Highlights

FedACT: Byzantine-Resilient Federated Learning with Explicit Attack Detection for Credit Scoring

Dengjia Li, Han Qiao, Chen Yang, Yuncheng Qiao

- A novel Byzantine detection framework providing explicit attack identification rather than implicit filtering for federated credit scoring.
- Three-zone classification with diversity-constrained committee voting accommodates data heterogeneity while maintaining 89.9% detection recall.
- Accuracy-driven TLBO aggregation with asymmetric reputation updates and Merkle-tree logging enables institutional accountability.
- Perfect precision (100%) on semantic attacks (backdoor and collusion) across four heterogeneous data scenarios.

FedACT: Byzantine-Resilient Federated Learning with Explicit Attack Detection for Credit Scoring

Dengjia Li[a,b], Han Qiao[a], Chen Yang[b], Yuncheng Qiao[c,d,*]

[a]*School of Economics and Management, University of Chinese Academy of Sciences, Beijing, 100190, China*
[b]*China National Clearing Center, The People's Bank of China, Beijing, 100048, China*
[c]*Business School, Shandong University of Technology, Zibo, 255000, China*
[d]*Key Laboratory of High-Performance Distributed Ledger and Digital Finance, Ministry of Education, Changsha, 410082, China*

Abstract

Federated learning enables privacy-preserving collaborative credit scoring, yet existing Byzantine-resilient aggregation methods focus solely on maintaining model accuracy without identifying malicious participants. This limitation is critical in financial applications where regulatory compliance and institutional accountability require explicit attack attribution. We propose FedACT (Federated Autoencoder-Committee-TLBO), a novel framework that prioritizes explicit Byzantine detection over implicit robustness. FedACT employs a three-stage defense pipeline: (1) autoencoder-based anomaly scoring with adaptive MAD thresholding partitions gradients into normal, uncertain, and anomalous zones; (2) diversity-constrained committee voting resolves borderline cases while accommodating legitimate data heterogeneity; and (3) TLBO-based reputation-weighted aggregation with Merkle-tree evidence logging provides accuracy-driven optimization and tamper-evident audit trails. Extensive experiments on two real-world credit datasets demonstrate that FedACT achieves 89.9% detection recall across twelve attack types, with perfect precision on semantic attacks including backdoor and collusion. While traditional robust aggregators achieve marginally higher model accuracy by silently filtering outliers, FedACT uniquely provides the explicit detection capability and auditability essential for regulated financial

[*] Corresponding author
Email address: qiaoyc@sdut.edu.cn (Yuncheng Qiao)

environments.

Keywords: Federated learning, Byzantine detection, Credit scoring, Anomaly detection, Auditability

1. Introduction

Credit scoring is fundamental to financial decision-making, yet traditional centralized approaches face increasing tension between model performance and data privacy requirements. Privacy regulations such as GDPR and China's Personal Information Protection Law impose strict constraints on cross-institutional data sharing, motivating the adoption of federated learning for collaborative credit modeling [1, 2]. In federated learning, institutions train models locally and share only gradient updates, enabling collective intelligence without raw data exchange.

However, the distributed nature of federated learning introduces vulnerability to Byzantine attacks, where malicious participants submit corrupted gradient updates to degrade model performance or inject backdoors [3]. This threat is particularly concerning in credit scoring: adversaries may seek to bias models toward approving high-risk applicants, and model failures can result in regulatory sanctions and reputational damage. The challenge is amplified by data heterogeneity inherent in cross-silo settings, where institutions serve distinct customer segments with non-IID data distributions that can mask or mimic malicious behavior [4].

Existing Byzantine-resilient methods fall into two paradigms: robust statistics and distance-based selection. Robust aggregators such as coordinate-wise median [5], trimmed mean, and geometric median [6] replace vulnerable averaging with estimators that tolerate outliers. Distance-based methods including Krum [3] and Bulyan [7] identify and exclude gradients far from the majority. While these approaches can maintain model accuracy under attack, they share a fundamental limitation: they provide no explicit identification of malicious participants. Outliers are silently filtered without attribution, leaving institutions unable to determine whether anomalies stem from attacks or legitimate heterogeneity, and providing no basis for accountability or regulatory reporting.

