TABULAR-TO-GRAPH DIFFUSION: STRUCTURED GENERATIVE OVERSAMPLING FOR IMBALANCED TABULAR DATA

Anonymous authors
Paper under double-blind review

ABSTRACT

Class imbalance presents severe challenges for supervised classification on tabular data. Traditional oversampling techniques synthesize minority records via linear interpolation, which often crosses decision boundaries in non-linear distributions. Tabular diffusion models alleviate this by learning generative densities; however, flattening records into one-dimensional vectors and initiating reverse trajectories from standard Gaussian noise across the full feature space frequently leads to uncalibrated score estimates in low-density regions, resulting in mode drift toward majority classes and damaged column correlations. We propose Tabular-to-Graph Diffusion (TGD), an oversampling framework designed to preserve feature dependencies under acute sample scarcity. TGD structures each minority record alongside its nearest minority neighbors into a local feature graph, where nodes represent tabular columns and node attribute vectors contain feature values across ordered neighboring records. A two-layer residual graph convolutional network updates feature representations using empirical correlation coefficients and a categorical mask that isolates mutually exclusive dummy variables, while linear layers project across neighbor coordinates to capture local variations. Reverse diffusion begins from a localized perturbation of the empirical neighborhood and integrates along deterministic trajectory paths, preventing synthetic records from drifting outside populated minority regions. Evaluations across eight benchmark datasets show that TGD achieves competitive downstream classification performance across heuristic interpolators, deep generative models, and vector-based tabular diffusion methods, establishing a practical trade-off between decision boundary adherence and multi-feature correlation preservation.

1 INTRODUCTION

Tabular representations remain foundational in critical decision-making systems including clinical diagnostics, fraud detection, and industrial monitoring Kebede et al. (2026); Panisson et al. (2026); Chai et al. (2026). In these domains, severe class skewness often biases supervised classifiers toward majority patterns Borisov et al. (2022); Lyu et al. (2026). Synthesizing minority instances to rebalance label distributions is an established paradigm to enhance classifier generalization. Heuristic oversampling approaches such as SMOTE Chawla et al. (2002) synthesize records along line segments under the premise that convex combinations of minority neighbors remain within the underlying class distribution Ganev et al. (2026); Zhang et al. (2026b). On complex tabular distributions with non-linear constraints, however, linear interpolation frequently intersects decision boundaries, placing synthetic samples in majority regions and degrading decision thresholds Yu & Chen (2026); Dong et al. (2026).

Deep generative models have also been applied to minority oversampling by parameterizing conditional density distributions through adversarial or variational objectives Xu et al. (2019). When operating exclusively on small minority subsets, the scarcity of minority records conflicts with the large parameter capacity of deep architectures. Adversarial training often encounters mode collapse, while variational autoencoders frequently smooth heterogeneous feature interactions Chandani & Dhiman (2026); Shu et al. (2026); Zhang et al. (2026a). Consequently, deep oversamplers struggle to capture non-linear joint structures given limited minority sample support.

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.

Diffusion models have recently demonstrated strong generative capabilities across continuous domains Chen et al. (2026). Adapting diffusion to tabular oversampling under acute minority scarcity, however, presents distinct structural challenges. Existing tabular diffusion frameworks, such as TabDDPM and TabSyn, are originally formulated for unconstrained whole-table generation, flattening each record into a one-dimensional continuous vector and initiating reverse trajectories from standard Gaussian noise across the entire feature space Akritidis & Bozanis (2026); Javaid et al. (2026). When repurposed for targeted minority oversampling with scarce data, this formulation encounters two practical limitations. First, treating flattened rows as isolated continuous vectors discards column-wise physical dependencies, compelling fully connected layers to memorize pairwise feature relationships without explicit structural guidance. Second, unconditional reverse diffusion initialized from pure white noise requires reliable score estimates across the entire input space. In data-sparse regions with minimal minority support, score matching becomes uncalibrated, causing sampling trajectories to drift into majority clusters and generating out-of-bounds numerical artifacts.

To address these limitations, we propose Tabular-to-Graph Diffusion (TGD), an oversampling framework that structures local minority neighborhoods into transposed feature graphs. For each minority record, TGD extracts a local patch consisting of its nearest minority neighbors and transposes the matrix such that graph nodes correspond to tabular columns and node attribute vectors store the ordered values across neighbor instances. The denoising network coordinates two functional operations on this representation: graph convolutions aggregate column dependencies using empirical Pearson correlation thresholds and categorical exclusivity masks, while linear projection layers model local numerical variations across the neighbor coordinates. Furthermore, reverse sampling is initialized from small Gaussian perturbations around the empirical neighborhood and integrated via deterministic ordinary differential equations, confining synthetic records to observed minority regions.

The main contributions of this work are summarized as follows:

- We propose TGD, an oversampling method for imbalanced tabular learning that structures local minority neighborhoods into transposed feature graphs, explicitly preserving column dependencies that are discarded by standard vector-based diffusion.
- We design a denoising network that couples correlation-weighted graph convolutions across tabular features with linear projections across neighbor coordinates, combined with a truncated deterministic sampling scheme that avoids trajectory drift in low-density regions.
- We conduct evaluations across eight benchmark datasets, demonstrating that TGD improves average downstream classification ranks while mitigating both the boundary-crossing issues of heuristic interpolation and the severe correlation degradation observed in vector diffusion baselines.

2 Related Work

Tabular oversampling has historically depended on linear geometric heuristics, exemplified by SMOTE Chawla et al. (2002) and related adaptive variants Zhong et al. (2026); Zhu & Zhang (2026); Liu et al. (2026). Although computationally efficient, linear convex combinations frequently generate invalid samples when class boundaries are non-convex or disconnected. Deep generative models learn parametric distributions via conditional adversarial networks or variational autoencoders Xu et al. (2019); Pongmarutai et al. (2026); Ren et al. (2025). Under extreme class skewness, deep architectures trained on limited minority samples often exhibit mode collapse and posterior oversmoothing.

Diffusion probabilistic models for tabular generation employ continuous-discrete noise schedules Wang et al. (2026); Mueller et al. (2025) or latent score matching Lin et al. (2025); Wang et al. (2025). These methods typically treat tabular rows as flat continuous vectors, relying entirely on unconstrained parameter weights to capture multi-variable interactions. Moreover, standard tabular diffusion relies on global reverse trajectories initialized from white Gaussian noise. When sample density is severely constrained, unconstrained reverse sampling can produce substantial variance in estimated scores across low-density regions.

Graph neural networks on tabular data model column interactions by treating features as interconnected nodes. Relational autoencoders such as GOGGLE Liu et al. (2023b) parameterize feature

dependencies through learnable adjacency matrices. Under small-sample constraints, global latent bottlenecks can smooth discrete coordinates. Supervised feature-graph models Guo et al. (2021); Shi et al. (2026) capture attribute interactions but are not designed for generative data oversampling. In contrast to these approaches, TGD directly structures local observation patches into feature graphs, combining correlation-guided convolutions, categorical masks, and truncated deterministic sampling to preserve local feature dependencies.

3 METHODOLOGY

Figure 1 illustrates the overall framework of TGD. TGD extracts localized minority patches, constructs transposed feature graphs governed by empirical correlation thresholds and discrete category masks, and executes deterministic reverse sampling from bounded neighborhood perturbations. Standard moment alignment and boundary projections are applied post-hoc to enforce valid mixed-type formats.

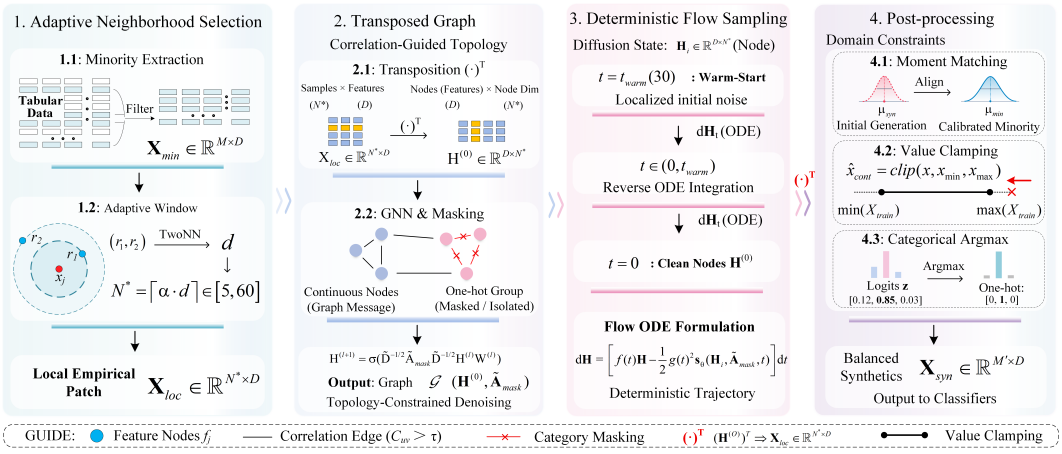


Figure 1: Overall architectural pipeline of Tabular-to-Graph Diffusion (TGD).

3.1 LOCAL PATCH EXTRACTION AND TRANSPOSED GRAPH REPRESENTATION

Let $\mathbf{X}_{\text{min}} \in \mathbb{R}^{M\times D}$ denote the standardized minority training partition, where M is the number of minority samples and D is the total feature dimension. Unlike image pixels that follow uniform spatial grids, tabular columns represent distinct variables with heterogeneous distributions. Directly constructing sample-level graphs when minority records are scarce leads to sparse adjacency structures that are sensitive to individual outliers.

TGD processes local groups of minority instances instead of isolated rows. For each minority anchor record, we identify its N^* nearest minority neighbors based on Euclidean distance, forming a local matrix $\mathbf{X}_{\text{loc}} \in \mathbb{R}^{N^*\times D}$. To provide a consistent geometric arrangement for subsequent network layers, rows in $\mathbf{X}_{\text{loc}}$ are sorted by ascending Euclidean distance to the anchor record. Transposing this ordered local matrix produces the graph representation:

$$\mathbf{H}^{(0)} = \mathbf{X}_{\text{loc}}^{\top} \in \mathbb{R}^{D\times N^*} \tag{1}$$

In this representation, the node set contains the D tabular features, and the node feature vector $\mathbf{h}_u^{(0)} \in \mathbb{R}^{N^*}$ contains the standardized values of feature u across the N^* ordered neighbors.

This formulation decouples feature-level interactions from sample-level variations. Graph convolutions operate over the D feature nodes to capture pairwise column correlations and mutual exclusivity between categorical categories. Along the N^* neighbor coordinates, linear projection layers model local continuous variations across neighboring records. Consequently, the network denoises the entire neighborhood matrix $\mathbf{H}_t$ jointly, retaining the joint distribution of local minority clusters.

3.2 Adaptive Neighborhood Scale Selection Via Intrinsic Dimensionality

Setting a static patch size across different tabular datasets either under-samples complex regions or merges disconnected distribution modes. We adaptively determine the neighborhood size N^* using the Two-Nearest-Neighbors (TwoNN) algorithm, which estimates the local intrinsic dimension d from the distribution of distance ratios between first and second nearest neighbors in $\mathbf{X}_{\text{min}}$.

To provide sufficient statistical support for estimating column correlations while preventing local patches from crossing low-density regions, the estimated intrinsic dimension d is scaled by a factor α to establish the dataset-level patch size N^*:

$$N^* = \text{clip}\left(\lceil \alpha \cdot d \rceil,\ 5,\ \min(60, M)\right) \tag{2}$$

The lower bound of 5 provides minimal degrees of freedom for sample correlation calculations, while the upper bound avoids aggregating heterogeneous modes or exceeding the available minority sample size M. Estimating d yields a unified patch window N^* that fixes the node attribute dimension for all graph instances within a given dataset.

3.3 Feature Dependency Graph and Category Masking

To capture linear dependencies within each local patch, empirical Pearson correlation coefficients are computed across the columns of $\mathbf{X}_{\text{loc}}$:

$$C_{uv} = \left|\text{Corr}\left(\mathbf{X}_{\text{loc},*,u},\ \mathbf{X}_{\text{loc},*,v}\right)\right| \tag{3}$$

A correlation threshold τ defines a sparse, correlation-weighted adjacency matrix:

$$A_{uv} = \begin{cases} C_{uv}, & \text{if } C_{uv} > \tau \text{ and } u \neq v \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

When the local patch size N^* is constrained, sample correlation estimates can exhibit higher variance. Setting the threshold $\tau = 0.8$ acts as a structural prior, pruning weak co-variations and preventing graph convolutions from degenerating into over-parameterized dense networks.

