DIDA: A VALUE-DRIVEN PARADIGM FOR DATA IMPORTANCE-BASED TABULAR DATA AUGMENTATION IN LOW-DATA HIGH-CONSTRAINT SCENARIOS

Anonymous authors
Paper under double-blind review

ABSTRACT

In regulated domains such as medical diagnosis and financial risk control, tabular datasets are often small, imbalanced, and subject to strong cross-feature constraints. Existing augmentation methods therefore face a difficult trade-off among preserving semantic consistency, increasing sample diversity, and keeping augmentation decisions auditable. We propose **DIDA** (**D**ata-**I**mportance driven **D**ifferential **A**ugmentation), a value-aware framework that differentiates augmentation at the sample-feature level using two complementary signals: feature-level SHAP importance and sample-level Shapley contribution. We formulate augmentation as diversity maximization under semantic-consistency constraints and show that the optimal perturbation magnitude decreases with sample contribution and scales inversely with feature importance (Theorem 1), providing a principled basis for value-dependent perturbation. DIDA further incorporates dynamic domain correction and constraint-guided categorical augmentation to preserve valid feature combinations. Across six public tabular datasets, six augmentation baselines, and ten downstream classifiers, DIDA improves average AUC-ROC, AUC-PR, and F1 by 1.94, 2.30, and 4.30 percentage points, respectively, over the strongest competing baseline. Its generated samples also achieve lower JS divergence and Wasserstein distance, together with higher information entropy and feature coverage, indicating that DIDA improves diversity without sacrificing distributional fidelity.

1 INTRODUCTION

Training data quality now dominates model performance, driving a shift from model-centric to data-centric AI (Zha et al., 2025; Jarrahi et al., 2023). Models of identical architecture generalize differently when trained on data of different quality, and the strength of large language models such as GPT-4 stems from systematic curation of high-quality training data rather than architectural novelty (OpenAI, 2023). However, in high-risk decision-making domains such as medical diagnosis and financial risk control, model construction faces a pronounced data bottleneck. High data acquisition costs and strict privacy regulation including HIPAA and GDPR leave too few effective samples to train complex models (Rajkomar et al., 2019). Compounding this issue, the low incidence of critical events such as disease onset and loan default induces severe class imbalance, where minority classes often account for less than 5% of the available data (He & Garcia, 2009). Domain expertise and regulatory requirements further impose hard restrictions on valid combinations of feature values. For instance, the pairing *gender* = *male* with *pregnancy status* = *positive* is physiologically impossible, while financial ratios must satisfy accounting identities. Synthetic samples violating these constraints can mislead downstream risk judgment (Chen et al., 2021). These are business-logic rules, distinct from algorithmic fairness constraints such as demographic parity; this paper targets the former, while the constraint-table framework can be extended to the latter. Data collected in such domains are also largely tabular, with any unstructured inputs reduced to tabular features via feature engineering (Borisov et al., 2024). These challenges combine to yield datasets that are small, heavily constrained, and imbalanced, preventing complex models from reliably learning the latent regularities underlying real-world scenarios.

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.

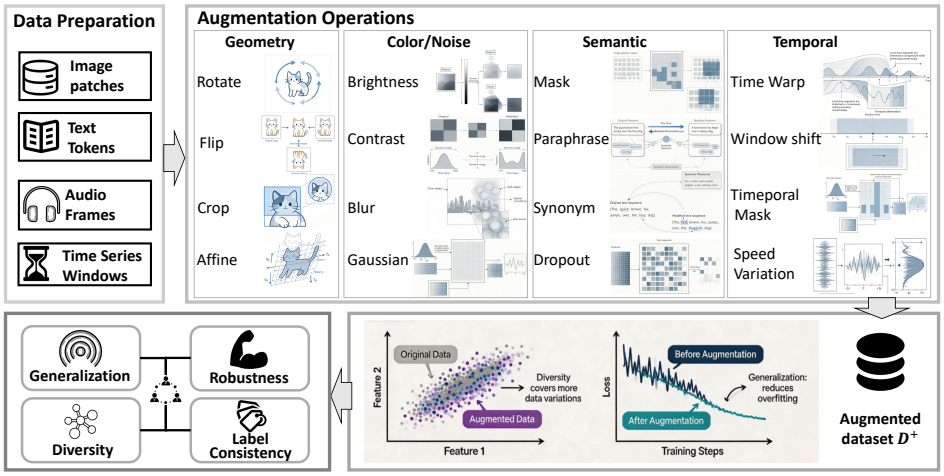

Figure 1: Background and overview: regulated tabular domains suffer from scarce samples, class imbalance, and hard feature constraints, which existing augmentation methods cannot satisfy at once, motivating value-driven differentiated augmentation.

Challenges. The fundamental limitation of existing augmentation methods (Figure 1) points to a single core tension: to augment effectively in high-constraint, small-sample scenarios, one must simultaneously satisfy *compliance* (not violating hard domain constraints), *auditability* (traceable augmentation decisions), and *differentiation* (allocating a limited budget according to data value). This gives rise to two key technical challenges.

Challenge 1: Each sample-feature unit requires a perturbation bound derived from two signals of different granularity, namely the sample-level Shapley contribution ds'_i (Ghorbani & Zou, 2019) and the feature-level SHAP importance b'_j (Lundberg & Lee, 2017), which together define a two-dimensional data value. This bound is required to be provably optimal and inversely proportional to that value: high-value units receive only light perturbation, while low-value units receive stronger perturbation, all within the domain-statistical boundary $[L_j, U_j]$. To the best of our knowledge, no existing tabular augmentation work establishes such a mapping.

Challenge 2: A single auditable data-value score has to govern two distinct mechanisms at the same time. Numerical perturbation needs to preserve value-range continuity and distribution consistency, and categorical replacement needs to respect conditional-probability constraints, with b'_j and ds'_i jointly determining the perturbation magnitude and the replacement probability. Every feature-value change also has to trace back to its decision basis, either a perturbation-function output or a constraint-table sample, recorded together with its source, value parameters, and conditional-probability distribution.

Contributions. We propose DIDA (Data-Importance driven Differential Augmentation), a differentiated tabular data augmentation framework driven uniformly by two-dimensional data value. Its core idea is to use data-value scores as the single basis for differentiated decisions, so that the augmentation intensity received by each sample-feature unit is inversely proportional to its importance to the downstream task. Our contributions are fourfold:

1) Value-aware augmentation theory. We formalize differentiated augmentation as sample-diversity maximization under a semantic-consistency constraint, and derive via KKT conditions a quantitative inverse relationship between optimal perturbation intensity and two-dimensional data value (Theorem 1).

2) Adaptive domain correction and differentiated numerical augmentation. We propose dynamic domain correction that extends the feature boundary only for out-of-domain samples with $ds'_i \geq \theta$, and a three-tier perturbation-round allocation driven by $V_{i,j} = \varphi_1 b'_j + \varphi_2 ds'_i$ that micro-perturbs high-value units and macro-perturbs low-value units.

3) Constraint-guided categorical augmentation. We design constraint-table construction based on a fused ANOVA–MI score, ensuring compliant categorical replacement via dual-threshold screening and a three-level fallback, sharing the unified scoring system with numerical augmentation.

4) Systematic empirical validation. On six public tabular datasets and ten classifiers, DIDA-augmented data on average outperform six mainstream baselines on AUC-ROC, AUC-PR, and F1, and the generated samples achieve superior distribution consistency, sample diversity, and feature coverage.

2 Related Work

Tabular augmentation methods span interpolation-based (SMOTE (Chawla et al., 2002), MixUp (Zhang et al., 2018)), deep generative (CTGAN (Xu et al., 2019), TabDDPM (Kotelnikov et al., 2023), MDLM (Isomura et al., 2025)), and implicit (TANDEM (Naor & Lindenbaum, 2025)) families, yet all are sample-agnostic, augmenting every sample alike even though Data Shapley (Ghorbani & Zou, 2019) shows contributions differ by orders of magnitude. Two capabilities DIDA builds on remain open. Valuation methods (Data Shapley, DVRL (Yoon et al., 2020), influence functions (Koh & Liang, 2017)) answer only which samples to keep or discard, not how to augment each sample differently; DIDA instead uses value to differentiate augmentation, adopting the efficient KNN-Shapley estimator ($O(N \log N)$, Appendix A). Constraint-aware generation relies on soft loss penalties or post-generation rule filtering; DIDA instead enforces constraints through a nonparametric conditional-probability constraint table (Appendix A.2). DIDA therefore differs from prior work in two ways: a KKT-derived quantitative mapping from data value to augmentation intensity (Theorem 1), and an explicit, traceable constraint table, with the full method comparison in Table 2.

3 Preliminaries

3.1 Formal Definitions

This section formalizes value-sensitive constrained optimization and derives via KKT the value-perturbation relationship (Theorem 1). The practical form is in Section 4. Consider a classification task over a tabular dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$ with $x_i = [x_i^{(\text{num})}; x_i^{(\text{cat})}]$, $m = m_{\text{num}} + m_{\text{cat}}$, $y_i \in \{1, \ldots, K\}$. Appendix B lists the notation. We first quantify data value from the feature dimension. Given a high-performance predictor f_{base} (XGBoost (Chen & Guestrin, 2016) / LightGBM (Ke et al., 2017)) pretrained on D, we compute the full-sample SHAP value matrix $\Phi \in \mathbb{R}^{N \times m}$ via TreeExplainer (Lundberg & Lee, 2017; Lundberg et al., 2020).

Definition 1 (Feature importance). The global importance of feature j is the mean of the absolute SHAP values over all samples, max-normalized to $[0, 1]$: $b_j = \frac{1}{N} \sum_{i=1}^{N} |\phi_{ij}|$ and $b'_j = \frac{b_j}{\max_{1 \le k \le m} b_k}$. b'_j captures the average marginal effect of feature j, and SHAP interaction values (Lundberg et al., 2020) may be substituted if strong interactions exist. Feature importance identifies which features matter overall, but samples can differ in contribution by orders of magnitude, so we must also quantify sample-level value.

Definition 2 (Sample contribution). Following Data Shapley (Ghorbani & Zou, 2019), each training sample is treated as a player in a cooperative game, and its Shapley value quantifies its marginal contribution to model performance:

$$ds_i = \sum_{S \subseteq D \setminus \{i\}} \frac{|S|!\ (|D| - |S| - 1)!}{|D|!} \left[U(S \cup \{i\}) - U(S) \right], \tag{1}$$

where $U(S)$ is the AUC-ROC on a validation set of a model trained on S. The Shapley value is the unique attribution scheme satisfying efficiency, symmetry, null player, and additivity. Exact computation is $O(2^N)$, so we adopt the KNN-Shapley approximation (Jia et al., 2019) ($O(N \log N)$, asymptotically unbiased, see Section 4.1). A min–max normalization with 99th-percentile truncation removes outliers: $ds'_i = \text{clip}\Big(\frac{ds_i - ds_{\min}}{ds_{99} - ds_{\min}},\ 0,\ 1\Big)$. Together b'_j and ds'_i, sharing a Shapley-axiomatic origin, form DIDA's two-dimensional value basis, encoded into the augmentation constraints via the following measures.

Definition 3 (Semantic Consistency). Define the feature-dimension weight vector $w = (b'_1, b'_2, \ldots, b'_m)$, and let $x_i = (a_1, a_2, \ldots, a_m)$ be the i-th original sample and $x_i^{\text{new}} = (a_1^{\text{new}}, a_2^{\text{new}}, \ldots, a_m^{\text{new}})$ its augmented version. On this basis, using weighted cosine similarity as the base measure and introducing a sample-level feature-deviation residual penalty, we construct the quantifiable semantic-consistency measure $\text{Consist}(x_i^{\text{new}}, x_i)$:

$$\text{Consist}(x_i^{\text{new}}, x_i) = \frac{\sum_j w_j \, x_{ij} \, x_{ij}^{\text{new}}}{\sqrt{\sum_j w_j x_{ij}^2} \sqrt{\sum_j w_j (x_{ij}^{\text{new}})^2}} - \frac{\gamma \cdot ds'_i}{m} \sum_j w_j \left| x_{ij} - x_{ij}^{\text{new}} \right|, \tag{2}$$

where $\gamma \in (0, 1]$ is a penalty-coefficient hyperparameter. The first term is feature-weighted cosine similarity and the second a feature-deviation residual penalty. Since $x_i^{\text{new}} = x_i + \delta_i$, Consist is an implicit function of the perturbation vector δ_i, expanded via the chain rule in the KKT derivation (see Appendix D). Beyond this semantic constraint, every feature value must lie within a legal range defined jointly by the data distribution and business rules.

Definition 4 (Complete Domain). Based on the original data D, the statistical boundary of feature j is determined by the interquartile range (IQR) rule: $L_j^{(\text{stat})} = Q1_j - 1.5 \cdot IQR_j, U_j^{(\text{stat})} = Q3_j + 1.5 \cdot IQR_j$. Combined with the business boundary $[L_j^{(\text{biz})}, U_j^{(\text{biz})}]$ from domain rules, the initial complete domain is their intersection $S_j = \left[L_j^{(\text{stat})}, U_j^{(\text{stat})}\right] \cap \left[L_j^{(\text{biz})}, U_j^{(\text{biz})}\right]$ (see Appendix C). When no business boundary is specified, S_j reduces to the statistical boundary alone; for small samples, a business boundary, when available, is preferred. S_j serves as the box constraint preceding feature perturbation, ensuring $x_{ij}^{\text{new}} \in S_j$ always holds. The value scores, semantic-consistency measure, and feasible boundary S_j are injected into the optimization below.

3.2 The Optimization Problem

For each numerical feature j of each sample i, let $\delta_{ij} \in \mathbb{R}^+$ be the perturbation variable with $x_{ij}^{\text{new}} = x_{ij} + \delta_{ij} \cdot R_{ij}$, $R_{ij} \in \{-1, +1\}$. Within the semantic-consistency and complete-domain constraints, a larger total perturbation magnitude yields higher data diversity, so DIDA formalizes differentiated augmentation as sample-diversity maximization: $\max \sum_{i,j} \delta_{ij}$ subject to $\text{Consist}(x_i^{\text{new}}, x_i) \geq 1 - \varepsilon_0$. Under a small-perturbation assumption, the penalty term is bounded and can be absorbed by $\varepsilon_0 \cdot ds'_i$ (Lemma D.1, Appendix D), so there exists $\gamma \in (0, 1]$ such that the penalty term is approximated by $\gamma\varepsilon_0 \cdot ds'_i$, leading to the following sufficient surrogate constraint

$$\max \sum_{i,j} \delta_{ij} \quad \text{s.t.} \quad \text{WeightedCos}(x_i^{\text{new}}, x_i) \geq 1 - \varepsilon_0 (1 - ds'_i). \tag{3}$$

After the simplification of Lemma D.1, the surrogate problem is a convex program whose closed-form solution follows from the KKT conditions, guaranteeing feasibility of the original semantic-consistency constraint.

Theorem 1. *Assume the perturbation intensity satisfies $\delta_{ij} \ll |x_{ij}|$ and every feature has positive normalized importance $b'_j > 0$. Then the closed-form solution of the sufficient surrogate diversity-maximization problem, which guarantees the original semantic-consistency constraint, decreases with ds'_i and is inversely proportional to b'_j, i.e., $\delta^*_{ij} \propto \sqrt{1 - ds'_i}/b'_j$ (see Figure 7 in Appendix D for an illustration, with the full proof therein).*

4 Method

Building on Theorem 1, this section operationalizes the theory: it first constructs an explicit perturbation function (Definition 5), then builds the three-stage DIDA pipeline, value scoring (Section 4.1), constraint-table construction (Section 4.2), and differentiated augmentation (Section 4.3); see Figure 2.