This implicit robustness paradigm is insufficient for regulated financial environments. Credit scoring systems require audit trails documenting how decisions were made and who participated in model training. When a defense mechanism silently excludes a gradient, there is no record of whether an attack occurred, which institution was responsible, or what evidence supported the exclusion. This opacity conflicts with regulatory expectations for explainability and accountability in automated financial decision-making.

We propose FedACT, a framework that shifts from implicit robustness to explicit detection. Rather than silently filtering outliers, FedACT explicitly identifies anomalous gradients, classifies them with calibrated uncertainty, and maintains tamper-evident records of all detection decisions. The framework comprises three stages: an autoencoder learns normal gradient manifolds and computes dual-metric anomaly scores, with MAD-based adaptive thresholding partitioning gradients into normal, uncertain, and anomalous zones; a diversity-constrained committee of dissimilar normal clients votes on borderline cases, reducing false positives from legitimate heterogeneity; and verified gradients aggregate via reputation-weighted optimization, with a dynamic reputation system providing long-term incentives and Merkle-tree logging ensuring auditability.

This design reflects a deliberate trade-off. Traditional robust aggregators achieve slightly higher model accuracy by aggressively filtering any gradient that deviates from the majority, but cannot distinguish attacks from heterogeneity. FedACT accepts marginally lower accuracy in exchange for explicit detection capability, enabling institutions to identify malicious participants, accumulate evidence over time through the reputation system, and provide auditors with verifiable records. In financial applications where accountability matters as much as accuracy, this trade-off is appropriate.

Our contributions are: (1) a three-stage Byzantine detection framework providing explicit attack identification with calibrated uncertainty for heterogeneous federated learning; (2) a diversity-constrained committee mechanism that reduces false positives from legitimate data heterogeneity; (3) a reputation system with asymmetric updates and Merkle-tree evidence logging for institutional accountability; and (4) comprehensive evaluation demonstrating 89.9% detection recall with perfect precision on semantic attacks.

2. Related Work

2.1. Byzantine Attacks in Federated Learning

Byzantine attacks in distributed systems date to the foundational work of Lamport et al. [8], who characterized the challenge of reaching consensus when participants may behave arbitrarily. In federated learning, these attacks have evolved from simple perturbations to sophisticated optimization-based strategies. Basic attacks include sign-flipping, Gaussian noise injection, and gradient scaling, which can degrade convergence when defenses assume honest majorities [9]. Optimization-based attacks explicitly craft updates to evade detection: ALIE generates perturbations within benign distribution tails [10], IPM manipulates inner products to fool distance-based methods [11], and MinMax solves constrained optimization to maximize damage while satisfying detectability constraints [12]. Semantic attacks achieve malicious objectives through gradients that may appear statistically normal, including backdoor injection [13], label corruption, and coordinated collusion among multiple malicious clients.

2.2. Byzantine-Resilient Aggregation

Defenses against Byzantine attacks employ several approaches. Robust statistics methods replace averaging with estimators having high breakdown points: coordinate-wise median provides 50% breakdown [5], trimmed mean discards extreme values, and geometric median minimizes sum of distances [6]. These methods assume honest gradients cluster tightly, an assumption violated under heterogeneity. Distance-based selection methods identify outliers through pairwise distances: Krum selects the gradient with minimum distance to its nearest neighbors [3], and Bulyan combines selection with trimmed aggregation [7]. Trust-anchored methods such as FLTrust leverage server-held clean data to compute trust scores [14], though requiring server-side data may be inappropriate in privacy-sensitive domains. Learning-based approaches using autoencoders or isolation forests can capture complex attack patterns but typically make binary decisions without handling the uncertainty inherent in heterogeneous settings [15].

A critical gap in existing methods is the absence of explicit detection capability. All approaches above focus on maintaining model accuracy by filtering outliers, but none provides attribution of which participants are malicious, evidence supporting detection decisions, or audit trails for regulatory compliance. This implicit robustness paradigm is insufficient for financial applications requiring accountability.

