathcal{A}$
Initialize encoder $\phi_\theta$, decoder $\psi_\theta$ with latent dim $d = \lceil 0.1 \cdot \dim(\mathbf{g}) \rceil$
for epoch = 1 to $E$ do
    $\mathcal{L} \leftarrow \frac{1}{|\mathcal{G}_{\text{hist}}|} \sum_{\mathbf{g} \in \mathcal{G}_{\text{hist}}} \|\psi_\theta(\phi_\theta(\mathbf{g})) - \mathbf{g}\|^2$
    Update $\theta$ via Adam($\mathcal{L}$, lr=$10^{-3}$)
end for
$\boldsymbol{\mu}_h \leftarrow |\mathcal{G}_{\text{hist}}|^{-1} \sum_{\mathbf{g} \in \mathcal{G}_{\text{hist}}} \phi_\theta(\mathbf{g})$    ▷ Latent centroid
for $i = 1$ to $N$ do
    $\ell_{\text{rec}} \leftarrow \|\mathbf{g}_i - \psi_\theta(\phi_\theta(\mathbf{g}_i))\|^2$;    $\ell_{\text{lat}} \leftarrow \|\phi_\theta(\mathbf{g}_i) - \boldsymbol{\mu}_h\|$
    $a_i \leftarrow \lambda \cdot \ell_{\text{rec}} / \max_j \ell_{\text{rec}}^{(j)} + (1 - \lambda) \cdot \ell_{\text{lat}} / \max_j \ell_{\text{lat}}^{(j)}$ ▷ $\lambda = 0.7$
end for
$\tau \leftarrow \text{med}(\mathbf{a}) + \kappa \cdot 1.4826 \cdot \text{MAD}(\mathbf{a})$    ▷ Adaptive threshold
$\mathcal{N} \leftarrow \{i : a_i < 0.7\tau\}$; $\mathcal{U} \leftarrow \{i : 0.7\tau \leq a_i < 1.5\tau\}$; $\mathcal{A} \leftarrow \{i : a_i \geq 1.5\tau\}$
return $\mathcal{N}, \mathcal{U}, \mathcal{A}$
```

where $v = 1$ indicates an anomalous vote. With self-exclusion (a client cannot vote on its own gradient), the decision rule is:

$$\text{label}(u) = \begin{cases} \text{anomalous} & \text{if } \sum_{c \neq u} v_{c \to u} > 0.5 \cdot |\mathcal{C} \setminus \{u\}| \\ \text{normal} & \text{otherwise} \end{cases} \tag{9}$$

Algorithm 2 Committee Selection and Voting

```
Require: Normal set $\mathcal{N}$, uncertain set $\mathcal{U}$, gradients $\{\mathbf{g}_i\}$, reputations $\{\rho_i\}$, $K = 5$, $\theta_v = 0.3$
Ensure: Updated sets $\mathcal{N}'$, $\mathcal{A}'$
$\mathcal{C} \leftarrow \{\arg\max_{i \in \mathcal{N}} \rho_i\}$    ▷ Select highest-reputation client
while $|\mathcal{C}| < K$ do
    $c \leftarrow \arg\min_{i \in \mathcal{N} \setminus \mathcal{C}} \max_{j \in \mathcal{C}} \cos(\mathbf{g}_i, \mathbf{g}_j)$
    $\mathcal{C} \leftarrow \mathcal{C} \cup \{c\}$
end while
for each $u \in \mathcal{U}$ do
    $v \leftarrow \sum_{c \in \mathcal{C}, c \neq u} \mathbf{1}[\cos(\mathbf{g}_u, \mathbf{g}_c) < \theta_v]$
    if $v > 0.5 \cdot |\mathcal{C} \setminus \{u\}|$ then
        $\mathcal{A}' \leftarrow \mathcal{A}' \cup \{u\}$
    else
        $\mathcal{N}' \leftarrow \mathcal{N}' \cup \{u\}$
    end if
end for
```

4.3. TLBO-Based Robust Aggregation

The final stage aggregates verified gradients using Teaching-Learning-Based Optimization (TLBO) [19], a population-based metaheuristic that iteratively refines candidate solutions without algorithm-specific parameters.

