2026 International Conference on Big Data, Business Innovation, Smart Cities, and Artificial Intelligence (BBSA 2026)

Article

Privacy-Preserving Techniques in Credit Risk Assessment: A Comparative Analysis of Differential Privacy, Federated Learning, and Homomorphic Encryption

Zhi Luo [1,*]

[1] Business Analytics, Columbia University, New York, NY, USA
* Correspondence: Zhi Luo, Business Analytics, Columbia University, New York, NY, USA

Abstract: The proliferation of machine learning in financial services has intensified concerns regarding consumer data privacy during credit risk assessment. This paper presents a comparative study of three privacy-preserving paradigms relevant to credit risk assessment: differential privacy, federated learning, and homomorphic encryption. Through empirical evaluation on a large-scale, representative credit dataset across multiple privacy-preserving configurations, we examine the privacy-utility trade-offs inherent in each approach. Our experimental framework assesses prediction accuracy, computational overhead, and privacy guarantees across multiple configurations. Results demonstrate that differential privacy with epsilon=8.65 achieves a balanced privacy-utility tradeoff with competitive discriminative performance, while homomorphic encryption preserves near-baseline predictive performance during encrypted inference. Federated learning enables collaborative model training across institutions without sharing raw data. The analysis reveals that privacy parameter selection critically impacts model utility, with differential privacy offering quantifiable privacy budgets and homomorphic encryption providing cryptographic security guarantees. These findings provide actionable guidance for financial institutions navigating regulatory compliance requirements while maintaining effective risk management capabilities.

Keywords: Privacy-preserving computation; Credit risk assessment; Differential privacy; Federated learning; Homomorphic encryption

Received: 26 February 2026
Revised: 21 April 2026
Accepted: 03 May 2026
Published: 06 May 2026

1. Introduction

1.1. Background and Motivation

The financial services industry has undergone a substantial transformation through the integration of machine learning algorithms for credit risk assessment. Financial institutions process vast quantities of sensitive consumer data daily, including income levels, employment history, transaction patterns, and debt obligations. Traditional centralized data processing architectures aggregate this information into unified repositories, creating concentrated privacy vulnerabilities. Recent data breaches affecting major credit bureaus have exposed millions of consumer records, underscoring the inadequacy of conventional security measures.

Regulatory frameworks worldwide have responded to these privacy concerns through stringent data protection legislation. The General Data Protection Regulation in Europe and the California Consumer Privacy Act in the United States impose strict requirements on the handling of financial data. The U.S. Consumer Financial Protection Bureau has issued guidance emphasizing the necessity of privacy-by-design principles in algorithmic credit scoring. These regulations mandate that financial institutions

implement technical safeguards to prevent unauthorized access while maintaining service quality.

The tension between data utility and privacy protection presents fundamental challenges for credit risk modeling. Differential privacy mechanisms introduce calibrated noise to protect individual records, establishing mathematical frameworks for quantifiable privacy guarantees [1]. Multi-party computation protocols enable collaborative analysis without revealing raw data, allowing institutions to jointly train models while maintaining data sovereignty [2]. Encryption-based approaches allow computation on ciphertext without decryption, providing cryptographic security for sensitive financial operations [3]. Each technique offers distinct advantages regarding privacy guarantees, computational efficiency, and integration complexity. Understanding these tradeoffs becomes essential for financial institutions designing compliant credit assessment systems.

1.2. Research Objectives and Contributions

This research conducts a rigorous comparative evaluation of privacy-preserving techniques applied to credit risk assessment. The primary objective is to measure privacy-utility trade-offs across implementations of differential privacy, federated learning, and homomorphic encryption. We quantify how privacy parameter selections impact prediction accuracy, processing latency, and privacy assurance levels. The analysis examines real-world deployment considerations, including scalability constraints, regulatory alignment, and operational integration requirements.

The study makes several contributions to privacy-preserving financial analytics. We establish empirical benchmarks comparing the performance of privacy techniques on public credit risk datasets, with the main comparative experiments conducted on the Home Credit dataset. Our experimental framework evaluates computational overhead across varying privacy configurations, providing practical guidance for implementation decisions. The research identifies optimal privacy budget allocations balancing regulatory compliance with predictive performance. We document the sensitivity of model accuracy to variations in privacy parameters, enabling informed trade-off decisions.

The comparative analysis synthesizes insights from distributed machine learning, cryptographic protocols, and statistical privacy theory. We assess the applicability of each technique to different institutional scenarios, considering factors such as data distribution patterns, collaboration requirements, and trust assumptions. The findings illuminate pathways for financial institutions to adopt privacy-enhancing technologies while preserving risk assessment capabilities. This work addresses critical knowledge gaps regarding the practical deployment of privacy-preserving credit scoring in regulated environments.

1.3. Paper Organization

The subsequent sections present a comprehensive analysis of privacy-preserving credit assessment techniques. Section 2 reviews foundational concepts in privacy-enhancing technologies and their application to financial machine learning. The discussion encompasses theoretical frameworks for differential privacy, architectural patterns for federated learning, and cryptographic principles underlying homomorphic encryption. Section 3 describes our experimental methodology, including privacy mechanism implementations, dataset characteristics, and evaluation metrics.

Section 4 presents empirical results from comparative experiments, analyzing prediction accuracy, privacy guarantees, and computational performance. The analysis examines privacy-utility curves across parameter ranges, identifying optimal configurations for different operational requirements. Section 5 synthesizes key findings, discusses practical implications for financial institutions, and outlines future research directions. The conclusion addresses limitations of current approaches and emerging opportunities in privacy-preserving credit analytics.

2. Related Work and Background

2.1. Privacy-Preserving Machine Learning Fundamentals

Privacy-enhancing technologies have evolved substantially over the past decade, driven by rising data breaches and regulatory pressures. The financial sector faces particular challenges in balancing analytical capabilities with consumer protection obligations. Traditional anonymization techniques, such as data masking and pseudonymization, provide limited protection against sophisticated re-identification attacks. Research has demonstrated that auxiliary information enables adversaries to link anonymized records to specific individuals, undermining privacy assurances.

Modern privacy-preserving approaches employ cryptographic and statistical mechanisms providing formal privacy guarantees. Recent studies have demonstrated the effectiveness of differential privacy in scenarios with data imbalance and heterogeneous distributions [4]. The framework quantifies privacy loss through epsilon parameters, enabling measurable privacy-utility tradeoffs. Smaller epsilon values provide stronger privacy but introduce more noise affecting model accuracy. Financial applications require careful calibration balancing regulatory requirements with predictive performance needs.

Secure multi-party computation protocols enable collaborative analysis across organizational boundaries without exposing raw data. Novel federated learning approaches with knowledge transfer have shown promise in credit scoring applications, achieving high accuracy while preserving data privacy [5]. These techniques allow multiple parties to jointly compute functions on their private inputs while revealing only the final result. Applications in credit assessment enable institutions to pool data resources for improved model training without compromising competitive advantages or violating data sharing restrictions. Hybrid approaches combining homomorphic encryption and differential privacy techniques provide enhanced security guarantees for federated learning paradigms [6].