For mixed-type tables, one-hot encoded columns belonging to the same categorical attribute are mutually exclusive. Propagating messages across these indicator columns can blur discrete representations. We define a categorical mask matrix $\mathbf{M}_{\text{cat}} \in \{0,1\}^{D\times D}$, where $M_{\text{cat},uv} = 1$ when features u and v originate from the same categorical attribute, and 0 otherwise. The masked adjacency matrix is computed as:

$$\tilde{\mathbf{A}}_{\text{mask}} = \mathbf{A} \odot (\mathbf{1} - \mathbf{M}_{\text{cat}}) \tag{5}$$

The resulting feature graph $\mathcal{G}(\mathbf{H}^{(0)}, \tilde{\mathbf{A}}_{\text{mask}})$ confines message passing to prominent feature dependencies while isolating mutually exclusive categorical indicators.

3.4 Denoising Network and Diffusion Training

The denoising network accepts the perturbed node matrix $\mathbf{H}_t \in \mathbb{R}^{D\times N^*}$, the masked adjacency matrix $\tilde{\mathbf{A}}_{\text{mask}}$, and the diffusion step t to estimate the injected perturbation noise.

The diffusion step $t \in [0, T]$ is transformed via sinusoidal embeddings and mapped by a multi-layer perceptron into a temporal conditioning vector $\mathbf{e}_t \in \mathbb{R}^{d_h}$:

$$\mathbf{e}_t = \text{MLP}\left(\text{Concat}\left[\sin\left(t \cdot 10000^{-\frac{2k}{d_h}}\right),\ \cos\left(t \cdot 10000^{-\frac{2k}{d_h}}\right)\right]\right) \tag{6}$$

The node attribute matrix $\mathbf{H}_t$ is projected into hidden states through a linear layer that operates across the N^* ordered coordinates: $\mathbf{H}^{(1)} = \text{SiLU}(\mathbf{H}_t\mathbf{W}_{\text{in}} + \mathbf{e}_{t,\text{node}})$, where $\mathbf{W}_{\text{in}} \in \mathbb{R}^{N^*\times d_h}$ and $\mathbf{e}_{t,\text{node}}$ represents the broadcast temporal embedding. The backbone then processes the representations through two residual graph convolutional blocks with edge weights $\tilde{\mathbf{A}}_{\text{mask}}$:

$$\mathbf{Z}^{(l)} = \tilde{\mathbf{D}}^{-\frac{1}{2}}(\tilde{\mathbf{A}}_{\text{mask}} + \mathbf{I}_D)\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{H}^{(l)}\mathbf{W}_{\text{g}}^{(l)} \tag{7}$$

$$\mathbf{H}^{(l+1)} = \text{SiLU}\left(\mathbf{Z}^{(l)} + \mathbf{e}_{t,\text{node}}\right) + \mathbf{H}^{(l)}, \quad l \in \{1, 2\} \tag{8}$$

where $\tilde{\mathbf{D}}$ denotes the degree matrix of $\tilde{\mathbf{A}}_{\text{mask}} + \mathbf{I}_D$, and $\mathbf{W}_{\text{g}}^{(l)} \in \mathbb{R}^{d_h \times d_h}$. An output projection $\mathbf{W}_{\text{out}} \in \mathbb{R}^{d_h \times N^*}$ maps hidden states back to predict the original noise: $\boldsymbol{\epsilon}_\theta = \mathbf{H}^{(3)}\mathbf{W}_{\text{out}}$.

The forward diffusion process progressively injects Gaussian noise into the node attribute matrix:

$$\mathbf{H}_t = \sqrt{\bar{\alpha}_t}\mathbf{H}^{(0)} + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{D \times N^*}) \tag{9}$$

The denoising network optimizes the mean squared error over the Frobenius norm:

$$\mathcal{L}(\theta) = \mathbb{E}_{t,\mathbf{H}^{(0)},\boldsymbol{\epsilon}}\left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{H}_t, \tilde{\mathbf{A}}_{\text{mask}}, t)\|_F^2\right] \tag{10}$$

3.5 Truncated Deterministic Sampling and Post-Processing

Standard diffusion models initiate reverse sampling from isotropic Gaussian noise across the unconstrained feature space. In class-imbalanced settings where minority instances are scarce, score estimates lack empirical calibration in unpopulated regions, which can drive trajectories toward majority modes or generate unrealistic feature combinations. To prevent off-target drift, TGD initializes reverse trajectories from a truncated noise perturbation around the empirical neighborhood patch:

$$\mathbf{H}_{t_{\text{warm}}} = \sqrt{\bar{\alpha}_{t_{\text{warm}}}}\mathbf{H}^{(0)} + \sqrt{1-\bar{\alpha}_{t_{\text{warm}}}}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{D \times N^*}) \tag{11}$$

Setting $t_{\text{warm}} = 30$ over $T = 100$ total steps introduces controlled stochastic variation to produce diverse instances while keeping reverse trajectories grounded within the empirical neighborhood.

Reverse updates follow deterministic probability flow dynamics with dispersion parameter $\eta = 0.0$:

$$\hat{\mathbf{H}}_0 = \frac{\mathbf{H}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_\theta(\mathbf{H}_t, \tilde{\mathbf{A}}_{\text{mask}}, t)}{\sqrt{\bar{\alpha}_t}} \tag{12}$$

$$\mathbf{H}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\hat{\mathbf{H}}_0 + \sqrt{1-\bar{\alpha}_{t-1}}\boldsymbol{\epsilon}_\theta(\mathbf{H}_t, \tilde{\mathbf{A}}_{\text{mask}}, t) \tag{13}$$

This trajectory corresponds to numerical integration of the probability flow ordinary differential equation:

$$\mathrm{d}\mathbf{H} = \left[f(t)\mathbf{H} - \frac{1}{2}g(t)^2\mathbf{s}_\theta(\mathbf{H}_t, \tilde{\mathbf{A}}_{\text{mask}}, t)\right]\mathrm{d}t \tag{14}$$

Omitting stochastic diffusion terms eliminates trajectory variance during reverse integration.

Following ordinary differential equation integration, matrix transposition returns the generated patch to tabular format:

$$\mathbf{X}_{\text{loc}} = \mathbf{H}_0^\top \in \mathbb{R}^{N^* \times D} \tag{15}$$

Because continuous diffusion operates in real coordinate space, standard domain constraints are applied to align synthetic records with physical feature intervals and discrete encodings.

Continuous attributes are adjusted to match the empirical mean and standard deviation of the training minority partition $\mathbf{X}_{\text{min}}$:

$$\hat{x} = \frac{x - \mu_{\text{syn}}}{\sigma_{\text{syn}}} \cdot \sigma_{\text{min}} + \mu_{\text{min}} \tag{16}$$

where μ_{min} and σ_{min} represent empirical minority statistics. Following moment alignment, continuous coordinates are clipped to observed training extremes $x_{\min}$ and $x_{\max}$ to prevent arithmetic extrapolation.

For one-hot encoded categorical columns, continuous network outputs are treated as unnormalized logits, assigning the active indicator to the maximum score coordinate:

$$\hat{\mathbf{z}} = \text{Argmax}(\mathbf{z}) \implies [0, \ldots, 1, \ldots, 0]^\top \tag{17}$$

Binary indicators are thresholded at 0.5. These operations project continuous outputs back into valid mixed-type formats. Synthesized instances from all local patches are then consolidated into the augmented dataset $\mathbf{X}_{\text{syn}} \in \mathbb{R}^{M' \times D}$ for downstream classifier evaluation.

4 Experimental Setup

Experiments are performed across eight real-world benchmark datasets obtained from Kaggle and the CLIMB repository, covering clinical diagnostics, churn tracking, fraud detection, and industrial sensing. Dataset statistics are listed in Table 1, while specific access links and preprocessing details are provided in Appendix A.

The evaluated baselines span three methodological categories. First, heuristic interpolation is represented by SMOTE Chawla et al. (2002) and ADASYN He et al. (2008). Second, deep generative architectures comprise CTGAN, TVAE Xu et al. (2019), and the relational graph generator GOGGLE Liu et al. (2023b). Third, tabular diffusion models include TabDDPM Kotelnikov et al. (2023), TabSyn Zhang et al. (2024), and TabDiff Shi et al. (2025).

All datasets are partitioned into 70% stratified training and 30% untouched testing partitions. Across all generative baselines, synthetic instances are generated to balance training labels toward an 80% parity ratio relative to the majority class, following the benchmark protocols detailed in Appendix B. The augmented training sets are subsequently evaluated across three representative base classifiers: Decision Tree, K-Nearest Neighbors, and Multi-Layer Perceptron. Performance is measured using balanced accuracy, AUC, and macro F1 score, with evaluation scores computed as the unweighted mean across the three downstream classifiers. Implementation details and training schedules are provided in Appendix B.

Dataset	Sizes	Imbalance Ratio (0:1)	Features (D)
Ol8	2534	0.94 : 0.06	72
Phoneme	5404	0.71 : 0.29	5
Churn	5000	0.86 : 0.14	33
BCCP	10000	0.80 : 0.20	11
SMP	10000	0.90 : 0.10	6
NamG	11183	0.98 : 0.02	6
OCI	12330	0.85 : 0.15	17
SD	20000	0.91 : 0.09	5

Table 1: Benchmark datasets.

Table 2: Evaluation results across eight benchmark datasets. Values denote mean$_{\text{std}}$ across three trials; parenthetical entries report relative metric rankings.