Let $\mathcal{N}'$ denote the verified normal set after detection and committee voting. Each gradient $\mathbf{g}_i \in \mathcal{N}'$ is treated as a learner. The fitness function measures alignment with the reputation-weighted target:

$$f(\mathbf{g}) = \cos(\mathbf{g}, \bar{\mathbf{g}}), \quad \bar{\mathbf{g}} = \sum_{i \in \mathcal{N}'} \omega_i \mathbf{g}_i, \quad \omega_i = \rho_i / \sum_j \rho_j \tag{10}$$

where ρ_i is client i's reputation score.

Teacher phase. Each learner moves toward the best-performing gradient (teacher) while moving away from the population mean:

$$\mathbf{g}_i' = \mathbf{g}_i + r \cdot (\mathbf{g}^* - T_F \cdot \bar{\boldsymbol{\mu}}), \quad r \sim \mathcal{U}(0,1), \ T_F \in \{1, 2\} \tag{11}$$

where $\mathbf{g}^* = \arg\max f(\mathbf{g})$ and $\bar{\boldsymbol{\mu}}$ is the population mean.

Learner phase. Learners interact pairwise, moving toward superior peers:

$$\mathbf{g}_i' = \mathbf{g}_i + r \cdot \text{sign}(f(\mathbf{g}_i) - f(\mathbf{g}_j)) \cdot (\mathbf{g}_i - \mathbf{g}_j) \tag{12}$$

Updates are accepted only if they improve fitness. After $T_{\text{TLBO}} = 10$ iterations, the final target $\bar{\mathbf{g}}$ becomes the aggregated gradient for model update.

Algorithm 3 TLBO-Based Gradient Aggregation

```
Require: Verified gradients $\{\mathbf{g}_i\}_{i \in \mathcal{N}'}$, reputations $\{\rho_i\}$, iterations $T = 10$
Ensure: Aggregated gradient $\bar{\mathbf{g}}$
$\omega_i \leftarrow \rho_i / \sum_j \rho_j$ for each $i \in \mathcal{N}'$    ▷ Reputation weights
$\bar{\mathbf{g}} \leftarrow \sum_i \omega_i \mathbf{g}_i$    ▷ Initial target
$\mathcal{P} \leftarrow \{\mathbf{g}_i\}_{i \in \mathcal{N}'}$    ▷ Learner population
for $t = 1$ to $T$ do
    $f(\mathbf{g}) \leftarrow \cos(\mathbf{g}, \bar{\mathbf{g}})$ for all $\mathbf{g} \in \mathcal{P}$    ▷ Fitness
    // Teacher phase
    $\mathbf{g}^* \leftarrow \arg\max_{\mathbf{g} \in \mathcal{P}} f(\mathbf{g})$;    $\bar{\boldsymbol{\mu}} \leftarrow |\mathcal{P}|^{-1} \sum_{\mathbf{g} \in \mathcal{P}} \mathbf{g}$
    for each $\mathbf{g}_i \in \mathcal{P}$ do
        $T_F \sim \text{Uniform}(\{1, 2\})$;    $r \sim \mathcal{U}(0, 1)$
        $\mathbf{g}_i' \leftarrow \mathbf{g}_i + r \cdot (\mathbf{g}^* - T_F \cdot \bar{\boldsymbol{\mu}})$
        if $f(\mathbf{g}_i') > f(\mathbf{g}_i)$ then $\mathbf{g}_i \leftarrow \mathbf{g}_i'$
        end if
    end for
    // Learner phase
    for each $\mathbf{g}_i \in \mathcal{P}$ do
        Sample $\mathbf{g}_j \in \mathcal{P} \setminus \{\mathbf{g}_i\}$;    $r \sim \mathcal{U}(0, 1)$
        $\mathbf{g}_i' \leftarrow \mathbf{g}_i + r \cdot \text{sign}(f(\mathbf{g}_i) - f(\mathbf{g}_j)) \cdot (\mathbf{g}_i - \mathbf{g}_j)$
        if $f(\mathbf{g}_i') > f(\mathbf{g}_i)$ then $\mathbf{g}_i \leftarrow \mathbf{g}_i'$
        end if
    end for
    $\bar{\mathbf{g}} \leftarrow |\mathcal{P}|^{-1} \sum_{\mathbf{g} \in \mathcal{P}} \mathbf{g}$    ▷ Update target
end for
return $\bar{\mathbf{g}}$
```

4.4. Reputation and Accountability Mechanisms

Reputation dynamics. Each client maintains a reputation score $\rho_i \in [0.1, 2.0]$, initialized at 1.0. After each round, reputations update asymmetrically:

$$\rho_i^{(t+1)} = \begin{cases} \min(\rho_i^{(t)} + 0.05 \cdot \xi_i, 2.0) & \text{if } i \in \mathcal{N}' \\ \max(0.7 \cdot \rho_i^{(t)}, 0.1) & \text{if } i \in \mathcal{A}' \end{cases} \tag{13}$$

where $\xi_i = \max(0, (\cos(\mathbf{g}_i, \bar{\mathbf{g}}) + 1)/2)$ is the contribution score based on alignment with the aggregated gradient. This design creates strategic deterrence: reputations grow slowly but decay rapidly, making sustained attacks costly.