2.3. Federated Learning for Credit Scoring

Credit scoring has evolved from logistic regression to gradient boosting and neural networks [16], with federated approaches emerging to address privacy constraints. Yang et al. [17] propose explainable federated learning with blockchain for credit modeling, addressing interpretability requirements. Vertical federated learning enables collaboration when institutions hold different features for overlapping customers [18]. However, Byzantine resilience in federated credit scoring remains underexplored, with existing work assuming honest participation and leaving systems vulnerable to strategic manipulation.

3. The FedACT Framework

3.1. Problem Formulation

Consider N financial institutions collaboratively training a credit scoring model $\mathbf{w} \in \mathbb{R}^d$. Each client i holds private data $\mathcal{D}_i = \{(\mathbf{x}_j, y_j)\}_{j=1}^{n_i}$ where $\mathbf{x}_j$ represents applicant features and $y_j \in \{0, 1\}$ indicates default status. The federated objective minimizes

$$F(\mathbf{w}) = \sum_{i=1}^{N} \frac{n_i}{n} F_i(\mathbf{w}), \quad F_i(\mathbf{w}) = \frac{1}{n_i} \sum_{j=1}^{n_i} \ell(\mathbf{w}; \mathbf{x}_j, y_j) \tag{1}$$

where ℓ is binary cross-entropy and $n = \sum_i n_i$. Training proceeds in rounds: the server broadcasts $\mathbf{w}^{(t)}$, clients perform local updates, and gradients $\mathbf{g}_i^{(t)} = \mathbf{w}^{(t)} - \mathbf{w}_i^{(t)}$ are aggregated.

We assume $M < N/2$ Byzantine clients with white-box knowledge of the defense mechanism, ability to submit arbitrary gradients, and potential for coordination. We evaluate against twelve attacks: three basic (sign-flip, Gaussian, scaling), five optimization-based (little, ALIE, IPM, MinMax, trim), and four semantic (label-flip, backdoor, free-rider, collision). Data heterogeneity includes IID baseline, label skew via Dirichlet allocation ($\alpha = 0.5$), feature distribution shifts, and power-law quantity skew.

FedACT defends against Byzantine attacks through three integrated stages, each implemented as a distinct algorithm. We present each stage with its algorithm embedded in the corresponding subsection.

3.2. Stage 1: Autoencoder-Based Anomaly Detection

The first stage learns a low-dimensional representation of benign gradient distributions using an autoencoder trained on historical normal gradients. For each incoming gradient $\mathbf{g}_i$, we compute two complementary metrics: reconstruction error $e_i = \|\mathbf{g}_i - \psi(\phi(\mathbf{g}_i))\|^2$ captures deviation from the learned manifold, while latent deviation $d_i = \|\phi(\mathbf{g}_i) - \boldsymbol{\mu}_z\|$ measures distance from the normal gradient centroid in latent space. These metrics are max-normalized and combined as $a_i = 0.7\tilde{e}_i + 0.3\tilde{d}_i$, weighting reconstruction error more heavily as it better captures structural violations.

Rather than making binary decisions, we partition gradients into three zones using MAD-based adaptive thresholding. The threshold $\tau = \text{med}(\mathbf{a}) + 3.5 \cdot 1.4826 \cdot \text{MAD}(\mathbf{a})$ adapts to the score distribution each round, with $k = 3.5$ chosen to balance detection sensitivity against false positives under data heterogeneity. A minimum threshold protection ensures $\tau \geq Q_{0.75}(\mathbf{a})$ when MAD is small, preventing over-detection when scores cluster tightly. Gradients with scores below 0.7τ are classified as normal, those above 1.5τ as anomalous, and intermediate scores fall into an uncertain zone requiring further adjudication. This three-zone approach acknowledges the fundamental difficulty of distinguishing sophisticated attacks from legitimate heterogeneity based on anomaly scores alone.

