FEDDCOL: FEDERATED DYNAMIC CONTINUAL OVERSAMPLING LEARNING FOR IMBALANCED TABULAR DATA CLASSIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Federated learning (FL) enables multiple clients to train a global model without sharing raw local data. However, clients usually face the problem of imbalanced class distributions, which may bias the global classifier toward the majority class. Existing data level approaches typically use static oversampling strategies, making it difficult to dynamically adjust minority class generation during federated training. To address this limitation, this study proposes Federated Dynamic Continual Oversampling Learning (FedDCOL). FedDCOL first characterizes the local data structure based on class overlap degree and then partitions local samples into safe, boundary, and noisy regions. Based on this structural information, FedDCOL develops a Structure-Constrained Diffusion Model (SC-Diff). By incorporating a base diffusion objective, a prototype relation constraint and an inter-class margin constraint, SC-Diff learns the data distribution while preserving the discriminative structure of the minority class. During federated training, SC-Diff is continually updated across communication rounds to adapt minority class generation, while structure selection guided by class overlap information retains reliable generated samples for classifier training. This transforms one-shot oversampling into a dynamic continual oversampling process. Experimental results on seven real world imbalanced tabular datasets show that FedDCOL outperforms six SOTA methods, achieving improvements of 2.23%, 12.41%, and 10.39% in AUC, F1-score, and G-mean over the strongest baselines, respectively. The particularly pronounced gains in F1-score and G-mean highlight the effectiveness of dynamic continual oversampling for minority class learning in federated tabular classification.

1 INTRODUCTION

Tabular data are widely used in critical domains such as financial risk control (Wen & Lou, 2024), clinical prediction (Wang et al., 2024), and cybersecurity (Fu et al., 2026). However, real world tabular datasets often exhibit class imbalance, with minority class samples being scarce. As a result, classifiers tend to be biased toward the majority class, leading to degraded performance on the minority class (Yang et al., 2024a). This issue is especially important in high-risk applications, where errors on minority class cases can incur substantial practical costs (He et al., 2024).

Meanwhile, real world data are often distributed across institutions or edge devices, and directly centralizing raw data may raise privacy concerns under data protection regulations such as the General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA) (Zhang et al., 2024b). Federated learning (FL) enables collaborative model training without centralizing raw data by keeping data on clients and aggregating model updates (McMahan et al., 2017; Ching et al., 2026), thereby reducing the privacy risks associated with direct data sharing. However, imbalanced class distributions are also prevalent across clients in federated settings (Wang et al., 2021; Dai et al., 2023), making minority class learning particularly challenging when data remain decentralized.

To address imbalanced class distributions in FL, existing approaches can be broadly categorized into the model architecture, aggregation strategy, and data level methods (Xiao et al., 2024; Wang et al., 2021; Ye et al., 2024). While model architecture and aggregation strategy methods mitigate imbalance through optimization, data level methods directly rebalance the training distribution. This

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

study focuses on data level methods, particularly oversampling, which increases minority class samples without removing majority class samples. Classical oversampling methods such as SMOTE (Chawla et al., 2002) generate samples through local linear interpolation, but may produce ambiguous or noisy samples in the presence of class overlap or small disjuncts, increasing uncertainty near class boundaries (Azhar et al., 2023). Deep generative models provide a more flexible alternative. GANs (Xie et al., 2025) and VAEs (Kingma & Welling, 2014) have been explored for minority class generation (Ding et al., 2025; Ren et al., 2025), but may suffer from training instability and posterior collapse, respectively (Zhang et al., 2023; Kinoshita et al., 2023). More recently, diffusion models (Ho et al., 2020) have emerged as a promising alternative by learning data distributions through progressive noising and iterative denoising, providing a more stable generative framework for minority class oversampling (Ren et al., 2026).

Although diffusion models provide a more stable approach to minority class oversampling, their application to federated tabular data with imbalanced class distributions remains challenging. One challenge is that such distributions are often accompanied by class overlap, whereas existing tabular diffusion models are primarily designed for general distribution modeling rather than explicitly preserving minority class structure or inter-class relationships (Kotelnikov et al., 2023; Shi et al., 2025). As a result, preserving sample structure during generation remains difficult (Um et al., 2024). Another challenge is that some federated data level methods treat sampling as a one-shot preprocessing step independent of subsequent model optimization (Hu et al., 2025). Since the generated minority class samples are fixed before training, they do not readily adapt to the evolving global model, limiting the coordination between oversampling and federated optimization.

To address the above challenges, this study proposes a Federated Dynamic Continual Oversampling Learning (FedDCOL) framework. FedDCOL treats minority class generation as a dynamic component of federated optimization, enabling the generative model and the classifier to evolve throughout communication rounds. To mitigate the generation ambiguity caused by class overlap, FedDCOL characterizes the local data structure using the class overlap degree and uses this structural information to construct a Structure-Constrained Diffusion Model (SC-Diff). SC-Diff augments the base diffusion objective with prototype relation and inter-class margin constraints to preserve sample structural information during generation. The updated SC-Diff then generates minority class candidates locally, which are selected based on their class overlap degree before being used for classifier training. By integrating generation, structure selection, and classifier learning into a unified federated training process, FedDCOL enables minority class generation to evolve with federated optimization.

The main contributions of this study are summarized as follows. 1) We introduce a class overlap degree to characterize local class overlap structure and develop SC-Diff, which incorporates this structure into tabular diffusion learning through prototype relation and inter-class margin constraints. 2) We propose FedDCOL to couple minority class generation, structure selection, and federated classifier optimization, enabling the generated minority class samples to dynamically adapt throughout federated training. 3) Extensive experiments across seven imbalanced tabular datasets demonstrate that FedDCOL improves AUC, F1-score, and G-mean on average, while sensitivity analyses demonstrate its robustness to parameter variations and ablation studies validate the effectiveness of its key design components.

2 Related Work

In FL, statistical heterogeneity is commonly reflected in non-IID data across clients, causing clients to optimize toward different local objectives and resulting in biased updates and degraded global performance. Such heterogeneity may arise from differences in feature distributions, class proportions, and local data sizes across clients, which further complicate global model training. Based on their primary mechanisms, existing approaches can be grouped into three categories: model architecture, aggregation strategy, and data level methods.

2.1 Model Architecture Methods

Model architecture methods adapt model design to address heterogeneous label distributions in FL. FedLC (Zhang et al., 2022) calibrates class logits based on local label distributions to reduce bias

caused by label skew. FedNH (Dai et al., 2023) employs class prototypes to improve representation learning under imbalanced class distributions. FedConcat (Diao et al., 2024) preserves local knowledge through model concatenation and client clustering under label skew. GBME (Zeng et al., 2023) estimates the global class prior from accumulated client gradients and constructs balanced experts, while FedLoGe (Xiao et al., 2024) combines shared representations, personalized classifiers, and feature realignment to improve both global and local performance. These methods mainly improve how models learn from existing local data. However, when minority class samples are limited on individual clients, model learning remains constrained by limited local data.

2.2 Aggregation Strategy Methods

Aggregation strategies modify client updates or server aggregation to mitigate the effects of distribution differences across clients. FedAvg (McMahan et al., 2017) performs weighted averaging based on local sample sizes, while FedDisco (Ye et al., 2023) further adjusts aggregation weights according to the discrepancy between local and global class distributions. Fed-GraB (Xiao et al., 2023) adaptively reweights client gradients according to an estimated global long-tailed prior. FedANAG (Zhang et al., 2025a) improves local update stability through Nesterov acceleration at both the client and server, while FedSWA (Liu et al., 2025) employs stochastic weight averaging to improve global generalization under highly heterogeneous data. These methods mitigate the impact of heterogeneous client updates on the global model, but when local bias is caused by limited minority class samples, adjusting aggregation alone cannot compensate for insufficient minority class information.

2.3 Data Level Methods

Data level methods directly improve local training data through information sharing and data generation or augmentation. FedFed (Yang et al., 2023) shares features relevant to model performance to reduce data differences across clients, while FedRDN (Yan et al., 2025) augments local data using distribution statistics collected across the federation to mitigate feature distribution shift. FedCOG (Ye et al., 2024) exploits knowledge from the global model to generate complementary data and reduce overfitting to heterogeneous local data, while GGEUR (Ma et al., 2025) uses geometric information from the global embedding distribution to guide local data generation. FedWGAN (Zhang et al., 2025b) employs a constrained WGAN to generate synthetic samples for mitigating data heterogeneity in federated cross modal retrieval, while GetFed (Bai et al., 2025) combines GAN-based data generation with differential privacy to improve FL under non-IID data. VQ-FedDiff (Yoon et al., 2025) further introduces client specific vector quantized conditioning into federated diffusion models for personalized data generation.

Compared with model architecture and aggregation strategy, data level methods enrich the local training data, providing a more direct way to address minority class scarcity. However, existing generative methods are mainly developed for general data heterogeneity in FL, with limited consideration of minority class structure under imbalanced class distributions and how data generation should adapt throughout federated optimization. To address this gap, FedDCOL introduces minority class structural constraints into federated diffusion generation and updates the generative model across communication rounds, enabling dynamic minority class generation throughout federated training.