2.2. Differential Privacy in Financial Applications

Differential privacy mechanisms add calibrated statistical noise to data or query results, obscuring individual contributions while preserving aggregate patterns. The Gaussian mechanism injects noise drawn from normal distributions scaled to query sensitivity. Privacy budget accounting tracks cumulative privacy loss across multiple queries, ensuring total leakage remains bounded. Privacy-preserving synthetic data generation using differential privacy has emerged as a practical approach for regulatory compliance in financial services, enabling data sharing while protecting individual privacy [7].

The privacy-utility tradeoff in differential privacy manifests through the relationship between epsilon values and model accuracy. Lowering epsilon provides stronger privacy guarantees but degrades prediction performance by increasing noise. Comprehensive surveys of advances and open problems in federated learning have explored optimal strategies for selecting privacy parameters, providing guidance for practitioners implementing privacy-preserving systems [8]. Advanced composition theorems enable privacy budget allocation across model training iterations, supporting iterative algorithms while maintaining overall privacy bounds.

Implementation challenges include determining appropriate sensitivity calibrations for different feature types and managing privacy budget consumption in deep learning pipelines. Recent research has developed adaptive privacy mechanisms adjusting noise levels based on data characteristics. Rényi differential privacy provides tighter privacy accounting for certain classes of algorithms, enabling better privacy-utility trade-offs. Financial institutions must consider regulatory interpretation of privacy parameters when deploying differential privacy systems.

2.3. Federated Learning and Homomorphic Encryption

Federated learning architectures distribute model training across multiple data holders, aggregating local updates without centralizing raw data. Horizontal federated learning applies to scenarios in which institutions hold different samples but share the same feature space. Vertical federated learning addresses cases where parties possess

different features for overlapping entities. Innovative approaches that combine federated learning with blockchain technology have demonstrated enhanced security and transparency in credit-scoring applications [9]. Credit assessment applications leverage federated learning to combine consumer data from banks, credit bureaus, and alternative data providers while maintaining data sovereignty.

Communication efficiency becomes critical in federated settings due to bandwidth constraints and latency requirements. Model compression techniques reduce update sizes transmitted between clients and coordination servers. Secure aggregation protocols employ cryptographic tools preventing the server from observing individual client contributions. Empirical studies have examined the performance of federated learning under various data heterogeneity conditions common in financial applications, with comprehensive evaluations demonstrating the effectiveness of privacy-preserving approaches [10]. Systematic literature reviews have synthesized key findings on federated learning techniques applied to credit risk management, identifying best practices and implementation challenges [11].

Homomorphic encryption schemes enable computation directly on encrypted data, producing encrypted results that can be decrypted to yield correct plaintext. Fully homomorphic encryption supports arbitrary computations but incurs substantial performance overhead. Partially homomorphic schemes optimize specific operations relevant to machine learning, such as addition and multiplication. Explainable federated learning frameworks integrating blockchain technology have been proposed for secure credit modeling, addressing interpretability requirements alongside privacy protection [12]. The computational costs remain significant, requiring hardware acceleration or algorithmic optimizations for practical deployment.

3. Methodology and Experimental Design

3.1. Privacy-Preserving Techniques Implementation

3.1.1. Differential Privacy Configuration

Our differential privacy implementation employs the Gaussian mechanism with Rényi differential privacy accounting to track cumulative privacy expenditure across training epochs. The experiments evaluate three target privacy regimes, corresponding to reported effective privacy budgets of 5.74, 8.65, and 14.13 under the adopted accounting configuration. For a query function f with L2-sensitivity Δf, the Gaussian mechanism adds noise sampled from $N(0, \sigma^2)$ where $\sigma = \Delta f \cdot \sqrt{(2\ln (1.25/\delta))}$ / epsilon. The delta parameter represents the failure probability and is set to 10^{-5} throughout the experiments.

Privacy accounting employs a moments-based accountant methodology to track Rényi divergence between adjacent dataset distributions. For gradient-based learning with batch sampling, the sensitivity Δf depends on the gradient clipping threshold C. We implement per-sample gradient clipping with C = 1.0, bounding the L2-norm of individual gradients before aggregation. The privacy cost accumulates across training iterations in accordance with standard privacy accounting procedures. In practice, the reported privacy budgets are tracked at the implementation level using RDP-based accounting with a fixed delta of 10^{-5}, rather than being derived solely from a simplified closed-form approximation.

The implementation integrates differential privacy into stochastic gradient descent through a DP-SGD-style training procedure adapted to the credit default prediction setting [13]. At each iteration t, the privatized gradient update is given by $gt = (1/B) \cdot \Sigma i \text{ clip}(\nabla L(x\_i), C) + N(0, \sigma^2 C^2 I)$, where B denotes the batch size, clip bounds the gradients to a C-norm, and the noise term ensures differential privacy. Learning rate scheduling adjusts to compensate for noise-induced variance in the gradient, thereby maintaining convergence stability. We evaluate three privacy regimes corresponding to epsilon values, measuring impact on credit default prediction accuracy.

3.1.2. Federated Learning Architecture

The federated learning framework distributes the primary Home Credit dataset across five simulated financial institutions, each maintaining a local partition of approximately 61,000--62,000 records under a non-IID allocation scheme. The architecture follows the horizontal federated learning paradigm, in which institutions share identical feature spaces but have distinct customer populations. A central aggregation server coordinates training rounds without accessing raw data, receiving only encrypted model updates from participants.

Each training round proceeds through local computation and secure aggregation phases. Local clients perform E=20 epochs of gradient descent on private datasets, computing model updates $\Delta wk = wk^{\{t+1\}} - wkt$ where wk represents client k's model parameters. Clients encrypt updates using additive secret sharing, splitting Δwk into shares transmitted through independent channels. The aggregation server combines encrypted shares, computing the global update: $\Delta wglobal = (1/K) \cdot \Sigma k\ nk/N \cdot \Delta wk$, where K denotes participating clients, nk indicates client k's sample count, and N represents total samples.

Secure aggregation employs threshold cryptography, preventing the server from observing individual contributions. Each client generates random pairwise masks that are shared with other participants, and constructs the masked update: $\Delta w\_k + \Sigma\_\{j \neq k\} (mask\_\{k,j\} - mask\_\{j,k\})$. The server aggregates masked updates, with pairwise masks canceling to reveal only the weighted average. Communication costs scale with model size and participant count, requiring bandwidth optimization through gradient compression and sparse update transmission.

Data heterogeneity across institutions introduces challenges for federated optimization. Non-IID data distributions cause client models to drift toward local optima, slowing convergence. We implement FedProx regularization adding a proximal term to local objectives: $Lk(w) + (\mu/2)||w - wglobal||^2$, where μ controls drift tolerance. This modification bounds local model deviation from the global model, improving convergence under heterogeneous data conditions. Recent advances in secure multiparty computation protocols based on homomorphic encryption have demonstrated effectiveness in blockchain applications, providing additional security layers for federated architectures [14]. The framework evaluates the number of communication rounds required for convergence, measuring trade-offs between accuracy and communication efficiency.