Task	Metric (↑)	Original	SMOTE	ADASYN	CTGAN	TVAE	TabDDPM	GOGGLE	TabSyn	TabDiff	TGD (Ours)
Ol8	Bal-Acc	$0.603_{0.000}$ (9)	$0.728_{0.000}$ (2)	$0.700_{0.000}$ (4)	$0.605_{0.012}$ (8)	$0.654_{0.034}$ (6)	$0.628_{0.011}$ (7)	$0.687_{0.016}$ (5)	$0.701_{0.010}$ (3)	$0.603_{0.014}$ (10)	$\mathbf{0.742_{0.006}}$ (1)
	AUC	$0.775_{0.000}$ (4)	$0.798_{0.000}$ (2)	$0.777_{0.000}$ (3)	$0.762_{0.006}$ (8)	$0.769_{0.018}$ (6)	$0.742_{0.009}$ (10)	$0.765_{0.007}$ (7)	$0.772_{0.011}$ (5)	$0.756_{0.010}$ (9)	$\mathbf{0.809_{0.004}}$ (1)
	F1	$0.298_{0.000}$ (7)	$0.368_{0.000}$ (2)	$0.326_{0.000}$ (6)	$0.288_{0.028}$ (8)	$0.331_{0.018}$ (4)	$0.271_{0.013}$ (10)	$0.343_{0.009}$ (3)	$0.327_{0.014}$ (5)	$0.275_{0.020}$ (9)	$\mathbf{0.385_{0.011}}$ (1)
Phoneme	Bal-Acc	$0.825_{0.000}$ (7)	$0.838_{0.000}$ (4)	$0.842_{0.000}$ (2)	$0.819_{0.009}$ (9)	$0.818_{0.002}$ (10)	$0.836_{0.002}$ (5)	$0.834_{0.003}$ (6)	$0.841_{0.001}$ (3)	$0.819_{0.007}$ (8)	$\mathbf{0.843_{0.001}}$ (1)
	AUC	$0.892_{0.000}$ (4)	$0.891_{0.000}$ (5)	$0.893_{0.000}$ (3)	$0.875_{0.006}$ (8)	$0.872_{0.002}$ (10)	$0.886_{0.002}$ (6)	$0.885_{0.002}$ (7)	$0.893_{0.001}$ (2)	$0.872_{0.003}$ (9)	$\mathbf{0.896_{0.002}}$ (1)
	F1	$0.754_{0.000}$ (6)	$0.764_{0.000}$ (2)	$0.756_{0.000}$ (5)	$0.737_{0.011}$ (8)	$0.733_{0.003}$ (10)	$0.756_{0.003}$ (4)	$0.753_{0.001}$ (7)	$\mathbf{0.765_{0.004}}$ (1)	$0.734_{0.007}$ (9)	$0.763_{0.003}$ (3)
Churn	Bal-Acc	$0.769_{0.000}$ (8)	$0.827_{0.000}$ (2)	$0.817_{0.000}$ (4)	$0.762_{0.003}$ (9)	$0.821_{0.000}$ (3)	$0.806_{0.003}$ (5)	$0.787_{0.006}$ (7)	$0.795_{0.007}$ (6)	$0.760_{0.002}$ (10)	$\mathbf{0.832_{0.002}}$ (1)
	AUC	$\mathbf{0.869_{0.000}}$ (1)	$0.863_{0.000}$ (4)	$0.854_{0.000}$ (9)	$0.859_{0.006}$ (8)	$0.861_{0.000}$ (7)	$0.852_{0.004}$ (10)	$0.867_{0.004}$ (3)	$0.861_{0.008}$ (5)	$0.861_{0.002}$ (6)	$0.867_{0.004}$ (2)
	F1	$0.634_{0.000}$ (6)	$0.646_{0.000}$ (5)	$0.616_{0.000}$ (9)	$0.621_{0.006}$ (7)	$0.661_{0.000}$ (1)	$\mathbf{0.597_{0.008}}$ (10)	$0.654_{0.007}$ (3)	$0.652_{0.020}$ (4)	$0.618_{0.005}$ (8)	$0.659_{0.008}$ (2)
BCCP	Bal-Acc	$0.683_{0.000}$ (10)	$0.715_{0.000}$ (4)	$0.714_{0.000}$ (5)	$0.695_{0.018}$ (8)	$0.700_{0.002}$ (7)	$0.721_{0.004}$ (2)	$0.710_{0.006}$ (6)	$0.717_{0.002}$ (3)	$0.694_{0.002}$ (9)	$\mathbf{0.725_{0.001}}$ (1)
	AUC	$0.772_{0.000}$ (6)	$0.768_{0.000}$ (7)	$0.764_{0.000}$ (9)	$0.740_{0.021}$ (10)	$0.778_{0.002}$ (3)	$0.777_{0.003}$ (5)	$0.780_{0.003}$ (2)	$\mathbf{0.782_{0.003}}$ (1)	$0.765_{0.001}$ (8)	$0.777_{0.004}$ (4)
	F1	$0.511_{0.000}$ (9)	$0.532_{0.000}$ (6)	$0.525_{0.000}$ (7)	$0.498_{0.019}$ (10)	$0.535_{0.003}$ (5)	$0.545_{0.006}$ (3)	$0.545_{0.008}$ (4)	$\mathbf{0.553_{0.003}}$ (1)	$0.521_{0.003}$ (8)	$0.547_{0.004}$ (2)
SMP	Bal-Acc	$0.943_{0.000}$ (8)	$0.966_{0.000}$ (3)	$0.965_{0.000}$ (4)	$0.899_{0.006}$ (9)	$0.952_{0.001}$ (7)	$0.963_{0.003}$ (5)	$0.952_{0.004}$ (6)	$0.968_{0.001}$ (2)	$0.887_{0.010}$ (10)	$\mathbf{0.972_{0.001}}$ (1)
	AUC	$0.988_{0.000}$ (5)	$0.989_{0.000}$ (4)	$0.988_{0.000}$ (7)	$0.957_{0.008}$ (9)	$0.988_{0.002}$ (6)	$0.989_{0.001}$ (3)	$0.983_{0.002}$ (8)	$\mathbf{0.991_{0.001}}$ (1)	$0.945_{0.008}$ (10)	$0.991_{0.000}$ (2)
	F1	$\mathbf{0.904_{0.000}}$ (1)	$0.862_{0.000}$ (3)	$0.862_{0.000}$ (3)	$0.635_{0.010}$ (9)	$0.886_{0.008}$ (2)	$0.791_{0.025}$ (7)	$0.754_{0.014}$ (8)	$0.857_{0.016}$ (5)	$0.618_{0.012}$ (10)	$0.851_{0.004}$ (6)
NamG	Bal-Acc	$0.763_{0.000}$ (10)	$0.902_{0.000}$ (2)	$0.884_{0.000}$ (4)	$0.790_{0.005}$ (9)	$0.806_{0.033}$ (7)	$0.887_{0.018}$ (3)	$0.830_{0.009}$ (5)	$0.829_{0.009}$ (6)	$0.804_{0.008}$ (8)	$\mathbf{0.920_{0.004}}$ (1)
	AUC	$0.872_{0.000}$ (9)	$0.916_{0.000}$ (3)	$0.902_{0.000}$ (4)	$0.836_{0.008}$ (10)	$0.887_{0.011}$ (5)	$0.918_{0.008}$ (2)	$0.880_{0.009}$ (6)	$0.874_{0.016}$ (8)	$0.875_{0.008}$ (7)	$\mathbf{0.936_{0.003}}$ (1)
	F1	$0.616_{0.000}$ (2)	$0.548_{0.000}$ (3)	$0.492_{0.000}$ (6)	$0.279_{0.071}$ (10)	$\mathbf{0.647_{0.022}}$ (1)	$0.478_{0.052}$ (9)	$0.480_{0.014}$ (8)	$0.513_{0.027}$ (5)	$0.514_{0.025}$ (4)	$0.482_{0.022}$ (7)
OCI	Bal-Acc	$0.706_{0.000}$ (10)	$0.759_{0.000}$ (2)	$0.756_{0.000}$ (4)	$0.709_{0.009}$ (9)	$0.718_{0.003}$ (7)	$0.752_{0.005}$ (5)	$0.732_{0.007}$ (6)	$0.758_{0.001}$ (3)	$0.713_{0.008}$ (8)	$\mathbf{0.762_{0.002}}$ (1)
	AUC	$0.795_{0.000}$ (8)	$0.805_{0.000}$ (4)	$0.801_{0.000}$ (5)	$0.771_{0.012}$ (10)	$0.798_{0.002}$ (7)	$0.806_{0.007}$ (3)	$0.801_{0.003}$ (6)	$\mathbf{0.813_{0.001}}$ (1)	$0.795_{0.003}$ (9)	$0.810_{0.001}$ (2)
	F1	$0.524_{0.000}$ (8)	$0.552_{0.000}$ (3)	$0.540_{0.000}$ (6)	$0.498_{0.012}$ (10)	$0.532_{0.003}$ (7)	$0.551_{0.005}$ (4)	$0.548_{0.006}$ (5)	$\mathbf{0.564_{0.006}}$ (1)	$0.524_{0.006}$ (9)	$0.561_{0.005}$ (2)
SD	Bal-Acc	$0.833_{0.000}$ (10)	$\mathbf{0.887_{0.000}}$ (1)	$0.885_{0.000}$ (3)	$0.836_{0.007}$ (8)	$0.834_{0.004}$ (9)	$0.868_{0.004}$ (4)	$0.846_{0.008}$ (7)	$0.862_{0.006}$ (5)	$0.847_{0.007}$ (6)	$0.886_{0.000}$ (2)
	AUC	$0.901_{0.000}$ (9)	$0.914_{0.000}$ (2)	$\mathbf{0.915_{0.000}}$ (1)	$0.887_{0.005}$ (10)	$0.904_{0.001}$ (6)	$0.905_{0.001}$ (4)	$0.902_{0.001}$ (8)	$0.905_{0.001}$ (5)	$0.903_{0.003}$ (7)	$0.912_{0.001}$ (3)
	F1	$0.713_{0.000}$ (7)	$0.707_{0.000}$ (8)	$0.639_{0.000}$ (9)	$0.540_{0.015}$ (10)	$0.715_{0.004}$ (6)	$0.729_{0.004}$ (2)	$0.722_{0.005}$ (5)	$0.729_{0.004}$ (3)	$0.725_{0.006}$ (4)	$\mathbf{0.733_{0.004}}$ (1)
Overall Mean Rank		6.833	3.458	5.083	8.917	5.917	5.333	5.750	3.500	8.125	**2.042**

5 Results and Analysis

5.1 Main Results and Statistical Significance

Table 2 reports downstream classification performance across eight benchmark datasets over three independent trials. TGD achieves the lowest overall average rank of 2.042 across all 24 evaluation tasks. On the primary metric of balanced accuracy, TGD secures an average rank of 1.125, ranking first across seven datasets and second on SD. In particular, TGD delivers substantial improvements on both high-dimensional tables such as Ol8 and acutely skewed distributions such as NamG. Conversely, unconstrained vector models including CTGAN, TabDiff, and TabDDPM suffer severe performance degradation due to mode collapse in adversarial optimization and unconstrained score drift in diffusion processes under small-sample regimes. On acutely skewed tables such as NamG (imbalance ratio 42:1), while TGD achieves leading balanced accuracy and AUC, the macro F1 score of all reliable methods is mathematically suppressed because the overwhelming majority

prior ($\pi_{\text{maj}} \approx 0.98$) heavily penalizes minor boundary explorations. In contrast, TVAE attains an elevated F1 score by collapsing predictions to an overly conservative core of minority instances, compromising broader distribution coverage as detailed in Appendix C.

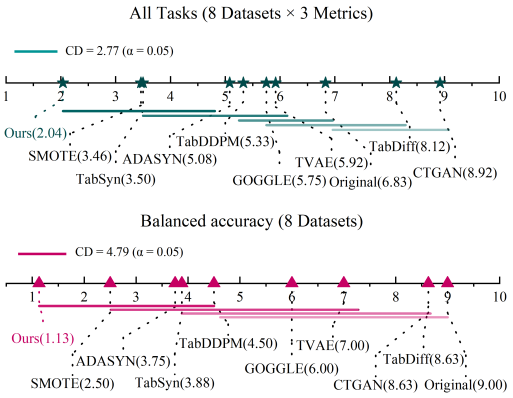


Figure 2: Nemenyi post-hoc critical difference diagram ($\alpha = 0.05$). Horizontal bold lines link cliques of models without statistically significant differences. Lower ranks indicate superior overall performance.

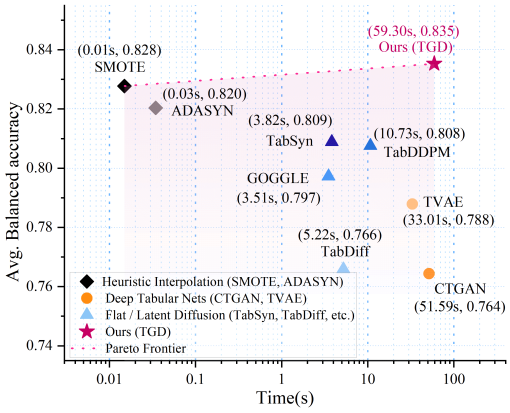


Figure 3: Pareto trade-off between Bal-Acc and average execution time per dataset on a logarithmic scale. The dashed line denotes the empirical efficiency frontier.

Non-parametric statistical tests confirm the significance of these empirical advantages. Across all 24 evaluation tasks, the Friedman test yields a test statistic of 105.90 at $p < 10^{-15}$, decisively rejecting the null hypothesis of equivalent performance. In the Nemenyi post-hoc critical difference analysis shown in Figure 2, under a critical difference threshold of 2.77, TGD shows statistically significant improvements over the baseline models whose mean ranks differ by more than this threshold.

Figure 3 illustrates the empirical Pareto trade-off between balanced accuracy and runtime on a logarithmic scale. The empirical Pareto frontier is formed exclusively by SMOTE, which maximizes execution efficiency, and TGD, which delivers the highest average balanced accuracy of 0.835. Deep parametric baselines such as CTGAN and TVAE incur computational overhead between 33 and 52 seconds while yielding inferior accuracy due to optimization instability on scarce minority observations. Heuristic SMOTE remains highly attractive for time-critical workflows due to its millisecond execution; however, TGD provides a viable alternative for scenarios where class boundary fidelity and mixed-type distribution quality are prioritized over training throughput. Similarly, standard residual diffusion TabDDPM requires an average of 10.73 seconds per dataset but remains strictly Pareto-dominated by SMOTE due to unconstrained score dispersion. By focusing computation on localized feature-graph updates, TGD achieves superior downstream accuracy with rapid convergence stabilizing between epochs 40 and 60, as detailed in the convergence profiles in Appendix D and complexity analysis in Appendix E.

5.2 Feature Covariance and Distribution Boundary Analysis

To evaluate the geometric and statistical fidelity of synthesized data, we examine low-dimensional distribution embeddings alongside inter-feature covariance residual structures.