3 Proposed Method: FedDCOL

3.1 Overview

Consider a federated system with K clients. The k-th client holds a local dataset $\mathcal{D}_k = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_k}$, where $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \{0, 1\}$, with $y = 1$ denoting the target minority class. After preprocessing, $\mathbf{x}_i$ denotes a mixed-type tabular vector consisting of min–max normalized numerical features and one-hot encoded categorical features. At communication round r, the server maintains the global SC-Diff parameters ϕ^r and the global classifier parameters θ^r, while the set of participating clients is denoted by $\mathcal{S}^r$. This study focuses on federated tabular classification with imbalanced class distributions and aims to improve the global classifier's learning capability for minority class samples by continually adapting minority class generation throughout federated

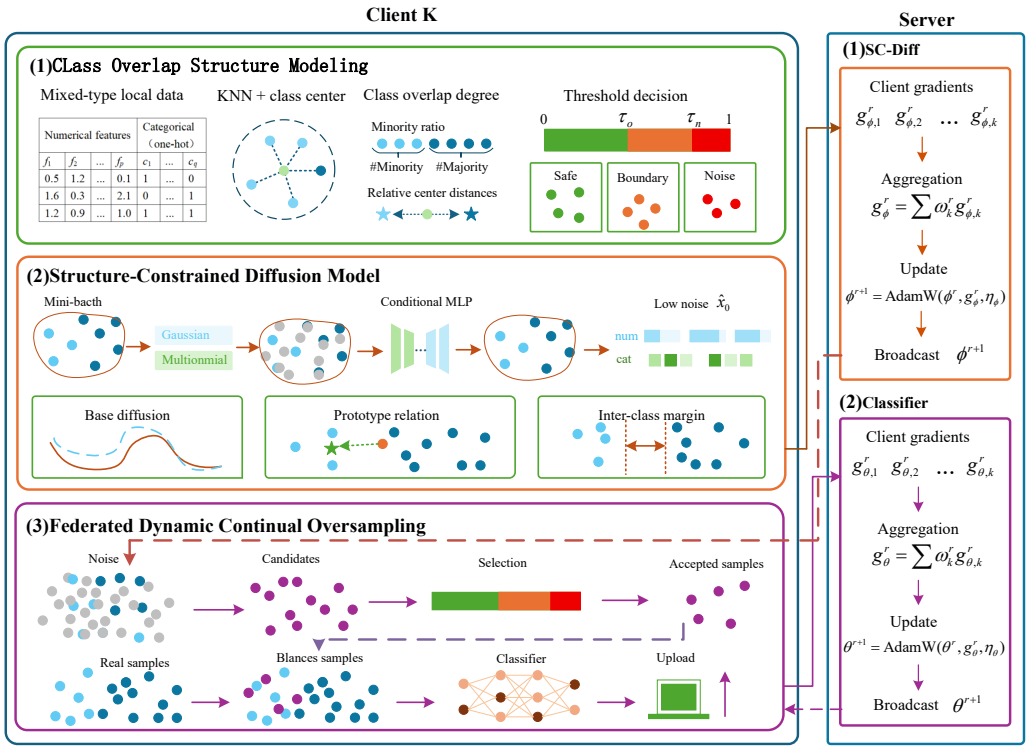


Figure 1: Overview of the proposed FedDCOL framework.

training. As shown in Figure 1, FedDCOL consists of three core components: class overlap structure modeling, SC-Diff, and dynamic continual oversampling.

3.2 Class Overlap Structure Modeling

Imbalanced class distributions can be accompanied by varying degrees of inter-class overlap: samples located near the class center can provide relatively stable class structure, whereas samples close to the opposite class tend to have higher boundary uncertainty. To characterize such geometric differences, FedDCOL computes the class overlap degree for each sample and accordingly partitions the local data into safe, boundary, and noisy regions.

For client k, consider a local sample $\mathbf{x}$ with label y. Let $\mathcal{N}_q(\mathbf{x})$ denote the set of its q nearest neighbors, and μ_y and $\mu_{\bar{y}}$ denote the local centers of the current class and the opposite class, respectively. This study defines the same class neighbor ratio and the relative class center distance as

$$s(\mathbf{x}) = \frac{1}{q} \sum_{j \in \mathcal{N}_q(\mathbf{x})} \mathbb{I}(y_j = y), \; d(\mathbf{x}) = \frac{\|\mathbf{x} - \mu_y\|_2}{\|\mathbf{x} - \mu_y\|_2 + \|\mathbf{x} - \mu_{\bar{y}}\|_2 + \varepsilon}. \tag{1}$$

The class overlap degree is defined as

$$O(\mathbf{x}) = \exp\left[-\frac{1}{2}\left(\frac{s(\mathbf{x})}{\max(d(\mathbf{x}), \varepsilon)}\right)^2\right], \tag{2}$$

where ε is a numerical stabilization term used to avoid numerical instability when the relative distance to the class center approaches zero. A smaller $O(\mathbf{x})$ indicates that the sample is more consistent with its own class, whereas a larger value indicates stronger overlap with the opposite class.

Given two thresholds τ_o and τ_n satisfying $0 < \tau_o < \tau_n < 1$, define:

$$(\mathbf{x}, y) \in \begin{cases} \mathcal{D}_k^{\text{safe}}, & O(\mathbf{x}) \le \tau_o, \\ \mathcal{D}_k^{\text{boundary}}, & \tau_o < O(\mathbf{x}) \le \tau_n, \\ \mathcal{D}_k^{\text{noisy}}, & O(\mathbf{x}) > \tau_n, \end{cases} \tag{3}$$

where $\tau_o = 0.4$ and $\tau_n = 0.9$ are determined through parameter sensitivity analysis. The detailed procedure for class overlap structure modeling is provided in Algorithm 1 of Appendix A.

3.3 Structure-Constrained Diffusion Model

Base Diffusion Objective. For numerical features, SC-Diff uses a Gaussian diffusion process to progressively add noise and trains the denoising network to predict the injected Gaussian noise. The training objective is

$$\mathcal{L}_{\text{gauss}} = \mathbb{E}_{\mathbf{x}_0, t, \boldsymbol{\epsilon}} \left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\phi(\mathbf{x}_t, t, y)\|_2^2 \right], \tag{4}$$

where $\mathbf{x}_0$ is the original sample, $\mathbf{x}_t$ is the noisy sample at timestep t, $\boldsymbol{\epsilon}$ is the injected Gaussian noise, and $\boldsymbol{\epsilon}_\phi$ is the noise prediction function parameterized by ϕ. For categorical features, a multinomial diffusion process progressively perturbs the one-hot representation and recovers the original discrete state by learning the reverse categorical distribution. Let J denote the number of categorical features and ℓ_{VB}^j the variational diffusion loss of the j-th categorical feature. Then,

$$\mathcal{L}_{\text{multi}} = \mathbb{E}_t \left[\sum_{j=1}^{J} \ell_{\text{VB}}^j(\mathbf{x}_0^j, \mathbf{x}_t^j, t) \right]. \tag{5}$$

The base diffusion objective for mixed-type tabular data is $\mathcal{L}_{\text{base}} = \mathcal{L}_{\text{gauss}} + \mathcal{L}_{\text{multi}}$. This objective captures the overall distributional characteristics and provides the foundation for incorporating additional structural constraints. The detailed diffusion process is provided in Appendix B.

Prototype Relation Constraint. To incorporate stable class structure into diffusion learning, SC-Diff constructs the class prototype set $\mathcal{P} = \mathcal{P}_0 \cup \mathcal{P}_1$, where $\mathcal{P}_c = \{h(\mathbf{x}) \mid (\mathbf{x}, y) \in \mathcal{D}_k^{\text{safe}}, y = c\}$ for $c \in \{0, 1\}$, $h(\cdot)$ denotes the representation produced by the last hidden layer of the classifier. For a sample $\mathbf{x}$ and a prototype $\mathbf{p} \in \mathcal{P}$, the normalized prototype distance is defined as

$$\delta(\mathbf{x}, \mathbf{p}) = \left\| \frac{h(\mathbf{x})}{\|h(\mathbf{x})\|_2} - \frac{\mathbf{p}}{\|\mathbf{p}\|_2} \right\|_2^2. \tag{6}$$

Let $\mathbf{r}(\mathbf{x}) = [\delta(\mathbf{x}, \mathbf{p})]_{\mathbf{p} \in \mathcal{P}}$ denote the relation vector of $\mathbf{x}$ with respect to the class prototypes. Accordingly, the prototype relation constraint is defined as

$$\mathcal{L}_{\text{proto}} = \frac{1}{|\mathcal{J}_p|} \sum_{\mathbf{x}_0 \in \mathcal{J}_p} \text{SmoothL1}(\mathbf{r}(\hat{\mathbf{x}}_0) - \mathbf{r}(\mathbf{x}_0)), \tag{7}$$

where $\hat{\mathbf{x}}_0$ denotes the denoised reconstruction, $\mathcal{J}_p$ denotes the set of minority class samples in $\mathcal{D}_k^{\text{rel}} = \mathcal{D}_k^{\text{safe}} \cup \mathcal{D}_k^{\text{boundary}}$, and the Smooth L1 loss is applied to each component of the relation error and averaged over all prototypes. By preserving the consistency of prototype relations between the original sample and its denoised reconstruction, this constraint encourages the diffusion process to preserve the class structure.