3.1.3. Homomorphic Encryption Mechanisms

The homomorphic encryption implementation utilizes the CKKS scheme, which supports approximate arithmetic on encrypted real numbers. The scheme represents plaintexts as polynomials in the ring $RQ = ZQ[X]/(X^{N+1}$, where $N=2^{14}$ provides the polynomial degree, and Q denotes the coefficient modulus. Encryption employs the public key (b, a) where $b = -a \cdot s + e \pmod Q$, with secret key s, public element a, and error e sampled from discrete Gaussian distributions.

For a plaintext message m, encryption produces ciphertext ct = (c0, c1) where $c0 = b \cdot u + e0 + \Delta m$ and $c1 = a \cdot u + e1$, with Δ representing the encoding scale factor, u denoting a random polynomial, and e0, e1 indicating fresh errors. Homomorphic addition and multiplication operations maintain the ring structure: ctadd = ct1 + ct2, and ctmult requires relinearization to control ciphertext size growth. The CKKS scheme supports depth-L computations with multiplicative depth determining circuit complexity.

Credit scoring inference over encrypted data proceeds by encrypting feature vectors and evaluating the trained model homomorphically. For a logistic regression model with weights w and bias b, prediction follows: $y = \sigma(w \cdot x + b)$ where σ denotes the sigmoid function. Polynomial approximation replaces the sigmoid with a degree-3 polynomial: $\sigma(z) \approx 0.5 + 0.25z - 0.0625z^3$, enabling homomorphic evaluation. The approximation maintains accuracy within ± 0.01 for $z \in [-5, 5]$, which covers the typical range of credit-scoring outputs.

Computational overhead stems from polynomial arithmetic operations and noise growth management. Each homomorphic multiplication increases noise magnitude, requiring bootstrapping for deep circuits. Our implementation employs parameter selection balancing security (λ=128 bits), precision (40-bit mantissa), and multiplicative depth (L=5). Performance optimization leverages SIMD packing, encoding multiple plaintexts into a single ciphertext, and amortizing encryption costs across batch predictions. Advanced techniques combining zero-knowledge proofs and functional encryption enable privacy-preserving distributed intelligent credit scoring with enhanced computational efficiency [15]. The framework measures inference latency and throughput by comparing encrypted and plaintext evaluations.

3.2. Experimental Setup and Datasets

3.2.1. Credit Risk Datasets

Experiments employ three publicly available credit datasets representing diverse lending scenarios and borrower characteristics. The German Credit dataset contains 1,000 loan records with 20 features, including account status, credit history, loan purpose, and employment duration. The Australian Credit dataset comprises 690 instances with 14 mixed numerical and categorical features. The Home Credit Default Risk dataset provides 307,511 application records with 122 features spanning demographic information, financial indicators, and historical credit behavior. Among the three datasets, the Home Credit Default Risk dataset serves as the primary benchmark for model training, privacy-utility comparison, and federated partition simulation because of its scale and richer feature space, while the German and Australian datasets are used as supplementary reference datasets for assessing the general applicability of the preprocessing and task setting.

Feature preprocessing standardizes numerical attributes to zero mean and unit variance, mitigating scale disparities. Categorical variables undergo one-hot encoding, expanding feature dimensionality while enabling compatibility with linear models. Missing value imputation employs median substitution for numerical features and mode replacement for categorical attributes. Class imbalance mitigation applies stratified sampling during train-test splitting, maintaining default rate proportions across partitions.

Data distribution for federated learning simulations partitions the Home Credit dataset across five institutions using a Dirichlet allocation with concentration parameter α = 0.5. This approach generates realistic non-IID distributions where institutions specialize in different borrower segments. Institution 1 receives predominantly low-risk applicants, Institution 3 handles high-risk subprime borrowers, while Institutions 2, 4, and 5 maintain mixed portfolios. Sample sizes range from 61,000 to 62,000 per institution, reflecting typical contributions from consortium members (As shown in Table 1).

Table 1. Dataset Characteristics and Distribution Statistics

Dataset	Total Samples	Features	Default Rate	Train Split	Test Split	Institutions
German Credit	1,000	20	30.0%	700	300	Single
Australian Credit	690	14	44.5%	483	207	Single
Home Credit	307,511	122	8.1%	246,009	61,502	5 (Federated)

Institution 1 (Low Risk)	61,502	122	4.2%	49,202	12,300	-
Institution 2 (Mixed)	61,502	122	8.0%	49,202	12,300	-
Institution 3 (High Risk)	61,502	122	14.5%	49,202	12,300	-
Institution 4 (Mixed)	61,502	122	7.8%	49,202	12,300	-
Institution 5 (Mixed)	61,503	122	8.2%	49,203	12,300	-

As summarized in Table 1, the institutional partitions are approximately balanced in size, while borrower composition differs across institutions due to the Dirichlet-based non-IID simulation.

3.2.2. Model Architecture and Training Configuration

For the primary Home Credit experiments, the base credit risk model employs a three-layer feedforward neural network with architecture [122, 64, 32, 1]. The input dimensionality is set to the Home Credit feature space after preprocessing, yielding 122 standardized input features in the primary experimental setting. Hidden layers utilize ReLU activation functions, introducing nonlinearity while maintaining gradient flow. The output layer employs sigmoid activation, producing default probability predictions in [0, 1]. Binary cross-entropy loss quantifies prediction errors: $L = -(1/N)\Sigma[yi \cdot \log(pi) + (1-yi) \cdot \log(1-pi)]$, where yi denotes true labels and pi represents predicted probabilities.

Training employs the Adam optimizer with initial learning rate $\eta=0.001$, exponential decay rates $\beta1=0.9$ and $\beta2=0.999$, and epsilon=10^{-8} for numerical stability. Batch size B=256 balances computational efficiency with gradient variance reduction. The centralized baseline model trains for 100 epochs, with early stopping based on validation loss; training halts when improvement stagnates for 10 consecutive epochs. This configuration achieves convergence within 60-80 epochs on the Home Credit dataset.

Differential privacy configurations modify the training procedure through gradient clipping and noise injection. Clipping threshold C=1.0 bounds per-sample gradient L2-norms before aggregation. Noise scale σ adjusts according to target epsilon values: {5.74: $\sigma=1.5$, 8.65: $\sigma=1.0$, 14.13: $\sigma=0.6$}. Privacy accounting tracks cumulative epsilon consumption across training iterations and terminates when the budget is exhausted. The learning rate is increased to $\eta=0.003$ to compensate for noise-induced gradient variance, thereby maintaining optimization progress.

Federated learning training spans 100 communication rounds, with each round conducting E=20 local epochs per institution. Local batch sizes match the centralized configuration at B=256. The FedProx proximal term uses $\mu=0.01$ to balance local adaptation and global consistency. Convergence monitoring evaluates aggregated model performance on held-out validation data every 5 rounds. Communication costs accumulate as the gradient size transmitted per round increases, measured in megabytes, accounting for floating-point precision (As shown in Table 2).