Figure 4 illustrates two-dimensional UMAP projections for Ol8, OCI, and NamG. On the 72-dimensional Ol8 benchmark, empirical majority and minority records are densely clustered within a narrow region along the right boundary. Standard generative baselines and diffusion models generate synthetic samples that scatter outside the empirical minority cluster and overlap with the majority class distribution. In contrast, TGD confines synthetic instances strictly within the genuine minority distribution. On the multi-category OCI dataset where minority instances form an isolated cluster, baseline models over-expand across majority decision boundaries, and TVAE generates isolated off-target outliers. TGD keeps synthesized records within the empirical minority region through localized neighborhood diffusion, while value clamping and categorical argmax enforce valid coordinate intervals and discrete formats. On NamG, characterized by an extreme imbalance ratio

of 42:1, TGD reconstructs samples along the narrow curved minority distribution, confirming that TwoNN adaptive neighborhoods prevent out-of-distribution extrapolation in highly skewed geometries. Geometric alignments across the remaining five benchmarks are provided in Appendix F.1.

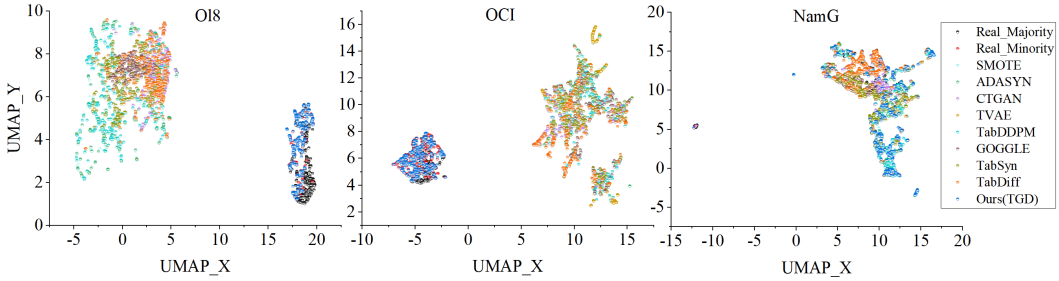


Figure 4: Two-dimensional UMAP embeddings. Black and red markers indicate true majority and minority samples, respectively; colored markers denote synthetic samples.

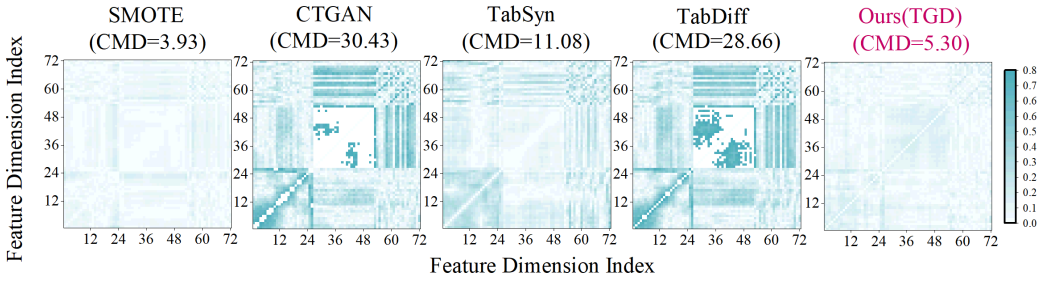


Figure 5: Heatmaps of absolute Pearson correlation residual matrices relative to true minority records on Ol8. Header labels report Frobenius covariance matrix distances (CMD).

Figure 5 presents heatmaps of absolute Pearson correlation residuals on Ol8 to evaluate pairwise statistical dependencies. Unconstrained generative baselines display severe correlation degradation, where CTGAN and TabDiff produce widespread cross-attribute grid distortions, resulting in elevated Frobenius covariance matrix distances exceeding 28. This degradation indicates that processing records as independent one-dimensional vectors fails to maintain inter-feature correlations when training samples are limited.

Comparing heuristic interpolation with TGD highlights distinct trade-offs between linear statistical metrics and non-linear boundary fidelity. While SMOTE achieves a lower covariance distance of 3.93 on Ol8 due to the inherent property of straight-line interpolation preserving pairwise moments, the low-dimensional embeddings in Figure 4 show that it generates synthetic instances directly within majority class clusters. TGD balances empirical correlation preservation with boundary precision, maintaining a low covariance distance of 5.30, whereas unconstrained diffusion models such as TabDiff degrade to 28.66. To verify that truncated reverse sampling does not merely replicate training coordinates, we evaluate the Euclidean Distance to Closest Record (DCR) relative to the minority training partition against the empirical test-to-train distance baseline. Across all benchmarks, synthetic instances exhibit positive minimum distances without coordinate duplication. Furthermore, the median DCR of generated records closely aligns with the empirical test baseline across both dense and sparse geometries (e.g., 0.329 vs. 0.245 on Phoneme, and 0.489 vs. 0.392 on NamG), confirming that TGD generates novel observations across minority clusters rather than memorizing local training points. Extended covariance heatmaps on NamG and Churn are reported in Appendix F.2, and complete DCR distributions are detailed in Appendix G.

5.3 Ablation Study

To isolate the contribution of each component within TGD, we evaluate six ablation variants across all benchmark tasks. The variant without graph message passing substitutes graph convolutions with independent linear transformations. The variant without TwoNN imposes a static neighborhood size

of 15. The variant without categorical masking removes the mask matrix to permit intra-group propagation. The variant without post-processing constraints omits boundary clamping and categorical argmax. The variant without deterministic flow substitutes ODE integration with stochastic Brownian sampling under noise coefficient 1.0. Finally, the variant without moment alignment omits empirical moment matching. Figure 6 presents overall mean ranks and 95% confidence intervals across all 24 evaluation tasks, with quantitative metrics detailed in Appendix H.

As shown in Figure 6, omitting post-processing constraints or reverting to stochastic Brownian sampling causes substantial performance degradation by generating out-of-bounds coordinates and boundary variance fluctuations.

When evaluating the variant without graph message passing, downstream balanced accuracy remains relatively stable on low-dimensional datasets. This behavior is directly attributable to the transposed patch formulation: even without graph convolutions, the input projection layers continue to operate along the node attribute axis, smoothing continuous variations across the ordered neighboring records. Consequently, the generated records stay confined to the minority neighborhood, preserving acceptable classification accuracy.

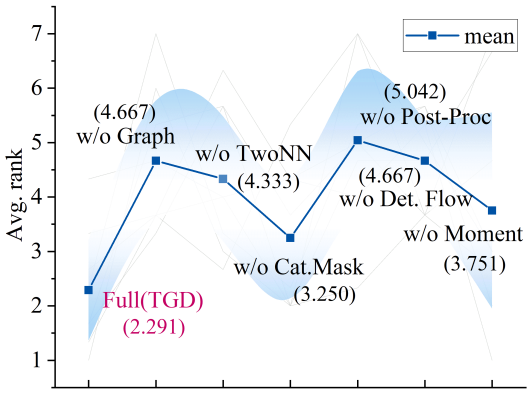


Figure 6: Mean rank trajectories and 95% confidence intervals across ablation variants over all evaluation tasks. Lower ranks indicate superior overall performance.

However, classification metrics alone do not reflect multi-variable correlation structures. As demonstrated by the covariance distance evaluations on high-dimensional benchmarks such as Ol8, removing graph convolutions causes a severe degradation in multi-attribute joint dependencies. The drop in overall ranking for the variant without graph message passing confirms that graph message passing is specifically required to preserve column correlations in higher-dimensional feature spaces, complementing the neighbor smoothing provided by the attribute axis.

Furthermore, fixed neighborhood scales degrade performance across distinct table distributions, as confirmed by the ranking drop of the variant without TwoNN. When N^* is fixed too small, local patches lack sufficient degrees of freedom to evaluate column covariance; conversely, setting N^* excessively large forces local approximations across complex decision thresholds, incorporating majority records into the empirical patch. TwoNN eliminates this trade-off by adapting the window size to local intrinsic dimensionality. Additionally, removing the categorical mask leads to intra-group message propagation across mutually exclusive dummy indicators, blurring discrete state boundaries and compromising valid mixed-type synthesis.

6 Conclusion

In this paper, we presented Tabular-to-Graph Diffusion (TGD), a generative oversampling framework designed for imbalanced tabular learning. By organizing localized minority observation patches into transposed feature graphs, TGD establishes a joint representation of features and local sample neighborhoods, avoiding the loss of column dependencies common in flat vector diffusion. Within this structure, the denoising network combines correlation-weighted, category-masked graph convolutions across features with linear projections across neighbor coordinates. Reverse trajectories are initialized from small perturbations around empirical neighborhoods and integrated deterministically, mitigating off-target drift in low-density regions. Standard post-processing steps further enforce empirical moment matching, coordinate interval clamping, and categorical discretization. Benchmark evaluations on eight datasets indicate that TGD attains favorable downstream classification performance across diverse data geometries, offering a balanced alternative that avoids both the decision-boundary violations of linear oversampling and the correlation collapse of unconstrained vector diffusion. Future work will investigate adaptive correlation thresholding strategies and extend transposed local diffusion to extreme multi-class long-tailed settings.

AI USE STATEMENT

In this work, we used generative AI tools for language polishing.

REPRODUCIBILITY STATEMENT

All code implementation details involved in this paper are available at https://anonymous.4open.science/r/TabulartoGraph-CED3, and their handling methods are correspondingly described in the main text and appendix.

REFERENCES

Leonidas Akritidis and Panayiotis Bozanis. A conditional gan for tabular data generation with probabilistic sampling of latent subspaces. *IEEE Transactions on Knowledge and Data Engineering*, 2026.

Vadim Borisov, Tobias Leemann, Kathrin Seßler, Johannes Haug, Martin Pawelczyk, and Gjergji Kasneci. Deep neural networks and tabular data: A survey. *IEEE transactions on neural networks and learning systems*, 35(6):7499–7519, 2022.

Guangzhao Chai, Youxi Wu, and Jing Liu. Cdaf: A contrastive distribution alignment framework for generative augmentation in imbalanced iiot intrusion detection. *IEEE Internet of Things Journal*, 2026.

Gaurav Bhushan Chandani and Gaurav Dhiman. Efficient generative adversarial networks via parity-guided sliding window kde. *IEEE Transactions on Artificial Intelligence*, 2026.

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.

Zhennan Chen, Junwei Zhu, Xu Chen, Jiangning Zhang, Xiaobin Hu, Hanzhen Zhao, Chengjie Wang, Jian Yang, and Ying Tai. Dip: Taming diffusion models in pixel space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 36136–36146, 2026.

Jingnan Dong, Boran Zhang, Shigen Shen, Jun Liu, Guangxia Xu, and Zhiquan Liu. Adaptive dbscan-enhanced oversampling for imbalanced network intrusion detection: A boundary-aware approach. *IEEE Transactions on Dependable and Secure Computing*, 2026.

Georgi Ganev, MohammadReza Nazari, Rees Davison, Amirhassan Dizche, Xinmin Wu, Ralph Abbey, Jorge Silva, and Emiliano De Cristofaro. Smote and mirrors: exposing privacy leakage from synthetic minority oversampling. In *International Conference on Learning Representations*, volume 2026, pp. 142401–142417, 2026.

Xiawei Guo, Yuhan Quan, Huan Zhao, Quanming Yao, Yong Li, and Weiwei Tu. Tabgnn: Multiplex graph neural network for tabular data prediction. *arXiv preprint arXiv:2108.09127*, 2021.

Haibo He, Yang Bai, Edwardo A Garcia, and Shutao Li. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*, pp. 1322–1328. Ieee, 2008.

Kumail Javaid, Meng Yuan, and Yukang Cui. Madiff-t: Multi-agent evolutionary calibration of diffusion models for privacy-audited tabular data generation. *Knowledge-Based Systems*, pp. 116515, 2026.

Yared Bitew Kebede, Henok Desalegn Shikur, and Ming-Der Yang. Enhancing class-imbalanced soil classification using an explainable tabular network and synthetic minority over-sampling framework. *Engineering Applications of Artificial Intelligence*, 183:116235, 2026.

Akim Kotelnikov, Dmitry Baranchuk, Ivan Rubachev, and Artem Babenko. Tabddpm: Modelling tabular data with diffusion models. In *International conference on machine learning*, pp. 17564–17579. PMLR, 2023.

Xiaofeng Lin, Chenheng Xu, Matthew Yang, and Guang Cheng. Ctsyn: A foundation model for cross tabular data generation. In *International Conference on Learning Representations*, volume 2025, pp. 47938–47959, 2025.

Guan-Horng Liu, Arash Vahdat, De-An Huang, Evangelos A Theodorou, Weili Nie, and Anima Anandkumar. I^2sb: Image-to-image schrödinger bridge. *arXiv preprint arXiv:2302.05872*, 2023a.