Inter-Class Margin Constraint. To prevent the denoising process from weakening the class separability of minority boundary samples, SC-Diff further constrains the relative inter-class margin between the original sample and its denoised reconstruction. Based on the prototype distance in Equation equation 6, the minority relative margin of $\mathbf{x}$ is defined as

$$m(\mathbf{x}) = \min_{\mathbf{p} \in \mathcal{P}_0} \delta(\mathbf{x}, \mathbf{p}) - \min_{\mathbf{p} \in \mathcal{P}_1} \delta(\mathbf{x}, \mathbf{p}). \tag{8}$$

For minority class samples, a larger $m(\mathbf{x})$ indicates a larger relative margin in favor of the minority class prototypes. The inter-class margin constraint is defined as

$$\mathcal{L}_{\text{marg}} = \frac{1}{|\mathcal{J}_m|} \sum_{\mathbf{x}_0 \in \mathcal{J}_m} \left\{ \tau \log \left[1 + \exp\left(\frac{m(\mathbf{x}_0) - m(\hat{\mathbf{x}}_0) - \Delta}{\tau} \right) \right] \right\}^2, \tag{9}$$

where $\mathcal{J}_m$ denotes the set of minority class samples in $\mathcal{D}_k^{\text{boundary}}$, $\Delta > 0$ denotes the allowed tolerance for margin degradation, and $\tau > 0$ denotes the smoothing temperature. The penalty increases as the reconstructed margin falls below the original margin minus the tolerance Δ, discouraging degradation of the relative inter-class margin during diffusion reconstruction.

Joint Objective. Combining the base diffusion objective, the prototype relation constraint, and the inter-class margin constraint, the SC-Diff training objective on each client is given by

$$\mathcal{L}_{\text{SC}} = \mathcal{L}_{\text{base}} + \lambda_p \mathcal{L}_{\text{proto}} + \lambda_m \mathcal{L}_{\text{marg}}, \tag{10}$$

where λ_p and λ_m control the relative contributions of the two structural constraints and are set to 0.3 and 0.1, respectively, based on the parameter sensitivity analysis. Under this joint objective, SC-Diff learns the local tabular distribution while preserving the structural information of minority class samples during denoising, supporting subsequent minority class generation. The detailed SC-Diff training procedure is provided in Algorithm 2 of Appendix A.

3.4 Federated Dynamic Continual Oversampling

Building on the class overlap structure modeling in Section 3.2 and the SC-Diff model in Section 3.3, FedDCOL jointly optimizes minority class generation and classifier training throughout the FL process.

Federated SC-Diff Optimization. Client k computes its local gradient as $\mathbf{g}_{\phi,k}^r = \nabla_\phi \mathcal{L}_{\text{SC},k}(\phi^r)$, with aggregation weight $\omega_k^r = n_k / \sum_{j \in \mathcal{S}^r} n_j$. The server updates the global SC-Diff parameters using

$$\mathbf{g}_\phi^r = \sum_{k \in \mathcal{S}^r} \omega_k^r \mathbf{g}_{\phi,k}^r, \ \phi^{r+1} = \text{AdamW}\big(\phi^r, \mathbf{g}_\phi^r, \eta_\phi\big). \tag{11}$$

Dynamic Generation and Structure Selection. Once minority class generation is activated, each participating client uses the latest SC-Diff to generate a candidate set $\mathcal{C}_k^r \sim p_{\phi^{r+1}}(\mathbf{x} \mid y = 1)$, whose size is determined by the local class imbalance. The $O(\mathbf{x})$ of each candidate is computed with respect to the local real dataset $\mathcal{D}_k$. The structure selection function $\mathcal{F}_k^{\text{str}}(\cdot)$ discards candidates assigned to the noisy region and retains up to the target number of remaining candidates with the smallest $O(\mathbf{x})$. The selected set is denoted by $\mathcal{A}_k^r = \mathcal{F}_k^{\text{str}}(\mathcal{C}_k^r; \mathcal{D}_k)$.

Federated Classifier Optimization. Let $a_k^r = |\mathcal{A}_k^r|$. The classifier is trained using $\mathcal{D}_k$ and $\mathcal{A}_k^r$:

$$\mathcal{L}_{\text{cls},k}^r(\theta) = \frac{\sum_{(\mathbf{x},y)\in\mathcal{D}_k} \text{CE}\big(f_\theta(\mathbf{x}), y\big) + \sum_{\tilde{\mathbf{x}}\in\mathcal{A}_k^r} \text{CE}\big(f_\theta(\tilde{\mathbf{x}}), 1\big)}{n_k + a_k^r}. \tag{12}$$

Client k computes $\mathbf{g}_{\theta,k}^r = \nabla_\theta \mathcal{L}_{\text{cls},k}^r(\theta^r)$, and the server updates the global classifier as

$$\mathbf{g}_\theta^r = \sum_{k \in \mathcal{S}^r} \omega_k^r \mathbf{g}_{\theta,k}^r, \ \theta^{r+1} = \text{AdamW}\big(\theta^r, \mathbf{g}_\theta^r, \eta_\theta\big). \tag{13}$$

As SC-Diff is continually updated, the generated and selected minority class samples change accordingly, transforming static oversampling into a dynamic continual process. The complete procedure is provided in Algorithm 3 of Appendix A.

4 Experiments and Analysis

4.1 Experimental Settings

Datasets. This study evaluates FedDCOL on seven publicly available imbalanced tabular datasets collected from the KEEL (Alcalá-Fdez et al., 2011) and UCI (Kelly et al., 2023) repositories. Detailed statistics for all datasets are provided in Table 5 of Appendix C.1.

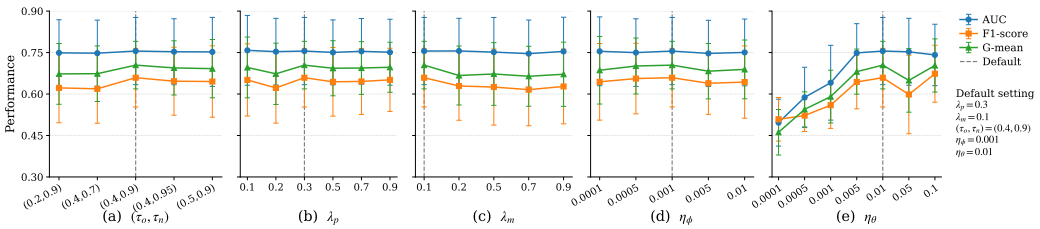


Figure 2: Parameter sensitivity analysis of FedDCOL on the *BUPA* dataset.

Baselines. This study compares FedDCOL against six recent FL baselines, grouped into three categories according to their primary mechanisms: 1) model architecture, including FedConcat (Diao et al., 2024) and UEFL (Zhang et al., 2025c); 2) aggregation strategy methods, including FedANAG (Zhang et al., 2025a) and FedSWA (Liu et al., 2025); and 3) data level methods, including FedWGAN (Zhang et al., 2025b) and FedKTL (Zhang et al., 2024a). Detailed descriptions of all baselines are provided in Appendix C.2.

Experimental configurations. The federated system consists of 5 clients, with full client participation in each communication round, and is trained for 150 communication rounds. On each client, the SC-Diff gradient is computed with a batch size of 128, and the server updates the global SC-Diff using AdamW with a learning rate of 0.001. The classifier is implemented as a two hidden layer MLP with hidden dimensions of 128 and 64 and ReLU activations. Each client computes the classifier gradient, and the server updates the global classifier using AdamW with a learning rate of 0.01. Detailed hyperparameter settings and additional implementation details are provided in Appendix C.3, with the complete settings summarized in Table 6.

Evaluation metrics and statistical tests. For imbalanced classification, this study adopts AUC, F1-score, and G-mean as the primary evaluation metrics (Yang et al., 2024b; Wang et al., 2025). AUC measures the overall discriminative ability of the classifier, F1-score balances precision and recall, and G-mean evaluates balanced performance across the majority and minority classes. To assess the statistical significance of performance differences across multiple datasets, the Friedman test is first applied, followed by the Nemenyi post hoc test when significant overall differences are observed. Detailed definitions of the statistical testing procedures are provided in Appendix C.4.

4.2 Parameter Sensitivity

This study evaluates the sensitivity of FedDCOL to the prototype relation weight λ_p, the inter-class margin weight λ_m, the class overlap structure thresholds τ_o and τ_n, and the learning rates η_ϕ and η_θ of SC-Diff and the classifier, respectively. Experiments are conducted on the *BUPA* dataset using AUC, F1-score, and G-mean as evaluation metrics. Figure 2 summarizes the sensitivity results, and the detailed results are provided in Table 7 of Appendix D.1. FedDCOL shows moderate robustness to λ_p, λ_m, τ_o and τ_n, although their effects differ across evaluation metrics. η_ϕ is also relatively stable within the tested range. In contrast, the classifier is substantially more sensitive to η_θ, and very small values lead to pronounced performance degradation.

4.3 Comparison with Baseline Methods

Overall performance. As shown in Tables 1, 2, and 3, FedDCOL achieves consistently strong and competitive performance across the seven imbalanced tabular datasets. Its average AUC, F1-score, and G-mean reach 0.825, 0.661, and 0.744, respectively, outperforming all six baseline methods. Compared with the strongest baseline for each metric, FedDCOL improves AUC over FedANAG by 2.23% and improves F1-score and G-mean over FedWGAN by 12.41% and 10.39%, respectively. The more pronounced gains in F1-score and G-mean indicate that FedDCOL is particularly effective for classification under imbalanced class distributions, while the improvement in AUC shows that these gains do not come at the expense of overall discriminative performance.

Performance under different imbalance ratios. FedDCOL generally achieves stronger performance when sufficient real minority class samples are available. On the six datasets with more than

Table 1: AUC comparison between FedDCOL and baseline methods across datasets.