Algorithm 1 Autoencoder-Based Three-Zone Detection

```
Require: Gradients $\mathbf{G} = \{\mathbf{g}_1, \ldots, \mathbf{g}_N\}$, autoencoder $(\phi, \psi)$
Ensure: Normal set $\mathcal{N}$, uncertain set $\mathcal{U}$, anomalous set $\mathcal{A}$
for each $\mathbf{g}_i \in \mathbf{G}$ do
    $e_i \leftarrow \|\mathbf{g}_i - \psi(\phi(\mathbf{g}_i))\|^2$
    $d_i \leftarrow \|\phi(\mathbf{g}_i) - \boldsymbol{\mu}_z\|$
    $a_i \leftarrow 0.7 \cdot \tilde{e}_i + 0.3 \cdot \tilde{d}_i$
end for
$\tau \leftarrow \text{med}(\mathbf{a}) + 3.5 \cdot 1.4826 \cdot \text{MAD}(\mathbf{a})$
$\mathcal{N} \leftarrow \{i : a_i < 0.7\tau\}$
$\mathcal{U} \leftarrow \{i : 0.7\tau \leq a_i < 1.5\tau\}$
$\mathcal{A} \leftarrow \{i : a_i \geq 1.5\tau\}$
return $\mathcal{N}, \mathcal{U}, \mathcal{A}$
```

3.3. Stage 2: Diversity-Constrained Committee Voting

Borderline cases in the uncertain zone are resolved through committee voting. We select $K = 5$ committee members from the normal set $\mathcal{N}$, maximizing diversity to ensure the committee represents varied but legitimate gradient directions. The first member has highest reputation; subsequent members are chosen to minimize maximum cosine similarity to already-selected members. Each committee member votes on whether an uncertain gradient is anomalous based on cosine similarity: if similarity falls below threshold $\gamma = 0.5$, the member votes to flag the gradient. The elevated threshold (compared to typical similarity thresholds) accounts for natural gradient divergence under data heterogeneity. Majority voting determines the final classification.

This mechanism addresses a key limitation of single-threshold detection. Under heterogeneity, some legitimate gradients will have elevated anomaly scores simply because they represent minority data distributions. A diverse committee drawn from normal clients provides multiple reference points, reducing false positives while maintaining sensitivity to genuine attacks that deviate from all normal directions. Additionally, our evidence accumulation mechanism requires consistent anomaly detection across multiple rounds (3 out of 5 consecutive rounds) before confirming a client as malicious, further reducing transient false positives.

Algorithm 2 Diversity-Constrained Committee Voting

```
**Require:** Normal set $\mathcal{N}$, uncertain set $\mathcal{U}$, gradients $\{\mathbf{g}_i\}$, reputations $\{\rho_i\}$
**Ensure:** Updated normal set $\mathcal{N}'$, anomalous set $\mathcal{A}'$
$c_1 \leftarrow \arg\max_{i \in \mathcal{N}} \rho_i$
$\mathcal{C} \leftarrow \{c_1\}$
**for** $k = 2$ to $K$ **do**
    $c_k \leftarrow \arg\min_{i \in \mathcal{N} \setminus \mathcal{C}} \max_{j \in \mathcal{C}} \cos(\mathbf{g}_i, \mathbf{g}_j)$
    $\mathcal{C} \leftarrow \mathcal{C} \cup \{c_k\}$
**end for**
$\mathcal{N}' \leftarrow \mathcal{N}$, $\mathcal{A}' \leftarrow \emptyset$
**for** each $u \in \mathcal{U}$ **do**
    votes $\leftarrow \sum_{c \in \mathcal{C}} \mathbf{1}[\cos(\mathbf{g}_u, \mathbf{g}_c) < 0.3]$
    **if** votes$/K \leq 0.5$ **then**
        $\mathcal{N}' \leftarrow \mathcal{N}' \cup \{u\}$
    **else**
        $\mathcal{A}' \leftarrow \mathcal{A}' \cup \{u\}$
    **end if**
**end for**
**return** $\mathcal{N}'$, $\mathcal{A}'$
```

3.4. Stage 3: TLBO-Based Robust Aggregation with Reputation

Verified gradients from $\mathcal{N}$ aggregate via Teaching-Learning-Based Optimization with reputation weighting. TLBO is a population-based metaheuristic that iteratively improves candidate solutions through two phases. In our adaptation, each gradient serves as a learner, and the optimization objective is to maximize model accuracy rather than gradient similarity.

The fitness function for each gradient is its contribution to test accuracy. We temporarily apply a gradient to the global model, evaluate accuracy on a validation subset, then restore the original parameters. In the teacher phase, the gradient achieving highest accuracy becomes the teacher, and other gradients move toward it. In the learner phase, pairs of gradients interact: each gradient moves toward better-performing peers and away from worse performers. After $T = 10$ iterations, this process yields the gradient that maximizes model accuracy.