Definition 5 (Perturbation Function). Based on Theorem 1, we construct the value-inverse perturbation function:

$$f(b'_j, ds'_i) = t_0 \left(1 - \left(\varphi_1 b'_j + \varphi_2 ds'_i\right)\right)^p + \varepsilon, \tag{4}$$

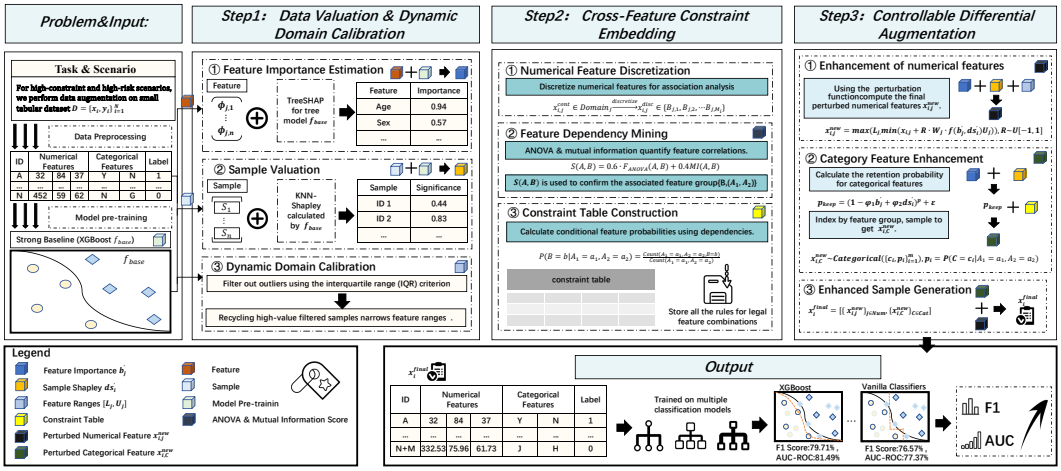

Figure 2: The DIDA pipeline: two-dimensional data value, combining feature-level SHAP importance and sample-level Shapley contribution, is estimated, injected into a constraint table, and used to drive differentiated numerical perturbation and categorical replacement that preserves high-value units and reshapes low-value units.

where t_0 is the maximum-perturbation anchor, φ_1, φ_2 the fusion weights (default $0.7/0.3$, with $\varphi_1 + \varphi_2 = 1$), $p > 0$ the perturbation-decay coefficient, and $\varepsilon > 0$ the minimum perturbation threshold. It satisfies $\partial f/\partial b'_j < 0$ and $\partial f/\partial ds'_i < 0$, consistent with Theorem 1, and degenerates to feature importance alone as $\varphi_2 \to 0$.

4.1 Value Scoring with Domain Correction

The first stage of the pipeline estimates the two value scores. A gradient-boosting tree model f_{base} (Chen & Guestrin, 2016; Ke et al., 2017) is pretrained on D. TreeExplainer (Lundberg & Lee, 2017) yields b'_j, and KNN-Shapley (Jia et al., 2019) yields ds'_i, together forming the two-dimensional value scores. Filtering solely by the initial complete domain S_j (Definition 4) risks discarding high-value marginal samples near the decision boundary, so we introduce dynamic domain correction (Figure 3). Let $G = \{x_i \mid \exists j,\ x_{ij} \notin S_j\}$ be the samples violating S_j, and with a contribution threshold θ (default 0.2), we select $G^* = \{x_i \in G \mid ds'_i \geq \theta\}$ and extend $S_j \leftarrow S_j \cup [\min_{x_i \in G^*} x_{ij},\ \max_{x_i \in G^*} x_{ij}]$ per feature. The to-be-augmented dataset is $D' = (D \setminus G) \cup G^*$, and negative-contribution samples ($ds'_i \approx 0$) are excluded (detailed algorithm in Appendix C.2).

4.2 Building the Constraint Table

Data-value quantification does not yet capture the inter-feature associations required for compliant categorical augmentation, so we construct the constraint table T as a rule base. Numerical features are discretized within their complete domain S_j (equal-width or equal-frequency binning (Chandrashekar & Sahin, 2014)) into M_j intervals retaining both statistical and business semantics. ANOVA and mutual information (MI) are normalized and fused into an association-strength score $S(A, B) = 0.6 \cdot F_{\text{norm}}(A, B) + 0.4 \cdot MI_{\text{norm}}(A, B)$. ANOVA captures monotonic associations and MI arbitrary dependence, with the weight favoring ANOVA since MI has large variance on small samples. Dual thresholds ($S \geq 0.1$, $p \leq 0.05$) select significant combinations, with a top-3 fallback if none qualify. For each selected joint feature group $\{B, (A_1, A_2, \dots)\}$, the conditional-probability distribution is computed:

$$P(B = b \mid A_1 = a_1, A_2 = a_2) = \frac{\text{Count}(A_1 = a_1, A_2 = a_2, B = b)}{\text{Count}(A_1 = a_1, A_2 = a_2)}. \tag{5}$$

All associations and conditional probabilities are organized into the constraint table T (detailed algorithm in Appendix C.3), hard-coding inter-feature dependencies and making categorical replacement traceable and business-compliant. Weight sensitivity is validated in Appendix E.

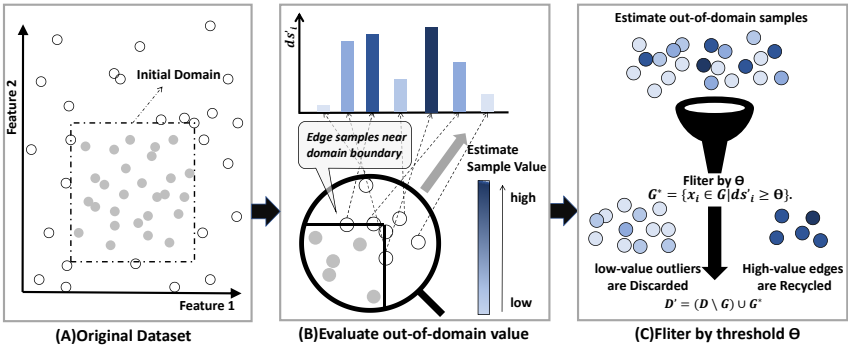

Figure 3: Dynamic domain correction: out-of-domain samples are gated by their contribution, so high-value edge samples extend the feature domain while near-zero-contribution samples are dropped, keeping scarce minority evidence that static bounds would discard.

4.3 Differentiated Augmentation

Given the value scores and the constraint table T, DIDA augments numerical then categorical features, producing the numerical dataset D_1 and then the complete augmented dataset D_2. The fusion weights φ_1, φ_2 of Section 3 are applied to b'_j and ds'_i, defining the cell-wise combined value $V_{ij} = \varphi_1 b'_j + \varphi_2 ds'_i$. Guided by this score, the principle of micro-perturbing high-value units and macro-perturbing low-value units is implemented along two dimensions, per-round magnitude and number of rounds. With thresholds $\theta_L < \theta_H$ (default $0.3/0.7$), high-value units ($V_{ij} \geq \theta_H$) receive multiple micro-rounds, medium-value units an intermediate number of rounds, and low-value units ($V_{ij} < \theta_L$) few macro-rounds. Candidate values are generated n_{ij} times within S_j via $f(b'_j, ds'_i)$ (Definition 5):

$$x_{ij}^{\text{new}} = \max\left(L_j,\ \min\left(x_{ij} + R \cdot |S_j| \cdot f(b'_j, ds'_i),\ U_j\right)\right), \qquad R \in \{-1, +1\}. \tag{6}$$

Duplicates are removed and a Cartesian product is taken over the m_{num} candidate sets. Retention per sample is regulated by ds'_i (quantile thresholds $h > l$, defaults $0.9/0.5$), keeping 3, 2, and 1 samples for high-, medium-, and low-contribution samples, respectively. When the total number of combinations is below the target retention count, all of them are kept, and otherwise sampling is without replacement, excluding combinations identical to the original sample. These thresholds and retention ratios are reference configurations, adjustable to a dataset's sample size, feature dimensionality, and class distribution. Categorical features and labels remain unchanged, yielding D_1 (detailed algorithm in Appendix C.4).

For categorical features, each sample is assigned values consistent with business logic, processing each categorical feature C in turn. The replacement probability follows Definition 5, $p_{\text{replace}}(i, C) = \min\big((1 - (\varphi_1 b'_C + \varphi_2 ds'_i))^p + \varepsilon,\ 1\big)$, and a random decision is made accordingly: if $u < p_{\text{replace}}(i, C)$ the value is replaced through the constraint table, otherwise $x_{i,C}^{\text{new}} \leftarrow x_{i,C}^{\text{orig}}$, with $u \sim U[0, 1]$. When a value is to be replaced, T is queried with the associated-feature values $A = (A_1 = a_1, A_2 = a_2, \dots)$ as index: if T holds the conditional distribution $P(C = c_k \mid A)$, sampling follows it. Otherwise a three-level fallback is triggered. The method relaxes the conditional constraint on the weakest relevant feature and retries the lookup, then uses the marginal distribution $P(C)$ as the sampling basis, and finally samples uniformly over the legal domain $\text{Uniform}(C)$, ensuring every query returns a legal value. Processed categorical features are concatenated with numerical features into $x_i^{\text{final}} = [\{x_{ij}^{\text{new}}\}_{j \in \text{Num}}, \{x_{iC}^{\text{new}}\}_{C \in \text{Cat}}]$, forming the augmented dataset $D_2 = D' \cup \{(x_i^{\text{final}}, y_i^{\text{final}})\}_{i=1}^{M}$ (detailed algorithm in Appendix C.5).

5 Experiments

5.1 Setup

We compare DIDA-augmented datasets against six augmentation baselines across ten heterogeneous classifiers to validate its effectiveness and robustness.

Datasets. Six public Kaggle tabular datasets (Heart, Alzheimers, Healthcare, Chronic Kidney, Teen, Urban Street) span medical health, mental health, and urban commerce; most are small-sample and half are class-imbalanced, matching the low-data regime DIDA targets. Basic statistics and preprocessing are in Appendix F (Table 10).

Baselines. Six mainstream methods cover oversampling (SMOTE, SMOTE-CDNN (Wang et al., 2023)) and deep generative models (CTGAN, TabDDPM, MDLM, TANDEM). All follow their original implementations with default hyperparameters and augment to the same target size (70% of the original training set), so downstream differences cannot be attributed to sample volume.

Downstream models. Ten classifiers spanning four mainstream learning paradigms are selected (tree ensembles, linear, instance-based, and tabular deep learning); the detailed list and hyperparameters appear in Appendix F.

Evaluation metrics. Generation quality is measured by distribution consistency (JS divergence, Wasserstein distance), sample diversity (information entropy), and feature coverage, while downstream performance uses AUC-ROC, AUC-PR, and F1, which suit class imbalance. Definitions are in Appendix F.

Experimental settings. A standardized pipeline uses stratified sampling with a $3 : 1 : 1$ train/validation/test split. Experiments repeat over 7 random seeds (10–70), reporting means and standard deviations, and significance is assessed with two-tailed paired t-tests at $\alpha = 0.05$ (Demšar, 2006). Full settings and DIDA's default hyperparameters are in Appendix F.

5.2 Main Results

Across six datasets and ten downstream classifiers, DIDA achieves the highest average AUC-ROC, AUC-PR, and F1 among all compared methods (Table 1), exceeding the strongest baseline by 1.94, 2.30, and 4.30 percentage points, respectively. The larger gains in AUC-PR and F1 are especially relevant for the imbalanced datasets considered here because both metrics are sensitive to minority-class errors. Since all augmentation methods generate the same number of synthetic samples, the improvement cannot be explained by a larger augmentation budget, indicating that performance depends on how this budget is distributed across samples and features. The generation-quality results support this interpretation (Figure 4): DIDA yields lower JS divergence and Wasserstein distance (Arjovsky et al., 2017) than all six baselines across the six datasets while achieving higher information entropy and feature coverage. This balance matters because greater diversity can easily come at the cost of distributional fidelity. DIDA controls this trade-off by assigning perturbation strength according to feature-level SHAP importance and sample-level Shapley contribution: high-value sample-feature units are perturbed conservatively to preserve task-relevant structure, whereas lower-value units are allowed to vary more broadly and contribute more to feature-space exploration. As a result, diversity increases without a comparable rise in distributional discrepancy. DIDA reaches full feature coverage on four datasets and remains close to full coverage on the other two while maintaining markedly lower divergence, showing that the additional coverage is not obtained by indiscriminately expanding the synthetic distribution. The categorical and domain-correction components further contribute to these results. Categorical values are sampled through the conditional constraint table rather than independently, keeping generated combinations consistent with feature dependencies observed in the data and reducing unsupported or invalid combinations, while dynamic domain correction retains high-value boundary samples that a static filter may remove, allowing informative marginal regions to remain available during augmentation. These mechanisms expand the usable feature space without weakening the structural constraints of the original data. The downstream gains are also not confined to a single predictor family (Figure 5). DIDA improves tree-based, linear, instance-based, and neural classifiers, although the magnitude varies: tree ensembles show the strongest improvements, whereas neural models respond more heterogeneously. A plausible explanation is that tree models rely on threshold-based partitions and are therefore more sensitive to synthetic samples that distort local feature distributions. By preserving high-value features while expanding lower-value regions, DIDA provides additional training samples with less local distributional distortion. Linear and instance-based models also benefit, although their gains are generally more moderate, while MLP and TabNet respond differently to the added coverage. DIDA does not produce a uniform performance shift; its effect depends on how each classifier exploits the augmented feature space. The downstream and generation-quality results point to the same mechanism: under the same augmentation budget, DIDA benefits from allocating perturbations according to data value rather than treating all samples and features uniformly, preserving high-value structure, exploring lower-value regions more aggressively, and constraining categorical changes through ob-

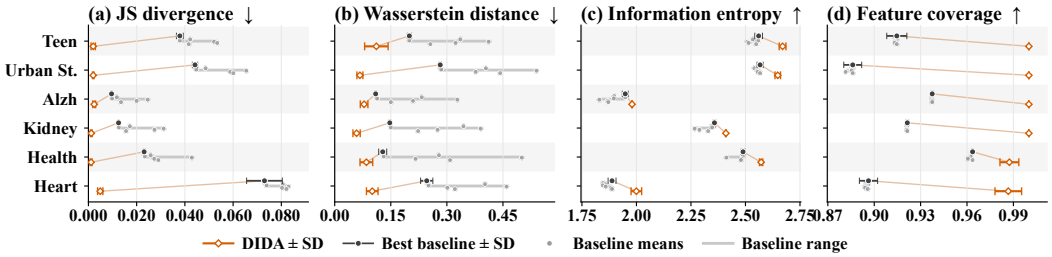

Figure 4: Generation-quality comparison across methods and datasets: DIDA attains lower JS divergence and Wasserstein distance, higher entropy, and higher feature coverage than the six baselines, showing value-aware placement preserves distribution consistency while expanding diversity.

Table 1: Downstream performance across methods. Avg std is the cross-classifier AUC-ROC standard deviation, and the best columns count ties, so their sums may exceed 60.

Method	Avg AUC-ROC	Avg AUC-PR	Avg F1	F1 best	AUC-ROC best	Avg std
DIDA	80.72	78.24	72.06	42	40	3.49
SMOTE	78.78	75.94	67.76	6	7	3.47
CTGAN	77.22	72.04	62.44	5	2	4.34
SMOTE-CDNN	78.72	75.84	67.48	7	9	3.42
TabDDPM	74.00	65.57	59.97	5	0	3.31
MDLM	77.95	72.93	62.32	2	3	4.45
TANDEM	77.39	71.98	62.73	1	1	4.37

served feature dependencies. DIDA can thus improve diversity and coverage while remaining closer to the original data distribution, and these gains translate into stronger downstream performance in the low-data, high-constraint setting considered here. Complete results and sensitivity analyses are provided in Appendix F.7 and Appendix E.