Table 2. Model Training Hyperparameters Across Privacy Configurations

Configuration	Learning Rate	Batch Size	Epochs/Rounds	Gradient Clip	Noise Scale	Proximal μ
Centralized Baseline	0.001	256	100	-	-	-
DP $\varepsilon = 5.74$	0.003	256	100	1.0	1.5	-
DP $\varepsilon = 8.65$	0.003	256	100	1.0	1.0	-
DP $\varepsilon = 14.13$	0.003	256	100	1.0	0.6	-
Federated (Standard)	0.001	256	100 rounds	-	-	0.01
Federated + DP	0.003	256	100 rounds	1.0	1.0	0.01
HE Inference Scenario (trained plaintext model)	0.001	256	100	-	-	-

Note: The homomorphic encryption configuration is reported as an encrypted inference setting built on a trained plaintext model, rather than as a separate end-to-end encrypted training pipeline.

3.2.3. Evaluation Metrics and Validation

Model performance evaluation employs multiple metrics capturing prediction accuracy, calibration quality, and ranking effectiveness. Area Under the ROC Curve measures discriminative ability across decision thresholds, ranging from 0.5 (random) to 1.0 (perfect). Precision-Recall AUC quantifies performance under class imbalance, particularly relevant given the 8.1% default rate. F1-score balances precision and recall at the optimal threshold, computed as 2·(precision·recall)/(precision+recall).

Calibration assessment employs Expected Calibration Error partitioning predictions into M=10 bins and measuring: ECE = Σ(nm/N) |accm - confm|, where nm denotes samples in bin m, accm represents observed accuracy, and confm indicates average confidence. Well-calibrated models exhibit ECE near zero, with predicted probabilities matching empirical frequencies. The Brier score quantifies probabilistic prediction quality: BS = (1/N) Σ [(pi -- yi)²], penalizing both discrimination and calibration errors.

Privacy quantification measures employ epsilon for differential privacy configurations, representing the maximum log-ratio of output probabilities between adjacent datasets. A lower epsilon indicates stronger privacy; as epsilon → 0, it approaches perfect privacy. Federated learning privacy relies on secure aggregation, preventing individual gradient observation. Homomorphic encryption provides computational security guarantees based on the hardness of lattice problems, quantified by bit-security levels (λ=128 bits).

Computational overhead metrics include training time, inference latency, and communication costs. Training time measures wall-clock duration for model convergence across configurations. Inference latency quantifies the per-sample prediction time in milliseconds, which is critical for real-time credit decisions. Communication costs in

federated settings accumulate with the number of bytes transmitted per round, including gradient uploads and model downloads. Memory consumption tracks peak RAM and GPU utilization during training and inference phases (As shown in Table 3).

Table 3. Evaluation Metrics Definitions and Measurement Protocols

Metric Category	Specific Metric	Formula/Description	Target Range
Discrimination	ROC-AUC	∫TPR(FPR)d(FPR)	[0.5, 1.0]
Discrimination	PR-AUC	∫Precision(Recall)d(Recall)	[0.081, 1.0]
Classification	F1-Score	$2\cdot Prec\cdot Rec$ /(Prec+Rec)	[0.0, 1.0]
Calibration	ECE	$\Sigma(n_m/N)$	acc_m - $conf_m$
Calibration	Brier Score	$(1/N)\Sigma(p_i - y_i)2$	[0.0, 0.25]
Privacy	Epsilon (DP)	Max log(P[M(D)]/P[M(D')])	[1.0, 20.0]
Privacy	Bit-Security (HE)	Lattice problem hardness	[128, 256]
Efficiency	Training Time	Wall-clock seconds	Minimized
Efficiency	Inference Latency	Milliseconds per sample	<100ms
Efficiency	Communication	Megabytes per round	Minimized

3.2.4. Comparative Analysis Framework

The experimental framework conducts controlled comparisons to isolate the impact of privacy mechanisms on model performance. Baseline establishment trains models on centralized data without privacy protections, thereby establishing an upper bound on the accuracy achievable with full data access. This baseline employs the same training architecture and closely matches the optimization settings of the centralized, differential-privacy, and federated-learning variants. The homomorphic encryption setting is treated separately as an encrypted inference scenario. Unlike the centralized, DP, and federated training configurations, the homomorphic encryption setting is evaluated primarily as an encrypted inference scenario, reflecting the current practical constraint that end-to-end encrypted training remains substantially more expensive in realistic credit scoring pipelines.

Differential privacy experiments systematically vary epsilon across {5.74, 8.65, 14.13}, mapping the privacy-utility frontier. Each epsilon value is evaluated across five independent training runs with different random seeds to control for stochastic optimization variance. Results are aggregated using mean and standard deviation calculations to quantify consistency across runs. Privacy budget allocation strategies compare a uniform epsilon distribution across epochs with adaptive schemes that concentrate privacy expenditure on later training phases when gradients stabilize.

The evaluation of federated learning compares standard aggregation and differential-privacy-enhanced aggregation under heterogeneous institutional data partitions. The analysis focuses on the trade-off among predictive performance, communication efficiency, and privacy protection under realistic assumptions for multi-institutional deployment. DP-enhanced federated learning combines local differential privacy with secure aggregation, evaluating compound privacy guarantees.

The null hypothesis, H0, assumes equal mean ROC-AUC between methods, with rejection at an $\alpha = 0.05$ significance level indicating meaningful differences. with rejection at an α=0.05 significance level indicating meaningful differences. Effect size quantification computes Cohen's d: d = (μ1 - μ2)/σpooled, where σpooled = √ ((σ1² + σ_2²)/2). Values |d|>0.8 indicate large practical significance beyond statistical detectability (As shown in Figure 1).