The reputation system provides long-term memory and incentives. Reputations $\rho_i \in [0.1, 2.0]$ update asymmetrically: normal clients receive additive rewards $\rho_i \leftarrow \rho_i + 0.05\xi_i$ proportional to their alignment with consensus, while detected anomalies suffer multiplicative penalties $\rho_i \leftarrow 0.7\rho_i$. This asymmetry means a single anomalous detection reduces reputation by 30%, requiring approximately six honest rounds to recover. Persistent attackers accumulate reputation damage, progressively reducing their influence even if they occasionally evade detection.

All detection decisions are logged to a Merkle tree, creating a tamper-evident audit trail. Each entry records the round, client identities, anomaly scores, zone classifications, and reputation updates.

The Merkle-tree hash chain ensures that any modification to historical records is immediately detectable, enabling auditors to verify the integrity of the complete detection history. This cryptographic commitment supports regulatory compliance requirements for financial model governance and provides irrefutable evidence for institutional accountability.

4. Experiments

4.1. Experimental Setup

We evaluate FedACT on two real-world credit scoring datasets: (1) UCI Credit Card Default [19] containing 30,000 Taiwan credit card clients with 23 features, and (2) Xinwang Bank dataset comprising 50,000 loan applicants with 35 features from a Chinese commercial bank. The credit scoring model is a multi-layer perceptron (MLP) with hidden dimensions [512, 256, 128, 64, 32, 16]. Training proceeds for 100 communication rounds with 5 local epochs per round, across $N = 10$ clients including $M = 3$ malicious attackers (30% Byzantine ratio). All experiments are repeated 3 times with different random seeds on NVIDIA RTX 4090 GPUs, and we report mean values.

4.2. Detection Performance

Table 1 reports FedACT's detection performance across attack categories, averaged over both datasets and four heterogeneity scenarios.

Table 1: Detection performance by attack category.

Category	Attack	Precision	Recall	F1
Basic	Sign-flip	0.302	0.939	0.457
	Gaussian	0.301	0.955	0.458
	Scaling	0.328	0.764	0.457
Optimization	Little	0.301	0.956	0.458
	ALIE	0.300	0.949	0.456
	IPM	0.431	0.925	0.582
	MinMax	0.312	0.883	0.458
	Trim	0.300	0.869	0.443
Semantic	Label-flip	0.301	0.955	0.458
	Backdoor	1.000	0.897	0.946
	Free-rider	0.300	0.950	0.456
	Collision	1.000	0.743	0.847
Overall		0.349	0.899	0.503

FedACT achieves 89.9% overall recall, detecting the vast majority of attacks across all categories. For semantic attacks, precision reaches 100% on backdoor and collision attacks, with F1 scores of 0.946 and 0.847 respectively. These attacks produce distinctive gradient patterns that the autoencoder captures effectively. The relatively lower precision on basic and optimization attacks reflects the conservative detection strategy: FedACT flags more gradients as suspicious to ensure high recall, accepting some false positives from legitimate heterogeneity. This trade-off is appropriate for financial applications where failing to detect an attack carries greater risk than occasionally flagging honest participants for review.

4.3. Comparison with Robust Aggregators

Table 2 compares model accuracy across defense methods under representative attacks.

Table 2: Model accuracy (%) under attacks. Best per row in bold.

Attack	Median	Trim	Krum	M-Krum	Bulyan	RFA	FedACT
Sign-flip	**84.75**	84.76	84.78	84.97	84.71	84.71	83.48
Gaussian	84.82	**84.90**	84.71	84.85	84.74	84.75	83.56
ALIE	84.67	84.71	84.79	**84.86**	84.82	84.66	83.66
Backdoor	84.78	84.77	**84.83**	**84.83**	84.78	84.64	82.67
Collision	84.80	**84.96**	84.70	84.70	84.86	84.86	73.12
Average	84.77	**84.83**	84.79	84.82	84.78	84.73	81.74

Traditional robust aggregators achieve approximately 3% higher accuracy than FedACT. This gap reflects a fundamental design difference: robust aggregators silently filter any gradient that deviates from the majority, effectively treating all heterogeneity as noise to be suppressed. FedACT explicitly detects and classifies anomalies, maintaining records of which participants were flagged and why. The accuracy trade-off is the cost of providing detection capability.