Tennison Liu, Zhaozhi Qian, Jeroen Berrevoets, and Mihaela van der Schaar. Goggle: Generative modelling for tabular data by learning relational structure. In *The Eleventh International Conference on Learning Representations*, 2023b.

Xiaoying Liu, Silei Xie, Jiangtao Yang, Shengnan Wu, Mengmeng Du, Yuan Ji, Hong Zheng, Huiping An, Wei Zhou, Huijing Lv, et al. Khoi-smote: An efficient oversampling technique based on k-means clustering and h-outlyingness index for imbalanced medical data. *Neurocomputing*, pp. 132643, 2026.

Jingyang Lyu, Kangjie Zhou, and Yiqiao Zhong. A statistical theory of overfitting for imbalanced classification. In *International Conference on Learning Representations*, volume 2026, pp. 154902–155004, 2026.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.

Markus Mueller, Kathrin Gruber, and Dennis Fok. Continuous diffusion for mixed-type tabular data. In *International Conference on Learning Representations*, volume 2025, pp. 39186–39225, 2025.

André Panisson, Adriano Fazzone, Carlo Abrate, Michele Starnini, and Walter Allasia. Safety-constrained classification under extreme imbalance: Application to payment filtering. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 7842–7853, 2026.

Sopanat Pongmarutai, Supanut Chaidee, Chanchanok Aramrat, Chaisiri Angkurawaranon, and Papangkorn Inkeaw. Chds: Geometry-based synthetic data generation for tabular data. *Knowledge-Based Systems*, pp. 116745, 2026.

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, 2025.

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*, volume 2025, pp. 37353–37375, 2025.

Ruxue Shi, Yili Wang, Mengnan Du, Hangting Ye, Yi Chang, and Xin Wang. Tarot: Task-adaptive refinement of llm-prior graphs for few-shot tabular learning. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 4229–4239, 2026.

Junwei Shu, Hantang Liu, Dawei Miao, Wenzheng Song, Mingyang Yuan, Wenjie Liu, Changgu Chen, Yang Li, and Changbo Wang. Vfxdb: A visual effects volume dataset and benchmark for vdb-native generative modeling. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pp. 1–11, 2026.

Brandon Trabucco, Kyle Doherty, Max Gurinas, and Ruslan Salakhutdinov. Effective data augmentation with diffusion models. In *International Conference on Learning Representations*, volume 2024, pp. 14590–14612, 2024.

Alex X Wang, Colin R Simpson, and Binh P Nguyen. Blending is all you need: Data-centric ensemble synthetic data. *Information Sciences*, 691:121610, 2025.

Yanyan Wang, Chenjie Wei, and Conghe Zhang. A generative deep learning framework for illegal ivory provenance via diffusion-based isotopic data augmentation. *Microchemical Journal*, pp. 118344, 2026.

Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Modeling tabular data using conditional gan. *Advances in neural information processing systems*, 32, 2019.

Zebin You, Shen Nie, Xiaolu Zhang, JUN ZHOU, Zhiwu Lu, Ji-Rong Wen, and Chongxuan Li. Llada-v: Large language diffusion models with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10093–10105, 2026.

Fuxiang Yu and Shengjian Chen. Risk-aware contrastive fraud detection network: Enhancing transaction risk identification in imbalanced systems. *Physica A: Statistical Mechanics and its Applications*, pp. 131970, 2026.

Gang Zhang, Ruinan Li, Yanghao Zhou, Jinyan Liu, Xiangyun Tang, and Longbing Cao. Patetdpm: Differentially private tabular data synthesis on long-tailed distribution. *IEEE Transactions on Knowledge and Data Engineering*, 2026a.

Hengrui Zhang, Jiani Zhang, Zhengyuan Shen, Balasubramaniam Srinivasan, Xiao Qin, Christos Faloutsos, Huzefa Rangwala, and George Karypis. Mixed-type tabular data synthesis with score-based diffusion in latent space. In *International Conference on Learning Representations*, volume 2024, pp. 52829–52857, 2024.

Yang Zhang, Jie Shi, Yanfang Liu, and Hong Zhao. Dafs: A distribution-aware hierarchical feature selection method for long-tailed classification. *Pattern Recognition*, pp. 113218, 2026b.

Kaiwen Zheng, Huayu Chen, Haotian Ye, Haoxiang Wang, Qinsheng Zhang, Kai Jiang, Hang Su, Stefano Ermon, Jun Zhu, and Ming-Yu Liu. Diffusionnft: Online diffusion reinforcement with forward process. In *International Conference on Learning Representations*, volume 2026, pp. 134129–134150, 2026.

Ying Zhong, Zhiliang Wang, Xingang Shi, Jiahai Yang, and Keqin Li. A unified oversampling-based imbalance processing framework for network traffic anomaly detection. *IEEE Transactions on Dependable and Secure Computing*, 2026.

Yaoqian Zhu and Ridong Zhang. New attention ensemble-based fault diagnosis for industrial processes under imbalanced finite data. *IEEE Transactions on Instrumentation and Measurement*, 75:1–8, 2026.

A Benchmark Datasets and Preprocessing Details

A.1 Data Sources and Reproducibility

The eight real-world benchmark datasets analyzed in this study were obtained from Kaggle and the OpenML/CLIMB benchmark suite. Full access links, raw data archives, and reproducible preprocessing scripts are publicly available at the link.

Following standard tabular benchmarking protocols, all continuous features across all evaluated datasets are standardized to zero mean and unit variance based on training set statistics, which is applied consistently across all oversampling baselines and downstream classifiers.

A.2 Feature Preprocessing Pipeline

To evaluate generative fidelity under authentic empirical data distributions without introducing artificial scaling distortions, all datasets undergo a unified preprocessing pipeline prior to model training.

String Sanitization and Type Disambiguation In raw tabular records, non-standard missing value indicators such as question marks, blank strings, or literal null tokens frequently contaminate continuous variables, causing numerical features to be parsed as string or object datatypes. We implement regular-expression matching to standardize corrupted entries into uniform missing values. To resolve datatype misclassification, columns where all observed non-missing entries can be losslessly parsed into floating-point numbers are restored to continuous numeric features; remaining columns are designated as discrete categorical variables.

Stratified Missing Value Imputation Missing values are addressed via a type-stratified imputation strategy:

- **Continuous Features**: Missing entries are imputed using the column-wise median computed over observed training values. If a continuous column contains no valid observations across all records, unobserved entries default to zero.
- **Categorical Features**: Missing values are imputed using the empirical mode of the column. In cases where no mode can be resolved, a dedicated missing token is assigned.

Dataset-Specific Transformations

- **OCI and BCCP**: String-valued categorical attributes are transformed into integer categorical indices via label encoding.
- **SMP**: Temporal timestamp features are converted into continuous elapsed numerical values to preserve differentiability during diffusion sampling.

Label Binarization and Class Alignment Instances with missing class labels are discarded. For binary classification standardization, class frequencies are computed on valid instances: the minority category is mapped to $y = 1$, and the majority category is mapped to $y = 0$.

Discrete Representation and Tensor Construction For benchmark datasets containing nominal categorical variables, discrete attributes are expanded into full binary one-hot indicator vectors without dropping reference categories. Retaining all coordinate dimensions is necessary for establishing the category mask matrix $\mathbf{M}_{\text{cat}}$ in the transposed feature graph as detailed in Section 3.3 and executing categorical argmax discretization during post-processing as detailed in Section 3.5. Conversely, for benchmarks utilizing integer-encoded categorical attributes such as OCI and BCCP, the discrete feature coordinates are directly maintained. The complete design matrix is subsequently cast into floating-point tensors to support graph neural network operations.

B Implementation Details and Hyperparameter Specifications

Experiments were executed on a workstation running Windows 11 equipped with an Intel Core i7-11800H processor, an NVIDIA RTX 3060 Laptop GPU, and 32 GB RAM.

To ensure complete reproducibility and fair evaluation, all baseline models and downstream discriminators are executed in strict accordance with our verified benchmark pipeline. In alignment with Section 4, heuristic interpolators such as SMOTE and ADASYN, alongside unconditioned diffusion frameworks including TabDDPM, TabSyn, TabDiff, and TGD, access only minority-class training records, strictly confining density modeling to empirical minority observations. Conversely, conditional deep generative baselines including CTGAN and TVAE follow their original formulations trained on full training partitions to perform class-conditional synthesis. Table 3 outlines the exact architectural configurations, training schedules, and hyperparameter specifications utilized across all evaluated models.

B.1 Hyperparameter Design Rationales and Generalization Protocol

To assess out-of-the-box generalization and avoid task-specific hyperparameter tuning, architectural and sampling parameters in TGD are kept constant across all eight benchmark datasets.

Localized Warm-Start Perturbation Step Rather than sweeping reverse steps empirically, setting $t_{\text{warm}} = 30$ over $T = 100$ forward steps is grounded in localized perturbation formulations Meng et al. (2021); Trabucco et al. (2024); Liu et al. (2023a). Under severe minority scarcity, intermediate perturbation confines initialization to the empirical support of the local neighborhood patch, avoiding the boundary drift of unconditioned Gaussian noise while providing sufficient variance to prevent record memorization as verified in Appendix G.

Correlation Filtering Threshold The threshold $\tau = 0.8$ serves as a structural regularizer to enforce topological sparsity. In localized patches where sample size N^* is constrained, sample correlation estimates can exhibit higher variance. Setting a relatively strict threshold acts as a strong sparsity prior, filtering out weak or noisy co-variations and ensuring that message passing is restricted to prominent feature dependencies where the resulting edge count remains significantly smaller than D^2.

Table 3: Detailed architectural specifications and hyperparameter configurations aligned with the verified benchmark pipeline.

Target	Category	Model	Architectural Specifications and Hyperparameters
80% Parity Ratio (All Oversamplers)	**Heuristic Interpolation**	SMOTE	Nearest neighbors: $k = 5$; Distance metric: Euclidean; Random state: 42; Target count: 80% of majority class; Software: `imbalanced-learn`
		ADASYN	Nearest neighbors: $k = 5$; Distance metric: Euclidean; Random state: 42; Target count: 80% of majority class; Software: `imbalanced-learn`
	Deep Generative Models	CTGAN	Generator and Discriminator: 2-layer MLP; Training epochs: 100; Batch size: 500 (default); Discrete columns: Extracted non-float attributes; Optimization: Adam with learning rate 2×10^{-4}
		TVAE	Encoder and Decoder: 2-layer MLP; Latent dimension: 12; Training epochs: 100; Batch size: 500 (default); Discrete columns: Extracted non-float attributes; Optimization: Adam with learning rate 1×10^{-3}
		GOGGLE	Relational VAE; Feature adjacency: Learnable matrix $\mathbf{A} = \sigma(\mathbf{W}_{\text{adj}}) \odot (\mathbf{1} - \mathbf{I})$; Hidden dimension: 64; Latent dimension: 16; Loss: MSE $+ 0.1 \times$ KL $+ 0.05 \times$ Sparsity; Training: Epochs: 80; Batch size: $\min(64, N_{\text{min}})$; Optimizer: Adam with learning rate 2×10^{-3}
	Tabular Diffusion Models	TabDDPM	Denoising backbone: 2-block Residual MLP with LayerNorm; Hidden dimension: 128; Time embedding: Sinusoidal with dimension 128; Activation: SiLU; Diffusion steps: $T = 100$; Variance schedule: Linear with $\beta_1 = 1 \times 10^{-4}$ and $\beta_T = 0.02$; Epochs: 80; Optimizer: AdamW with learning rate 2×10^{-3} and weight decay 1×10^{-4}; Reverse sampling: 100-step stochastic DDPM Markov chain initialized from $\mathcal{N}(\mathbf{0}, \mathbf{I})$
		TabSyn	VAE stage: 2-layer MLP (hidden dimension 64) with SiLU; Latent dimension: $\max(4, \min(16, D))$; VAE training: Epochs: 60; Optimizer: Adam with learning rate 3×10^{-3}; Diffusion stage: 3-layer MLP with hidden dimension 64; Diffusion steps: $T = 40$; Linear schedule; Diffusion training: Epochs: 60; Optimizer: Adam with learning rate 2×10^{-3}; Batch size: $\min(64, N_{\text{min}})$; Reverse sampling: 40-step stochastic DDPM Markov chain in latent space
		TabDiff	Backbone: 1-layer Multihead Attention with 4 heads and hidden dimension 64; Schedule: Feature-wise learnable rate factor $\boldsymbol{\gamma} \in \mathbb{R}^D$; Time embedding: SiLU MLP; Training: Epochs: 80; Batch size: $\min(64, N_{\text{min}})$; Optimizer: AdamW (lr $= 2 \times 10^{-3}$, weight decay 1×10^{-4}); Reverse sampling: 40-step deterministic probability flow ODE integration
	Proposed Method (TGD)	**TGD (Ours)**	Neighborhood scale estimation: TwoNN expansion factor $\alpha = 1.8$; Window bounds: $N^*_{\text{min}} = 5$, $N^*_{\text{max}} = 60$; Feature topology: Pearson correlation threshold $\tau = 0.8$; Categorical mask $\mathbf{M}_{\text{cat}} \in \{0, 1\}^{D \times D}$; Denoising backbone: 2-block Residual GCN; Hidden dimension: 64; Node projection: Linear $\mathbf{W}_{\text{in}}$, $\mathbf{W}_{\text{out}}$; GCN weight $\mathbf{W}_g$; Diffusion training: Steps: $T = 100$; Epochs: 100; Batch size: 32; Optimizer: AdamW (lr $= 1 \times 10^{-3}$, weight decay 1×10^{-4}); Reverse sampling: Warm-start step $t_{\text{warm}} = 30$; Deterministic ODE trajectory with $\eta = 0.0$
–	**Downstream Classifiers**	Decision Tree	Splitting criterion: Gini impurity; Class weight: balanced; Random state: 42
		K-Nearest Neighbors	Number of neighbors: $k = 5$; Metric: Euclidean; Parallel jobs: all available cores
		Multi-Layer Perceptron	Hidden layer sizes: 128 and 64; Activation: ReLU; Early stopping: enabled; Max iterations: 200; Random state: 42