Methods	BUPA	Spambase	Diabetes	German	ILPD	Parkinsons	Blood Transfusion	Average
FedConcat	0.597 ± 0.133	0.970 ± 0.007	0.791 ± 0.050	0.642 ± 0.035	0.655 ± 0.095	0.865 ± 0.054	0.738 ± 0.042	0.751
UEFL	0.503 ± 0.019	0.972 ± 0.013	0.722 ± 0.092	0.653 ± 0.050	0.492 ± 0.065	0.688 ± 0.172	0.581 ± 0.069	0.659
FedANAG	0.698 ± 0.123	$\mathbf{0.980 \pm 0.005}$	0.830 ± 0.046	0.713 ± 0.054	0.736 ± 0.078	0.941 ± 0.046	$\mathbf{0.751 \pm 0.051}$	0.807
FedSWA	0.616 ± 0.144	0.973 ± 0.006	0.803 ± 0.045	0.684 ± 0.047	0.681 ± 0.079	0.864 ± 0.068	0.729 ± 0.033	0.764
FedWGAN	0.716 ± 0.102	0.972 ± 0.005	0.804 ± 0.037	0.690 ± 0.059	0.731 ± 0.058	0.957 ± 0.041	0.737 ± 0.060	0.801
FedKTL	0.639 ± 0.079	0.942 ± 0.015	0.704 ± 0.057	0.608 ± 0.030	0.645 ± 0.064	0.784 ± 0.071	0.707 ± 0.056	0.718
FedDCOL	$\mathbf{0.755 \pm 0.122}$	$\mathbf{0.980 \pm 0.006}$	$\mathbf{0.831 \pm 0.046}$	$\mathbf{0.741 \pm 0.059}$	$\mathbf{0.750 \pm 0.072}$	$\mathbf{0.966 \pm 0.047}$	0.749 ± 0.041	**0.825**

Table 2: F1-score comparison between FedDCOL and baseline methods across datasets.

Methods	BUPA	Spambase	Diabetes	German	ILPD	Parkinsons	Blood Transfusion	Average
FedConcat	0.038 ± 0.084	0.898 ± 0.020	0.284 ± 0.213	0.074 ± 0.075	0.000 ± 0.000	0.543 ± 0.220	0.056 ± 0.125	0.270
UEFL	0.000 ± 0.000	0.912 ± 0.019	0.477 ± 0.305	0.389 ± 0.211	0.000 ± 0.000	0.537 ± 0.172	0.148 ± 0.154	0.352
FedANAG	0.528 ± 0.139	0.915 ± 0.022	0.623 ± 0.043	0.463 ± 0.107	0.333 ± 0.176	0.736 ± 0.184	0.338 ± 0.160	0.562
FedSWA	0.024 ± 0.074	0.906 ± 0.019	0.264 ± 0.227	0.052 ± 0.082	0.000 ± 0.000	0.563 ± 0.235	0.031 ± 0.068	0.263
FedWGAN	0.611 ± 0.084	0.904 ± 0.014	0.611 ± 0.085	0.455 ± 0.073	0.392 ± 0.098	0.786 ± 0.187	0.360 ± 0.114	0.588
FedKTL	0.405 ± 0.104	0.850 ± 0.032	0.375 ± 0.085	0.349 ± 0.063	0.212 ± 0.049	0.592 ± 0.107	0.222 ± 0.051	0.429
FedDCOL	$\mathbf{0.659 \pm 0.106}$	$\mathbf{0.922 \pm 0.017}$	$\mathbf{0.641 \pm 0.059}$	$\mathbf{0.556 \pm 0.074}$	$\mathbf{0.560 \pm 0.079}$	$\mathbf{0.809 \pm 0.166}$	$\mathbf{0.482 \pm 0.099}$	**0.661**

100 minority class samples, FedDCOL outperforms the strongest baseline in F1-score, with an average relative improvement of 18.06%. For example, on ILPD (IR = 2.49, Minority = 167), FedDCOL improves F1-score and G-mean over the strongest baseline by 42.86% and 28.63%, respectively. Nevertheless, on Parkinsons (IR = 3.06, Minority = 48), FedDCOL still improves G-mean over the strongest baseline by 4.92%. These results indicate that FedDCOL remains effective across different imbalance settings, although its performance is influenced by the amount of real minority class data available.

Comparison with different categories of baselines. Compared with model architecture and aggregation strategy methods, FedDCOL directly enriches minority class samples and shows more pronounced advantages on imbalance sensitive metrics such as F1-score and G-mean. More importantly, compared with FedWGAN, the strongest data-level baseline that also employs data generation, FedDCOL further improves AUC, F1-score, and G-mean by 3.00%, 12.41%, and 10.39%, respectively. These results suggest that the performance gains of FedDCOL cannot be explained solely by increasing the number of minority class samples, but are also consistent with its design of dynamically generating minority class samples during federated training and applying structure selection.

4.4 Statistical Test Results

This study compares the overall performance of FedDCOL and the baselines using average ranks, the Friedman test, and the Nemenyi post hoc test. FedDCOL achieves average ranks of 1.286, 1.000, and 1.000 on AUC, F1-score, and G-mean, respectively, ranking first on all three metrics. The complete ranking results are reported in Table 8 of Appendix D.2. The Friedman test reveals significant differences among the compared methods on three metrics at the significance level of $\alpha = 0.05$. The subsequent Nemenyi post hoc test yields a critical difference of $CD = 3.4044$, with the detailed results shown in Figure 3. Overall, FedDCOL achieves the best average rank across all three evaluation metrics, and its advantages on F1-score and G-mean reach statistical significance in multiple baseline comparisons, with detailed results provided in Table 9 of Appendix D.2.

4.5 Ablation Study

To evaluate the contribution of each component of FedDCOL, this study designs three ablation variants: **w/o Class Overlap Modeling** removes the sample structure partitioning and directly uses the original local data to train SC-Diff; **w/o Structure Constraints** removes the prototype relation con-

Table 3: G-mean comparison between FedDCOL and baseline methods across datasets.

Methods	BUPA	Spambase	Diabetes	German	ILPD	Parkinsons	Blood Transfusion	Average
FedConcat	0.064 ± 0.137	0.913 ± 0.018	0.374 ± 0.238	0.154 ± 0.141	0.000 ± 0.000	0.613 ± 0.235	0.081 ± 0.173	0.314
UEFL	0.000 ± 0.000	0.923 ± 0.017	0.551 ± 0.301	0.490 ± 0.261	0.000 ± 0.000	0.667 ± 0.143	0.241 ± 0.196	0.410
FedANAG	0.604 ± 0.108	0.929 ± 0.019	0.698 ± 0.031	0.584 ± 0.091	0.465 ± 0.164	0.794 ± 0.142	0.453 ± 0.190	0.647
FedSWA	0.037 ± 0.116	0.921 ± 0.017	0.352 ± 0.248	0.103 ± 0.144	0.000 ± 0.000	0.612 ± 0.238	0.057 ± 0.122	0.297
FedWGAN	0.665 ± 0.070	0.921 ± 0.013	0.694 ± 0.067	0.586 ± 0.061	0.531 ± 0.079	0.833 ± 0.151	0.490 ± 0.095	0.674
FedKTL	0.443 ± 0.110	0.872 ± 0.028	0.441 ± 0.095	0.485 ± 0.061	0.285 ± 0.058	0.682 ± 0.103	0.307 ± 0.071	0.502
FedDCOL	**0.705** ± **0.086**	**0.938** ± **0.015**	**0.713** ± **0.050**	**0.662** ± **0.066**	**0.683** ± **0.064**	**0.874** ± **0.141**	**0.630** ± **0.098**	**0.744**

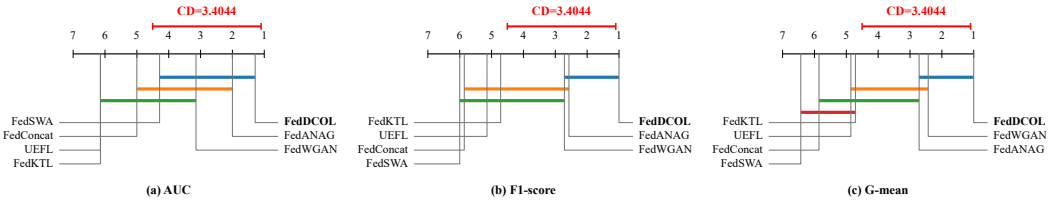


Figure 3: Nemenyi post hoc test results on AUC, F1-score, and G-mean.

straint $\mathcal{L}_{\text{proto}}$ and the inter-class margin constraint $\mathcal{L}_{\text{marg}}$, retaining only the base diffusion loss; **w/o Dynamic Continual Oversampling** removes the dynamic update of SC-Diff during the communication process, trains the generator only at the initial stage, and repeatedly reuses the generated samples in subsequent rounds.

Table 4 reports the average results of FedDCOL and its three ablation variants across seven datasets. The full FedDCOL achieves AUC, F1-score, and G-mean values of 0.825, 0.661, and 0.744, respectively, outperforming all ablation variants. In particular, removing Class Overlap Modeling, Structure Constraints, and Dynamic Continual Oversampling leads to F1-score decreases of 0.026, 0.022, and 0.019, respectively, and G-mean decreases of 0.020, 0.016, and 0.015. Compared with AUC, the more pronounced degradation in F1-score and G-mean suggests that these components are particularly beneficial for minority class recognition under imbalanced class distributions, with the detailed results reported in Table 10 of Appendix D.3.

Table 4: Ablation study of FedDCOL.