Notably, FedACT shows significant accuracy degradation under collision attacks (73.12%), where coordinated malicious clients form their own apparent majority. This scenario challenges all methods but particularly affects FedACT's committee voting when colluding attackers contaminate the normal set. Future work should address collusion-resistant committee selection.

4.4. Heterogeneity Robustness

Table 3 demonstrates stable detection performance across data heterogeneity scenarios.

Detection recall remains stable at approximately 90% across all heterogeneity types, demonstrating that the three-zone classification and committee voting effectively accommodate legitimate gradient variation. The slightly

Table 3: Performance under heterogeneity scenarios.

Scenario	Precision	Recall	F1	Accuracy
IID	0.425	0.899	0.537	80.16
Label skew	0.434	0.899	0.544	83.36
Feature skew	0.437	0.905	0.546	81.91
Quantity skew	0.430	0.892	0.531	81.69
Average	0.431	0.899	0.540	81.78

higher precision under heterogeneous scenarios compared to IID may seem counterintuitive but reflects that heterogeneity makes attacks more distinctive: when honest gradients vary naturally, malicious gradients that deviate in unusual ways become more identifiable.

4.5. Ablation Study

Table 4 isolates the contribution of each component.

Table 4: Ablation study results.

Configuration	Precision	Recall	F1	Accuracy
FedACT (full)	0.349	0.899	0.503	81.78
Without autoencoder	0.285	0.712	0.407	82.45
Without committee	0.312	0.921	0.467	80.92
Without TLBO	0.349	0.899	0.503	80.45
FedAvg (no defense)	—	—	—	78.23

The autoencoder is essential: removing it reduces recall by 18.7 percentage points, demonstrating that learned representations capture attack patterns that simpler heuristics miss. The committee mechanism improves precision by 3.7 points while slightly reducing recall, confirming its role in filtering false positives from heterogeneity. TLBO aggregation improves accuracy by 1.3 points compared to simple averaging but does not affect detection metrics.

5. Conclusion

This paper presents FedACT, a Byzantine-resilient federated learning framework that prioritizes explicit attack detection over implicit robustness for credit scoring applications. By integrating autoencoder-based anomaly scoring with three-zone classification, diversity-constrained committee voting, and accuracy-driven TLBO aggregation with Merkle-tree evidence logging, FedACT achieves 89.9% detection recall with perfect precision (100%) on semantic attacks including backdoor and collusion. While traditional robust aggregators achieve marginally higher model accuracy through silent outlier filtering, FedACT uniquely provides the explicit detection capability and auditability essential for regulated financial environments.

The framework represents a deliberate and principled trade-off appropriate for financial applications where institutional accountability matters as much as model performance. With increasing regulatory requirements that automated decision systems maintain comprehensive audit trails and provide explanations for their behavior, FedACT addresses this critical need by explicitly identifying anomalous participants, accumulating evidence through the asymmetric reputation system, and maintaining tamper-evident cryptographic records for auditors.

Limitations of the current work include vulnerability to sophisticated collusion attacks where malicious clients form apparent majorities, and the inherent tension between detection sensitivity and false positive rates under extreme data heterogeneity. Future research directions include collusion-resistant committee selection mechanisms, adversarial training for improved autoencoder robustness, and formal convergence guarantees under Byzantine presence with theoretical attack resilience bounds.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 72171073).

References

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

[2] G. Long, Y. Tan, J. Jiang, C. Zhang, Federated learning for open banking, arXiv preprint arXiv:2004.10316 (2020).

[3] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, J. Stainer, Machine learning with adversaries: Byzantine tolerant gradient descent, in: Advances in Neural Information Processing Systems, volume 30, 2017, pp. 119–129.

[4] Q. Li, Y. Diao, Q. Chen, B. He, Federated learning on non-iid data silos: An experimental study, IEEE Transactions on Parallel and Distributed Systems 33 (2022) 3446–3457.