C Extended Metric Discrepancy and Decision Threshold Analysis

Benchmark evaluations in Table 2 display systematic variations across balanced accuracy, AUC, and F1 score. These patterns stem from the mathematical formulations of the metrics, the magnification of false positive penalties under severe class skewness, and the properties of static operating thresholds.

C.1 Base-Rate Invariance versus False Positive Penalties

The divergence between balanced accuracy and macro F1 score reflects contrasting sensitivities to marginal class proportions. Let TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively. The metrics are defined as:

$$\text{Bal-Acc} = \frac{1}{2}\left(\frac{\text{TP}}{\text{TP} + \text{FN}} + \frac{\text{TN}}{\text{TN} + \text{FP}}\right) = \frac{1}{2}\left(\text{TPR} + \text{TNR}\right) \tag{18}$$

$$\text{F1} = \frac{2 \cdot \text{TP}}{2 \cdot \text{TP} + \text{FP} + \text{FN}} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{19}$$

where TPR and TNR denote the true positive rate representing minority recall and true negative rate representing majority specificity. Balanced accuracy assigns equal weight to each class independently of the marginal class prior, rendering it base-rate invariant.

In contrast, the F1 score depends directly on empirical precision. Expressing precision as a function of the minority prior $\pi_{\text{min}} = P(y = 1)$ and majority prior $\pi_{\text{maj}} = P(y = 0) = 1 - \pi_{\text{min}}$ yields:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{\pi_{\text{min}} \cdot \text{TPR}}{\pi_{\text{min}} \cdot \text{TPR} + (1 - \pi_{\text{min}}) \cdot \text{FPR}} \tag{20}$$

where $\text{FPR} = \frac{\text{FP}}{\text{TN}+\text{FP}}$ is the false positive rate. Under severe class imbalance, such as in NamG where $\pi_{\text{min}} \approx 0.02$ and $\pi_{\text{maj}} \approx 0.98$, the empirical imbalance ratio is 42 to 1 and the prior odds $\pi_{\text{maj}}/\pi_{\text{min}}$ reach approximately 49. In this regime, the majority class dominates the denominator by a factor of 49. Consequently, even a highly conservative false positive rate of $\text{FPR} = 0.03$ alongside complete minority recall with true positive rate of 1.0 severely suppresses precision:

$$\text{Precision} \approx \frac{0.02 \times 1.0}{0.02 \times 1.0 + 0.98 \times 0.03} \approx 0.405 \tag{21}$$

Under this condition, the resulting F1 score cannot exceed $\frac{2\times0.405\times1.0}{0.405+1.0} \approx 0.576$, even while balanced accuracy reaches $\frac{1.0+(1-0.03)}{2} = 0.985$.

This relationship accounts for the empirical observations in Table 2. On NamG, TGD attains the leading positions with balanced accuracy of 0.920 and AUC of 0.936, reflecting effective distribution reconstruction. Under extreme base-rate skewness, however, slight boundary shifts across majority-class regions reduce F1 to 0.482. Conversely, TVAE yields an elevated F1 of 0.647 alongside a lower balanced accuracy of 0.806, indicating that positive classifications remain confined to a compact core of minority instances without capturing broader minority modes.

C.2 Global Monotonic Ranking versus Static Decision Thresholding

A second structural divergence occurs between AUC and threshold-dependent metrics. Downstream classification labels $\hat{y} \in \{0, 1\}$ are assigned by comparing predicted posterior probabilities $p(\mathbf{x}) = P(y = 1 \mid \mathbf{x})$ against a fixed operating threshold $\theta = 0.5$:

$$\hat{y} = \mathbb{I}\left(p(\mathbf{x}) \geq \theta\right) \tag{22}$$

In contrast, AUC measures the probability that a randomly chosen minority instance $\mathbf{x}^+$ receives a higher predicted score than an arbitrary majority instance $\mathbf{x}^-$, integrating over all potential threshold values:

$$\text{AUC} = P\left(p(\mathbf{x}^+) > p(\mathbf{x}^-)\right) \tag{23}$$

On the Churn benchmark, the unaugmented baseline attains a leading AUC of 0.869 by preserving monotonic ordering, yet falls to Rank 8 on balanced accuracy at 0.769 and Rank 6 on F1 at 0.634. When discriminators are trained on unaugmented imbalanced data, majority-class evidence compresses output probabilities, concentrating minority scores within the 0.15 to 0.35 range. Although these values reliably exceed majority scores to secure high AUC, they remain below the static $\theta = 0.5$ boundary, inducing false negatives and degrading balanced accuracy.

Conversely, TVAE achieves high balanced accuracy of 0.821 at Rank 3 and a leading F1 of 0.661 at Rank 1 on Churn while recording an inferior AUC of 0.861 at Rank 7. Variational posteriors tend toward over-smoothing, which alters relative confidence ranks across extreme quantiles while shifting average predictions across the $\theta = 0.5$ threshold.

C.3 Conservative Prediction Behaviors under Extreme Imbalance

Elevated F1 scores in degenerate or unaugmented baselines frequently arise from prediction conservatism. On the SMP benchmark, the unaugmented baseline achieves a leading F1 score of 0.904, yet records a low balanced accuracy of 0.943 at Rank 8. Similarly, GOGGLE on the 72-dimensional Ol8 benchmark achieves an F1 score of 0.343 at Rank 3 despite lower balanced accuracy of 0.687 at Rank 5 and AUC of 0.765 at Rank 7.

Under severe class skewness, models suffering from posterior over-smoothing or optimization constraints restrict positive predictions to high-confidence sub-regions $\hat{\mathcal{S}}_{\text{min}} \subset \mathcal{S}^*_{\text{min}}$. While this suppresses false positives and raises empirical precision, it compromises overall minority recall. Evaluating imbalanced synthesis therefore requires inspecting both base-rate invariant metrics such as balanced accuracy and AUC alongside precision-sensitive metrics such as F1, as extreme class skewness disproportionately penalizes minor boundary explorations under static decision thresholds.

C.4 Distribution Mode Preservation across Multi-Modal Data

The dual-mode geometry of the SD benchmark illustrates the properties of localized neighborhood modeling relative to linear interpolation. As shown in the low-dimensional embeddings in Figure 8, the ground-truth minority class on SD comprises two disconnected clusters separated by intermediate majority instances.

Heuristic methods such as SMOTE and ADASYN perform linear interpolation across the input feature space:

$$\mathbf{x}_{\text{syn}} = \mathbf{x}_i + \lambda(\mathbf{x}_j - \mathbf{x}_i), \quad \lambda \sim \mathcal{U}(0, 1) \tag{24}$$

When $\mathbf{x}_i$ and $\mathbf{x}_j$ originate from distinct clusters, synthetic instances $\mathbf{x}_{\text{syn}}$ cut across intermediate majority regions. While Euclidean convex combinations expand the minority decision envelope to achieve high recall, interpolating across intermediate majority space generates out-of-distribution instances. On held-out test sets, these bridge artifacts induce false positives that depress precision and lower the resulting F1 ranks, where SMOTE ranks eighth at 0.707 and ADASYN ranks ninth at 0.639.

In contrast, TGD restricts generative trajectories to locally verified observation neighborhoods via TwoNN estimation, reconstructing both disconnected modes without synthesizing spurious bridges across decision thresholds. Consequently, TGD attains top-tier balanced accuracy of 0.886 at Rank 2 and AUC of 0.912 at Rank 3 while securing the leading F1 score of 0.733 at Rank 1, verifying that feature-graph regularized diffusion successfully balances minority recall with precision fidelity.

D Training Convergence and Loss Dynamics

Figure 7 documents the empirical score-matching loss trajectories across all eight benchmark datasets during the 100 training epochs of TGD, plotted from the mini-batch Frobenius norm objective in Eq. equation 10.

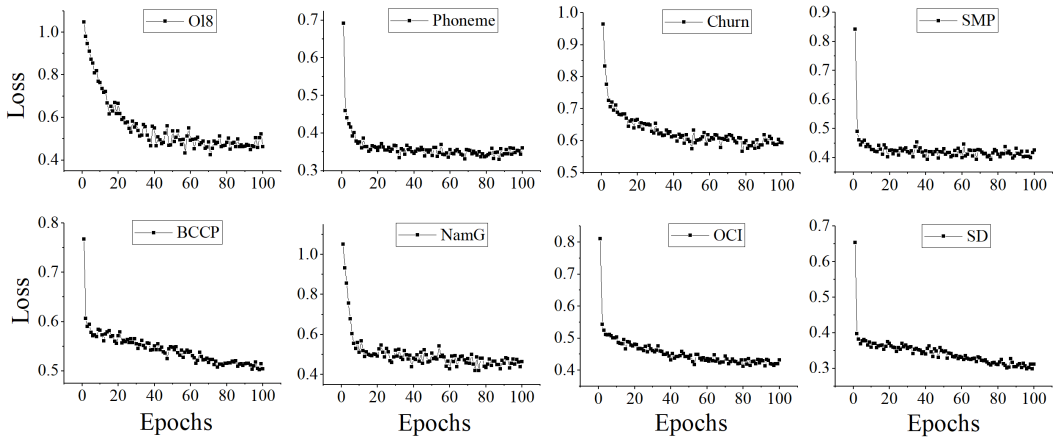


Figure 7: Empirical training loss trajectories of TGD across all eight benchmark datasets over 100 epochs. The y-axis denotes the average mini-batch mean squared error of noise estimation, and the x-axis indicates training epochs.

Optimization progresses through rapid error reduction within the initial 15 to 20 epochs across all tasks, transitioning into steady asymptotic plateaus between epochs 40 and 60 where mean squared error stabilizes in the range from 0.30 to 0.50. The smooth trajectories verify that gradient updates remain stable without divergence or loss spikes across both low-dimensional and high-dimensional datasets, confirming that a fixed budget of 100 epochs supports complete score convergence without task-specific early stopping.

E Theoretical Computational Complexity and Scalability Analysis

To evaluate the algorithmic efficiency and scalability of Tabular-to-Graph Diffusion, we characterize the computational complexity across its sequential processing stages: adaptive neighborhood scale estimation, transposed graph construction, reverse probability flow ODE sampling, and domain post-processing constraints.

Let M denote the number of minority-class training instances, D the total number of feature nodes, N^* the adaptively estimated neighborhood window size bounded between 5 and 60, and d_h the hidden dimension of graph representations set to 64.

E.1 Adaptive Neighborhood Scale Estimation

The TwoNN algorithm estimates local intrinsic dimensionality across the standardized minority subset $\mathbf{X}_{\text{min}} \in \mathbb{R}^{M \times D}$.