5.3 Ablation Study

As shown in Figure 6, removing any major component degrades DIDA on AUC-ROC, AUC-PR, or F1 (Appendix F, Table 13), but the differences reveal distinct roles rather than a uniform contribution from each module. The largest loss occurs when value-driven count regulation is replaced by fixed retention. Since the two value coordinates are still available in this variant, the degradation shows that estimating data value is not sufficient by itself: the value estimate must also determine how augmentation capacity is allocated across samples. Count regulation therefore acts as the mechanism that converts sample contribution into differentiated augmentation, preventing informative and redundant samples from receiving the same retention budget and thereby preserving the intended value-aware allocation. The two value signals play complementary roles. Removing feature importance causes a larger decline than removing sample contribution, suggesting that feature-level importance is more critical for controlling perturbation magnitude because changes to influential features directly affect task-relevant structure. Sample contribution instead differentiates how individual observations participate in augmentation. Their combination therefore enables joint control at the sample–feature level, which is the key distinction of DIDA's two-dimensional value representation. This complementarity also explains why neither value coordinate can fully substitute for the other. Removing the constraint table mainly reduces F1, indicating that categorical changes require structural control in addition to value-aware perturbation. By conditioning replacements on observed cross-feature dependencies, the table suppresses unsupported combinations and preserves useful categorical structure instead of merely increasing categorical variation. The lower variation of the complete model across random seeds further shows that these modules are complementary, not redundant. Value estimation identifies what should be protected; count regulation decides where that protection is spent; and the constraint table governs how categorical changes are introduced. Their joint effect explains why the complete DIDA is both more accurate and more stable than any single ablated variant.

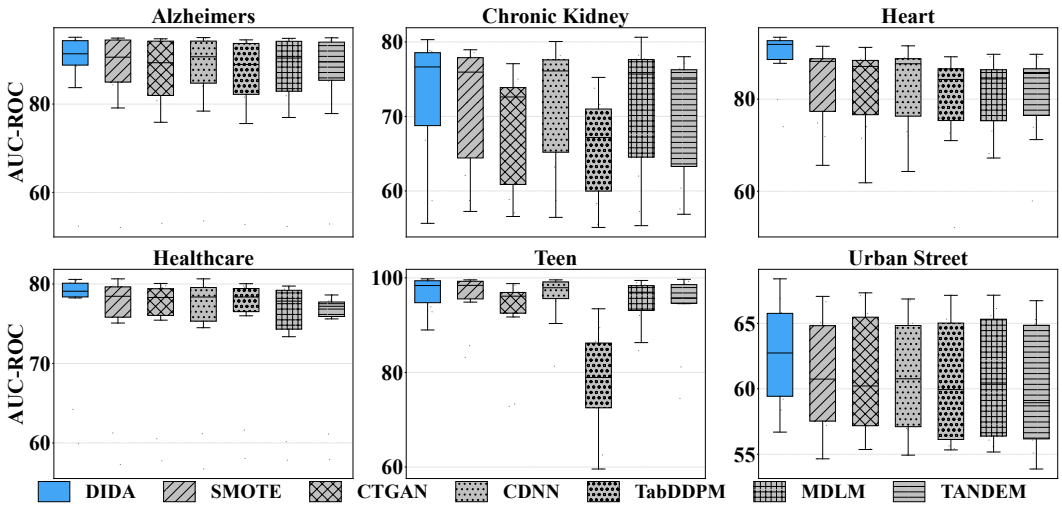

Figure 5: AUC-ROC box plots across datasets and methods: DIDA's median lies above the baselines with tighter whiskers across nearly every dataset, indicating the gains are stable across seeds rather than an artifact of a favorable one.

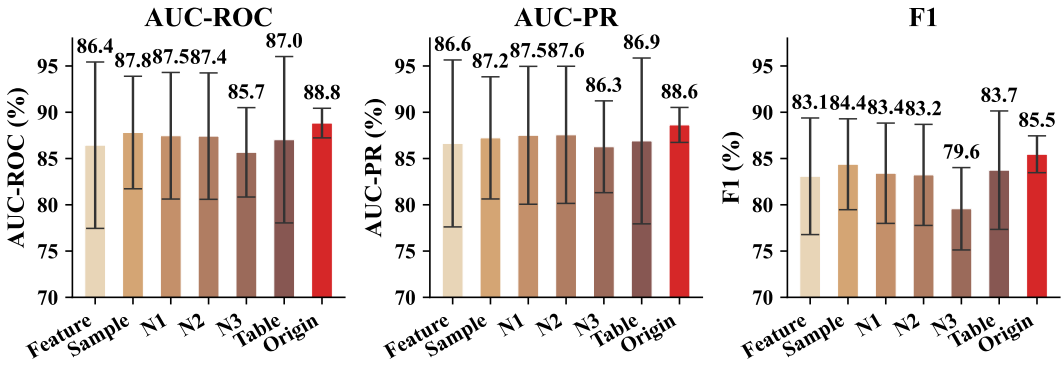

Figure 6: Ablation study on Heart: removing the joint-drive mechanism, the dynamic count regulation, or the constraint table each degrades all three metrics, confirming every module contributes and the full model is the most stable across seeds.

6 Conclusion

We introduced DIDA to recast tabular data augmentation in low-data, high-constraint settings as a value-aware allocation problem rather than uniform sample synthesis. Its central principle is that augmentation freedom should be distributed according to the task value of each sample-feature unit: task-relevant structure should be protected, whereas less critical regions can be explored more aggressively. The KKT-derived value–perturbation relationship provides a theoretical basis for this principle, while dynamic domain correction and constraint-guided categorical augmentation make it operational and traceable. The empirical evidence supports a broader conclusion: effective augmentation depends not simply on how much synthetic data is generated, but on where perturbation is introduced and which structural dependencies are preserved. DIDA is currently limited by its dependence on base-model value estimates and by a constraint table learned from observed combinations, which may omit valid but unseen dependencies. Future work will investigate model-robust and online data valuation, richer constraint acquisition from domain knowledge, and the extension of value-aware augmentation beyond tabular data.

AI USE STATEMENT

In this work, we used generative AI tools for language polishing, manuscript refinement, and literature search. We have not used AI for core research ideas, experimental implementation, or the initial draft. All AI-assisted content has been carefully reviewed by the authors, who take full responsibility for the final work.

ETHICS STATEMENT

This work uses only public benchmark datasets and does not involve human subjects, personally identifiable information, or the release of sensitive data. The proposed method is designed for constraint-aware tabular data augmentation in regulated domains and does not introduce fairness or privacy risks beyond those of the underlying data.

REPRODUCIBILITY STATEMENT

To support reproducibility, we release an anonymized implementation of DIDA at `https://anonymous.4open.science/r/DIDA-F8C5`. The complete experimental protocol, including dataset preprocessing, data splits, baseline settings, downstream-model hyperparameters, and DIDA configurations, is documented in Appendix F. All experiments are repeated over 7 fixed random seeds (10–70), and pseudocode for the core algorithms is provided in Appendix C.

REFERENCES

Sercan Ö. Arik and Tomas Pfister. TabNet: Attentive interpretable tabular learning. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 35, pp. 6679–6687, 2021. doi: 10.1609/aaai.v35i8.16826.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70, pp. 214–223, 2017. doi: 10.48550/arXiv.1701.07875.

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, 2024. doi: 10.1109/tnnls.2022.3229161.

Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001. doi: 10.1023/A:1010933404324.

Gavin C. Cawley and Nicola L. C. Talbot. On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research (JMLR)*, 11:2079–2107, 2010. URL `https://jmlr.org/papers/v11/cawley10a.html`.

Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014. doi: 10.1016/j.compeleceng.2013.11.024.

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.

Irene Y. Chen, Emma Pierson, Sherri Rose, Shalmali Joshi, Kadija Ferryman, and Marzyeh Ghassemi. Ethical machine learning in healthcare. *Annual Review of Biomedical Data Science*, 4: 123–144, 2021. doi: 10.1146/annurev-biodatasci-092820-114757.

Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 785–794, 2016. doi: 10.1145/2939672.2939785.

Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research (JMLR)*, 7:1–30, 2006. URL `https://jmlr.org/papers/v7/demsar06a.html`.

Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, 2001. doi: 10.1214/aos/1013203451.

Amirata Ghorbani and James Zou. Data shapley: Equitable valuation of data for machine learning. In *International Conference on Machine Learning (ICML)*, volume 97, pp. 2242–2251, 2019. doi: 10.48550/arXiv.1904.02868.

Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, 2022. doi: 10.48550/arXiv.2207.08815.

Haibo He and Edwardo A. Garcia. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 21(9):1263–1284, 2009. doi: 10.1109/tkde.2008.239.

Tokimasa Isomura, Ryotaro Shimizu, and Masayuki Goto. Generating realistic synthetic tabular data with integrated LLM and diffusion models. *Neurocomputing*, 655:131371, 2025. doi: 10.1016/j.neucom.2025.131371.

Mohammad Hossein Jarrahi, Ali Memariani, and Shion Guha. The principles of data-centric AI. *Communications of the ACM*, 66(8):84–92, 2023. doi: 10.1145/3571724.

Ruoxi Jia, David Dao, Boxin Wang, Frances Ann Hubis, Nezihe Merve Gürel, Bo Li, Ce Zhang, Costas J. Spanos, and Dawn Song. Efficient task-specific data valuation for nearest neighbor algorithms. *Proceedings of the VLDB Endowment (PVLDB)*, 12(11):1610–1623, 2019. doi: 10.14778/3342263.3342637.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pp. 3146–3154, 2017.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning (ICML)*, volume 70, pp. 1885–1894, 2017. doi: 10.48550/arXiv.1703.04730.

Akim Kotelnikov, Dmitry Baranchuk, Ivan Rubachev, and Artem Babenko. TabDDPM: Modelling tabular data with diffusion models. In *International Conference on Machine Learning (ICML)*, volume 202, pp. 17564–17579, 2023. doi: 10.48550/arXiv.2209.15421.

Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pp. 4765–4774, 2017. doi: 10.48550/arXiv.1705.07874.

Scott M. Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1):56–67, 2020. doi: 10.1038/s42256-019-0138-9.

Erel Naor and Ofir Lindenbaum. Hybrid autoencoders for tabular data: Leveraging model-based augmentation in low-label settings. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 105365–105384, 2025. doi: 10.52202/085713-3178.

OpenAI. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. doi: 10.48550/arXiv.2303.08774.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research (JMLR)*, 12:2825–2830, 2011. URL `https://jmlr.org/papers/v12/pedregosa11a.html`.

Alvin Rajkomar, Jeffrey Dean, and Isaac Kohane. Machine learning in medicine. *New England Journal of Medicine*, 380(14):1347–1358, 2019. doi: 10.1056/nejmra1814259.

Alex X. Wang, Stefanka S. Chukova, and Binh P. Nguyen. Synthetic minority oversampling using edited displacement-based k-nearest neighbors. *Applied Soft Computing*, 148:110895, 2023. doi: 10.1016/j.asoc.2023.110895.

Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Modeling tabular data using conditional GAN. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pp. 7333–7343, 2019. doi: 10.48550/arXiv.1907.00503.

Jinsung Yoon, Sercan Ö. Arik, and Tomas Pfister. Data valuation using reinforcement learning. In *International Conference on Machine Learning (ICML)*, volume 119, pp. 10842–10851, 2020. doi: 10.48550/arXiv.1909.11671.

Daochen Zha, Zaid Pervaiz Bhat, Kwei-Herng Lai, Fan Yang, Zhimeng Jiang, Shaochen Zhong, and Xia Hu. Data-centric artificial intelligence: A survey. *ACM Computing Surveys*, 57(5):1–42, 2025. doi: 10.1145/3711118.

Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations (ICLR)*, 2018. doi: 10.48550/arXiv.1710.09412.

A Related Work Extended

This appendix consolidates the two related-work lines surveyed in Section 2, sample-level data valuation and constraint-aware data generation, whose technical details and comparisons are presented below.

A.1 Data Valuation

This subsection details and compares the data-valuation methods mentioned in Section 2. Data valuation is the theoretical prerequisite of DIDA's differentiated augmentation mechanism, and its implementation details are concentrated here for the reader's convenience.

A.1.1 Data Shapley

Ghorbani & Zou (2019) introduced the Shapley value from cooperative game theory into machine-learning data valuation, defining from an axiomatic perspective the fair marginal contribution of each training sample to model performance. For a dataset D of N training samples, the Data Shapley value of sample x_i is defined in Eq. (1) of Section 3, where $U(S)$ is the performance (e.g., AUC-ROC) on a validation set of a model trained on subset S. This definition satisfies the four axioms of Shapley values, namely efficiency ($\sum_i ds_i = U(D) - U(\varnothing)$), symmetry (samples with equal contributions receive equal valuations), null player (zero-contribution samples receive zero valuation), and additivity (valuations over multiple datasets compose linearly), providing formal guarantees for the valuation results at the axiomatic level. However, exact computation of Data Shapley requires enumerating all 2^N subsets and training a model for each, costing $O(2^N \cdot T)$ (where T is single-training time), which is infeasible for $N > 20$.

A.1.2 DVRL

DVRL, proposed by Yoon et al. (2020), uses reinforcement learning to train a valuation network (the Data Value Estimator), modeling the valuation of each training sample as a Markov decision process: the valuation network outputs the probability that a sample is selected into the training subset, the performance of the model trained on that subset serves as the reward signal, and the valuation-network parameters are optimized via policy gradients. DVRL learns the valuation function end-to-end through reinforcement learning, avoiding the exponential cost ($O(2^N)$) of exact Shapley values, and completes valuation within an acceptable time on medium-scale datasets with $N > 1000$. The cost of this method is that the valuation network requires independent training, introducing additional hyperparameters (learning rate, network structure, etc.), and the valuation result depends on the choice of the validation set: if the validation set does not represent the true data distribution, the valuation will exhibit systematic bias.

Moreover, DVRL's valuation scores lack the axiomatic guarantees of Shapley values (efficiency, symmetry, null player, additivity), and its economic interpretability is weaker than Shapley-based methods.

A.1.3 INFLUENCE FUNCTIONS

Koh & Liang (2017) use influence functions from robust statistics to estimate the first-order approximation of the perturbation to model parameters caused by removing a single training sample: $I(x_i) = -H^{-1}\nabla L(x_i, \theta^*)$, where H is the Hessian matrix of the model over the training set and $\nabla L(x_i, \theta^*)$ is the gradient of sample x_i at the optimal parameters θ^*. This method requires no retraining: after a single training pass yielding θ^*, all sample influence scores are computed via matrix-vector products, costing $O(Np^2 + p^3)$ (p being the number of model parameters). It relies on three strong assumptions: the loss is twice differentiable at θ^* with nonsingular H, the effect of removing a single sample on parameters is well approximated by a first-order Taylor expansion, and the inverse Hessian can be stably estimated via conjugate gradients. For tree-ensemble models on tabular data (XGBoost, LightGBM), the loss is not everywhere differentiable and the parameter count is enormous (all tree leaf-node weights), making influence functions difficult to apply directly, which is the core reason DIDA chooses the KNN-Shapley approximation over influence functions.

A.1.4 WHY KNN-SHAPLEY

As noted in Section 2, the fundamental limitation of existing data-valuation methods is that they only answer the diagnostic question of which samples matter, with downstream applications concentrated on data selection, i.e., identifying and removing low-value samples (Ghorbani & Zou, 2019; Yoon et al., 2020; Koh & Liang, 2017). In small-sample scenarios where data are already scarce, directly discarding samples is unacceptable, and more critically, none of the existing valuation methods addresses the generative question of how to augment data differentially according to value. DIDA establishes, precisely at this intersection, a quantitative mapping from value scores to augmentation decisions.

Across the three dimensions above, DIDA adopts KNN-Shapley (Jia et al., 2019) as the sample-contribution estimator: its $O(N \log N)$ complexity is deployable, it inherits the four Shapley axioms, and its model-agnostic decision-space embedding fits tree-ensemble base models; its limitations (dependence on the nearest-neighbor assumption, sensitivity to the distance metric) and their mitigation are discussed in Section 4.1.