Method	AUC	F1-score	G-mean
FedDCOL	**0.825**	**0.661**	**0.744**
w/o Class Overlap Modeling	0.815	0.635	0.724
w/o Structure Constraints	0.817	0.639	0.727
w/o Dynamic Continual Oversampling	0.817	0.642	0.728

5 Conclusion

To address imbalanced class distributions in FL for tabular classification, this study proposes FedDCOL, which integrates local structure mining, SC-Diff optimization, dynamic local minority class sample generation, structure selection, and federated classifier learning. FedDCOL partitions samples from both classes into safe, boundary, and noisy regions, and uses the reliable regions to construct stable prototypes while retaining the complete local dataset for base diffusion learning. This study develops a Structure-Constrained Diffusion Model (SC-Diff) that consists of a base diffusion loss, a prototype relation constraint $\mathcal{L}_{\text{proto}}$, and an inter-class margin constraint $\mathcal{L}_{\text{marg}}$. The server aggregates client-side SC-Diff gradients to update the shared SC-Diff parameters. Subsequently, all participating clients load the latest parameters into their local SC-Diff models, generate new minority class samples locally, and perform structure selection before incorporating the selected samples into local classifier training. Therefore, the dynamic and continual nature of FedDCOL stems from the repeated SC-Diff parameter updates and local minority class sample regeneration throughout federated training. In future work, differential privacy and secure aggregation will be further introduced to enhance the privacy preserving capability of FedDCOL in practical applications.

AI USE STATEMENT

Generative AI tools were used solely to assist with language polishing and improving the clarity of the manuscript. They were not used for method design or experimental design. All AI-assisted text was reviewed and verified by the authors, who take full responsibility for the final content of the paper.

REFERENCES

Jesús Alcalá-Fdez, Alberto Fernández, Julián Luengo, Joaquín Derrac, Salvador García, Luciano Sánchez, and Francisco Herrera. KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic and Soft Computing*, 17(2-3):255–287, 2011.

Nur Athirah Azhar, Muhammad Syafiq Mohd Pozi, Aniza Mohamed Din, and Adam Jatowt. An investigation of SMOTE-based methods for imbalanced datasets with data complexity analysis. *IEEE Transactions on Knowledge and Data Engineering*, 35(7):6651–6672, 2023. doi: 10.1109/TKDE.2022.3179381.

Hao Bai, Kun He, Yuqing Li, Jing Chen, Haowei Li, Zhongmou Liu, Xuanang Yang, and Ruiying Du. GetFed: Accurate, differentially private federated learning with GAN-based data generation. *IEEE Transactions on Dependable and Secure Computing*, 22(6):7463–7478, 2025. doi: 10.1109/TDSC.2025.3597635.

Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002. doi: 10.1613/jair.953.

Cheng-Wei Ching, Xin Chen, Taehwan Kim, Jian-Jhih Kuo, Dilma Da Silva, and Liting Hu. Totoro+: An adaptive and scalable edge federated learning system. *IEEE Transactions on Parallel and Distributed Systems*, 37(7):1740–1757, 2026. doi: 10.1109/TPDS.2026.3696917.

Yutong Dai, Zeyuan Chen, Junnan Li, Shelby Heinecke, Lichao Sun, and Ran Xu. Tackling data heterogeneity in federated learning with class prototypes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 7314–7322, 2023. doi: 10.1609/aaai.v37i6.25891.

Yiqun Diao, Qinbin Li, and Bingsheng He. Exploiting label skews in federated learning with model concatenation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 11784–11792, 2024. doi: 10.1609/aaai.v38i10.29063.

Hongwei Ding, Nana Huang, Qi Tao, Jiaqi Liang, and Xiaohui Cui. Balancing act: MDGAN for imbalanced tabular data synthesis. *IEEE Transactions on Knowledge and Data Engineering*, 37 (11):6304–6319, 2025. doi: 10.1109/TKDE.2025.3607862.

Jiadong Fu, Jiang Fang, Jiyan Sun, Shangyuan Zhuang, Yinlong Liu, and Zhiqiang Lv. Toward efficient distributed network security: A lightweight multitask traffic analysis framework. *IEEE Transactions on Networking*, 34:2271–2286, 2026. doi: 10.1109/TON.2025.3643832.

Yu-Cheng He, Yao-Xiang Ding, Han-Jia Ye, and Zhi-Hua Zhou. Learning only when it matters: Cost-aware long-tailed classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 12411–12420, 2024.

Jonathan Ho, Ajay N. Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020.

Xiaowen Hu, Ming Zheng, Rui Zhu, Xuan Zhang, and Zhi Jin. Fed-OLF: Federated oversampling learning framework for imbalanced software defect prediction under privacy protection. *IEEE Transactions on Reliability*, 74(3):3266–3280, 2025. doi: 10.1109/TR.2024.3524064.

Markelle Kelly, Rachel Longjohn, and Kolby Nottingham. The UCI machine learning repository. University of California, Irvine, 2023.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.

Yuri Kinoshita, Kenta Oono, Kenji Fukumizu, Yuichi Yoshida, and Shin-Ichi Maeda. Controlling posterior collapse by an inverse Lipschitz constraint on the decoder network. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 17041–17060, 2023.

Akim Kotelnikov, Dmitry Baranchuk, Ivan Rubachev, and Artem Babenko. TabDDPM: Modelling tabular data with diffusion models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 17564–17579, 2023.

Junkang Liu, Yuanyuan Liu, Fanhua Shang, Hongying Liu, Jin Liu, and Wei Feng. Improving generalization in federated learning with highly heterogeneous data via momentum-based stochastic controlled weight averaging. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 38894–38939, 2025.

Yanbiao Ma, Wei Dai, Wenke Huang, and Jiayi Chen. Geometric knowledge-guided localized global distribution alignment for federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20958–20968, 2025. doi: 10.1109/CVPR52734.2025.01952.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54, pp. 1273–1282, 2017.

Shiqi Ren, Jinliang Ding, Cuie Yang, and Yiu-ming Cheung. GOIO: Generative oversampling approach to class imbalance and overlap of tabular data. *IEEE Transactions on Knowledge and Data Engineering*, 37(11):6450–6463, 2025.

Shiqi Ren, Jinliang Ding, and Yiu-ming Cheung. Diffusion GAN-based oversampling for imbalanced tabular data. *IEEE Transactions on Knowledge and Data Engineering*, 38(2):983–996, 2026. doi: 10.1109/TKDE.2025.3639433.

Juntong Shi, Minkai Xu, Harper Hua, Hengrui Zhang, Stefano Ermon, and Jure Leskovec. TabDiff: A mixed-type diffusion model for tabular data generation. In *International Conference on Learning Representations*, 2025.

Soobin Um, Suhyeon Lee, and Jong Chul Ye. Don't play favorites: Minority guidance for diffusion models. In *International Conference on Learning Representations*, 2024.

Lei Wang, Shanshan Huang, Chunyuan Zheng, Jun Liao, Xiaofei Zhu, Haoxuan Li, and Li Liu. Mitigating data imbalance in time series classification based on counterfactual minority samples augmentation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2962–2973, 2025.

Lixu Wang, Shichao Xu, Xiao Wang, and Qi Zhu. Addressing class imbalance in federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 10165–10173, 2021. doi: 10.1609/aaai.v35i11.17219.

Zifeng Wang, Chufan Gao, Cao Xiao, and Jimeng Sun. MediTab: Scaling medical tabular data predictors via data consolidation, enrichment, and refinement. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pp. 6062–6070, 2024. doi: 10.24963/ijcai.2024/670.

Chengyao Wen and Yin Lou. On finding bi-objective pareto-optimal fraud prevention rule sets for fintech applications. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 5959–5968, 2024. doi: 10.1145/3637528.3671521.

Zikai Xiao, Zihan Chen, Songshang Liu, Hualiang Wang, Yang Feng, Jin Hao, Joey Tianyi Zhou, Jian Wu, Howard Hua Yang, and Zuozhu Liu. Fed-GraB: Federated long-tailed learning with self-adjusting gradient balancer. In *Advances in Neural Information Processing Systems*, volume 36, pp. 77745–77757, 2023.

Zikai Xiao, Zihan Chen, Liyinglan Liu, Yang Feng, Joey Tianyi Zhou, Jian Wu, Wanlu Liu, Howard Hao Yang, and Zuozhu Liu. FedLoGe: Joint local and generic federated learning under long-tailed data. In *International Conference on Learning Representations*, 2024.

Yu Xie, Junkai Shan, Lifei Wei, Jiamin Yao, and MengChu Zhou. GAN-based hybrid sampling method for transaction fraud detection. *IEEE Transactions on Knowledge and Data Engineering*, 37(10):5905–5918, 2025. doi: 10.1109/TKDE.2025.3589885.

Yunlu Yan, Huazhu Fu, Yuexiang Li, Jinheng Xie, Jun Ma, Guang Yang, and Lei Zhu. A simple data augmentation for feature distribution skewed federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 25749–25758, 2025. doi: 10.1109/CVPR52734.2025.02398.

June Yong Yang, Geondo Park, Joowon Kim, Hyeongwon Jang, and Eunho Yang. Language-interfaced tabular oversampling via progressive imputation and self-authentication. In *International Conference on Learning Representations*, pp. 21524–21546, 2024a.

Kaixiang Yang, Zhiwen Yu, Wuxing Chen, Zefeng Liang, and C. L. Philip Chen. Solving the imbalanced problem by metric learning and oversampling. *IEEE Transactions on Knowledge and Data Engineering*, 36(12):9294–9307, 2024b. doi: 10.1109/TKDE.2024.3419834.

Zhiqin Yang, Yonggang Zhang, Yu Zheng, Xinmei Tian, Hao Peng, Tongliang Liu, and Bo Han. FedFed: Feature distillation against data heterogeneity in federated learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 60397–60428, 2023.