- **Nearest Neighbor Retrieval**: Computing the first and second nearest neighbors across minority samples requires $\mathcal{O}(M^2 D)$ operations via pairwise scans, or $\mathcal{O}(M \log M \cdot D)$ using spatial search trees in low-to-medium dimensions.
- **Closed-Form Dimension Fitting**: Sorting distance ratios μ and executing one-dimensional linear regression requires $\mathcal{O}(M \log M + M)$ operations.

TwoNN estimation is executed once across the minority partition to establish the shared window size N^*, bypassing the majority-class sample size N_{maj} entirely.

E.2 Transposed Graph Construction and Topological Masking

For each minority patch $\mathbf{X}_{\text{loc}} \in \mathbb{R}^{N^* \times D}$, TGD constructs a feature-node graph $\mathcal{G}(\mathbf{H}^{(0)}, \tilde{\mathbf{A}}_{\text{mask}})$:

- **Local Patch Extraction**: Extracting N^* nearest neighbors for all M anchors requires $\mathcal{O}(M \cdot N^* D)$ retrieval operations.
- **Empirical Covariance Computation**: Evaluating pairwise column correlations across $\mathbf{X}_{\text{loc}}$ requires $\mathcal{O}(N^* \cdot D^2)$ operations.
- **Categorical Masking and Thresholding**: Element-wise masking with $(\mathbf{1} - \mathbf{M}_{\text{cat}})$ and applying hard thresholding τ takes $\mathcal{O}(D^2)$ operations.

Because the correlation cutoff with threshold $\tau = 0.8$ removes weak correlations, the resulting adjacency matrix remains sparse, ensuring that the number of edges remains significantly smaller than D^2.

E.3 Denoising Network and Reverse ODE Sampling

The denoising backbone comprises linear projections and two residual graph convolutional blocks.

- **Graph Convolution Propagation**: For a feature graph with D nodes and $|\mathcal{E}|$ edges, normalized message passing requires $\mathcal{O}(|\mathcal{E}| \cdot d_h)$ operations, while linear node projections require $\mathcal{O}(D \cdot N^* d_h + D \cdot d_h^2)$.
- **Reverse Integration Trajectory**: Deterministic probability flow integration executes from warm-start step t_{warm} to 0. The sampling cost across t_{warm} steps scales as:

$$\mathcal{O}\left(t_{\text{warm}} \cdot \left(|\mathcal{E}| \cdot d_h + D \cdot d_h^2 + D \cdot N^* d_h\right)\right) \tag{25}$$

With fixed parameters $t_{\text{warm}} = 30$, $d_h = 64$, and $N^* \leq 60$, reverse integration scales linearly with feature dimension D and edge density $|\mathcal{E}|$. In contrast to Transformer-based tabular diffusion architectures such as TabDiff, which incur $\mathcal{O}(T \cdot D^2 d_h)$ quadratic attention costs across all columns, sparse graph message passing in TGD maintains favorable computational scaling in higher feature dimensions.

E.4 Domain Post-Processing Constraints

Post-processing operates coordinate-wise across synthesized records:

- **Minority Moment Matching**: Canonicalizing empirical mean and variance requires $\mathcal{O}(D)$ operations per record.
- **Continuous Value Clamping**: Bounding continuous columns by observed training extremes requires $\mathcal{O}(D_{\text{cont}})$ operations per record.
- **Categorical Argmax Discretization**: Determining maximum scores within discrete one-hot groups requires $\mathcal{O}(D_{\text{cat}})$ operations per record.

The post-processing stage incurs linear computational overhead of $\mathcal{O}(D)$.

E.5 Overall Scalability and Imbalance Robustness

A practical property of TGD is its restriction to minority-class instances during training and graph modeling. In extreme imbalance settings such as NamG where the minority class comprises two percent of the data, standard deep generative frameworks process the entire joint distribution of size $N = N_{\text{maj}} + M$, incurring optimization overhead across uninformative majority regions. In contrast, TGD models localized observation patches exclusively on the minority subset $M \ll N$, avoiding parameter saturation and explaining its monotonic convergence within 50 epochs under a 100-epoch budget and moderate 59.30-second average runtime on the Pareto efficiency frontier.

F Extended Distribution Geometry and Feature Covariance Analysis

F.1 Extended Low-Dimensional Distribution Embeddings

Figure 8 displays two-dimensional UMAP projections for the remaining five benchmark datasets: Phoneme, BCCP, Churn, SMP, and SD. These visualizations provide further empirical verification of the boundary preservation characterized in Section 5.2.

Across tasks exhibiting distinct isolated minority clusters such as BCCP and Churn, ground-truth minority records reside within compact islands separated from the majority distribution, where X lies between 15 and 22 on BCCP and near 15 and 20 on Churn. Standard heuristic oversamplers such as SMOTE and ADASYN alongside unconstrained generative architectures including CTGAN, TVAE, TabSyn, and TabDiff synthesize instances that diffuse broadly into majority-class regions where X ranges from -10 to 5. In contrast, TGD consistently restricts generation to the empirical minority distribution.

On SMP and SD, where the minority class follows narrow curved feature sub-distributions or multi-modal disconnected bands, baseline models generate extensive out-of-distribution instances in un-populated regions between minority modes. TGD reconstructs both disconnected minority modes on SD located across separate coordinate intervals without generating spurious bridges across majority clusters, confirming the boundary adaptation of TwoNN adaptive neighborhood estimation.

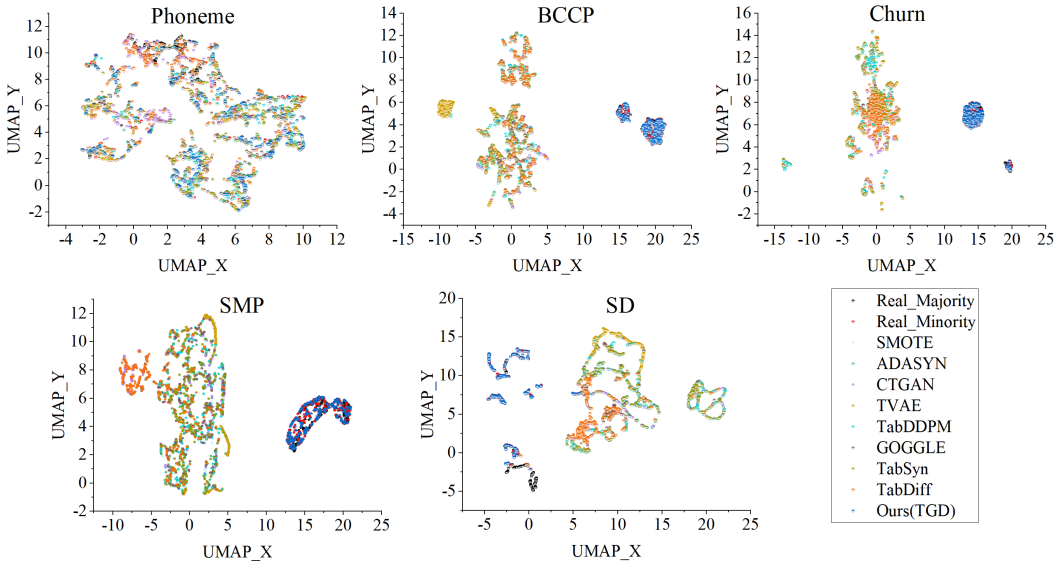
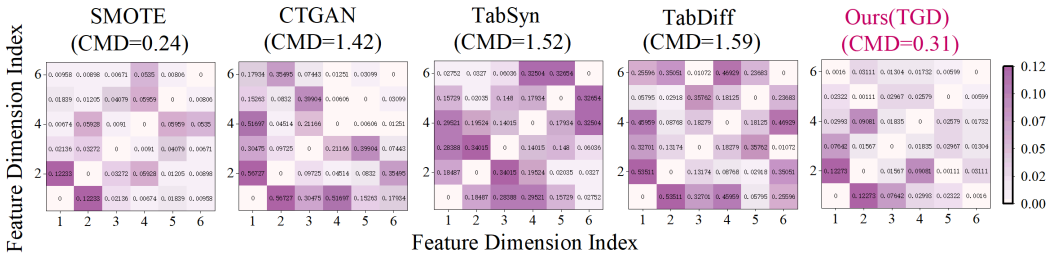


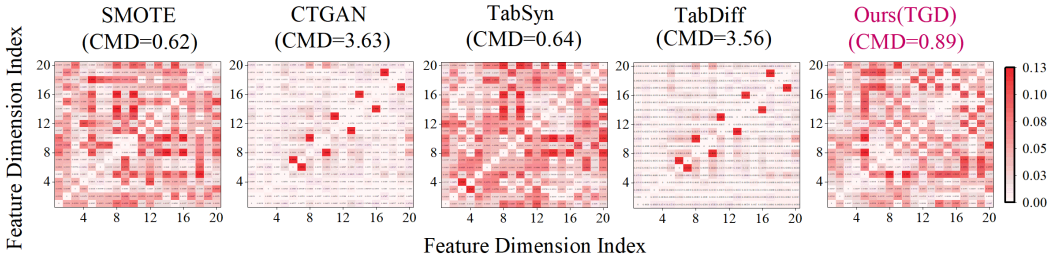
Figure 8: Two-dimensional UMAP distribution embeddings across five benchmark datasets (Phoneme, BCCP, Churn, SMP, and SD). Red markers denote ground-truth minority instances, black markers denote majority instances, and colored markers represent synthetic instances generated by evaluated oversampling and generative models.

F.2 Multi-Scale Feature Covariance Residual Analysis

To demonstrate that the feature-level covariance fidelity of TGD generalizes across varying feature dimensionalities, Figure 9 illustrates absolute Pearson correlation residual heatmaps and corresponding Frobenius covariance matrix distances on the compact 6-dimensional NamG benchmark and the medium-dimensional Churn benchmark displaying its primary 20 attributes.



(a) Absolute Pearson correlation residual heatmaps on the low-dimensional NamG dataset ($D = 6$).



(b) Absolute Pearson correlation residual heatmaps on the medium-dimensional Churn dataset.

Figure 9: Heatmaps of absolute Pearson correlation residual matrices relative to true minority records on NamG and Churn. Headers report Frobenius covariance matrix distances (CMD). Lower CMD values indicate superior multi-variable statistical fidelity.

On the compact 6-dimensional NamG dataset shown in Figure 9a, vector-flattened generative models display elevated covariance error matrices, with CTGAN, TabSyn, and TabDiff yielding CMD values of 1.42, 1.52, and 1.59, respectively. Heuristic SMOTE attains a low CMD of 0.24, as pairwise linear interpolation inherently preserves linear expectations along straight Euclidean segments; however, it tends to cut across non-convex boundaries into majority regions as visualized in Figure 4. TGD preserves the non-convex distribution geometry while maintaining a low covariance distance of 0.31, substantially outperforming vector diffusion baselines on multi-variable correlation preservation.

On the medium-dimensional Churn benchmark shown in Figure 9b, CTGAN and TabDiff produce substantial correlation distortions across multiple attribute pairs, resulting in CMD values of 3.63 and 3.56. In contrast, TGD maintains low residual values across the entire matrix, securing a CMD of 0.89. In combination with the high-dimensional Ol8 benchmark with 72 features and a CMD of 5.30 analyzed in Section 5.2, these results confirm that transposed topological diffusion robustly retains multi-feature covariance structures across low, medium, and high-dimensional tabular spaces.

G Novelty and Empirical Memorization Analysis

Because TGD initializes reverse sampling from a localized perturbation at step $t_{\text{warm}} = 30$ rather than an isotropic Gaussian distribution, we empirically evaluate whether the model synthesizes genuinely novel records or memorizes training coordinates.

For each synthetic minority instance $\mathbf{x}_{\text{syn}}$, we compute the Euclidean Distance to Closest Record, termed DCR, relative to the standardized minority training partition $\mathbf{X}_{\text{train, min}}$:

$$\text{DCR}(\mathbf{x}_{\text{syn}}) = \min_{\mathbf{x}_i \in \mathbf{X}_{\text{train, min}}} \|\mathbf{x}_{\text{syn}} - \mathbf{x}_i\|_2 \tag{26}$$

As an authentic empirical baseline, we simultaneously evaluate the intrinsic distance distribution between held-out test minority records $\mathbf{x}_{\text{test}}$ and the training minority partition:

$$\text{DCR}(\mathbf{x}_{\text{test}}) = \min_{\mathbf{x}_i \in \mathbf{X}_{\text{train, min}}} \|\mathbf{x}_{\text{test}} - \mathbf{x}_i\|_2 \tag{27}$$

Table 4 reports the novelty metrics across all eight benchmark datasets. While minimum distances are naturally small in dense low-dimensional spaces (e.g., 0.0040 on SD and 0.0417 on NamG), exact zero replication does not occur. More importantly, the median distance of synthetic instances is of the same order of magnitude as the empirical test baseline across all dimensional scales. On dense benchmarks including Phoneme, NamG, and SD, the synthetic medians closely match the test baselines, demonstrating exploration across the minority data distribution. On high-dimensional or sparse tables such as Ol8 and Churn, the synthetic median remains moderately below the test baseline, reflecting that TwoNN-guided local patch diffusion confines synthetic samples within the empirical support of the minority class without drifting into unpopulated external regions.