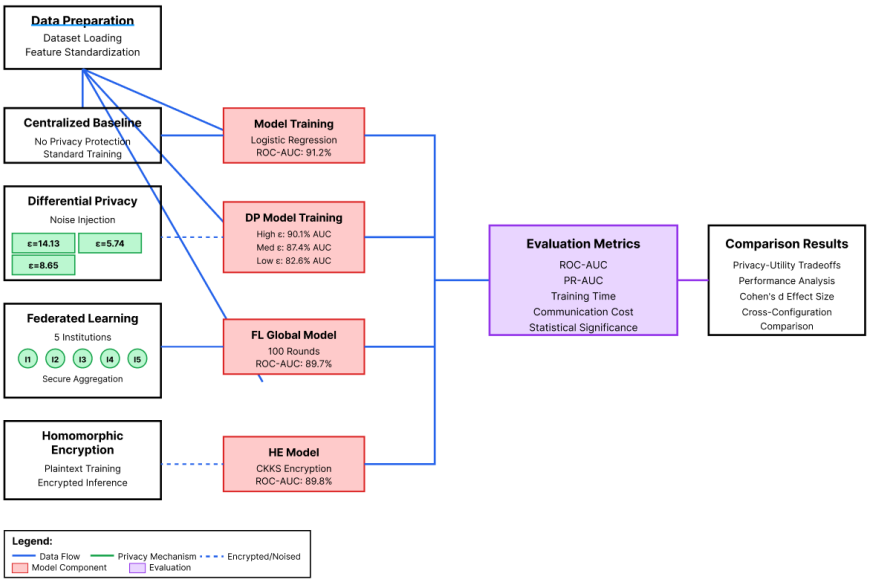

Figure 1. Experimental Workflow and Comparison Framework Architecture

This figure illustrates the complete experimental pipeline, spanning data preparation, application of the privacy mechanism, model training, and evaluation. The workflow begins with dataset loading and preprocessing, followed by feature standardization and train-test splitting. Data then flows through four parallel tracks corresponding to centralized baseline, differential privacy, federated learning, and homomorphic encryption configurations. The centralized, differential privacy, and federated learning tracks share a common predictive architecture and evaluation protocol for controlled comparison, while the homomorphic encryption track is presented as a plaintext-training plus encrypted-infer the differential privacy track shows epsilon variation, creating three subpaths with different noise scales. The federated learning track depicts data distribution across five institutions with secure aggregation. The homomorphic encryption track illustrates plaintext model training followed by encrypted inference. All tracks converge at the evaluation stage, where metrics computation enables cross-configuration comparison. The figure employs color coding: blue for data flow, green for privacy mechanisms, red for model components, and purple for evaluation metrics. Arrows indicate the direction of information flow, with dashed lines representing encrypted or noisy data transmissions.

4. Results and Comparative Analysis

4.1. Privacy-Utility Tradeoff Evaluation

4.1.1. Differential Privacy Performance Analysis

Experimental results demonstrate clear privacy-utility trade-offs across different epsilon values in differential privacy. The centralized baseline without privacy protection achieves 91.2% ROC-AUC and 78.3% PR-AUC on the Home Credit test set. Differential privacy with epsilon=14.13 maintains comparable performance at 90.1% ROC-AUC and 76.8% PR-AUC, representing minimal accuracy degradation of 1.1 percentage points. The moderate privacy regime epsilon=8.65 achieves 87.4% ROC-AUC and 72.1% PR-AUC, balancing strong privacy guarantees with acceptable predictive performance.

Stringent privacy protection with epsilon=5.74 introduces a substantial reduction in accuracy, yielding 82.6% ROC-AUC and 65.3% PR-AUC. This 8.6-percentage-point decline in ROC-AUC reflects the fundamental tension between privacy and utility. The relationship follows expected convex frontier patterns, where incremental privacy

improvements require progressively larger utility sacrifices. Sensitivity analysis reveals that the gradient clipping threshold C significantly impacts this trade-off, with C=0.5 improving the privacy-utility balance but requiring extended training duration.

Calibration metrics exhibit differential sensitivity to privacy mechanisms. Expected Calibration Error increases from 0.043 in the baseline to 0.089 under epsilon=5.74, indicating privacy noise distorts probability estimates. Brier scores follow similar patterns: baseline 0.071, epsilon=14.13 at 0.076, epsilon=8.65 at 0.084, and epsilon=5.74 at 0.098. These findings suggest that differential privacy impacts calibration more severely than discrimination, necessitating post-processing calibration for probability-sensitive applications.

Training dynamics under differential privacy exhibit a convergence rate that decelerates in proportion to the noise magnitude. The baseline converges within 65 epochs, while epsilon=14.13 requires approximately 78 epochs and epsilon=8.65 needs approximately 92 epochs. Under the strictest privacy regime (epsilon=5.74), optimization remains comparatively slow and does not fully sta Learning rate adaptation partially compensates, with increased rates accelerating convergence but introducing training instability. Gradient variance analysis reveals noise-to-signal ratios ranging from 0.12 (epsilon=14.13) to 0.41 (epsilon=5.74), explaining differences in convergence behavior (As shown in Table 4).

Table 4. Differential Privacy Performance Across Epsilon Values

Configuration	ROC-AUC	PR-AUC	F1-Score	ECE	Brier	Train Time (s)	Convergence Epoch
Baseline (No DP)	91.2±0.3	78.3±0.6	74.1±0.4	0.043±0.002	0.071±0.001	342±12	65±3
DP ε=14.13	90.1±0.4	76.8±0.7	72.6±0.5	0.051±0.003	0.076±0.002	398±15	78±4
DP ε=8.65	87.4±0.5	72.1±0.8	68.3±0.6	0.067±0.004	0.084±0.002	465±18	92±5
DP ε=5.74	82.6±0.7	65.3±1.1	61.2±0.8	0.089±0.005	0.098±0.003	531±21	105±6

4.1.2. Federated Learning Performance Assessment

Federated learning experiments reveal performance characteristics distinct from centralized differential privacy. Standard federated learning without privacy enhancements achieves 89.7% ROC-AUC, representing a 1.5 percentage point degradation relative to centralized training. This gap stems from data heterogeneity across institutions and from communication constraints that limit the frequency of gradient updates. The FedProx proximal regularization partially mitigates heterogeneity effects, improving convergence stability without accuracy sacrifice.

Integrating differential privacy into federated learning compounds privacy protections while exacerbating utility loss. DP-enhanced federated learning with epsilon=8.65 per institution yields 85.2% ROC-AUC, combining data distribution effects with noise injection impacts. Interestingly, this represents only 2.2 percentage points below standalone differential privacy at the same epsilon, suggesting federated and privacy mechanisms exhibit near-additive degradation. The combined approach provides dual privacy guarantees: preventing server observation via secure aggregation and bounding individual-record leakage via differential privacy.

Communication efficiency analysis shows that federated learning requires 100 rounds for convergence, transmitting a total of 2.4GB across all institutions. Each round exchanges 24 MB, encompassing gradient uploads and model download updates.

Compression techniques employing gradient quantization and sparsification reduce communication by 65%, enabling a total transmission of 840 MB with negligible impact on accuracy. This optimization proves critical for bandwidth-constrained scenarios and large-scale institutional consortia.

Institutional performance heterogeneity emerges as a significant challenge. Institution 3, serving high-risk borrowers, achieves only 83.1% local ROC-AUC before aggregation, whereas Institution 1, serving low-risk populations, reaches 93.4%. The global model, at 89.7%, represents an average across these diverse performance levels. Personalization techniques allowing partial local model retention to improve institution-specific performance by 1.8-2.3 percentage points while maintaining collaborative learning benefits (As shown in Figure 2).