Rui Ye, Mingkai Xu, Jianyu Wang, Chenxin Xu, Siheng Chen, and Yanfeng Wang. FedDisco: Federated learning with discrepancy-aware collaboration. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 39879–39902, 2023.

Rui Ye, Yaxin Du, Zhenyang Ni, Yanfeng Wang, and Siheng Chen. Fake it till make it: Federated learning with consensus-oriented generation. In *International Conference on Learning Representations*, 2024.

Tehrim Yoon, Minyoung Hwang, and Eunho Yang. VQ-FedDiff: Federated learning algorithm of diffusion models with client-specific vector-quantized conditioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(12):11863–11873, 2025. doi: 10.1109/TPAMI.2025.3602282.

Yaopei Zeng, Lei Liu, Li Liu, Li Shen, Shaoguo Liu, and Baoyuan Wu. Global balanced experts for federated long-tailed learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4815–4825, 2023. doi: 10.1109/ICCV51070.2023.00444.

Hao Zhang, Chenglin Li, Wenrui Dai, Ziyang Zheng, Junni Zou, and Hongkai Xiong. Stabilizing and accelerating federated learning on heterogeneous data with partial client participation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(1):67–83, 2025a. doi: 10.1109/TPAMI.2024.3469188.

Jianqing Zhang, Yang Liu, Yang Hua, and Jian Cao. An upload-efficient scheme for transferring knowledge from a server-side pre-trained generator to clients in heterogeneous federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12109–12119, 2024a.

Jie Zhang, Zhiqi Li, Bo Li, Jianghe Xu, Shuang Wu, Shouhong Ding, and Chao Wu. Federated learning with label distribution skew via logits calibration. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pp. 26311–26329, 2022.

Mingxue Zhang, Wei Meng, You Zhou, and Kui Ren. CSChecker: Revisiting GDPR and CCPA compliance of cookie banners on the web. In *Proceedings of the IEEE/ACM 46th International Conference on Software Engineering*, pp. 1–12, 2024b. doi: 10.1145/3597503.3639159.

Tianlong Zhang, Zhe Xue, Adnan Mahmood, Junping Du, Yuchen Dong, Shilong Ou, Lang Feng, Ming-Hsuan Yang, and Yuankai Qi. Generating synthetic data for unsupervised federated learning of cross-modal retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 22569–22577, 2025b. doi: 10.1609/aaai.v39i21.34415.

Tianyi Zhang, Yu Cao, and Dianbo Liu. Uncertainty-based extensible codebook for discrete federated learning in heterogeneous data silos. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 74565–74582, 2025c.

Wentian Zhang, Haozhe Liu, Bing Li, Jinheng Xie, Yawen Huang, Yuexiang Li, Yefeng Zheng, and Bernard Ghanem. Dynamically masked discriminator for GANs. In *Advances in Neural Information Processing Systems*, volume 36, pp. 23094–23114, 2023. doi: 10.52202/075280-1002.

A ALGORITHMS

In this section, this study provides an algorithmic illustration of the proposed FedDCOL. Algorithm 1 and Algorithm 2 present the class overlap structure modeling and SC-Diff, respectively. Algorithm 3 presents FedDCOL's federated dynamic continual oversampling procedure.

Algorithm 1 Class overlap structure modeling

Input: Local dataset $\mathcal{D}_k$, neighbors q, thresholds τ_o, τ_n
Output: $\mathcal{D}_k^{\text{safe}}, \mathcal{D}_k^{\text{boundary}}, \mathcal{D}_k^{\text{noisy}}$

Compute class centers $\mu_y, \mu_{\bar{y}}$
for each $(\mathbf{x}, y) \in \mathcal{D}_k$ **do**
  Find the q nearest neighbors $\mathcal{N}_q(\mathbf{x})$
  $s(\mathbf{x}) \leftarrow q^{-1} \sum_{j \in \mathcal{N}_q(\mathbf{x})} \mathbb{I}(y_j = y)$
  $d(\mathbf{x}) \leftarrow \frac{\|\mathbf{x}-\mu_y\|_2}{\|\mathbf{x}-\mu_y\|_2 + \|\mathbf{x}-\mu_{\bar{y}}\|_2 + \varepsilon}$
  $O(\mathbf{x}) \leftarrow \exp\left[-\frac{1}{2}\left(\frac{s(\mathbf{x})}{\max(d(\mathbf{x}),\varepsilon)}\right)^2\right]$
  Assign the sample to safe, boundary, or noisy regions according to τ_o and τ_n
end for
return $\mathcal{D}_k^{\text{safe}}, \mathcal{D}_k^{\text{boundary}}, \mathcal{D}_k^{\text{noisy}}$

Algorithm 2 Local SC-Diff training procedure

Input: Global SC-Diff parameters ϕ^r, local dataset $\mathcal{D}_k$, safe and boundary sets $\mathcal{D}_k^{\text{safe}}$ and $\mathcal{D}_k^{\text{boundary}}$, loss weights λ_p, λ_m
Output: Local SC-Diff gradient $\mathbf{g}_{\phi,k}^r$

Construct prototype sets $\mathcal{P}_0$ and $\mathcal{P}_1$ from $\mathcal{D}_k^{\text{safe}}$
Sample mini-batch $\mathcal{B} \subset \mathcal{D}_k$ and diffusion timestep t
Sample noisy samples $\mathbf{x}_t$ and obtain reconstruction $\hat{\mathbf{x}}_0$
Compute base diffusion loss $\mathcal{L}_{\text{base}}$ on $\mathcal{B}$
Select $\mathcal{J}_p = \{\mathbf{x}_0 \in \mathcal{B} \cap (\mathcal{D}_k^{\text{safe}} \cup \mathcal{D}_k^{\text{boundary}}) : y = 1\}$, $\mathcal{J}_m = \{\mathbf{x}_0 \in \mathcal{B} \cap \mathcal{D}_k^{\text{boundary}} : y = 1\}$
Compute $\mathcal{L}_{\text{proto}}(\mathcal{J}_p, \mathcal{P}_0, \mathcal{P}_1)$ and $\mathcal{L}_{\text{marg}}(\mathcal{J}_m, \mathcal{P}_0, \mathcal{P}_1)$
$\mathcal{L}_{\text{SC}} = \mathcal{L}_{\text{base}} + \lambda_p \mathcal{L}_{\text{proto}} + \lambda_m \mathcal{L}_{\text{marg}}$
$\mathbf{g}_{\phi,k}^r \leftarrow \nabla_\phi \mathcal{L}_{\text{SC}}$ at parameters ϕ^r
return $\mathbf{g}_{\phi,k}^r$

Algorithm 3 Training procedure of FedDCOL

Input: Client datasets $\{\mathcal{D}_k\}_{k=1}^K$, communication rounds R, structural thresholds τ_o, τ_n, loss weights λ_p, λ_m
Output: Global classifier parameters θ^R and global SC-Diff parameters ϕ^R

Initialize global SC-Diff parameters ϕ^0 and global classifier parameters θ^0
for each client $k = 1, \ldots, K$ **do**
  $(\mathcal{D}_k^{\text{safe}}, \mathcal{D}_k^{\text{boundary}}, \mathcal{D}_k^{\text{noisy}}) \leftarrow Algorithm1(\mathcal{D}_k, q, \tau_o, \tau_n)$
end for
for communication round $r = 0, \ldots, R-1$ **do**
  Select participating clients $\mathcal{S}^r$ and broadcast ϕ^r, θ^r
  for each $k \in \mathcal{S}^r$ in parallel **do**
    $\mathbf{g}_{\phi,k}^r \leftarrow Algorithm2(\phi^r, \mathcal{D}_k, \mathcal{D}_k^{\text{safe}}, \mathcal{D}_k^{\text{boundary}}, \lambda_p, \lambda_m)$
  end for
  $\omega_k^r \leftarrow n_k / \sum_{j \in \mathcal{S}^r} n_j$, $\mathbf{g}_\phi^r \leftarrow \sum_{k \in \mathcal{S}^r} \omega_k^r \mathbf{g}_{\phi,k}^r$, $\phi^{r+1} \leftarrow \text{AdamW}(\phi^r, \mathbf{g}_\phi^r, \eta_\phi)$
  Broadcast the updated ϕ^{r+1} to participating clients
  for each $k \in \mathcal{S}^r$ in parallel **do**
    Sample $\mathcal{C}_k^r \sim p_{\phi^{r+1}}(\mathbf{x} \mid y = 1)$, $\mathcal{A}_k^r \leftarrow \mathcal{F}_k^{\text{str}}(\mathcal{C}_k^r; \mathcal{D}_k)$
    $\mathbf{g}_{\theta,k}^r \leftarrow \nabla_\theta \mathcal{L}_{\text{cls},k}^r(\theta^r; \mathcal{D}_k, \mathcal{A}_k^r)$
  end for
  $\mathbf{g}_\theta^r \leftarrow \sum_{k \in \mathcal{S}^r} \omega_k^r \mathbf{g}_{\theta,k}^r$, $\theta^{r+1} \leftarrow \text{AdamW}(\theta^r, \mathbf{g}_\theta^r, \eta_\theta)$
end for
return θ^R, ϕ^R

B Detailed Diffusion Process

SC-Diff models numerical and categorical features using Gaussian and multinomial diffusion, respectively. The corresponding base diffusion objectives are introduced in the main text. This section further describes the forward perturbation and reverse denoising processes. Let $\mathbf{x}_0 = [\mathbf{x}_0^{\text{num}}, \mathbf{x}_0^{\text{cat}}]$ denote the original sample, y its class label, and T the total number of diffusion steps.