Table 4: Evaluation of sample novelty and memorization via Distance to Closest Record (DCR) across eight benchmark datasets. All metrics are computed in the standardized feature space.

Dataset	**Min DCR (Syn)**	**5th Pct. DCR (Syn)**	**Median DCR (Syn)**	**Median DCR (Test GT)**
Ol8 ($D = 72$)	1.7321	2.1125	2.4772	4.0356
Phoneme ($D = 5$)	0.0692	0.1681	0.3287	0.2448
Churn ($D = 33$)	0.5543	0.9651	1.3715	3.0944
BCCP ($D = 11$)	0.1671	0.4634	0.8023	1.4961
SMP ($D = 6$)	0.1019	0.2780	0.4804	0.6114
NamG ($D = 6$)	0.0417	0.2478	0.4894	0.3915
OCI ($D = 17$)	0.3224	0.6565	1.0302	1.5067
SD ($D = 5$)	0.0040	0.0655	0.2012	0.1546

H Ablation Study Details and Parameter Sensitivity

H.1 Per-Dataset Quantitative Ablation Results

Table 5 provides the comprehensive quantitative results across all eight benchmark datasets for the six ablation variants and the integrated TGD model. Each metric reports the mean and standard deviation across three independent runs, with relative ranks indicated numerically.

H.2 Neighborhood Window Size Sensitivity

Figure 10 displays downstream classification performance under static neighborhood sizes with values 5, 15, and 35 compared to TwoNN adaptive neighborhood estimation. When set too small with $N = 5$, the local sample support lacks sufficient statistical degrees of freedom to estimate local column covariance structures. Conversely, setting the window excessively large with $N = 35$ causes localized linear approximations to degrade across complex class boundaries, incorporating majority-class instances across decision thresholds. TwoNN automatically calibrates the expansion window $N^* = \text{clip}(\lceil \alpha \cdot d \rceil, 5, \min(60, M))$ according to the estimated local intrinsic dimensionality d, avoiding manual hyperparameter tuning across disparate tabular datasets.

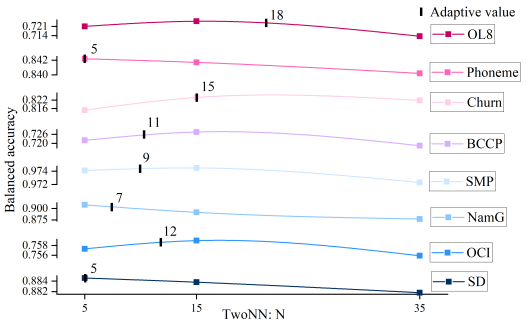


Figure 10: Ablation of neighborhood window sizes under fixed parameter settings ($N \in \{5, 15, 35\}$).

Table 5: Quantitative ablation results across eight benchmark datasets evaluated on an independent ablation trial suite. Parenthetical values indicate relative metric rankings within each dataset.

Dataset	Variant	Bal-Acc (↑)	AUC (↑)	F1 (↑)	Mean Rank (↓)
Ol8	Full (Ours)	0.738 ± 0.001 (2)	0.818 ± 0.004 (2)	$\mathbf{0.402 \pm 0.007}$ (1)	**1.667**
	w/o Graph (MLP)	0.716 ± 0.007 (6)	0.807 ± 0.013 (5)	0.383 ± 0.014 (5)	5.333
	w/o TwoNN	0.720 ± 0.006 (5)	0.805 ± 0.005 (6)	0.378 ± 0.004 (6)	5.667
	w/o Cat. Mask	0.731 ± 0.006 (3)	$\mathbf{0.821 \pm 0.005}$ (1)	0.393 ± 0.001 (2)	2.000
	w/o Post-Proc.	0.714 ± 0.014 (7)	0.804 ± 0.009 (7)	0.371 ± 0.011 (7)	7.000
	w/o Det. Flow	0.726 ± 0.005 (4)	0.812 ± 0.007 (4)	0.389 ± 0.005 (3)	3.667
	w/o Moment	$\mathbf{0.745 \pm 0.003}$ (1)	0.815 ± 0.001 (3)	0.389 ± 0.006 (4)	2.667
Phoneme	Full (Ours)	$\mathbf{0.846 \pm 0.001}$ (1)	$\mathbf{0.898 \pm 0.001}$ (1)	$\mathbf{0.772 \pm 0.002}$ (1)	**1.000**
	w/o Graph (MLP)	0.840 ± 0.001 (6)	0.894 ± 0.001 (5)	0.764 ± 0.002 (7)	6.000
	w/o TwoNN	0.841 ± 0.001 (3)	0.895 ± 0.002 (3)	0.768 ± 0.001 (3)	3.000
	w/o Cat. Mask	0.844 ± 0.005 (2)	0.898 ± 0.005 (2)	0.770 ± 0.006 (2)	2.000
	w/o Post-Proc.	0.841 ± 0.004 (5)	0.895 ± 0.004 (4)	0.766 ± 0.004 (5)	4.667
	w/o Det. Flow	0.841 ± 0.001 (4)	0.893 ± 0.003 (6)	0.767 ± 0.003 (4)	4.667
	w/o Moment	0.840 ± 0.003 (7)	0.892 ± 0.001 (7)	0.765 ± 0.004 (6)	6.667
Churn	Full (Ours)	$\mathbf{0.831 \pm 0.000}$ (1)	0.866 ± 0.003 (2)	0.660 ± 0.004 (2)	**1.667**
	w/o Graph (MLP)	0.827 ± 0.005 (3)	0.862 ± 0.005 (4)	0.659 ± 0.010 (3)	3.333
	w/o TwoNN	0.820 ± 0.005 (6)	0.861 ± 0.002 (6)	0.652 ± 0.017 (5)	5.667
	w/o Cat. Mask	0.825 ± 0.004 (4)	0.865 ± 0.004 (3)	0.656 ± 0.004 (4)	3.667
	w/o Post-Proc.	0.809 ± 0.007 (7)	0.859 ± 0.003 (7)	$\mathbf{0.675 \pm 0.014}$ (1)	5.000
	w/o Det. Flow	0.821 ± 0.000 (5)	0.861 ± 0.003 (5)	0.646 ± 0.000 (7)	5.667
	w/o Moment	0.828 ± 0.002 (2)	$\mathbf{0.870 \pm 0.000}$ (1)	0.651 ± 0.002 (6)	3.000
NamG	Full (Ours)	0.915 ± 0.004 (2)	$\mathbf{0.938 \pm 0.002}$ (1)	0.503 ± 0.011 (6)	3.000
	w/o Graph (MLP)	0.906 ± 0.008 (3)	0.938 ± 0.013 (3)	0.504 ± 0.022 (5)	3.667
	w/o TwoNN	0.892 ± 0.015 (6)	0.932 ± 0.011 (6)	0.487 ± 0.046 (7)	6.333
	w/o Cat. Mask	0.904 ± 0.013 (5)	0.933 ± 0.006 (5)	0.512 ± 0.011 (3)	4.333
	w/o Post-Proc.	0.905 ± 0.018 (4)	0.934 ± 0.008 (4)	0.507 ± 0.027 (4)	4.000
	w/o Det. Flow	0.883 ± 0.012 (7)	0.930 ± 0.011 (7)	0.513 ± 0.005 (2)	5.333
	w/o Moment	$\mathbf{0.916 \pm 0.006}$ (1)	0.938 ± 0.006 (2)	$\mathbf{0.556 \pm 0.019}$ (1)	**1.333**
BCCP	Full (Ours)	$\mathbf{0.727 \pm 0.002}$ (1)	0.778 ± 0.001 (2)	$\mathbf{0.548 \pm 0.002}$ (1)	**1.333**
	w/o Graph (MLP)	0.722 ± 0.006 (4)	0.776 ± 0.004 (4)	0.541 ± 0.005 (3)	3.667
	w/o TwoNN	0.723 ± 0.005 (3)	0.776 ± 0.002 (3)	0.547 ± 0.007 (2)	2.667
	w/o Cat. Mask	0.722 ± 0.001 (6)	0.774 ± 0.000 (5)	0.540 ± 0.001 (5)	5.333
	w/o Post-Proc.	0.715 ± 0.003 (7)	0.770 ± 0.002 (7)	0.534 ± 0.005 (7)	7.000
	w/o Det. Flow	0.722 ± 0.005 (5)	0.774 ± 0.008 (6)	0.541 ± 0.012 (4)	5.000
	w/o Moment	0.726 ± 0.004 (2)	$\mathbf{0.779 \pm 0.003}$ (1)	0.539 ± 0.005 (6)	3.000
SMP	Full (Ours)	0.974 ± 0.000 (4)	0.990 ± 0.000 (6)	0.862 ± 0.006 (3)	4.333
	w/o Graph (MLP)	0.974 ± 0.001 (3)	0.990 ± 0.000 (5)	0.857 ± 0.001 (6)	4.667
	w/o TwoNN	0.974 ± 0.001 (5)	0.991 ± 0.000 (2)	0.861 ± 0.002 (5)	4.000
	w/o Cat. Mask	$\mathbf{0.975 \pm 0.001}$ (1)	0.991 ± 0.001 (4)	$\mathbf{0.865 \pm 0.005}$ (1)	**2.000**
	w/o Post-Proc.	0.974 ± 0.001 (2)	0.991 ± 0.001 (3)	0.863 ± 0.003 (2)	2.333
	w/o Det. Flow	0.974 ± 0.002 (6)	$\mathbf{0.991 \pm 0.000}$ (1)	0.862 ± 0.003 (4)	3.667
	w/o Moment	0.960 ± 0.002 (7)	0.986 ± 0.002 (7)	0.792 ± 0.010 (7)	7.000
OCI	Full (Ours)	$\mathbf{0.762 \pm 0.003}$ (1)	0.811 ± 0.002 (2)	0.559 ± 0.004 (4)	2.333
	w/o Graph (MLP)	0.750 ± 0.002 (7)	0.808 ± 0.003 (7)	0.554 ± 0.005 (7)	7.000
	w/o TwoNN	0.757 ± 0.002 (3)	0.810 ± 0.003 (4)	0.560 ± 0.004 (3)	3.333
	w/o Cat. Mask	0.755 ± 0.001 (4)	$\mathbf{0.812 \pm 0.002}$ (1)	0.561 ± 0.002 (2)	2.333
	w/o Post-Proc.	0.753 ± 0.001 (5)	0.810 ± 0.004 (5)	0.557 ± 0.004 (6)	5.333
	w/o Det. Flow	0.752 ± 0.001 (6)	0.810 ± 0.002 (6)	0.558 ± 0.001 (5)	5.667
	w/o Moment	0.758 ± 0.001 (2)	0.811 ± 0.000 (3)	$\mathbf{0.564 \pm 0.005}$ (1)	**2.000**
SD	Full (Ours)	$\mathbf{0.886 \pm 0.001}$ (1)	0.909 ± 0.000 (5)	0.735 ± 0.002 (2)	**2.667**
	w/o Graph (MLP)	0.880 ± 0.003 (5)	$\mathbf{0.910 \pm 0.002}$ (1)	0.732 ± 0.004 (5)	3.667
	w/o TwoNN	0.883 ± 0.002 (2)	0.909 ± 0.001 (4)	0.732 ± 0.003 (6)	4.000
	w/o Cat. Mask	0.883 ± 0.001 (3)	0.909 ± 0.000 (3)	0.731 ± 0.003 (7)	4.333
	w/o Post-Proc.	0.877 ± 0.002 (7)	0.907 ± 0.000 (7)	$\mathbf{0.738 \pm 0.001}$ (1)	5.000
	w/o Det. Flow	0.878 ± 0.001 (6)	0.909 ± 0.002 (2)	0.734 ± 0.004 (3)	3.667
	w/o Moment	0.882 ± 0.001 (4)	0.908 ± 0.001 (6)	0.733 ± 0.002 (4)	4.667