Merkle-tree evidence. For auditability, detection results $\{(i, a_i, \text{label}_i)\}$ at each round are hashed into a Merkle tree. The root hash $h^{(t)}$ is appended to an evidence chain $\mathcal{E} = \{h^{(1)}, \dots, h^{(T)}\}$, enabling tamper-evident logging and $O(\log N)$ verification for dispute resolution.

4.5. Complete FedACT Pipeline

Algorithm 4 presents the complete FedACT pipeline integrating all components.

Algorithm 4 Complete FedACT Framework

```
Require: Clients $\{1, \dots, N\}$, global model $\mathbf{w}^{(0)}$, rounds $T$
Ensure: Final model $\mathbf{w}^{(T)}$, evidence chain $\mathcal{E}$
Initialize reputations $\rho_i \leftarrow 1.0$ for all $i$
Initialize evidence chain $\mathcal{E} \leftarrow \emptyset$
for round $t = 1$ to $T$ do
    // Client-side: Local training
    for each client $i$ in parallel do
        Receive $\mathbf{w}^{(t-1)}$ from server
        Compute gradient $\mathbf{g}_i^{(t)} \leftarrow \nabla F_i(\mathbf{w}^{(t-1)})$
        Send $\mathbf{g}_i^{(t)}$ to server
    end for
    // Server-side: FedACT defense
    $\mathcal{N}, \mathcal{U}, \mathcal{A} \leftarrow$ AutoencoderDetection($\{\mathbf{g}_i^{(t)}\}, \mathcal{G}_{\text{hist}}$)
    if $t > 5$ and $|\mathcal{U}| > 0$ then
        $\mathcal{N}', \mathcal{A}' \leftarrow$ CommitteeVoting($\mathcal{N}, \mathcal{U}, \{\mathbf{g}_i\}, \{\rho_i\}$)  ▷ Alg. 2
    else
        $\mathcal{N}' \leftarrow \mathcal{N} \cup \mathcal{U}$;    $\mathcal{A}' \leftarrow \mathcal{A}$  ▷ Warm-up: detector-only
    end if
    $\bar{\mathbf{g}} \leftarrow$ TLBOAggregation($\{\mathbf{g}_i\}_{i \in \mathcal{N}'}, \{\rho_i\}$)
    // Update model
    $\mathbf{w}^{(t)} \leftarrow \mathbf{w}^{(t-1)} - \eta \cdot \bar{\mathbf{g}}$
    // Update reputations
    for each $i \in \mathcal{N}'$ do
        $c_i \leftarrow \max\left(0, (\cos(\mathbf{g}_i^{(t)}, \bar{\mathbf{g}}) + 1)/2\right)$  ▷ Alignment contribution
        $\rho_i \leftarrow \min(\rho_i + 0.05 \cdot c_i, 2.0)$
    end for
    for each $i \in \mathcal{A}'$ do
        $\rho_i \leftarrow \max(\rho_i \times 0.7, 0.1)$
    end for
    // Record evidence
    $h^{(t)} \leftarrow$ MerkleRoot($\{(i, a_i, \ell_i)\}_{i=1}^{N}$)
    $\mathcal{E} \leftarrow \mathcal{E} \cup \{h^{(t)}\}$
    Update: $\mathcal{G}_{\text{hist}} \leftarrow \mathcal{G}_{\text{hist}} \cup \{\mathbf{g}_i^{(t)}\}_{i \in \mathcal{N}'}$
end for
return $\mathbf{w}^{(T)}, \mathcal{E}$
```

5. Experiments

We evaluate FedACT on four research questions: (RQ1) detection effectiveness across attack types; (RQ2) comparison with baseline defenses; (RQ3) robustness under data heterogeneity; (RQ4) component contributions via ablation.