A.2 CONSTRAINT-AWARE GENERATION

This subsection supplements the technical details of the constraint-aware data generation methods mentioned briefly in Section 2, covering the comparison of three technical routes: soft-penalty constraints, post-processing rule filtering, and DIDA's constraint table.

A.2.1 OVERVIEW

The domain constraints of tabular data can be summarized into three categories. Semantic constraints are logic rules among features defined by domain knowledge, for example blood pressure systolic $>$ diastolic, and gender $=$ male implying pregnancy indicator $=$ not applicable. Statistical constraints are feature conditional distributions estimated from training data, for example the conditional-probability constraint of $P(x_{(j)} \mid x_{(k)})$. Domain constraints are physical or business boundaries on feature values, for example age $\in [0, 120]$, blood glucose $\in [0, 30]$ mmol/L. The goal of constraint-aware data augmentation is to ensure, during synthetic-sample generation, that all augmented samples simultaneously satisfy the above three classes of constraints, avoiding synthetic data that are physically impossible or business non-compliant.

A.2.2 SOFT PENALTY METHODS

Soft-penalty methods introduce a penalty term for constraint violation into the loss function of a deep generative model, so that the model implicitly learns constraint satisfaction during training: $L_{\text{total}} = L_{\text{gen}} + \lambda \cdot L_{\text{constraint}}$. CTGAN (Xu et al., 2019) and TabDDPM (Kotelnikov et al., 2023) do not define an explicit constraint loss, but their conditional generation mechanisms (conditional

vector + training-by-sampling) provide a degree of statistical constraint satisfaction through implicit fitting of the conditional distributions in the training data. The advantage of such methods is that no explicit modeling of all constraint rules is required, and the model automatically learns inter-feature dependencies from data. However, constraint satisfaction is only statistical: for rare feature combinations the model may generate constraint-violating samples, and the degree of satisfaction depends on the model's convergence quality and the coverage of the constraint space by the training data. In high-regulation settings (medical diagnosis, credit approval), statistical approximation is usually insufficient for compliance review.

A.2.3 POST-HOC FILTERING

Post-processing methods filter or correct synthetic samples through a predefined business-rule base after generation. A typical flow first generates a batch of candidate samples unconditionally or weakly conditionally, then checks constraint violations sample by sample, and finally discards violating samples or applies rule-driven correction to violating features. TANDEM (Naor & Lindenbaum, 2025) relies on data-distribution constraints after latent-space transformation but provides no explicit rule-enforcement mechanism. The advantage of such methods is that rule enforcement is deterministic and constraint satisfaction can be strictly verified. The core problem is that corrected samples may deviate from the original data distribution, introducing distribution shift: for example, correcting only the occupation field of a sample violating an age-occupation constraint may break the joint-distribution consistency among age-income-occupation in that sample. Moreover, the maintenance cost of large rule bases grows superlinearly with the number of constraints.

A.2.4 CONSTRAINT-TABLE METHOD

DIDA's constraint-table method takes a technical route different from the above two: a nonparametric conditional probability table (CPT) hard-codes inter-feature dependencies. For each categorical feature C, the constraint table stores, under each combination of relevant features (selected by the ANOVA–MI fused score), the legal value set of C and the corresponding conditional probabilities. During augmentation, categorical-feature replacement is completed by deterministic table lookup: given the current sample's relevant-feature value combination, the legal value list of C is retrieved from the constraint table, and a replacement value is sampled by conditional-probability-weighted sampling. When no match exists for the current value combination (an out-of-table query), a three-level fallback guarantees a legal return value (Algorithm 5). This scheme has four advantages. Constraint satisfaction is deterministic: every legal query returns a feature-value combination verified in the training data, independent of model-training convergence quality. The three-level fallback guarantees a 100% query-return rate, so constraint conflicts never interrupt generation. Constraint-table construction is decoupled from the augmentation process, since the table is built once and reused across multiple augmentation rounds with constant computational overhead. Compared with post-processing correction, the table is embedded inside the augmentation process, avoiding the distribution shift of the generate-correct pipeline.

A.2.5 METHOD COMPARISON

Table 2 summarizes the qualitative capabilities of the main tabular data augmentation methods discussed in Section 2, complementing the constraint-oriented survey above.

Two dimensions of Table 2 are worth reading together. The generative models (CTGAN, TabDDPM, MDLM) earn three stars in categorical handling and constraint awareness, yet their small-sample suitability stays at one or two, since mode collapse and dependence on pretrained weights are exactly the failure modes that scarce data amplifies. The SMOTE family sits at the opposite corner: robust on small data but confined to linear interpolation, with categorical support that never exceeds two stars. DIDA's uniform three-star profile is therefore structural rather than coincidental: value-driven differentiation supplies the small-sample adaptability the generative models lack, while the constraint table provides the deterministic guarantees the SMOTE family never attempts. The one limitation we list, that the value of generated data may fluctuate, is inherited from the KNN-Shapley estimator and is quantified in Appendix E.

Table 2: Comparison of main data augmentation methods. Interp. denotes interpretability, Cat. denotes categorical augmentation, Small denotes small-data suitability, and Cons. denotes constraint awareness.

Method	Interp.	Small	Cat.	Cons.	Strengths	Limitations
SMOTE	⋆⋆⋆	⋆	⋆	⋆	Simple, efficient, easy to deploy	Linear-only, misses nonlinearities, noisy in high dims
SMOTE-CDNN	⋆⋆	⋆	⋆⋆	⋆⋆	Nonlinear mapping, more plausible samples	Neighbor-constrained, overfits on small data
CTGAN	⋆	⋆	⋆⋆⋆	⋆⋆⋆	Mixed features, strong diversity and fit	Unstable, mode collapse, poor small-sample generalization
TabDDPM	⋆	⋆	⋆⋆⋆	⋆⋆⋆	Strong high-order modeling, high quality	High compute, slow inference, overfits small data
MDLM	⋆	⋆⋆	⋆⋆⋆	⋆⋆⋆	LLM priors, strong semantics and transfer	Pretrained-weight dependence, costly, weak control
TANDEM	⋆⋆	⋆⋆⋆	⋆⋆⋆	⋆⋆	Interpretable, robust to small data, high fidelity	Hyperparameter tuning, lower diversity
DIDA	⋆⋆⋆	⋆⋆⋆	⋆⋆⋆	⋆⋆⋆	Value-driven, constraint-aware, small-sample adaptive	Value of generated data may fluctuate

Table 3: Core symbols used throughout the paper.

Symbol	Definition/Formula	Description
D	$D = \{(x_i, y_i)\}_{i=1}^{N}$	Tabular dataset
x_i	$x_i = [x_i^{(\text{num})}; x_i^{(\text{cat})}]$	Feature vector of sample i
y_i	$y_i \in \{1, \ldots, K\}$	Class label
m	$m = m_{\text{num}} + m_{\text{cat}}$	Total number of features
b'_j	$b'_j \in [0, 1]$	Normalized global feature importance (SHAP mean)
ds'_i	$ds'_i \in [0, 1]$	Normalized sample contribution (KNN-Shapley)
$\text{Consist}(x_i^{\text{new}}, x_i)$	Eq. (2)	Semantic-consistency measure
S_j	$S_j = [L_j^{(\text{stat})}, U_j^{(\text{stat})}] \cap [L_j^{(\text{biz})}, U_j^{(\text{biz})}]$	Complete domain of feature j
δ_{ij}	$\delta_{ij} \in \mathbb{R}$	Perturbation intensity of feature j of sample i
R_{ij}	$R_{ij} \in \{-1, +1\}$	Random perturbation direction
ε_0	$\varepsilon_0 \in (0, 1)$	Semantic-consistency threshold
δ_{ij}^*	$\delta_{ij}^* \propto \sqrt{1 - ds'_i}/b'_j$	Optimal perturbation intensity (Thm. 1)
$f(b'_j, ds'_i)$	$f(b'_j, ds'_i) = t_0[1 - (\varphi_1 b'_j + \varphi_2 ds'_i)]^p + \varepsilon$	Differentiated perturbation intensity
t_0	$t_0 > 0$	Maximum-perturbation anchor
φ_1, φ_2	$\varphi_1 + \varphi_2 = 1,\ \varphi_1, \varphi_2 \in [0, 1]$	Feature/sample fusion weights
p	$p \geq 1$	Decay coefficient
ε	$\varepsilon > 0$	Minimum perturbation threshold
p_{replace}	$p_{\text{replace}} = \min([1 - (\varphi_1 b'_C + \varphi_2 ds'_i)]^p + \varepsilon,\ 1)$	Categorical-feature replacement probability
T	$T : P(B = b \mid A = a)$	Conditional-probability constraint table

B Core Symbol Table

Table 3 summarizes the core symbols used across Section 3. One-off intermediate quantities are not listed.

Most of Section 3 can be read with only the first two data-value rows in mind: the normalized importances b'_j and contributions ds'_i are the two coordinates of the value that governs every decision in the framework, and the rows that follow are its concrete consequences, where δ_{ij}^* instantiates that

Algorithm 1 KNN-Shapley sample contribution computation for tree ensembles.

Input: pretrained tree-ensemble model f_{base} (with H decision trees)
Output: normalized sample contributions $\{ds'_i\}_{i=1}^{N}$

for each sample x_i in D **do**
  for each tree $h = 1$ to H **do**
    $l_{i,h} \leftarrow f_{\text{base}}.\text{tree}[h].\text{apply}(x_i)$ // leaf index
  end for
  $l_i \leftarrow [l_{i,1}, l_{i,2}, \ldots, l_{i,H}]$ // decision-space embedding
end for
$K \leftarrow \lfloor \log_2 N \rfloor$ // adaptive neighbor count
for each $h = 1$ to H **do**
  $w_h^t \leftarrow (1/h) / \sum_{k=1}^{H} (1/k)$ // tree decay weight
end for
for each sample i in D **do**
  $ds_i \leftarrow 0$
  for each sample j in D, $j \neq i$ **do**
    $d(i,j) \leftarrow \sum_{h=1}^{H} w_h^t \cdot \mathbb{I}(l_{i,h} \neq l_{j,h}) + \mathbb{I}(y_i \neq y_j) \cdot \mu$
  end for
  for each class $c \in \{1, \ldots, Y\}$ **do**
    $N_c \leftarrow$ top K/Y samples of class c by $d(i, \cdot)$ ascending
  end for
  $N_i \leftarrow$ union of N_c for all c // directed neighbor set
  sort N_i by $d(i, \cdot)$ ascending, assign rank $r = 1, 2, \ldots, K$
  for each neighbor j_r in N_i (rank r) **do**
    $w_r^s \leftarrow (1/r) / \sum_{k=1}^{K} (1/k)$ // Shapley-aligned weight
    $ds_i \leftarrow ds_i + w_r^s \cdot [\mathbb{I}(y_{j_r} = y_i) - \mathbb{I}(y_{j_r} \neq y_i)]$
  end for
  $ds_i \leftarrow \text{clip}(ds_i, -1, 1)$
end for
$ds_{\min} \leftarrow \min_i ds_i$
$ds_{99} \leftarrow \text{Percentile}_{99}(ds_i)$
for each $i = 1$ to N **do**
  $ds'_i \leftarrow \text{clip}((ds_i - ds_{\min})/(ds_{99} - ds_{\min}), 0, 1)$
end for
return $\{ds'_i\}_{i=1}^{N}$

value in the numerical branch, p_{replace} in the categorical branch, and the fused intensity $f(b'_j, ds'_i)$ is the operational quantity the augmentation algorithms actually compute. Listing each definition beside its formula lets the derivations in Section 3 and the pseudocode in Appendix C be checked against a single source rather than reconstructed from prose.

C Algorithms

C.1 KNN-Shapley Computation

Algorithm 1 converts an exact Data-Shapley oracle, costing $O(2^N)$ and therefore infeasible, into an $O(N \log N)$ approximation. Every sample is first embedded into the *tree decision space*: the pretrained tree-ensemble model f_{base} (XGBoost (Chen & Guestrin, 2016) or LightGBM (Ke et al., 2017)) assigns x_i to a leaf $l_{i,h}$ of each tree h, yielding the H-dimensional embedding $l_i = [l_{i,1}, \ldots, l_{i,H}]$. Tree ensembles are used because they generalize strongly on small- and medium-scale tabular data and consume numerical and encoded categorical features directly (Grinsztajn et al., 2022). The same pretrained model produces the feature-level SHAP importance b'_j (Section 4.1), so the two value coordinates share a single feature representation.

Similarity is then measured by a weighted Hamming distance $d(i,j) = \sum_h w_h^t \, \mathbb{I}(l_{i,h} \neq l_{j,h}) + \mathbb{I}(y_i \neq y_j)\, \mu$, and directed sampling over a per-class neighbor budget replaces the all-pairs comparison, reducing the quadratic cost to $O(N \log N)$ (Jia et al., 2019). The valuation-method comparison is in Appendix A. Three choices are the key engineering decisions. The adaptive neighborhood $K = \lfloor \log_2 N \rfloor$ keeps the search small on scarce data without collapsing on larger datasets. The tree decay weight $w_h^t \propto 1/h$ discounts deeper trees, whose leaf assignments overfit individual samples and contribute noisy distances. The neighbor weight $w_r^s \propto 1/r$ mirrors the harmonic weighting of the Shapley formula, so closer same-class neighbors dominate the score.

The final normalization, clipping to $[-1, 1]$ and then rescaling against the 99th percentile, turns a raw, scale-dependent count into the bounded $[0, 1]$ quantity ds'_i that Theorem 1 and the augmentation branch both consume. The resulting score locates the high-value rare edge samples that a

Algorithm 2 Dynamic domain correction.

Input: original dataset D, initial complete domain $\{S_j\}_{j=1}^{m}$ (Def. 4), normalized sample contributions $\{ds'_i\}_{i=1}^{N}$, threshold θ (default 0.2)
Output: corrected complete domain $\{S_j\}_{j=1}^{m}$, to-be-augmented dataset D'

$G \leftarrow \{x_i \in D \mid \exists j,\ x_{ij} \notin S_j\}$ // samples violating the domain boundary
$G^* \leftarrow \{x_i \in G \mid ds'_i \geq \theta\}$ // select high-value anomalous samples
if $G^* \neq \varnothing$ **then**
  for each feature $j = 1$ to m **do**
    $L_j^* \leftarrow \min_{x_i \in G^*} x_{ij}$
    $U_j^* \leftarrow \max_{x_i \in G^*} x_{ij}$
    $L_j \leftarrow \min(L_j, L_j^*)$
    $U_j \leftarrow \max(U_j, U_j^*)$
    $S_j \leftarrow [L_j, U_j]$ // extend the complete domain
  end for
end if
$D' \leftarrow (D \setminus G) \cup G^*$ // recover high-value marginal samples
return S_j, D'

static complete domain (Definition 4) would otherwise discard, which is precisely what makes the dynamic correction of Algorithm 2 necessary.

C.2 Dynamic Domain Correction

For completeness, Definition 4's initial complete domain reduces to the pure statistical boundary when no business boundary is specified. For $n < 30$ the business boundary is recommended to take precedence.

The purpose of Algorithm 2 is to recover data, not merely to clean it. A boundary built once on the raw distribution (Definition 4) is necessarily myopic: a sample can fall outside the statistical boundary because it is a genuine, high-value edge case whose decision-boundary information the model has not yet absorbed, not because it is noise. If such samples are silently dropped before augmentation, the pipeline discards precisely the rare minority evidence that augmentation exists to produce, and every downstream synthetic point inherits an artificially narrowed feasible region.