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

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

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

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

[9] M. Fang, X. Cao, J. Jia, N. Gong, Local model poisoning attacks to byzantine-robust federated learning, in: 29th USENIX Security Symposium, 2020, pp. 1605–1622.

[10] M. Baruch, G. Baruch, Y. Goldberg, A little is enough: Circumventing defenses for distributed learning, in: Advances in Neural Information Processing Systems, volume 32, 2019, pp. 8635–8645.

[11] C. Xie, O. Koyejo, I. Gupta, Fall of empires: Breaking byzantine-tolerant sgd by inner product manipulation, in: Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence, 2020, pp. 261–270.

[12] V. Shejwalkar, A. Houmansadr, Manipulating the byzantine: Optimizing model poisoning attacks and defenses for federated learning, in:

Proceedings of the Network and Distributed System Security Symposium, 2021.

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

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

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

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

[17] F. Yang, M. Z. Abedin, P. Hájek, An explainable federated learning and blockchain-based secure credit modeling method, European Journal of Operational Research 317 (2024) 449–467.

[18] Y. Chen, X. Liu, T. Wang, C. Niu, Q. Yang, Fedbcd: A communication-efficient collaborative learning framework for distributed features, IEEE Transactions on Signal Processing 70 (2022) 4277–4290.

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

Algorithm 3 TLBO Aggregation with Reputation Updates

```
Require: Normal gradients $\{\mathbf{g}_i : i \in \mathcal{N}'\}$, reputations $\{\rho_i\}$, iterations $T$
Ensure: Aggregated gradient $\mathbf{g}^*$, updated reputations $\{\rho'_i\}$
Initialize learners: $\mathcal{L} \leftarrow \{\mathbf{g}_i : i \in \mathcal{N}'\}$
for $t = 1$ to $T$ do
    // Compute fitness (accuracy) for each learner
    for each $\mathbf{l}_i \in \mathcal{L}$ do
        $f_i \leftarrow \text{EvaluateAccuracy}(\mathbf{l}_i)$
    end for
    // Teacher phase: learn from best accuracy
    teacher $\leftarrow \arg\max_{\mathbf{l} \in \mathcal{L}} f(\mathbf{l})$
    $\bar{\mathbf{l}} \leftarrow \frac{1}{|\mathcal{L}|} \sum_{\mathbf{l} \in \mathcal{L}} \mathbf{l}$
    $TF \sim \text{Uniform}\{1, 2\}$
    for each $\mathbf{l}_i \in \mathcal{L}$ do
        $\mathbf{l}'_i \leftarrow \mathbf{l}_i + r(\text{teacher} - TF \cdot \bar{\mathbf{l}})$ where $r \sim \mathcal{U}(0, 1)$
        if EvaluateAccuracy($\mathbf{l}'_i$) $> f_i$ then
            $\mathbf{l}_i \leftarrow \mathbf{l}'_i$
        end if
    end for
    // Learner phase: mutual learning based on accuracy
    for each $\mathbf{l}_i \in \mathcal{L}$ do
        $j \sim \text{Uniform}(\{k : k \neq i\})$
        $\mathbf{l}'_i \leftarrow \mathbf{l}_i + r(\mathbf{l}_j - \mathbf{l}_i)$ if $f_j > f_i$, else $\mathbf{l}_i + r(\mathbf{l}_i - \mathbf{l}_j)$
        if EvaluateAccuracy($\mathbf{l}'_i$) $> f_i$ then
            $\mathbf{l}_i \leftarrow \mathbf{l}'_i$
        end if
    end for
end for
$\mathbf{g}^* \leftarrow \arg\max_{\mathbf{l} \in \mathcal{L}} \text{EvaluateAccuracy}(\mathbf{l})$
// Update reputations
for each client $i$ do
    if $i \in \mathcal{N}'$ then
        $\xi_i \leftarrow (\cos(\mathbf{g}_i, \mathbf{g}^*) + 1)/2$
        $\rho'_i \leftarrow \min(\rho_i + 0.05\xi_i, 2.0)$
    else
        $\rho'_i \leftarrow \max(\rho_i \times 0.7, 0.1)$
    end if
end for
return $\mathbf{g}^*$, $\{\rho'_i\}$
```