5.1. Setup

Datasets. We use two credit scoring datasets: UCI Credit Card Default [20] (30,000 samples, 23 features, 22.1% default rate) and Xinwang Bank (50,000 samples, 35 features, 15.3% default rate).

Table 2
FedACT detection performance (UCI, label skew, 30% attackers)

Attack Type	Precision	Recall	F1-Score
Basic Attacks			
Sign-flipping	–	–	–
Gaussian noise	–	–	–
Scaling	–	–	–
Sophisticated Attacks			
Little	–	–	–
ALIE	–	–	–
IPM	–	–	–
MinMax	–	–	–
Trim attack	–	–	–
Other Attacks			
Label-flipping	–	–	–
Backdoor	–	–	–
Free-rider	–	–	–
Collusion	–	–	–
Average	–	–	–

Heterogeneity scenarios. Data is partitioned across $N = 10$ clients under four settings: IID (uniform random), label skew (Dirichlet $\beta = 0.5$), feature skew (partial feature overlap), and quantity skew (power-law distribution).

Model and training. Three-layer MLP (Input→128→64→1) with ReLU, Dropout(0.3), and Sigmoid output. Training uses binary cross-entropy, Adam ($\eta = 10^{-3}$), batch size 64, $E = 5$ local epochs, $T = 100$ rounds.

Attacks. We evaluate 12 Byzantine attacks: basic (sign-flip, Gaussian, scaling), sophisticated (ALIE [10], IPM [11], MinMax [12]), and others (label-flip, backdoor [13], free-riding, collusion). Default attacker ratio: $M = 3$ of $N = 10$ (30%).

Baselines. FedAvg [1] (undefended), Median [7], Trimmed-Mean [7], Krum/Multi-Krum [5], Bulyan [8], RFA [15].

Metrics. Detection: Precision, Recall, F1. Model: Accuracy, AUC-ROC, Accuracy Preservation ($\text{Acc}_{\text{defended}}/\text{Acc}_{\text{clean}}$).

5.2. Detection Effectiveness (RQ1)

Table 2 reports FedACT's detection performance on UCI dataset under label skew with 30% attackers. Basic perturbation attacks are detected with high precision and recall due to their pronounced deviation from the learned gradient manifold. Optimization-based attacks (ALIE, MinMax) are more challenging as they explicitly minimize statistical distinguishability, yet FedACT maintains competitive F1 scores through the combination of reconstruction-based and latent-deviation metrics.

FedACT achieves strong detection performance across attack families, with simple perturbation attacks being easier to detect than optimization-based attacks that explicitly mimic benign statistics.

Table 3
Model accuracy comparison across defense methods (UCI, label skew, 30% attackers)

Attack	None	Median	TrimMean	Krum	Multi-Krum	Bulyan	RFA	FedACT
No attack	–	–	–	–	–	–	–	–
Sign-flip	–	–	–	–	–	–	–	–
Gaussian	–	–	–	–	–	–	–	–
Scaling	–	–	–	–	–	–	–	–
ALIE	–	–	–	–	–	–	–	–
IPM	–	–	–	–	–	–	–	–
MinMax	–	–	–	–	–	–	–	–
Label-flip	–	–	–	–	–	–	–	–
Backdoor	–	–	–	–	–	–	–	–
Average	–	–	–	–	–	–	–	–
Preserve %	–	–	–	–	–	–	–	–

Table 4
FedACT performance under heterogeneity scenarios (MinMax attack)

Heterogeneity	Precision	Recall	Accuracy	AUC
IID	–	–	–	–
Label skew	–	–	–	–
Feature skew	–	–	–	–
Quantity skew	–	–	–	–
Average	–	–	–	–

5.3. Comparison with Baselines (RQ2)

Table 3 compares model accuracy under various attacks (30% attackers, label skew).