The algorithm therefore partitions the out-of-domain set G by contribution, not by distance. Samples with $ds'_i \geq \theta$ (default 0.2) are treated as recoverable edge cases: they are kept in the augmentation set D', and their per-feature extrema L_j^*, U_j^* are folded into the complete domain, so $S_j \leftarrow [\min(L_j, L_j^*), \max(U_j, U_j^*)]$ widens only where high-value evidence actually lies. Negative- or near-zero-contribution samples ($ds'_i \approx 0$) are excluded by the same threshold. This resolves the small-data tension raised in Section 4.1: deleting anomalies would erase exactly the scarce minority evidence that augmentation exists to preserve, so domain correction is gated by value, not by rule.

C.3 Constraint-Table Construction

Before associations are mined, numerical features are discretized inside their complete domain S_j: near-uniform features use equal-width binning and skewed features use equal-frequency binning (Chandrashekar & Sahin, 2014), mapping continuous values to M_j intervals that preserve both statistical and business semantics. This gives numerical and categorical features a unified symbolic representation, so a single association-mining pass covers the entire feature set.

Algorithm 3 then combines ANOVA and mutual information because neither is trustworthy on tabular data alone: the F-test captures monotonic association but misses arbitrary nonlinear dependence, while mutual information detects general dependence but is unstable on small samples (hence the ANOVA-favoring weights $0.6/0.4$). The weighted fused score with its joint threshold ($S \geq 0.1$ and $p \leq 0.05$) suppresses spurious feature pairs before they reach the table, and the fallback to the top-scoring candidates guarantees that even a feature with no statistically significant association still receives a table entry, so the categorical sampler downstream never encounters an empty lookup.

Each selected joint group $\{B, (A_1, A_2, \dots)\}$ contributes a nonparametric conditional-probability table $P(B = b \mid A_1 = a_1, A_2 = a_2)$, hard-coding the inter-feature dependency. This contrasts with the soft-penalty terms of deep generative baselines: a lookup table makes every categorical replacement traceable to a specific observed combination and business-compliant by construction, which

Algorithm 3 ANOVA–MI constraint-table construction.

Input: training set D, significance $\alpha = 0.05$, ANOVA weight $\omega_A = 0.6$, MI weight $\omega_M = 0.4$
Output: constraint table T

$\tilde{D} = \text{discretize_numerical}(D)$ // see Section 4.2
$T = \text{empty map}$
for each target feature B in categorical($\tilde{D}$) **do**
  candidates = []
  for each feature $A \neq B$ **do**
    $F_{\text{stat}}, p_{\text{val}} = \text{ANOVA_F_test}(B, A)$
    $mi = \text{MutualInformation}(B, A)$
    $S = \omega_A \cdot F_{\text{norm}} + \omega_M \cdot mi_{\text{norm}}$ // fused score
    candidates.append((A, S, p_{val}))
  end for
  $\mathcal{A} = \{A \mid (A, S, p) \in \text{candidates},\ S \geq 0.1 \text{ AND } p \leq \alpha\}$ // joint relevant-feature set
  if $\mathcal{A} = \varnothing$ **then**
    $\mathcal{A} =$ top-3(candidates, by S) // sparse-combination fallback
  end if
  for each $(a_1, a_2, \ldots, b)$ in observed_joint_combinations($\tilde{D}$, $\mathcal{A}$, B) **do**
    $P = \text{Count}(B = b, A_1 = a_1, A_2 = a_2, \ldots)/\text{Count}(A_1 = a_1, A_2 = a_2, \ldots)$
    $T[(B, a_1, a_2, \ldots)] = T[(B, a_1, a_2, \ldots)] \cup \{(b, P)\}$
  end for
end for
return T

is the auditability requirement of Section 4.2. The fused-score weights are validated separately in Appendix E.

C.4 Numerical-Feature Augmentation

Algorithm 4 is where Theorem 1 becomes procedure. The per-cell value $V_{ij} = \varphi_1 b'_j + \varphi_2 ds'_i$ governs two quantities at once: how many perturbations to draw ($n_{ij} \in \{1, 3, 5\}$) and how many to retain ($M_i \in \{1, 2, 3\}$), so high-value cells ($V_{ij} \geq \theta_H$, default 0.7) receive multiple micro-rounds that explore their neighborhood finely and are preserved generously, while low-value cells ($V_{ij} < \theta_L$, default 0.3) are perturbed once and kept at most once, which is the micro-perturb high value, macro-perturb low value principle of Section 4.3. Each candidate is generated inside the complete domain S_j by the value-inverse perturbation function of Definition 5 and clipped to $[L_j, U_j]$, so no synthetic value ever leaves the feasible region.

The per-feature candidate sets are combined by Cartesian product, then pruned by deduplication and by removing the original combination. Retention per sample is regulated by ds'_i (quantile thresholds $h > l$, defaults $0.9/0.5$): high-contribution samples keep 3 candidates, medium keep 2, and low keep 1, so augmentation capacity concentrates on the samples the valuation deems informative. This bounds the growth of D_1 and guarantees that the original sample, the most semantically safe point, is never re-issued as a synthetic one. The thresholds and retention ratios are reference configurations, adjustable to a dataset's sample size, feature dimensionality, and class distribution.

C.5 Categorical-Feature Augmentation

Algorithm 5 extends the same value-inverse principle to categorical features. The replacement probability $p_{\text{replace}} = \min((1 - (\varphi_1 b'_C + \varphi_2 ds'_i))^p + \varepsilon,\ 1)$ reuses the numerical perturbation form of Definition 5, so a categorical feature that is unimportant or belongs to a low-contribution sample is rarely touched, which is the same conservative instinct that protects the minority class in the numerical branch. A random decision against p_{replace} then decides, for each feature and sample independently, whether the original value is retained or replaced through the constraint table.

Replacement is table-driven and total. The associated-feature values $A = (A_1 = a_1, A_2 = a_2, \ldots)$ index the table. An exact hit samples the most faithful legal value, a relaxed lookup drops the weakest conditioning feature when the exact combination is unseen, the marginal distribution $P(C)$ is the second fallback, and a uniform draw over the legal domain $\text{Uniform}(C)$ is the last resort. Because the fallback chain always terminates at a legal value, every query returns a business-compliant category. The processed features are concatenated with the numerical branch into x_i^{final} and merged with D' into D_2.

Algorithm 4 Differentiated numerical-feature augmentation.

Input: to-be-augmented dataset D', complete domain $\{S_j = [L_j, U_j]\}_{j=1}^{m}$, feature importance $\{b'_j\}$, sample contributions $\{ds'_i\}$, hyperparameters $\varphi_1 = 0.7, \varphi_2 = 0.3, p, \varepsilon, t_0$ (Def. 5), thresholds $\theta_L = 0.3, \theta_H = 0.7, h = 0.9, l = 0.5$
Output: numerical augmented dataset D_1

$K_{\text{num}} \leftarrow$ number of numerical features
$D_1 \leftarrow \varnothing$
for each sample x_i in D' **do**
  for each numerical feature $j = 1$ to K_{num} **do**
    $V_{ij} \leftarrow \varphi_1 \cdot b'_j + \varphi_2 \cdot ds'_i$ // cell combined value
    $f(b'_j, ds'_i) \leftarrow t_0 \cdot (1 - V_{ij})^p + \varepsilon$ // perturbation function (Def. 5)
    if $V_{ij} \geq \theta_H$ **then**
      $n_{ij} \leftarrow 5$ // high value: multiple micro-rounds
    else if $V_{ij} \geq \theta_L$ **then**
      $n_{ij} \leftarrow 3$ // medium value
    else
      $n_{ij} \leftarrow 1$ // low value: few macro-rounds
    end if
    $X_{ij} \leftarrow \varnothing$
    for round $= 1$ to n_{ij} **do**
      $R \leftarrow \text{RandomChoice}(\{-1, +1\})$
      $x_{\text{new}} \leftarrow \max(L_j, \min(x_{ij} + R \cdot |S_j| \cdot f(b'_j, ds'_i), U_j))$
      $X_{ij} \leftarrow X_{ij} \cup \{x_{\text{new}}\}$
    end for
    $X_{ij} \leftarrow \text{RemoveDuplicates}(X_{ij})$
  end for
  $X_i \leftarrow X_{i,1} \times X_{i,2} \times \cdots \times X_{i,K_{\text{num}}}$ // Cartesian product
  $X_i \leftarrow X_i \setminus \{[x_{i,1}, \ldots, x_{i,K_{\text{num}}}]\}$ // exclude original combination
  if $ds'_i \geq h$ **then**
    $M_i \leftarrow 3$ // high contribution: keep 3
  else if $ds'_i \geq l$ **then**
    $M_i \leftarrow 2$ // medium contribution: keep 2
  else
    $M_i \leftarrow 1$ // low contribution: keep 1
  end if
  if $|X_i| > M_i$ **then**
    $X_i \leftarrow \text{SampleWithoutReplacement}(X_i, M_i)$
  end if
  for each x^{new} in X_i **do**
    $x^{\text{new}}.\text{cat_features} \leftarrow x_i.\text{cat_features}$ // categorical features retained
    $x^{\text{new}}.\text{label} \leftarrow y_i$
    $D_1 \leftarrow D_1 \cup \{x^{\text{new}}\}$
  end for
end for
return D_1

D Proof of Theorem 1

Figure 7 previews the result ahead of the derivation: the optimal perturbation surface decreases in both coordinates of the data value, so high-value cells are only lightly touched while low-value cells receive the strongest perturbation, which is the quantitative form of DIDA's differential principle. The formal argument follows.

Lemma D.1 (Penalty absorption lemma). Under the small-perturbation assumption (the premise of Theorem 1), for any given semantic-consistency threshold $\varepsilon_0 \in (0, 1)$, there exists a penalty coefficient $\gamma \in (0, 1]$ and a maximum perturbation bound $\delta_{\max} = \max_{i,j} \delta_{ij}$ such that for any sample i:

$$\frac{\gamma \cdot ds'_i}{m} \cdot w^\top |x_i - x_i^{\text{new}}| \leq \varepsilon_0 \cdot ds'_i. \tag{7}$$

Moreover, the original semantic-consistency constraint $\text{Consist}(x_i^{\text{new}}, x_i) \geq 1 - \varepsilon_0$ can be guaranteed by the following tighter sufficient condition:

$$\text{WeightedCos}(x_i^{\text{new}}, x_i) \geq 1 - \varepsilon_0 \,(1 - ds'_i)\,. \tag{8}$$

Proof.

Algorithm 5 Constraint-guided categorical-feature augmentation.

Input: numerical augmented dataset D_1, to-be-augmented dataset D', constraint table T, feature importance $\{b'_j\}$, sample contributions $\{ds'_i\}$, fusion weights φ_1, φ_2, exponent p, floor ε
Output: fully augmented dataset D_2

$D_2 = \varnothing$
for each (x_i, y_i) in D_1 **do**
  $\hat{x} = \mathrm{copy}(x_i)$
  for each categorical feature C **do**
    $p_{\mathrm{replace}} = \min((1 - (\varphi_1 b'_C + \varphi_2 ds'_i))^p + \varepsilon,\ 1)$
    if $\mathrm{Random}() < p_{\mathrm{replace}}$ **then**
      $A = \mathrm{joint_feature_values}(x_i, C, T)$ // query key
      if $A \in T$ **then**
        $\hat{x}[C] = \mathrm{CategoricalSample}(T[A])$
      else
        $A' \leftarrow \mathrm{relax_weakest}(A, T)$ // fallback 1: relax weakest constraint
        if $A' \in T$ **then**
          $\hat{x}[C] = \mathrm{CategoricalSample}(T[A'])$
        else if $\mathrm{has_marginal}(C)$ **then**
          $\hat{x}[C] = \mathrm{MarginalSample}(C)$ // fallback 2: marginal distribution
        else
          $\hat{x}[C] = \mathrm{UniformSample}(C)$ // fallback 3: uniform over legal domain
        end if
      end if
    else
      $\hat{x}[C] = x_i[C]$ // retain original
    end if
  end for
  $\hat{x} = (\hat{x}[\mathrm{num}], \hat{x}[\mathrm{cat}])$
  $D_2 = D_2 \cup \{(\hat{x}, y_i)\}$
end for
$D_2 = D_2 \cup D'$ // merge original
return D_2

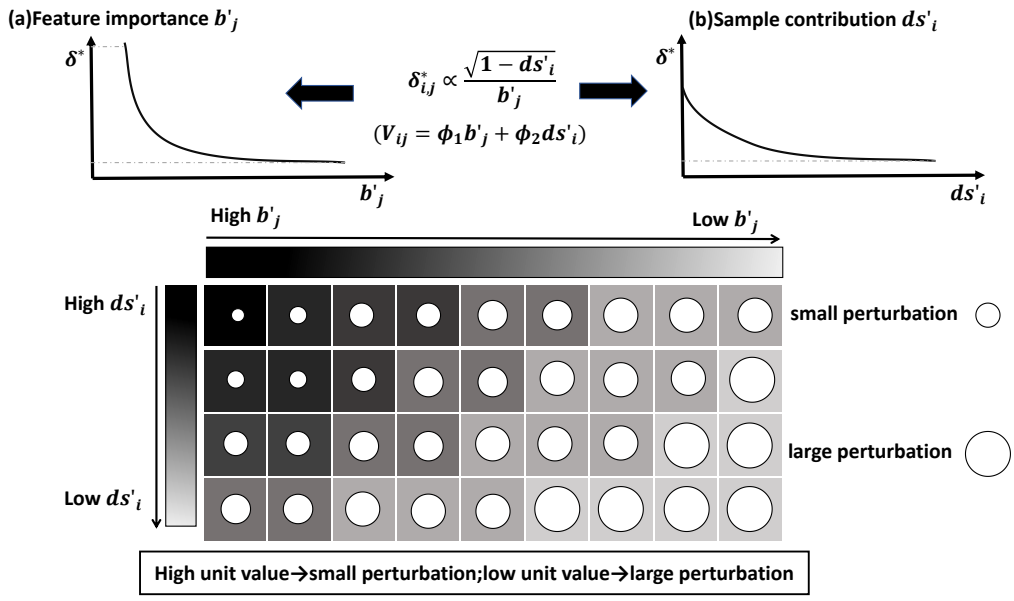

Figure 7: Illustration of Theorem 1: the optimal perturbation intensity δ^*_{ij} decreases with the two-dimensional data value (b'_j, ds'_i), so high-value cells are only lightly touched while low-value cells receive the strongest perturbation.

Step 1 (upper bound of the penalty term). From the small-perturbation assumption and $b'_j \in [0, 1]$, together with $\delta_{ij} \le \delta_{\max}$, we relax the weighted perturbation term:

$$w^\top |x_i - x_i^{\mathrm{new}}| = \sum_j b'_j \delta_{ij} \le \sum_j \delta_{\max} = m \cdot \delta_{\max}. \tag{9}$$

Substituting into the penalty term:

$$\frac{\gamma \cdot ds_i'}{m} \cdot w^\top |x_i - x_i^{\text{new}}| \le \frac{\gamma \cdot ds_i'}{m} \cdot m\delta_{\max} = \gamma \cdot ds_i' \cdot \delta_{\max}. \tag{10}$$

Step 2 (existence construction of γ). For Eq. (7) to hold, it suffices that $\gamma \cdot \delta_{\max} \le \varepsilon_0$, i.e., $\gamma \le \varepsilon_0/\delta_{\max}$. Construct the feasible penalty coefficient:

$$\gamma = \min\left\{1,\ \frac{\varepsilon_0}{\delta_{\max}}\right\}. \tag{11}$$

Two cases verify legality:

(i) If $\delta_{\max} \ge \varepsilon_0$, then $\varepsilon_0/\delta_{\max} \in (0, 1]$, and taking $\gamma = \varepsilon_0/\delta_{\max}$ satisfies $\gamma \cdot \delta_{\max} = \varepsilon_0$.

(ii) If $\delta_{\max} < \varepsilon_0$, taking $\gamma = 1$ satisfies $\gamma \cdot \delta_{\max} = \delta_{\max} < \varepsilon_0$.