B.1 Gaussian Diffusion for Numerical Features

For numerical features, SC-Diff adopts a Gaussian diffusion process. During the forward process, Gaussian noise is progressively introduced into the original numerical features $\mathbf{x}_0^{\text{num}}$. Let $t \in \{1, \dots, T\}$ denote the diffusion step and $\beta_t \in (0, 1)$ the noise variance at t, and define

$$\alpha_t = 1 - \beta_t, \ \bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s. \tag{14}$$

The distribution from the original numerical features $\mathbf{x}_0^{\text{num}}$ to an arbitrary t is given by

$$q(\mathbf{x}_t^{\text{num}} \mid \mathbf{x}_0^{\text{num}}) = \mathcal{N}\left(\mathbf{x}_t^{\text{num}}; \sqrt{\bar{\alpha}_t}\mathbf{x}_0^{\text{num}}, (1 - \bar{\alpha}_t)\mathbf{I}\right). \tag{15}$$

A noisy sample at an arbitrary t can then be directly constructed as

$$\mathbf{x}_t^{\text{num}} = \sqrt{\bar{\alpha}_t}\mathbf{x}_0^{\text{num}} + \sqrt{1 - \bar{\alpha}_t}\,\boldsymbol{\epsilon}, \ \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{16}$$

During the reverse denoising process, SC-Diff uses the conditional noise predictor $\boldsymbol{\epsilon}_\phi(\mathbf{x}_t, t, y)$ to estimate the Gaussian noise introduced during the forward process and progressively recover a less noisy state $\mathbf{x}_{t-1}^{\text{num}}$ from the current noisy state $\mathbf{x}_t^{\text{num}}$. The corresponding reverse denoising distribution is written as

$$p_\phi(\mathbf{x}_{t-1}^{\text{num}} \mid \mathbf{x}_t, t, y) = \mathcal{N}\left(\mathbf{x}_{t-1}^{\text{num}}; \boldsymbol{\mu}_\phi(\mathbf{x}_t, t, y), \sigma_t^2\mathbf{I}\right), \tag{17}$$

where σ_t^2 denotes the variance used at reverse t, and the reverse mean is determined from the predicted noise as

$$\boldsymbol{\mu}_\phi(\mathbf{x}_t, t, y) = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t^{\text{num}} - \frac{\beta_t}{\sqrt{1 - \bar{\alpha}_t}}\boldsymbol{\epsilon}_\phi(\mathbf{x}_t, t, y)\right). \tag{18}$$

Iteratively applying the above reverse denoising process from $t = T$ to $t = 1$ progressively recovers the numerical features from noise.

B.2 Multinomial Diffusion for Categorical Features

For categorical features, SC-Diff adopts a multinomial diffusion process that perturbs discrete variables directly in the discrete state space. Let the j-th categorical feature contain K_j categories and be represented by a one-hot vector $\mathbf{x}_0^j \in \{0, 1\}^{K_j}$.

During the forward process, the categorical state is gradually driven toward a uniform categorical distribution. Given the cumulative noise coefficient $\bar{\alpha}_t$, the state distribution at t is

$$q(\mathbf{x}_t^j \mid \mathbf{x}_0^j) = \text{Cat}\left(\bar{\alpha}_t\mathbf{x}_0^j + \frac{1 - \bar{\alpha}_t}{K_j}\mathbf{1}\right). \tag{19}$$

Consequently, information about the original category is progressively weakened as t increases, while the categorical probabilities approach a uniform distribution.

During the reverse denoising process, the model learns the reverse categorical distribution of the j-th feature from the current mixed-type noisy state $\mathbf{x}_t$, diffusion step t, and class condition y:

$$p_\phi(\mathbf{x}_{t-1}^j \mid \mathbf{x}_t, t, y) = \text{Cat}\left(\mathbf{x}_{t-1}^j; \boldsymbol{\pi}_\phi^j(\mathbf{x}_t, t, y)\right), \tag{20}$$

where $\boldsymbol{\pi}_\phi^j(\mathbf{x}_t, t, y) \in \Delta^{K_j - 1}$ denotes the predicted probability vector over the K_j categories. By progressively performing reverse denoising from $t = T$ to $t = 1$, the categorical variables are gradually recovered from noisy states to discrete categorical states.

C Details of Experimental Setups

C.1 Datasets

Table 5: Dataset statistics.

Dataset	Features	Samples	Majority	Minority	IR
BUPA	6	344	200	144	1.39
Spambase	57	4596	2785	1811	1.54
Diabetes	8	768	500	268	1.87
German	20	999	699	300	2.33
ILPD	10	583	416	167	2.49
Parkinsons	22	195	147	48	3.06
Blood Transfusion	4	748	570	178	3.20

This study uses seven real world imbalanced tabular datasets: BUPA, Spambase, Diabetes, German, ILPD, Parkinsons, and Blood Transfusion. These datasets vary substantially in data scale, feature dimension, and imbalance ratio (IR), containing 4–57 features, 195–4,596 samples, and 48–1,811 minority-class instances, with IR ranging from 1.39 to 3.20. Table 5 provides the overall statistics of the datasets, including the number of features, samples, majority class instances, minority class instances, and IR.

C.2 Baselines

This study compares FedDCOL with six recent FL baselines, grouped into model architecture, aggregation strategy, and data level methods. When available, official implementations are adopted and adapted to tabular classification when necessary. The data level methods serve as more direct baselines for FedDCOL, as they also mitigate imbalanced class distributions in FL through data generation or augmentation.

Model architecture methods.

FedConcat (Diao et al., 2024) clusters clients by label distributions, collaboratively trains a model within each cluster, and concatenates cluster models as the basis of the global model, preserving local knowledge under label skew.

UEFL (Zhang et al., 2025c) dynamically maps latent features to trainable discrete vectors, assesses predictive uncertainty, and extends the codebook for high uncertainty clients to better accommodate heterogeneous data distributions.

Aggregation strategy methods.

FedANAG (Zhang et al., 2025a) applies Nesterov accelerated gradient at both server and client levels, alternating global and local momentum updates while using global momentum to improve local-update stability under data heterogeneity.

FedSWA (Liu et al., 2025) incorporates stochastic weight averaging into FL to seek flatter minima and improve global model generalization under highly heterogeneous data.

Data level methods.

FedWGAN (Zhang et al., 2025b) trains local WGANs with direct and prototype-based contrastive losses plus Jensen–Shannon divergence, then generates and filters synthetic data at the server.

FedKTL (Zhang et al., 2024a) aligns client prototypes with the valid latent domain of a server side pre-trained generator and uses prototypical image–vector pairs for supervised knowledge transfer to clients.

C.3 EXPERIMENTAL CONFIGURATIONS

Table 6: Hyperparameter settings of FedDCOL.

Component	Hyperparameter	Setting
Structure constraints	Number of neighbors q	5
	Overlap threshold τ_o	0.4
	Noise threshold τ_n	0.9
	Margin tolerance Δ	0.1
	Smoothing temperature τ	0.05
SC-Diff	Diffusion steps T	256
	Denoising MLP hidden dimensions	$[128, 256]$
	SC-Diff learning rate η_ϕ	0.001
	Prototype relation weight λ_p	0.3
	Inter-class margin weight λ_m	0.1
Classifier	Classifier learning rate η_θ	0.01
	Dropout	0.1
	Hidden dimensions	$[128, 64]$
Dynamic oversampling	Generation starting round	40
	Synthetic balance ratio	0.5
	Candidate multiplier	3.0
Federated training	Number of clients K	5
	Client participation	100%
	Maximum communication rounds	150
	Training batch size	128

To simulate statistical heterogeneity across clients, the training samples are partitioned using a Dirichlet distribution with the concentration parameter set to 1 by default. For each dataset, FedDCOL and all competing methods are evaluated using stratified 10-fold cross-validation. In each fold, the held-out fold is used exclusively for testing, while 20% of the remaining training data is stratified as a validation set. All preprocessing operations are fitted using only the training portion and subsequently applied to the validation and test sets, preventing information leakage. A fixed random seed is used for data splitting, client partitioning, and model initialization. The mean and standard deviation across the ten test folds are reported for all methods.

Additional implementation details. When $\mathcal{J}_p$ or $\mathcal{J}_m$ is empty, the corresponding structural loss is set to zero. If no generated candidate is retained, $\mathcal{A}_k^r = \varnothing$ and the classifier is trained using only the local real dataset $\mathcal{D}_k$.

During the early communication rounds, SC-Diff is still being optimized, and the generated minority class samples may therefore have limited quality and structural reliability. To avoid introducing unreliable generated minority class samples prematurely, minority class generation is activated from communication round 40 and is performed in each subsequent round using the latest global SC-Diff. For each client, the target number of generated minority class samples is set to 50% of the local majority–minority sample count gap. A candidate pool three times the target size is then generated, from which samples are selected using the structure selection for classifier training. To prevent generated minority class samples from dominating classifier optimization, the number of selected generated minority class samples is limited to 75% of the local real training set size.

C.4 STATISTICAL TESTS

To further assess whether the performance differences between FedDCOL and the baseline methods are statistically significant across multiple datasets, the Friedman test and the Nemenyi post hoc test are employed. The methods are ranked according to their test performance on each dataset, with rank 1 assigned to the best-performing method. Tied methods are assigned the corresponding average rank. The Friedman test is then performed based on the average ranks of all methods across datasets to determine whether there is a statistically significant overall difference among them. The significance level is set to $\alpha = 0.05$. When $p < \alpha$, the null hypothesis of no significant performance difference among the methods is rejected. If the Friedman test indicates a significant

overall difference, the Nemenyi post hoc test is subsequently applied for pairwise comparisons. Two methods are considered significantly different if the difference between their average ranks exceeds the critical difference (CD).