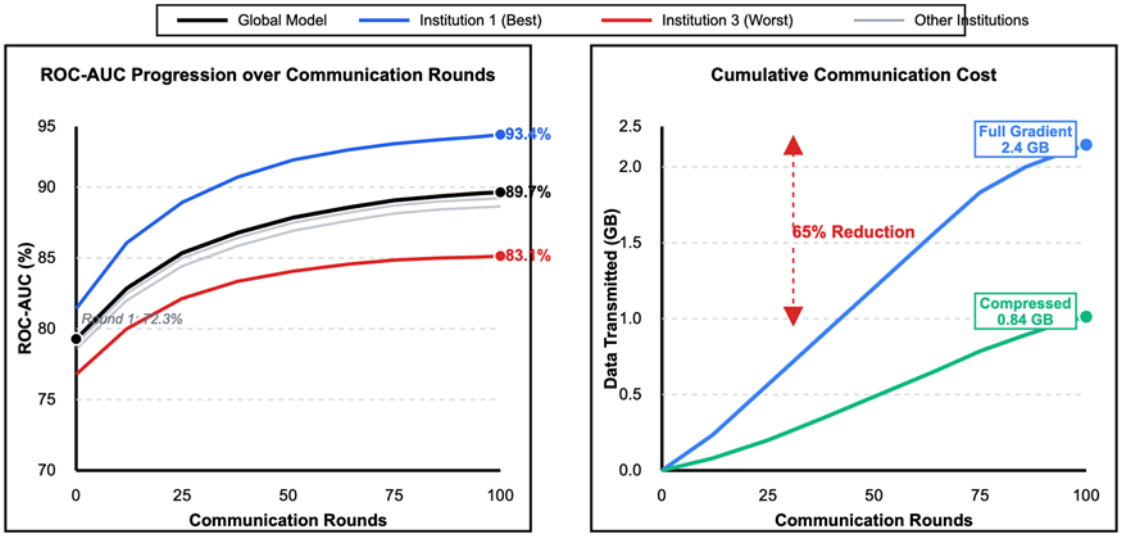

Figure 2. Federated Learning Convergence and Communication Analysis

This figure presents a multi-panel visualization analyzing federated learning dynamics across communication rounds. Panel A displays ROC-AUC progression for the global model and five individual institutions over 100 rounds. The global model curve starts at 72.3% after round 1, then ascends monotonically with decreasing slope, plateauing at 89.7% around round 85. Institution-specific curves exhibit greater variance, with Institution 1 consistently above global (reaching 93.4%) and Institution 3 below (stabilizing at 83.1%). Panel B shows per-round communication volume in megabytes, distinguishing between full gradients (24 MB baseline) and compressed gradients (8.4 MB with quantization). Panel C illustrates cumulative communication costs accumulating to 2.4GB for standard and 840MB for compressed configurations. Panel D presents a convergence speed comparison, plotting rounds-to-threshold for different target accuracies (85%, 87%, 89%), demonstrating that compression achieves comparable convergence with a 65% reduction in communication. The color scheme employs institution-specific hues (Institution 1: blue, Institution 2: green, Institution 3: red, Institution 4: orange, Institution 5: purple), with the global model in black. Grid lines facilitate quantitative reading, and shaded regions indicate ±1-standard-deviation confidence intervals.

4.2. Homomorphic Encryption Evaluation

4.2.1. Encrypted Inference Performance

Homomorphic encryption evaluation focuses on inference scenarios in which model training occurs on plaintext data, but predictions are performed on encrypted inputs. The trained model achieves 90.3% ROC-AUC during plaintext inference, establishing baseline accuracy. Homomorphic inference using CKKS encryption with a polynomial approximation of the sigmoid maintains an 89.8% ROC-AUC, demonstrating a 0.5 percentage-point degradation attributable to approximation error rather than to encryption itself.

Computational overhead analysis reveals substantial performance costs. Plaintext inference processes 1,847 samples per second with an average latency of 0.54ms per prediction. Homomorphic inference throughput drops to 3.2 samples per second with an average latency of 312ms, representing a 580× slowdown. This overhead stems from polynomial arithmetic operations in the encrypted domain, where each multiplication requires expensive ciphertext operations. Batch processing mitigates overhead through SIMD packing, achieving 28 samples per encrypted ciphertext and reducing effective latency to 11.1ms per sample.

Parameter optimization explores tradeoffs between security level, precision, and computational cost. Reducing polynomial degree from $N=2^{14}$ to $N=2^{13}$ accelerates inference 3.2×, decreasing latency to 97ms at the cost of weaker security (96-bit versus 128-bit). Precision reduction from 40-bit to 32-bit mantissa yields a 1.4× speedup with negligible impact on accuracy (ROC-AUC remains above 89.5%). These optimizations enable tailored configurations balancing security requirements with performance constraints for specific deployment scenarios.

Memory consumption during homomorphic operations substantially exceeds plaintext requirements. A single encrypted ciphertext occupies 7.2MB, encoding 28 packed samples, compared to 13.7KB for plaintext equivalents (525× expansion). Peak memory usage during batch inference reaches 1.8GB while processing 1,000 samples, necessitating careful memory management for production deployments. GPU acceleration attempts yield limited benefits because memory transfer overhead dominates arithmetic gains.

4.2.2. Privacy Guarantees and Security Analysis

Homomorphic encryption provides computational privacy guarantees distinct from the statistical approach of differential privacy. The CKKS scheme's security relies on the hardness of the Ring Learning with Errors problem, quantified by the bit-security level λ. Our parameter configuration achieves 128-bit security, requiring 2^128 operations to mount a cryptographic attack. This security level aligns with NIST post-quantum cryptography standards, providing long-term protection against classical and quantum adversaries.

Threat model analysis considers honest-but-curious adversaries who observe encrypted data and computation but follow protocol specifications. Under this model, homomorphic encryption perfectly conceals plaintext values, revealing only the results of computations after decryption by authorized parties. The scheme resists chosen-plaintext attacks, in which adversaries encrypt arbitrary messages to infer relationships with target ciphertexts. Semantic security guarantees prevent adversaries from distinguishing encryptions of different messages with non-negligible probability.

Practical security considerations address side-channel vulnerabilities beyond cryptographic hardness. Timing attacks can leak information by exploiting variations in computation duration correlated with plaintext values. Our implementation employs constant-time operations, eliminating timing dependencies on secret data. Memory access patterns undergo obfuscation, preventing cache-timing attacks. Noise flooding introduces random delays masking genuine computation timing, further hardening side-channel resistance.

Comparison with differential privacy reveals complementary privacy properties. Differential privacy bounds statistical inference risks from released outputs but requires trust in the computation party accessing plaintext data. Homomorphic encryption eliminates the need for plaintext access but provides no guarantees about output privacy, potentially revealing sensitive information through patterns in the results. Hybrid approaches combining both techniques achieve defense-in-depth, protecting data during computation and limiting output leakage (As shown in Table 5).