Hence there always exists $\gamma \in (0, 1]$ such that Eq. (7) holds:

$$\frac{\gamma \cdot ds_i'}{m} \cdot w^\top |x_i - x_i^{\text{new}}| \le \varepsilon_0 \cdot ds_i'. \tag{12}$$

By the definition of semantic consistency, together with the penalty-term upper bound:

$$\text{Consist}(x_i^{\text{new}}, x_i) \ge \text{WeightedCos}(x_i^{\text{new}}, x_i) - \varepsilon_0 \cdot ds_i'. \tag{13}$$

If Eq. (8) holds, substituting the weighted-cosine lower bound:

$$\text{Consist} \ge [1 - \varepsilon_0(1 - ds_i')] - \varepsilon_0 ds_i' = 1 - \varepsilon_0. \tag{14}$$

Therefore Eq. (8) is a sufficient condition for the original semantic-consistency constraint.

In DIDA's practical deployment, the maximum perturbation bound $\delta_{\max}$ is controlled by the anchor hyperparameter t_0 (Definition 5). Under the conventional configuration $t_0 \in [0.05, 0.3]$, the corresponding $\delta_{\max}$ is of order $10^{-2} \sim 10^{-1}$, ensuring that a feasible $\gamma \in (0, 1]$ always exists. In engineering practice, $\gamma = \varepsilon_0/t_0$ is recommended as the default penalty coefficient.

Proof of Theorem 1. By Definition 3, the semantic-consistency function is:

$$\text{WeightedCos}(x_i^{\text{new}}, x_i) = \frac{\sum_{j=1}^m b_j' \, x_{ij} \, x_{ij}^{\text{new}}}{\sqrt{\sum_{j=1}^m b_j' x_{ij}^2} \cdot \sqrt{\sum_{j=1}^m b_j' \left(x_{ij}^{\text{new}}\right)^2}}. \tag{15}$$

For the small-perturbation scenario $\delta_{ij} \ll |x_{ij}|$, let $x_{ij}^{\text{new}} = x_{ij} + \delta_{ij} \cdot R_{ij}$ with $R_{ij} \in \{-1, +1\}$. A second-order Taylor expansion of the weighted cosine about x_i gives

$$\text{WeightedCos}(x_i^{\text{new}}, x_i) \approx 1 - \frac{1}{2} \cdot \frac{\sum_{j=1}^m b_j' \, \delta_{ij}^2 R_{ij}^2}{\sum_{j=1}^m b_j' x_{ij}^2} + \frac{1}{2}\left(\frac{\sum_{j=1}^m b_j' x_{ij} \, \delta_{ij} R_{ij}}{\sum_{j=1}^m b_j' x_{ij}^2}\right)^2. \tag{16}$$

The projection term (the last addend) is non-negative, so dropping it yields a stricter sufficient condition. Since $R_{ij} \in \{-1, +1\}$, $\mathbb{E}[R_{ij}^2] = 1$. Defining the normalization factor

$$A_i = \sum_{j=1}^m b_j' x_{ij}^2, \tag{17}$$

we obtain the sufficient bound

$$\text{WeightedCos}(x_i^{\text{new}}, x_i) \ge 1 - \frac{1}{2A_i} \sum_{j=1}^m b_j' \delta_{ij}^2. \tag{18}$$

Combining with the constraint $\text{WeightedCos}(x_i^{\text{new}}, x_i) \ge 1 - \varepsilon_0(1 - ds_i')$:

$$\sum_{j=1}^m b_j' \delta_{ij}^2 \le 2A_i \varepsilon_0 \left(1 - ds_i'\right). \tag{19}$$

Define the constraint upper bound:

$$C_i = \kappa\left(1 - ds_i'\right), \qquad \kappa = 2A_i\varepsilon_0. \tag{20}$$

Thus, for each sample i, the optimization problem simplifies to:

$$\max \sum_{j=1}^{m} \delta_{ij} \quad \text{s.t.} \quad \sum_{j=1}^{m} b_j' \delta_{ij}^2 \le \kappa\left(1 - ds_i'\right). \tag{21}$$

Construct the Lagrangian:

$$\mathcal{L}\left(\delta_{i,1}, \ldots, \delta_{i,m}, \lambda, \mu_1, \ldots, \mu_m\right) = \sum_{j=1}^{m} \delta_{ij} - \lambda\left(\sum_{j=1}^{m} b_j' \delta_{ij}^2 - \kappa\left(1 - ds_i'\right)\right) + \sum_{j=1}^{m} \mu_j \delta_{ij}, \tag{22}$$

where $\lambda \ge 0$, $\mu_j \ge 0$ are Lagrange multipliers.

The KKT optimality conditions are as follows.

Stationarity:

$$\frac{\partial \mathcal{L}}{\partial \delta_{ij}} = 1 - 2\lambda b_j' \delta_{ij} + \mu_j = 0, \quad \forall j. \tag{23}$$

Primal feasibility:

$$\sum_{j=1}^{m} b_j' \delta_{ij}^2 \le \kappa\left(1 - ds_i'\right), \quad \delta_{ij} \ge 0. \tag{24}$$

Dual feasibility:

$$\lambda \ge 0, \quad \mu_j \ge 0, \quad \forall j. \tag{25}$$

Complementary slackness:

$$\lambda\left(\sum_{j=1}^{m} b_j' \delta_{ij}^2 - \kappa\left(1 - ds_i'\right)\right) = 0, \qquad \mu_j \delta_{ij} = 0, \quad \forall j. \tag{26}$$

Because the objective is linear and increasing, the optimum lies on the constraint boundary (perturbation as large as possible to maximize diversity). Hence:

$$\lambda > 0, \qquad \sum_{j=1}^{m} b_j' \delta_{ij}^2 = \kappa\left(1 - ds_i'\right). \tag{27}$$

If $\delta_{ij} > 0$, then $\mu_j = 0$. From stationarity:

$$1 - 2\lambda b_j' \delta_{ij} = 0 \quad \Rightarrow \quad \delta_{ij} = \frac{1}{2\lambda b_j'}. \tag{28}$$

Substituting into the boundary condition:

$$\sum_{j=1}^{m} b_j' \left(\frac{1}{2\lambda b_j'}\right)^2 = \kappa\left(1 - ds_i'\right). \tag{29}$$

Solving for λ:

$$\lambda = \frac{1}{2\sqrt{\kappa\left(1 - ds_i'\right)}} \sqrt{\sum_{j=1}^{m} \frac{1}{b_j'}}. \tag{30}$$

Substituting back:

$$\delta_{ij} = \frac{1}{b_j'} \cdot \frac{\sqrt{\kappa\left(1 - ds_i'\right)}}{\sqrt{\sum_{k=1}^{m} \frac{1}{b_k'}}}. \tag{31}$$

Therefore:

$$\delta_{ij} = Z \cdot \frac{\sqrt{1 - ds_i'}}{b_j'}, \tag{32}$$

which establishes $\delta_{ij}^* \propto \frac{\sqrt{1-ds_i'}}{b_j'}$, completing the proof of Theorem 1. ■

Table 4: ANOVA/MI weight-scan results (including constraint-table intermediate statistics).

ANOVA	MI	Heart ROC	Heart F1	Kidney ROC	Kidney F1	Heart #r	Kidney #r
0.8	0.2	0.8511	0.8016	0.8283	0.9618	317	314
0.7	0.3	0.8511	0.8016	0.8283	0.9618	317	314
0.6	0.4	0.8511	0.8016	0.8276	0.9628	317	246
0.5	0.5	0.8511	0.8016	0.8276	0.9628	317	246
0.4	0.6	0.8511	0.8016	0.8292	0.9613	317	246
0.3	0.7	0.8511	0.8016	0.8358	0.9620	317	174
0.2	0.8	0.8511	0.8016	0.8317	0.9620	317	126
Range		0.0000	0.0000	0.0082	0.0015	0	188

From Theorem 1 to the practical perturbation function. Definition 5 resolves Theorem 1's three deployment constraints ($b' \to 0$ divergence, normalization coupling, uncontrollable decay) while preserving monotonicity.

E Sensitivity Analysis

E.1 Design Rationale

The weighted combination of ANOVA and MI forms the core of the feature-association score in DIDA's constraint-table construction: $\text{Score}(f_i, f_j) = \omega_{ANOVA} \cdot ANOVA_{\text{norm}}(f_i, f_j) + \omega_{MI} \cdot MI_{\text{norm}}(f_i, f_j)$, where $ANOVA_{\text{norm}}$ and MI_{norm} are the F/χ^2 and mutual-information statistics, respectively, normalized to $[0, 1]$, with $\omega_{ANOVA} + \omega_{MI} = 1$. After dual-threshold screening (score ≥ 0.1, $p \leq 0.05$), the top-3 associated feature pairs are selected to build the constraint table guiding categorical-feature augmentation. The core question of this subsection is whether the relative weights ω_{MI} and ω_{ANOVA} significantly affect DIDA's final classification performance. If the effect is negligible, they can be removed from the search space.

E.2 Experimental Results

Table 4 reports the full weight-scan results.

Heart, strict zero sensitivity: ROC range $= 0.0000$, F1 range $= 0.0000$. The seven weight settings produce exactly the same constraint table (317 rules), hence identical augmented data. The root cause is that, on Heart's six categorical features, the MI normalized ranking and the F-test ranking are highly consistent, so the top-3 feature-pair set after dual-threshold screening is unaffected by $\omega_{MI}/\omega_{ANOVA}$, and when two association measures agree on ranking, the weighted combination degenerates into a ranking-invariant function.

Kidney, minimal sensitivity: ROC range $= 0.0082$ ($0.8276 \to 0.8358$), F1 range $= 0.0015$ ($0.9613 \to 0.9628$). The number of constraint-table rules shrinks from 314 to 126 ($\Delta = 188$) as ω_{ANOVA} decreases, reflecting that the F/χ^2 test is more sensitive to Kidney's 18-dimensional categorical features. However, the change in constraint-table size does not translate into performance differences: on one hand, the filtered rules are mostly weak constraints (conditional probability < 0.6, weak type), and on the other hand, DIDA's three-level buffer mechanism (dual threshold $\to$ top-3 truncation $\to$ probability sampling) smooths constraint-table differences. $\omega_{MI} = 0.7$ (174 rules) achieves the best ROC $= 0.8358$, hinting that constraint-table pruning (suppressing weak-association noise) may be better than enrichment.

E.3 Why It's Robust

The above results show that the effect of the ANOVA–MI weighted combination on DIDA's final classification performance is negligible in both statistical and practical senses: the ROC range is 0.0000 on Heart and 0.0082 on Kidney, both far below the marginal effects of the core hyperparameters (ε, t_0). We recommend fixing the ANOVA/MI weights at $(0.6, 0.4)$ and permanently removing them from the hyperparameter search space.

Table 5: Global statistics over 700 full-factorial configurations.

Statistic	Heart AUC-ROC	Heart F1-score	Kidney AUC-ROC	Kidney F1-score
Mean	0.8566	0.8075	0.7311	0.9524
Std	0.002712	0.003188	0.008394	0.003775
Min	0.8472	0.7972	0.7080	0.9401
Max	0.8650	0.8187	0.7539	0.9595
Range	0.0178	0.0215	0.0459	0.0194
Relative fluct. (%)	2.1	2.7	6.3	2.0

E.4 Hyperparameter Sensitivity

E.4.1 Experimental Setting

To evaluate DIDA's robustness to hyperparameters, a full-factorial experiment is conducted on the Heart and Kidney datasets: $\Phi = \{\varphi_1\} \times \{t_0\} \times \{p\} \times \{\varepsilon\} = 7 \times 5 \times 5 \times 4 = 700$ configurations. ANOVA/MI is fixed at $(0.6, 0.4)$, with its independence verified in Appendix E. $\varphi_1 \in \{0.2, \ldots, 0.8\}$ covers the sample-dominant, balanced, and feature-dominant paradigms. $t_0 \in \{0.05, 0.1, 0.15, 0.2, 0.3\}$ controls perturbation magnitude. $p \in \{1.0, 1.5, 2.0, 2.5, 3.0\}$ controls the decay rate. $\varepsilon \in \{0.005, 0.01, 0.02, 0.05\}$ ensures minimum effective perturbation. Each configuration is evaluated independently on the 10 heterogeneous classifiers, taking the AUC-ROC and F1 means and cross-model standard deviations. A design limitation is that ANOVA/MI is fixed at its default value, assuming no strong interaction with the perturbation parameters (subsequent validation in Appendix E). The two parameter classes are searched independently, which may underestimate joint effects if cross-class interactions exist. Given computational feasibility, independent search is a reasonable simplification to control the combinatorial explosion (the complete 5-parameter space is $7 \times 5 \times 5 \times 4 \times 7 = 4900$ configurations). The experiment strips the long-tail optimization and business-constraint fusion modules to avoid interference from non-core factors.

E.4.2 Global Sensitivity Panorama

The 700-configuration global statistics are as follows (Table 5). On Heart, the ROC mean is 0.8566 (SD $= 0.0027$), range 0.0178, relative fluctuation 2.08%. On Kidney, the ROC mean is 0.7311 (SD $= 0.0084$), range 0.0459, relative fluctuation 6.28%. On both datasets the ROC standard deviations across configurations are < 0.01 and relative fluctuations $< 7\%$, indicating that the DIDA framework is intrinsically robust to hyperparameter choice. This robustness stems from the smoothness of the perturbation formula: parameters pass through continuous functions with no abrupt thresholds. Kidney's relative fluctuation is roughly 3 times Heart's, consistent with its higher feature dimensionality (51 vs 11): more features enlarge the parameter-influence space. The ROC gaps between the global best and worst configurations (Heart: 0.0178, Kidney: 0.0459) equal $0.14\times$ and $0.47\times$ the respective cross-model ROC standard deviations, directly quantifying the marginal-gain upper bound of hyperparameter tuning relative to model selection.

The headline of Table 5 is the relative fluctuation: under 3% on Heart and under 7% on Kidney across all 700 configurations. That a method whose core operation is per-sample, value-dependent perturbation stays within a 0.02 ROC band on Heart and a 0.05 band on Kidney while its four key hyperparameters are swept end to end is the strongest available evidence that DIDA's advantage is structural rather than tuned. Kidney's roughly threefold larger fluctuation aligns with its higher dimensionality (51 versus 11 features), which widens the space over which parameter choices can act.

E.4.3 Single-Parameter Marginal Effects

For each parameter, the other three are fixed at their defaults ($\varphi_1 = 0.7, t_0 = 0.1, p = 2.0, \varepsilon = 0.01$) and varied one at a time. Two-sided t-tests evaluate the statistical significance between extreme-value groups ($^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$, n.s. not significant). φ_1 has the smallest effect size and unstable direction: on Heart, the sample-dominant $\varphi_1 < 0.5$ (ROC $= 0.8565$) slightly beats the feature-dominant $\varphi_1 > 0.5$ (0.8549), while on Kidney the feature-dominant (0.7344) beats the sample-dominant (0.7218), in opposite directions. On both datasets it does not reach

Table 6: $t_0 \times p$ interaction matrix (Heart, $\varphi_1 = 0.7$, $\varepsilon = 0.01$).

$t_0 \setminus p$	1.0	1.5	2.0	2.5	3.0
0.05	0.8504	0.8563	0.8558	0.8551	0.8566
0.1	0.8549	0.8594	0.8541	0.8497	0.8568
0.15	0.8541	0.8568	0.8583	0.8556	0.8595
0.2	0.8526	0.8546	0.8569	0.8579	0.8585
0.3	0.8599	0.8616	0.8597	0.8575	0.8584

Table 7: $t_0 \times p$ interaction matrix (Kidney, $\varphi_1 = 0.7$, $\varepsilon = 0.01$).