FedACT maintains high accuracy by combining explicit anomaly detection with robust aggregation. Robust statistics-based defenses offer partial protection but may be evaded by carefully crafted attacks. Distance-based methods can be stressed by heterogeneity, while FedACT's detection and uncertainty resolution mitigate this.

5.4. Robustness Under Heterogeneity (RQ3)

Table 4 evaluates FedACT under different heterogeneity types (MinMax attack, 30% attackers).

FedACT maintains stable performance across heterogeneity scenarios. Feature skew presents the greatest challenge due to structural gradient differences; the uncertainty zone mitigates false positives from legitimate heterogeneity. Quantity skew is addressed through reputation-weighted aggregation.

Table 5 shows graceful degradation as attacker ratio increases from 10% to 45%. The three-stage pipeline maintains detection effectiveness near the honest-majority boundary.

FedACT degrades gracefully as attacker ratio increases by combining conservative filtering, uncertainty handling, and robust aggregation over verified updates.

Table 5
Performance under varying attacker ratios (MinMax attack, label skew)

Attacker %	Precision	Recall	F1-Score	Accuracy
10%	–	–	–	–
20%	–	–	–	–
30%	–	–	–	–
35%	–	–	–	–
40%	–	–	–	–
45%	–	–	–	–

Table 6
Ablation study: FedACT component contributions

Configuration	Precision	Recall	Accuracy	F1
Part 1: Component Ablation				
FedACT_Full	–	–	–	–
w/o_Autoencoder	–	–	–	–
w/o_Committee	–	–	–	–

5.5. Ablation Study (RQ4)

Table 6 isolates component contributions under MinMax attack (30% attackers, label skew). Removing the autoencoder significantly degrades detection of clearly anomalous updates. Removing committee voting increases false positives in borderline cases where heterogeneity resembles adversarial behavior.

Table 7 compares aggregation algorithms without defense components. Metaheuristic methods (FedTLBO, FedGWO, FedPSO) outperform standard FL algorithms (FedAvg, FedProx) but remain vulnerable to sophisticated attacks without explicit detection.

Table 7
Aggregation algorithm comparison without defense components (Only_Agg)

Algorithm	Sign-flip	ALIE	MinMax	Average
FedAvg	–	–	–	–
FedProx	–	–	–	–
SCAFFOLD	–	–	–	–
MOON	–	–	–	–
FedPSO	–	–	–	–
FedGWO	–	–	–	–
FedTLBO	–	–	–	–

Table 8
Parameter sensitivity (MinMax attack, 30% attackers)

Parameter	Values	Accuracy	Detection F1
c_{lower}	0.5	–	–
	0.7	–	–
	0.9	–	–
c_{upper}	1.2	–	–
	1.5	–	–
	1.8	–	–
Committee size K	3	–	–
	5	–	–
	7	–	–
TLBO iterations	5	–	–
	10	–	–
	20	–	–

Removing the autoencoder impacts filtering of clearly anomalous updates; removing committee voting affects borderline decisions where heterogeneity resembles adversarial behavior.

Table 7 presents the aggregation algorithm comparison results (Only_Agg, no defense components, three representative attacks).

Metaheuristic aggregation improves robustness but without detection remains susceptible to crafted attacks. FedACT combines both so that optimization operates on verified updates.

5.6. Sensitivity Analysis

Table 8 shows FedACT is robust to hyperparameter choices. The default configuration ($c_{\text{lower}} = 0.7$, $c_{\text{upper}} = 1.5$, $K = 5$, $T_{\text{TLBO}} = 10$) balances detection sensitivity with false positive avoidance across tested ranges.

FedACT is relatively robust to parameter choices. Default values ($c_{lower} = 0.7, c_{upper} = 1.5, K = 5, T_{TLBO} = 10$) provide good balance between detection sensitivity and false positive avoidance.

6. Conclusion

This paper proposed FedACT, a Byzantine-resilient federated learning framework that addresses the heterogeneity-tolerance gap in cross-silo credit scoring. By integrating autoencoder-based anomaly detection, diversity-constrained committee voting, and TLBO-based robust aggregation with reputation-driven incentives, FedACT achieves effective attack detection without the IID assumptions that limit existing defenses. Experimental evaluation across twelve attack types and four heterogeneity scenarios validates the contribution of each component.