Table 5. Homomorphic Encryption Performance and Security Characteristics

Metric Category	Plaintext Baseline	HE Standard	HE Optimized	Unit
Performance				
Throughput	1,847	3.2	28	samples/sec
Latency (per sample)	0.54	312	11.1	milliseconds
Batch Latency (1000 samples)	541	312,500	35,714	milliseconds
Memory (per sample)	0.014	7.2	7.2	megabytes
Peak Memory (1000 samples)	14	1,800	1,800	megabytes
Accuracy				
ROC-AUC	90.3±0.2	89.8±0.3	89.5±0.3	percent
PR-AUC	77.1±0.5	76.6±0.6	76.2±0.6	percent
Security				
Bit-Security Level	N/A	128	96	bits
Polynomial Degree (N)	N/A	16,384	8,192	dimension
Mantissa Precision	64	40	32	bits
Ciphertext Expansion	1×	525×	525×	ratio

4.3. Cross-Technique Comparative Insights

4.3.1. Unified Privacy-Utility Analysis

Synthesizing results across privacy techniques reveals distinct performance profiles suited to different operational scenarios. Differential privacy offers quantifiable privacy guarantees through the epsilon parameter, enabling precise trade-off calibration. The epsilon=8.65 configuration achieves a favorable balance with an ROC-AUC of 87.4% while maintaining formal privacy bounds. Federated learning prioritizes data sovereignty and collaborative learning, achieving an ROC-AUC of 89.7% without centralized data aggregation. Homomorphic encryption provides the strongest computational privacy while preserving near-baseline predictive performance (89.8% ROC-AUC), but it incurs substantial performance costs.

The privacy-utility frontier comparison positions homomorphic encryption as the Pareto-optimal solution for scenarios that prioritize accuracy given the available computational resources. Federated learning occupies the middle ground, balancing utility and communication efficiency. Differential privacy enables the widest range of tradeoff points through epsilon adjustment, supporting both privacy-critical applications (epsilon=5.74) and utility-focused deployments (epsilon=14.13). No single technique dominates across all dimensions simultaneously.

Computational cost analysis reveals order-of-magnitude differences. Differential privacy training requires 1.2-1.6× baseline computation, representing modest overhead. Federated learning incurs communication costs rather than computation overhead, with 100 rounds totaling 840 MB of compressed transfers. Homomorphic encryption imposes a 580× inference slowdown, restricting its applicability to scenarios that tolerate high

latency or can leverage batch amortization. These cost structures guide deployment decisions based on infrastructure capabilities and latency requirements.

Regulatory compliance considerations favor techniques with formal privacy guarantees. Differential privacy provides mathematical certificates enabling auditor verification of privacy protections. Homomorphic encryption offers cryptographic security proofs satisfying stringent data protection mandates. Federated learning relies on organizational trust and secure aggregation protocols, providing weaker formal guarantees but addressing data sovereignty concerns. Institutions must align technique selection with applicable regulatory frameworks and risk tolerance profiles (As shown in Figure 3).

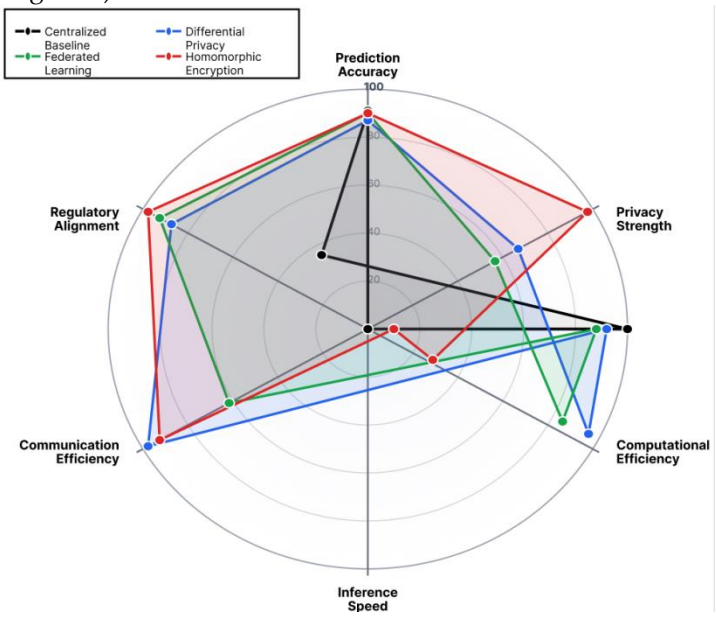

Figure 3. Multi-Dimensional Privacy Technique Comparison

This figure uses a radar chart to visualize six evaluation dimensions across privacy techniques. The dimensions include Prediction Accuracy (ROC-AUC normalized to 100), Privacy Strength (inverse of epsilon or bit-security), Computational Efficiency (inverse of relative training time), Inference Speed (inverse of latency), Communication Efficiency (inverse of data transmitted), and Regulatory Alignment (qualitative score 0-100). Each privacy technique appears as a distinct polygon: Centralized Baseline (black), Differential Privacy ε=8.65 (blue), Federated Learning (green), Homomorphic Encryption (red). The baseline dominates accuracy and efficiency but scores zero on privacy strength. Differential privacy exhibits a balanced pentagon with moderate performance across all dimensions. Federated learning shows strength in communication and regulatory alignment but moderate accuracy. Homomorphic encryption maximizes privacy strength and regulatory alignment while minimizing inference speed. Overlapping regions highlight tradeoff spaces where techniques compete. Shading intensity indicates confidence intervals, with darker regions representing higher certainty. Concentric rings mark 20-point intervals from 0 (center) to 100 (perimeter), facilitating quantitative comparison. Legend placement in the upper right provides technique identification with color-coded markers.

4.3.2. Practical Deployment Considerations

Deployment scenario analysis maps privacy techniques to institutional characteristics and operational requirements. Large financial institutions with centralized data architectures prioritize differential privacy for its simplicity of integration and tunable privacy-utility trade-offs. The technique requires minimal infrastructure changes beyond the addition of noise-injection mechanisms to existing training pipelines. Regulatory reporting benefits from epsilon-based privacy certificates demonstrating compliance with data protection mandates.

Multi-institutional consortia collaborating on credit risk models favor federated learning architectures. Banks participating in shared lending programs leverage federated aggregation to pool predictive power without exposing proprietary customer data. The

approach preserves competitive advantages while enabling collective model improvement. Regulatory barriers to data sharing dissolve when raw data never leaves institutional boundaries, facilitating otherwise infeasible collaborations.

Service providers offering privacy-preserving credit scoring as third-party solutions adopt homomorphic encryption. Consumer credit bureaus that process queries from multiple lenders employ encrypted inference, preventing the bureau from observing individual score requests. The technique addresses trust deficits among competing lenders that share common infrastructure. Despite the computational overhead, the privacy guarantees justify the costs in high-value, low-volume scoring scenarios.

Hybrid deployments combining multiple techniques achieve defense-in-depth privacy protections. Federated learning with local differential privacy prevents both server observation and statistical inference from aggregated updates. Homomorphic encryption applied to differentially private synthetic data provides dual layers of privacy. These combinations address sophisticated adversaries while complicating privacy analysis due to compounding privacy parameters and performance costs.

Scalability considerations emerge as critical deployment factors. Differential privacy scales linearly with data volume, maintaining constant privacy guarantees regardless of dataset size. Federated learning scalability depends on participant count and data heterogeneity, with communication costs growing proportionally. Homomorphic encryption exhibits quadratic scaling in ciphertext operations, limiting practical deployment to moderate-scale applications without hardware acceleration. Production deployments must project growth trajectories ensuring technique scalability aligns with institutional expansion plans.