$t_0 \setminus p$	1.0	1.5	2.0	2.5	3.0
0.05	0.7180	0.7274	0.7299	0.7306	0.7171
0.1	0.7279	0.7236	0.7333	0.7275	0.7291
0.15	0.7317	0.7338	0.7315	0.7330	0.7248
0.2	0.7255	0.7306	0.7355	0.7261	0.7280
0.3	0.7456	0.7426	0.7404	0.7282	0.7323

statistical significance (Heart $t = -1.24$, $p = 0.216$, and Kidney $t = -1.52$, $p = 0.130$), giving it the lowest tuning value. t_0 has the largest effect size with a consistently positive direction: on Kidney, ROC rises from 0.7299 to 0.7404 as t_0 goes from 0.05 to 0.3 ($\Delta = 0.0105$, $t = -3.42$, $p = 0.0007^{***}$), and on Heart $t_0 = 0.3$ is optimal (ROC $= 0.8597$), and the search interval $[0.1, 0.3]$ is recommended. p has the strongest dataset dependence: the optimum is $p = 1.5$ on Heart (ROC $= 0.8594$) and $p = 2.0$ on Kidney (0.7333), with $p = 1.5$–2.0 both feasible. At $p \geq 2.5$ Heart degrades (ROC $= 0.8497$), as strong decay leaves high-contribution samples almost unperturbed. ε is the only parameter with a cross-dataset consistent, significant, and positive unidirectional effect: Heart ROC $0.8568 \to 0.8606$ ($\Delta = 0.0038$, $t = -4.5^{***}$), Kidney $0.7272 \to 0.7456$ ($\Delta = 0.0184$, $t = -6.2^{***}$). Overall priority: $\varepsilon > t_0 > p > \varphi_1$, and tuning ε and t_0 gives the most reliable gains.

E.4.4 Parameter-Interaction Effects

The full-cross design reveals interaction patterns undetectable by independent grid search. The $t_0 \times p$ interaction shows weak synergy: larger t_0 with moderate p (1.5–2.0) performs best (Tables 6 and 7). On Heart, $t_0 = 0.3, p = 1.5$ gives ROC $= 0.8616$, with interaction increment ≈ 0.0017. The mechanism is that a large t_0 provides sufficient perturbation magnitude while a moderate p preserves contribution-discrimination power. The $\varepsilon \times t_0$ interaction is significantly complementary on Kidney (Table 8): at $\varepsilon = 0.005$ the ROC range of t_0 is 0.0169, shrinking to 0.0136 at $\varepsilon = 0.05$, indicating that a larger additive baseline perturbation can partially substitute for the multiplicative coefficient, reducing dependence on the precise setting of t_0. The practical implication of the absence of strong interactions is that the ROC gap between the full-cross optimum and the default-plus-independent-marginal optimum is only 0.0032 (Heart), meaning independent marginal tuning already approaches over 97% of the global optimum. The theoretical completeness of the full-cross design translates into stronger confidence in parameter robustness.

Read together, the three matrices (Tables 6, 7, 8) show interactions that are real but weak and benign. In the $t_0 \times p$ matrices the optimum is a gentle plateau, where the region of large t_0 with moderate p stays within about 0.002 of the global peak across several adjacent cells, so no single parameter pair is critical. In the $\varepsilon \times t_0$ matrix the rows visibly flatten as ε grows, the statistical signature of the substitution described above: a larger additive baseline partially frees the method from its dependence on the exact multiplicative anchor.

E.4.5 Cross-Model Stability

The core conclusion of this section is that model selection matters far more than hyperparameter selection. On Heart, the cross-model ROC variance (0.0159) far exceeds Kidney's (0.0097), and within a single configuration the best and worst model ROC can differ by 0.40. The cross-model ROC range (0.40) is 23 times the hyperparameter ROC range (0.0178) on Heart, and 7 times on Kidney (0.31 vs 0.0459). The model-wise ROC means and ranges are summarized in Table 9. In

Table 8: $\varepsilon \times t_0$ interaction matrix (Kidney, $\varphi_1 = 0.7$, $p = 2.0$).

$\varepsilon \setminus t_0$	0.05	0.1	0.15	0.2	0.3
0.005	0.7221	0.7390	0.7371	0.7224	0.7330
0.01	0.7299	0.7333	0.7315	0.7355	0.7404
0.02	0.7277	0.7272	0.7308	0.7178	0.7227
0.05	0.7348	0.7456	0.7451	0.7320	0.7357

Table 9: Cross-model performance ranking (averaged over 700 configurations).

Rank	Model	Heart ROC mean±std	Heart ROC range	Kidney ROC mean±std	Kidney ROC range
1	Random Forest	0.9345 ± 0.0034	$[0.9243, 0.9444]$	0.8090 ± 0.0172	$[0.7597, 0.8583]$
2	Extra Trees	0.9370 ± 0.0040	$[0.9237, 0.9486]$	0.8009 ± 0.0231	$[0.7346, 0.8660]$
3	XGBoost	0.9264 ± 0.0043	$[0.9091, 0.9394]$	0.8090 ± 0.0152	$[0.7687, 0.8624]$
4	Gradient Boosting	0.9339 ± 0.0040	$[0.9217, 0.9450]$	0.7946 ± 0.0176	$[0.7432, 0.8429]$
5	AdaBoost	0.9337 ± 0.0025	$[0.9253, 0.9405]$	0.7745 ± 0.0181	$[0.7129, 0.8275]$
6	Logistic Regression	0.9170 ± 0.0015	$[0.9114, 0.9206]$	0.7888 ± 0.0080	$[0.7677, 0.8138]$
7	Neural Network	0.9181 ± 0.0024	$[0.9108, 0.9235]$	0.7270 ± 0.0267	$[0.6509, 0.8038]$
8	Decision Tree	0.7899 ± 0.0207	$[0.7348, 0.8656]$	0.6687 ± 0.0375	$[0.5750, 0.7940]$
9	k-NN	0.7408 ± 0.0113	$[0.7070, 0.7835]$	0.6428 ± 0.0273	$[0.5378, 0.7526]$
10	TabNet	0.5350 ± 0.0000	$[0.5350, 0.5350]$	0.4961 ± 0.0162	$[0.4254, 0.5424]$

practical deployment, model selection and ensembling should be prioritized over DIDA parameter fine-tuning. On Kidney, the F1–ROC gap (F1 ≈ 0.951 vs ROC ≈ 0.733, $\Delta \approx +0.218$) reflects the high class prevalence ($\approx 62.5\%$). F1 is threshold-dependent while AUC-ROC is threshold-free, so both metrics are reported.

Table 9 reorders the practical priorities of this section. On Heart the ROC spread runs from Extra Trees' 0.9370 down to TabNet's 0.5350, a 0.40 gap that is 23 times the hyperparameter ROC range, so no amount of DIDA tuning can compensate for a weak downstream classifier. The tree ensembles occupy the top five positions on both datasets, exactly the regime where DIDA's leaf-based embedding and constraint-table sampling are most compatible, while TabNet's flat 0.5350 (zero variance) reveals a model that collapses on small tabular data regardless of augmentation quality.

E.4.6 Recommended Configuration

The default hyperparameters ($t_0 = 0.1, p = 2.0, \varepsilon = 0.01, \varphi_1 = 0.7$) behave stably on both datasets (Heart ROC $= 0.8541$, Kidney ROC $= 0.7333$), with gaps to the optima of only $\Delta = 0.0077$ (Heart) and $\Delta = 0.0180$ (Kidney). Tuning priority follows the marginal-gain order. ε is tuned first over $[0.01, 0.05]$ in steps of 0.01 and is the most reliable lever, with a ROC gain of roughly $+0.005 \sim 0.008$. t_0 comes next over $[0.1, 0.3]$ in steps of 0.05, contributing $+0.002 \sim 0.004$ and never exceeding 0.3. p is a low-priority refinement over $[1.0, 2.0]$ in steps of 0.5, gaining below $+0.004$, with $[1.0, 1.5]$ preferred for high-dimensional data. φ_1 and the ANOVA/MI weights gain under $+0.001$ and are fixed at their defaults (0.7 and $0.6/0.4$, respectively). In practice only $5 \times 5 \times 3 = 75$ configurations ($\varepsilon \times t_0 \times p$) need searching, an 89% reduction from the full 4-parameter 700 configurations while retaining over 97% of the global optimum. ANOVA/MI tuning is not recommended, as independently verified in Appendix E (Heart ROC range 0.0000, Kidney 0.0082).

This recipe condenses the section: tune ε and t_0 first, treat p as a low-priority refinement, and freeze φ_1 and the ANOVA/MI weights entirely. The 89% reduction in search cost, 75 configurations instead of 700, retaining over 97% of the global optimum, is the practical payoff of the robustness established throughout this appendix, and it is what makes DIDA deployable in exactly the small-data, resource-constrained settings where exhaustive hyperparameter search is least affordable.

F Experimental Details

F.1 Dataset Details

This experiment selects six public tabular datasets from Kaggle as benchmarks, spanning the three domains of medical health, mental health, and urban commerce: the Teen Mental Health Dataset

Table 10: Summary of the six benchmark datasets.

Dataset	Sample size	Num. features	Cat. features	Balanced
Teen	1200	9	3	Yes
Heart	918	5	6	No
Alzheimers	2149	15	17	No
Healthcare	15000	12	5	Yes
Urban Street	22000	11	8	Yes
Chronic Kidney	1659	33	18	No

(Teen), the Heart Disease Dataset (Heart), the Alzheimer's Disease Dataset (Alzheimers), the Healthcare Synthetic Dataset (Healthcare), the Urban Street Food Vendor Survival Dataset (Urban Street), and the Chronic Kidney Disease Dataset (Chronic Kidney). Basic information (sample size, feature dimension, class distribution, and domain) is summarized in Table 10.

The six datasets span three domains but deliberately concentrate in the small-sample regime: four of the six contain fewer than 2,200 samples, while the two largest (Healthcare and Urban Street, 15,000 and 22,000) act as stress tests for whether the value-driven mechanism degrades once the scarcity assumption relaxes. Half of the datasets are class-imbalanced, which is why AUC-ROC and F1 are both reported downstream and why the contribution-sensitive augmentation branch is exercised rather than merely asserted.

F.2 Data Preprocessing

A standardized preprocessing pipeline is applied. ID/serial fields with no business meaning are removed first. For categorical features, missing values are filled with the mode, then converted to numeric inputs via label encoding. For numerical features, missing values are filled with the median to avoid interference from extreme values on feature-distribution statistics. Dataset splitting uses stratified random sampling, ensuring the class distribution of each split matches the original dataset, and the train/validation/test ratio is $3:1:1$. For the larger, higher-quality Urban Street dataset, to balance computational efficiency and experimental precision, 2,000 samples are drawn as the training set and 2,000 as the validation set via stratified sampling, with the remaining samples as the test set, keeping class distributions consistent.

F.3 Protocol

Augmented-sample-size control. To eliminate the interference of sample-size differences on downstream performance, the number of newly generated samples of all baselines is set uniformly to 70% of the original training-set size, ensuring the augmented training-set sizes are exactly consistent across methods.

Downstream models. The ten downstream classifiers are: tree ensembles (AdaBoost, decision tree, extra trees, gradient boosting (Friedman, 2001), random forest (Breiman, 2001), XGBoost (Chen & Guestrin, 2016)), a linear model (logistic regression), instance-based learning (k-NN), and tabular deep learning (MLP, TabNet (Arik & Pfister, 2021)), all with public default configurations not tuned per augmented dataset, to avoid introducing artificial performance bias.

DIDA default hyperparameters. Perturbation-related: anchor $t_0 = 0.1$, decay $p = 2.0$, minimum threshold $\varepsilon = 0.01$, fusion weights $\varphi_1 = 0.7, \varphi_2 = 0.3$. Constraint-table-related: MI weight $= 0.4$, ANOVA weight $= 0.6$. These defaults are validated by the 700-configuration full-cross sensitivity analysis of Appendix E, with relative fluctuation $< 7\%$ on two datasets.

Base-model training. The pretrained XGBoost serving as the base model for DIDA's feature-importance and sample-contribution computation is tuned via grid search with 5-fold cross-validation (Cawley & Talbot, 2010).

F.4 ENVIRONMENT

All experiments run in the same hardware environment. Software environment: Python 3.9, PyTorch 1.13, TensorFlow 2.10, Scikit-learn (Pedregosa et al., 2011), SHAP (Lundberg & Lee, 2017), XGBoost (Chen & Guestrin, 2016), LightGBM (Ke et al., 2017). To control for unrelated variables, the preprocessing pipeline, input data format, and downstream-model training logic of all augmentation methods are fully unified. All methods' generated samples are inverse-encoded back to the original data format, excluding the influence of format differences on performance evaluation.

F.5 STATISTICAL VALIDATION

A single experiment fixes the global random seed at 10, and all deep-learning frameworks (PyTorch, TensorFlow) are set to deterministic mode, ensuring full reproducibility of a single run. To validate statistical significance, the entire experimental pipeline is repeated with seeds $10, 20, 30, 40, 50, 60, 70$ (7 times), and the mean and standard deviation of the 7 runs are reported, excluding random-error interference. Statistical-significance testing uses the paired t-test (two-tailed) (Demšar, 2006) with significance level $\alpha = 0.05$.

F.6 EVALUATION METRICS

This appendix summarizes all evaluation metrics used in the Section 5 experiments and their mathematical definitions. Let the real training-set feature matrix be $D \in \mathbb{R}^{N\times m}$ (N samples, m feature dimensions), with the marginal distribution of the d-th feature denoted P_d, and let the augmented generated set be $D_2 \in \mathbb{R}^{\tilde{N}\times m}$, with marginal distribution Q_d.

F.6.1 GENERATION-QUALITY METRICS

Average JS divergence (Jensen–Shannon divergence). Measures the symmetric difference between the real and generated distributions, addressing the asymmetry of KL divergence. Averaged over all features:

$$\mathrm{JS}_{\mathrm{avg}} = \frac{1}{m}\sum_{d=1}^{m}\frac{1}{2}\left[\mathrm{KL}(P_d \parallel M_d) + \mathrm{KL}(Q_d \parallel M_d)\right], \tag{33}$$

where $M_d = (P_d + Q_d)/2$ is the average distribution and $\mathrm{KL}(\cdot \parallel \cdot)$ is the Kullback–Leibler divergence. Range $[0, 1]$, and smaller is better.

Average Wasserstein distance. Its sensitivity to differences between non-overlapping distributions is significantly better than that of JS divergence, suiting tabular features. Averaged over all features:

$$W_{\mathrm{avg}} = \frac{1}{m}\sum_{d=1}^{m}\inf_{\gamma\in\Pi(P_d,Q_d)}\mathbb{E}_{(x,y)\sim\gamma}\left[\|x-y\|\right], \tag{34}$$

where $\Pi(P_d, Q_d)$ is the set of all joint distributions with marginals P_d and Q_d. Non-negative, and smaller is better.

Average information entropy. Measures the feature diversity of generated samples (continuous features are discretized by equal-frequency binning). Averaged over all features:

$$H_{\mathrm{avg}} = \frac{1}{m}\sum_{d=1}^{m}H(Q_d), \tag{35}$$

where $H(\cdot)$ is the entropy of a discrete distribution. Non-negative, and higher is better.

Average feature coverage rate. Measures the coverage of generated samples over the real feature-value space. Discrete features count value coverage directly, and continuous features count bin coverage after equal-frequency binning. Averaged over all features:

$$C_{\mathrm{avg}} = \frac{1}{m}\sum_{d=1}^{m}\frac{|\mathrm{Val}(\tilde{X}_d)\cap\mathrm{Val}(X_d)|}{|\mathrm{Val}(X_d)|}, \tag{36}$$

where $\mathrm{Val}(\cdot)$ is the discrete value set of a feature (bin identifiers for continuous features). Range $[0, 1]$, and higher is better.