From a governance perspective, FedACT separates detection, adjudication, and aggregation into auditable stages with Merkle-tree evidence recording, operationalizing accountability requirements for regulated financial consortia.

Future work includes adversarially robust training against autoencoder-aware evasion, continual adaptation for concept drift, communication-efficient variants, extension to vertical FL, and formal convergence analysis.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 72171073) and the Research Fund of Hunan University. The authors thank the anonymous reviewers for their constructive feedback.

References

[1] McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A., 2017. Communication-efficient learning of deep networks from decentralized data, in: Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, pp. 1273–1282.

[2] Yang, Q., Liu, Y., Chen, T., Tong, Y., 2019. Federated machine learning: Concept and applications. ACM Transactions on Intelligent Systems and Technology 10, 1–19.

[3] Lessmann, S., Baesens, B., Seow, H.V., Thomas, L.C., 2015. Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. European Journal of Operational Research 247, 124–136.

[4] Voigt, P., Von dem Bussche, A., 2017. The EU General Data Protection Regulation (GDPR): A Practical Guide. Springer.

[5] Blanchard, P., El Mhamdi, E.M., Guerraoui, R., Stainer, J., 2017. Machine learning with adversaries: Byzantine tolerant gradient descent, in: Advances in Neural Information Processing Systems, pp. 119–129.

[6] Lamport, L., Shostak, R., Pease, M., 1982. The byzantine generals problem. ACM Transactions on Programming Languages and Systems 4, 382–401.

[7] Yin, D., Chen, Y., Kannan, R., Bartlett, P., 2018. Byzantine-robust distributed learning: Towards optimal statistical rates, in: International Conference on Machine Learning, pp. 5650–5659.

[8] El-Mhamdi, E.M., Guerraoui, R., Rouault, S., 2018. The hidden vulnerability of distributed learning in byzantium, in: International Conference on Machine Learning, pp. 3521–3530.

[9] Karimireddy, S.P., He, L., Jaggi, M., 2020. Byzantine-robust learning on heterogeneous datasets via bucketing. arXiv preprint arXiv:2006.09365 doi:10.48550/arXiv.2006.09365.

[10] Baruch, M., Baruch, G., Goldberg, Y., 2019. A little is enough: Circumventing defenses for distributed learning, in: Advances in Neural Information Processing Systems, pp. 8635–8645.

[11] Xie, C., Koyejo, O., Gupta, I., 2020. Fall of empires: Breaking byzantine-tolerant sgd by inner product manipulation, in: Proceedings

of the 36th Conference on Uncertainty in Artificial Intelligence, pp. 261–270.

[12] Shejwalkar, V., Houmansadr, A., 2021. Manipulating the byzantine: Optimizing model poisoning attacks and defenses for federated learning, in: Proceedings of the Network and Distributed System Security Symposium.

[13] Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., Shmatikov, V., 2020. How to backdoor federated learning, in: International Conference on Artificial Intelligence and Statistics, pp. 2938–2948.

[14] Wang, H., Sreenivasan, K., Rajput, S., Vishwakarma, H., Avestimehr, S., Papailiopoulos, D., 2020. Attack of the tails: Yes, you really can backdoor federated learning, in: Advances in Neural Information Processing Systems, pp. 16070–16084.

[15] Pillutla, K., Kakade, S.M., Harchaoui, Z., 2019. Robust aggregation for federated learning. arXiv preprint arXiv:1912.13445 .

[16] Cao, X., Fang, M., Liu, J., Gong, N.Z., 2021. Fltrust: Byzantine-robust federated learning via trust bootstrapping, in: Proceedings of the Network and Distributed System Security Symposium.

[17] Li, W., Xu, F., Liu, J., 2023. Autofl: Automatic byzantine-resilient federated learning via isolation forests, in: Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 1242–1252.

[18] Zhang, Z., Cao, Y., Jia, J., 2022. Autoencoder-based anomaly detection for byzantine attack in federated learning. IEEE Transactions on Neural Networks and Learning Systems 34, 8853–8867.

[19] Rao, R.V., Savsani, V.J., Vakharia, D., 2011. Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems. Computer-Aided Design 43, 303–315.

[20] Yeh, I.C., Lien, C.h., 2009. The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. Expert Systems with Applications 36, 2473–2480.