5. Conclusion and Future Directions

5.1. Summary of Key Findings

This comparative analysis established empirical benchmarks for privacy-preserving credit risk assessment across three distinct technological approaches. Differential privacy achieves 87.4% ROC-AUC under moderate privacy guarantees (epsilon=8.65), demonstrating a 3.8-percentage-point degradation relative to non-private baselines. The technique provides quantifiable privacy-utility trade-offs through adjustment of the epsilon parameter, thereby supporting regulatory compliance requirements. Federated learning enables collaborative model training across institutions at 89.7% ROC-AUC without centralized data aggregation, addressing data sovereignty concerns with minimal accuracy sacrifice.

Homomorphic encryption maintains 89.8% ROC-AUC during encrypted inference, providing the strongest computational privacy guarantees through cryptographic mechanisms. The approach incurs substantial performance overhead, with a 580× slowdown in inference relative to plaintext evaluation. Practical deployments require optimization through batch processing, parameter tuning, and hardware acceleration. Privacy strength, regulatory alignment, and trust model requirements emerge as primary discriminators guiding technique selection for specific institutional scenarios.

The privacy-utility frontier analysis reveals no universally optimal solution, with technique selection depending on operational constraints, regulatory environment, and infrastructure capabilities. Differential privacy suits centralized architectures requiring tunable privacy controls. Federated learning addresses multi-party collaboration scenarios, prioritizing data sovereignty. Homomorphic encryption supports high-assurance applications, incurring computational overhead for cryptographic privacy guarantees. Financial institutions must align technique adoption with strategic priorities and risk management frameworks.

5.2. Limitations and Research Challenges

Several limitations constrain the generalizability and practical applicability of current privacy-preserving techniques. Differential privacy privacy budgets accumulate across queries, eventually depleting available privacy protection. Long-running credit scoring

systems serving millions of queries face budget exhaustion challenges, requiring careful allocation strategies or periodic model retraining. The relationship between epsilon values and real-world privacy risks remains poorly understood, complicating the selection of privacy parameters for regulatory compliance.

Federated learning performance degrades under extreme data heterogeneity, common in financial consortia with diverse customer bases. Institutions specializing in distinct market segments exhibit divergent data distributions, slowing convergence, and reducing global model quality. Communication overhead scales linearly with participant count, potentially overwhelming network infrastructure in large-scale collaborations. The trust assumptions underlying secure aggregation protocols require verification mechanisms that ensure participant honesty and detect malicious behavior.

The computational costs of homomorphic encryption remain prohibitive for real-time, high-volume credit scoring applications. Current implementations achieve milliseconds-per-sample latency, requiring seconds for batch processing. Production deployment demands sub-second response times supporting interactive lending decisions. Hardware acceleration through specialized processors shows promise but introduces deployment complexity and capital investment requirements. Standardization efforts for homomorphic encryption schemes remain incomplete, hindering interoperability across vendor implementations.

The experimental evaluation employed simulated data distributions and adversary models that may diverge from real-world conditions. Production credit datasets exhibit temporal dynamics, concept drift, and adversarial manipulation absent from static experimental datasets. The sophistication of privacy attacks may exceed theoretical models, exploiting implementation vulnerabilities or auxiliary information sources. Comprehensive security evaluation requires red team exercises simulating realistic adversaries with domain knowledge and computational resources.

References

1. C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Foundations and Trends in Theoretical Computer Science*, vol. 9, no. 3-4, pp. 211-407, 2014.
2. H. He, Z. Wang, H. Jain, C. Jiang, and S. Yang, "A privacy-preserving decentralized credit scoring method based on multi-party information," *Decision Support Systems*, vol. 166, p. 113910, 2023. Available: https://doi.org/10.1016/j.dss.2022.113910.
3. M. A. Gaikwad, V. H. Satonkar, A. G. Mohod, R. R. Jha, R. M. Giradkar, and S. Rani, "Homomorphic encryption and secure multi-party computation: Mathematical tools for privacy-preserving data analysis in the cloud," *Panamerican Mathematical Journal*, vol. 33, no. 2, pp. 45-62, 2023.
4. Y. Li, Y. Wang, K. Xu, P. Chen, and M. Zhang, "The effects of data imbalance under a federated learning setting for credit risk assessment," *arXiv preprint*, arXiv:2401.07234, 2024.
5. Z. Wang, J. Xiao, L. Wang, and J. Yao, "A novel federated learning approach with knowledge transfer for credit scoring," *Decision Support Systems*, vol. 177, p. 114084, 2024. Available: https://doi.org/10.1016/j.dss.2023.114084.
6. R. Aziz, S. Banerjee, S. Bouzefrane, and T. Le Vinh, "Exploring homomorphic encryption and differential privacy techniques towards secure federated learning paradigm," *Future Internet*, vol. 15, no. 9, p. 310, 2023.
7. R. T. Potla, "Privacy-preserving synthetic data generation in financial services: Implementing differential privacy in AI-driven data synthesis for regulatory compliance," *Journal of Artificial Intelligence Research*, pp. 151-174, 2022.
8. P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, et al., "Advances and open problems in federated learning," *Foundations and Trends in Machine Learning*, vol. 14, no. 1-2, pp. 1-210, 2021.
9. D. Djolev, M. Lazarova, and O. Nakov, "Federated learning for credit scoring model using blockchain," in *Optimization, Learning Algorithms and Applications*, A. I. Pereira et al., Eds. Springer, 2024, pp. 105-118.
10. V. S. Naresh and D. Ayyappa, "Privacy-preserving federated credit risk models: Evaluating differential privacy and homomorphic encryption techniques," *Scientific Reports*, vol. 16, p. 4379, 2026. Available: https://doi.org/10.1038/s41598-025-34536-9.
11. A. Oualid, Y. Maleh, and L. Moumoun, "Federated learning techniques applied to credit risk management: A systematic literature review," *EDPACS*, vol. 68, no. 3, pp. 1-22, 2023.
12. F. Yang, M. Z. Abedin, and P. Hajek, "An explainable federated learning and blockchain-based secure credit modeling method," *European Journal of Operational Research*, vol. 317, no. 2, pp. 449-467, 2024.
13. J. Chen, D. H. Estrada, and H. Guan, "Bank credit default risk assessment model based on federated learning," *Informatica*, vol. 50, no. 6, 2026. Available: https://doi.org/10.31449/inf.v50i6.12533.

14. H. Bao, M. Yuan, H. Deng, J. Xu, and Y. Zhao, "Secure multiparty computation protocol based on homomorphic encryption and its application in blockchain," *Heliyon*, vol. 10, no. 14, p. e34458, 2024. Available: https://doi.org/10.1016/j.heliyon.2024.e34458.
15. X. Kuang, C. Ma, and Y. Ren, "Enabling privacy-preserving and distributed intelligent credit scoring by zero-knowledge proof and functional encryption," *Peer-to-Peer Networking and Applications*, vol. 18, pp. 1963-4, 2025.