D Details of Experimental Results

D.1 Parameter Sensitivity

Table 7: Parameter sensitivity analysis of FedDCOL.

Hyperparameter	Parameter Setting	AUC	F1-score	G-mean
(τ_o, τ_n)	$(0.2, 0.9)$	0.749 ± 0.120	0.622 ± 0.126	0.673 ± 0.110
	$(0.4, 0.7)$	0.748 ± 0.120	0.619 ± 0.125	0.674 ± 0.101
	$(\mathbf{0.4}, \mathbf{0.9})$	$\mathbf{0.755 \pm 0.122}$	$\mathbf{0.659 \pm 0.106}$	$\mathbf{0.705 \pm 0.086}$
	$(0.4, 0.95)$	0.753 ± 0.119	0.646 ± 0.123	0.695 ± 0.098
	$(0.5, 0.9)$	0.752 ± 0.125	0.645 ± 0.129	0.692 ± 0.105
λ_p	0.1	$\mathbf{0.758 \pm 0.126}$	0.651 ± 0.131	0.696 ± 0.110
	0.2	0.753 ± 0.121	0.622 ± 0.127	0.673 ± 0.111
	$\mathbf{0.3}$	0.755 ± 0.122	$\mathbf{0.659 \pm 0.106}$	$\mathbf{0.705 \pm 0.086}$
	0.5	0.751 ± 0.118	0.644 ± 0.124	0.693 ± 0.100
	0.7	0.754 ± 0.119	0.646 ± 0.119	0.694 ± 0.095
	0.9	0.751 ± 0.119	0.651 ± 0.114	0.697 ± 0.091
λ_m	$\mathbf{0.1}$	0.755 ± 0.122	$\mathbf{0.659 \pm 0.106}$	$\mathbf{0.705 \pm 0.086}$
	0.2	$\mathbf{0.756 \pm 0.118}$	0.629 ± 0.124	0.667 ± 0.107
	0.5	0.752 ± 0.125	0.625 ± 0.138	0.673 ± 0.113
	0.7	0.746 ± 0.122	0.616 ± 0.131	0.664 ± 0.109
	0.9	0.754 ± 0.124	0.627 ± 0.135	0.672 ± 0.116
η_ϕ	0.0001	$\mathbf{0.755 \pm 0.124}$	0.644 ± 0.139	0.686 ± 0.122
	0.0005	0.750 ± 0.122	0.656 ± 0.128	0.701 ± 0.101
	$\mathbf{0.001}$	$\mathbf{0.755 \pm 0.122}$	$\mathbf{0.659 \pm 0.106}$	$\mathbf{0.705 \pm 0.086}$
	0.005	0.747 ± 0.120	0.639 ± 0.112	0.683 ± 0.100
	0.01	0.751 ± 0.121	0.643 ± 0.131	0.689 ± 0.106
η_θ	0.0001	0.497 ± 0.085	0.509 ± 0.079	0.462 ± 0.082
	0.0005	0.588 ± 0.109	0.522 ± 0.058	0.545 ± 0.063
	0.001	0.641 ± 0.135	0.560 ± 0.084	0.591 ± 0.096
	0.005	0.749 ± 0.106	0.644 ± 0.097	0.681 ± 0.082
	$\mathbf{0.01}$	$\mathbf{0.755 \pm 0.122}$	0.659 ± 0.106	$\mathbf{0.705 \pm 0.086}$
	0.05	0.753 ± 0.121	0.599 ± 0.142	0.649 ± 0.115
	0.1	0.741 ± 0.111	$\mathbf{0.673 \pm 0.103}$	0.703 ± 0.096

For each experiment, only the hyperparameter under investigation is varied, while all others are fixed to their default values. Table 7 provides the detailed numerical results. For the prototype relation weight, $\lambda_p = 0.3$ achieves the highest mean F1-score and G-mean, while AUC varies only slightly across the tested values. For the inter-class margin weight, AUC remains relatively stable, whereas F1-score and G-mean generally decrease when λ_m exceeds 0.1, supporting $\lambda_m = 0.1$ as the default. For the structure thresholds, $(\tau_o, \tau_n) = (0.4, 0.9)$ yields the highest mean values for all three metrics. The SC-Diff learning rate is relatively stable between 0.0005 and 0.001, with $\eta_\phi = 0.001$ achieving the highest mean F1-score and G-mean. In contrast, the classifier is substantially more sensitive to its learning rate: very small values lead to pronounced performance degradation. $\eta_\theta = 0.01$ achieves the highest mean AUC and G-mean, whereas $\eta_\theta = 0.1$ gives the highest mean F1-score.

D.2 Statistical Test Results

Table 8: Statistical test results of the seven methods on three evaluation metrics.

Metric	FedConcat	UEFL	FedANAG	FedSWA	FedWGAN	FedKTL	FedDCOL	χ_F^2	F_F
AUC	5.000	6.143	2.000	4.286	3.143	6.143	**1.286**	33.551	23.826
F1-score	5.857	5.143	2.571	6.000	2.714	4.714	**1.000**	33.278	22.894
G-mean	5.857	4.857	2.714	6.429	2.429	4.714	**1.000**	35.938	35.571

Table 9: Pairwise comparisons of FedDCOL with baselines. FedDCOL performs best on all three metrics; only statistically significant pairwise wins are listed.

Metric	**Baselines significantly outperformed by FedDCOL**
AUC	FedConcat, UEFL, FedKTL
F1-score	FedConcat, UEFL, FedSWA, FedKTL
G-mean	FedConcat, UEFL, FedSWA, FedKTL

Table 8 provides the complete average ranks and test statistics for all seven methods, while Table 9 reports the pairwise comparisons for which the improvements of FedDCOL are statistically significant. On AUC, FedDCOL significantly outperforms FedConcat, UEFL, and FedKTL, whereas the differences from the remaining baselines do not reach the significance threshold. For F1-score and G-mean, the range of significant comparisons becomes broader, with FedDCOL significantly outperforming FedConcat, UEFL, FedSWA, and FedKTL. In contrast, the differences between FedDCOL and FedANAG or FedWGAN do not reach statistical significance for AUC, F1-score, or G-mean. Notably, FedWGAN is based on data generation, and its competitive performance suggests that this strategy can effectively mitigate imbalanced class distributions in FL.

D.3 Ablation Results

Table 10: Detailed ablation results across seven datasets.

Dataset	Metric	FedDCOL	w/o Class Overlap Modeling	w/o Structure Constraints	w/o Dynamic Continual Oversampling
BUPA	AUC	**0.755 ± 0.122**	0.741 ± 0.111	0.747 ± 0.110	0.748 ± 0.110
	F1-score	**0.659 ± 0.106**	0.627 ± 0.115	0.635 ± 0.127	0.647 ± 0.099
	G-mean	**0.705 ± 0.086**	0.675 ± 0.093	0.679 ± 0.101	0.690 ± 0.087
Spambase	AUC	**0.980 ± 0.006**	0.978 ± 0.006	0.978 ± 0.006	0.978 ± 0.006
	F1-score	**0.922 ± 0.017**	0.913 ± 0.017	0.916 ± 0.017	0.915 ± 0.016
	G-mean	**0.938 ± 0.015**	0.931 ± 0.014	0.933 ± 0.013	0.933 ± 0.013
Diabetes	AUC	**0.831 ± 0.046**	0.830 ± 0.046	0.821 ± 0.048	0.821 ± 0.048
	F1-score	0.641 ± 0.059	0.642 ± 0.070	0.641 ± 0.071	**0.642 ± 0.071**
	G-mean	**0.713 ± 0.050**	0.712 ± 0.059	0.710 ± 0.061	0.710 ± 0.060
German	AUC	**0.741 ± 0.059**	0.732 ± 0.054	0.741 ± 0.054	0.740 ± 0.055
	F1-score	**0.556 ± 0.074**	0.542 ± 0.080	0.541 ± 0.063	0.544 ± 0.074
	G-mean	**0.662 ± 0.066**	0.654 ± 0.064	0.656 ± 0.054	0.655 ± 0.061
ILPD	AUC	0.750 ± 0.072	**0.752 ± 0.074**	0.750 ± 0.074	0.750 ± 0.074
	F1-score	**0.560 ± 0.079**	0.554 ± 0.083	0.542 ± 0.086	0.559 ± 0.085
	G-mean	**0.683 ± 0.064**	0.676 ± 0.067	0.677 ± 0.070	0.683 ± 0.069
Parkinsons	AUC	**0.966 ± 0.047**	0.930 ± 0.061	0.935 ± 0.057	0.937 ± 0.058
	F1-score	**0.809 ± 0.166**	0.724 ± 0.176	0.718 ± 0.180	0.711 ± 0.182
	G-mean	**0.874 ± 0.141**	0.792 ± 0.151	0.799 ± 0.148	0.796 ± 0.148
Blood Transfusion	AUC	**0.749 ± 0.041**	0.743 ± 0.039	0.743 ± 0.038	0.742 ± 0.040
	F1-score	**0.482 ± 0.099**	0.444 ± 0.106	0.479 ± 0.111	0.478 ± 0.112
	G-mean	0.630 ± 0.098	0.626 ± 0.099	**0.636 ± 0.102**	0.629 ± 0.102