Table 11: Generation-quality comparison across methods (6 datasets), values scaled by 100. Lower is better for JS divergence and Wasserstein distance, and higher is better for information entropy and feature coverage. Shaded bold = DIDA, and underline = best baseline.

Method	Avg JS Divergence	Avg Wasserstein Distance	Avg Information Entropy	Feature Coverage Rate	Dataset
DIDA	**0.20 ± 0.09**	**11.14 ± 3.14**	**267.02 ± 1.55**	**100 ± 0.00**	
SMOTE	3.79 ± 0.14	20.05 ± 0.51	256.04 ± 1.76	91.32 ± 0.70	
CTGAN	4.22 ± 0.20	33.59 ± 5.04	253.35 ± 1.85	91.46 ± 0.65	
SMOTE-CDNN	3.79 ± 0.13	19.96 ± 0.55	255.89 ± 1.83	91.32 ± 0.70	**Teen**
TabDDPM	5.21 ± 0.29	41.19 ± 4.23	250.78 ± 3.03	91.32 ± 0.70	
MDLM	5.34 ± 0.31	32.33 ± 1.86	251.47 ± 1.89	91.32 ± 0.70	
TANDEM	4.16 ± 0.18	25.58 ± 1.37	254.92 ± 2.38	91.46 ± 0.65	
DIDA	**0.20 ± 0.03**	**6.74 ± 0.80**	**264.81 ± 1.27**	**100 ± 0.00**	
SMOTE	4.42 ± 0.11	28.22 ± 0.35	255.92 ± 1.42	88.54 ± 0.64	
CTGAN	4.86 ± 0.16	40.44 ± 2.22	254.06 ± 1.13	88.61 ± 0.58	
SMOTE-CDNN	4.47 ± 0.10	28.50 ± 0.50	255.56 ± 1.43	88.54 ± 0.64	**Urban Street**
TabDDPM	6.55 ± 0.27	54.00 ± 4.44	255.41 ± 2.41	88.54 ± 0.64	
MDLM	5.88 ± 0.13	37.71 ± 0.42	255.61 ± 1.32	88.15 ± 0.83	
TANDEM	6.01 ± 0.28	44.26 ± 2.09	256.71 ± 1.38	88.61 ± 0.58	
DIDA	**0.24 ± 0.06**	**7.86 ± 1.02**	**197.98 ± 0.18**	**100 ± 0.00**	
SMOTE	0.96 ± 0.02	10.95 ± 0.21	189.81 ± 0.27	93.75 ± 0.07	
CTGAN	1.18 ± 0.14	23.29 ± 2.56	194.87 ± 1.35	93.75 ± 0.06	
SMOTE-CDNN	0.98 ± 0.02	11.33 ± 0.25	189.6 ± 0.25	93.75 ± 0.07	**Alzheimers**
TabDDPM	2.46 ± 0.16	32.81 ± 2.63	182.97 ± 0.65	93.74 ± 0.06	
MDLM	2.00 ± 0.04	20.94 ± 0.47	186.77 ± 0.39	93.74 ± 0.06	
TANDEM	1.35 ± 0.05	15.02 ± 1.00	187.11 ± 0.38	93.75 ± 0.06	
DIDA	**0.12 ± 0.04**	**5.85 ± 0.97**	**241.02 ± 0.18**	**100 ± 0.00**	
SMOTE	1.25 ± 0.01	14.73 ± 0.29	234.77 ± 0.07	92.13 ± 0.09	
CTGAN	1.72 ± 0.11	34.43 ± 3.27	235.78 ± 0.58	92.14 ± 0.09	
SMOTE-CDNN	1.26 ± 0.01	14.99 ± 0.41	234.69 ± 0.08	92.13 ± 0.09	**Chronic Kidney**
TabDDPM	3.12 ± 0.10	39.05 ± 1.08	226.68 ± 0.49	92.09 ± 0.09	
MDLM	2.74 ± 0.02	27.52 ± 0.29	228.88 ± 0.20	92.09 ± 0.09	
TANDEM	1.56 ± 0.09	22.31 ± 1.54	232.9 ± 0.34	92.14 ± 0.09	
DIDA	**0.11 ± 0.04**	**8.47 ± 1.70**	**257.18 ± 0.76**	**98.74 ± 0.61**	
SMOTE	2.31 ± 0.04	12.84 ± 1.07	248.89 ± 0.87	96.24 ± 0.13	
CTGAN	2.58 ± 0.19	27.88 ± 3.92	248.77 ± 1.31	96.29 ± 0.12	
SMOTE-CDNN	2.34 ± 0.06	13.09 ± 1.30	248.75 ± 0.95	96.23 ± 0.11	**Healthcare**
TabDDPM	4.29 ± 0.30	50.06 ± 1.96	241.23 ± 1.66	96.06 ± 0.19	
MDLM	2.73 ± 0.10	21.63 ± 0.67	248.07 ± 0.76	96.06 ± 0.19	
TANDEM	2.90 ± 0.18	30.85 ± 1.13	247.92 ± 1.21	96.36 ± 0.05	
DIDA	**0.49 ± 0.11**	**10.04 ± 1.55**	**199.96 ± 2.43**	**98.67 ± 0.86**	
SMOTE	7.30 ± 0.74	24.61 ± 1.64	184.72 ± 2.37	89.47 ± 0.56	
CTGAN	8.06 ± 0.40	40.20 ± 1.81	185.76 ± 2.00	89.63 ± 0.59	
SMOTE-CDNN	7.39 ± 0.70	25.09 ± 1.45	184.54 ± 2.78	89.46 ± 0.39	**Heart**
TabDDPM	8.30 ± 0.25	45.98 ± 4.28	185.89 ± 2.53	89.41 ± 0.47	
MDLM	8.03 ± 0.42	30.14 ± 1.06	188.27 ± 1.84	89.41 ± 0.47	
TANDEM	8.21 ± 0.47	32.15 ± 0.83	188.81 ± 1.81	89.58 ± 0.51	

F.6.2 Downstream Classification Metrics

AUC-ROC. The area under the receiver operating characteristic curve, measuring the overall ranking ability of the model on positive and negative samples. Range $[0, 1]$, and higher is better.

F1 Score. A composite metric balancing precision and recall: $\mathrm{F1} = 2PR/(P + R)$, with $P = TP/(TP + FP)$ the precision and $R = TP/(TP + FN)$ the recall, where TP, FP, FN are the true-positive, false-positive, and false-negative counts, respectively. Range $[0, 1]$, and higher is better.

F.7 Full Results

This appendix summarizes the complete comparison results of generation quality and downstream classification performance from the Section 5 experiments. Table 11 reports the generation-quality metrics of DIDA and the 6 baselines on 6 datasets, and Table 12 reports the complete downstream performance of all methods across 6 datasets $\times$ 10 classifiers, providing detailed support for the summarized results of Section 5.2.

F.7.1 Generation-Quality Comparison

Table 11 reports the generation-quality comparison of DIDA and the 6 baselines on 6 datasets, over four metrics: average JS divergence, average Wasserstein distance, average information entropy, and feature coverage rate, where lower is better for the first two and higher is better for the last two.

The quality table separates DIDA from the baselines on the divergence metrics. DIDA's average JS divergence stays below 0.5 on all six datasets. The strongest baseline figure on any dataset is 0.96 (SMOTE on Alzheimers, against DIDA's 0.24 there), and elsewhere baseline JS values rise as high as 8.3 on Heart. The Wasserstein gap is wider still, with DIDA roughly 1.4 to 4.2 times smaller than the nearest competitor across the six datasets. Both gaps point to the same mechanism: constraint-table sampling and value-aware perturbation keep synthetic samples inside the real distribution rather than merely near it. The coverage column tells the complementary story. DIDA reaches 100% on four datasets and 98–99% on the other two, while baselines saturate around 88–96%, the direct footprint of the three-level fallback's guarantee that no legal value is ever unreachable.

F.7.2 Downstream-Performance Comparison

Table 12 reports the complete downstream performance (60 scenarios), and Figures 5, 8 and 9 give the corresponding box-plot view.

Table 12: Complete downstream-performance comparison: mean ± std of AUC-ROC, AUC-PR and F1 Score of DIDA and the 6 baselines across 6 datasets × 10 classifiers (60 scenarios). Bold = best, underline = second-best per (classifier, dataset, metric), and DIDA rows are shaded.

Method	Alzheimers Disease Data			Chronic Kidney Disease data			Heart			Healthcare Synthetic Data			Teen Mental Health Dataset			Urban Street Food Vendor Survival Dataset			Model
	AUC-ROC (Mean±Std)	AUC-PR (Mean±Std)	F1 (Mean±Std)	AUC-ROC (Mean±Std)	AUC-PR (Mean±Std)	F1 (Mean±Std)	AUC-ROC (Mean±Std)	AUC-PR (Mean±Std)	F1 (Mean±Std)	AUC-ROC (Mean±Std)	AUC-PR (Mean±Std)	F1 (Mean±Std)	AUC-ROC (Mean±Std)	AUC-PR (Mean±Std)	F1 (Mean±Std)	AUC-ROC (Mean±Std)	AUC-PR (Mean±Std)	F1 (Mean±Std)	
DIDA	93.62 ± 1.00	89.58 ± 1.52	**87.48 ± 2.36**	76.10 ± 1.95	96.29 ± 0.59	**95.95 ± 0.16**	**92.55 ± 0.86**	**92.66 ± 1.16**	**88.74 ± 1.53**	**79.88 ± 0.59**	**75.78 ± 0.48**	**67.43 ± 0.67**	99.57 ± 1.13	**97.92 ± 5.51**	94.55 ± 7.71	65.81 ± 0.9	59.03 ± 0.68	44.92 ± 3.79	AdaBoost
SMOTE	93.93 ± 1.17	90.35 ± 1.72	86.78 ± 1.06	75.49 ± 2.54	96.22 ± 0.75	95.66 ± 0.45	88.73 ± 1.80	88.16 ± 3.21	68.28 ± 7.23	79.28 ± 0.70	75.43 ± 0.65	63.59 ± 1.71	99.58 ± 0.89	96.16 ± 5.95	93.51 ± 4.44	66.65 ± 1.53	59.19 ± 0.96	43.08 ± 2.60	
CTGAN	93.12 ± 1.09	89.00 ± 2.25	83.11 ± 6.15	72.32 ± 3.33	96.02 ± 0.56	95.89 ± 0.29	88.68 ± 2.06	88.38 ± 2.78	63.18 ± 5.83	78.96 ± 0.65	75.16 ± 0.58	62.73 ± 1.60	95.93 ± 2.78	53.3 ± 18.55	31.77 ± 15.49	**67.11 ± 1.02**	**59.20 ± 0.56**	**44.93 ± 4.51**	
SMOTE-CDNN	93.71 ± 0.90	90.11 ± 2.09	87.50 ± 1.7	**76.16 ± 3.47**	96.34 ± 0.79	95.72 ± 0.55	88.22 ± 2.48	87.67 ± 3.65	69.19 ± 4.53	79.27 ± 0.69	75.49 ± 0.75	63.51 ± 1.75	**99.58 ± 0.72**	96.14 ± 6.60	**96.10 ± 4.86**	66.71 ± 1.03	58.90 ± 0.62	43.07 ± 3.94	
TabDDPM	92.71 ± 0.90	88.89 ± 1.41	83.02 ± 2.38	67.66 ± 4.78	95.46 ± 0.66	95.70 ± 0.40	86.68 ± 3.42	87.98 ± 3.50	80.65 ± 1.83	78.93 ± 0.42	74.80 ± 0.25	63.57 ± 1.05	72.67 ± 6.60	6.35 ± 1.90	0.00 ± 0.00	65.05 ± 2.51	56.64 ± 2.09	36.98 ± 5.44	
MDLM	**94.10 ± 1.04**	**91.11 ± 2.41**	87.18 ± 1.7	76.07 ± 4.58	**96.55 ± 0.72**	95.82 ± 0.31	86.49 ± 2.88	87.03 ± 3.51	68.31 ± 4.20	78.79 ± 0.89	74.75 ± 0.80	56.24 ± 2.00	96.2 ± 2.34	49.9 ± 18.63	18.99 ± 13.19	66.13 ± 1.73	58.41 ± 1.26	37.87 ± 5.58	
TANDEM	92.64 ± 0.94	88.2 ± 1.4	83.32 ± 1.42	75.03 ± 4.17	96.14 ± 1.04	95.90 ± 0.36	86.00 ± 1.64	85.88 ± 2.57	71.67 ± 2.71	76.81 ± 1.33	72.47 ± 1.67	55.52 ± 6.34	95.01 ± 4.39	44.48 ± 30.16	33.43 ± 32.07	65.29 ± 1.48	57.10 ± 1.52	40.86 ± 3.86	
DIDA	**89.54 ± 1.82**	**79.75 ± 2.77**	**86.5 ± 2.2**	**66.81 ± 3.01**	**94.46 ± 0.48**	**94.53 ± 0.80**	**79.91 ± 2.48**	**77.25 ± 2.61**	**82.40 ± 2.15**	**64.22 ± 2.77**	**53.31 ± 2.68**	**59.50 ± 2.87**	88.98 ± 11.87	67.15 ± 31.89	78.37 ± 24.23	**56.69 ± 0.50**	**46.47 ± 0.34**	**49.42 ± 1.14**	
SMOTE	84.34 ± 3.11	68.31 ± 4.28	79.09 ± 3.65	62.09 ± 3.75	93.72 ± 0.58	92.61 ± 0.80	74.85 ± 3.66	74.10 ± 3.58	73.47 ± 4.28	61.26 ± 0.91	50.89 ± 0.74	55.17 ± 1.10	83.18 ± 11.01	62.54 ± 28.78	74.81 ± 22.52	54.65 ± 1.33	45.15 ± 0.84	47.75 ± 0.94	
CTGAN	80.77 ± 1.46	64.98 ± 2.1	75.08 ± 1.8	58.84 ± 5.9	93.22 ± 0.92	93.57 ± 1.19	71.54 ± 3.47	71.32 ± 2.98	68.28 ± 5.03	60.53 ± 0.51	50.32 ± 0.45	54.20 ± 0.60	72.83 ± 7.57	23.4 ± 14.17	43.42 ± 15.22	55.36 ± 1.22	45.60 ± 0.76	48.10 ± 1.70	
SMOTE-CDNN	84.04 ± 2.14	68.29 ± 3.43	78.88 ± 2.65	63.2 ± 5.13	93.89 ± 0.79	92.49 ± 0.90	72.95 ± 3.25	72.06 ± 2.80	71.83 ± 4.40	61.16 ± 0.85	50.78 ± 0.65	55.17 ± 1.23	**90.35 ± 7.48**	**73.49 ± 15.03**	**84.41 ± 10.04**	54.93 ± 0.83	45.32 ± 0.52	47.80 ± 0.59	Decision tree
TabDDPM	81.57 ± 2.2	65.55 ± 2.99	75.99 ± 2.74	58.3 ± 6.66	93.14 ± 1.02	91.11 ± 2.03	71.00 ± 5.44	70.07 ± 4.58	70.84 ± 6.99	61.62 ± 0.76	51.21 ± 0.58	55.24 ± 1.24	72.47 ± 8.38	12.13 ± 5.97	28.63 ± 8.8	55.64 ± 0.45	45.76 ± 0.28	48.28 ± 0.89	
MDLM	81.87 ± 2.13	65.41 ± 2.91	76.26 ± 2.59	62 ± 4.75	93.71 ± 0.75	91.70 ± 1.30	67.22 ± 7.45	67.25 ± 5.69	64.99 ± 10.75	60.17 ± 1.04	50.07 ± 0.84	53.48 ± 1.11	84.62 ± 15.39	38.22 ± 29.08	55.37 ± 21.52	55.17 ± 1.16	45.47 ± 0.71	48.38 ± 1.61	
TANDEM	85.97 ± 2.8	71.97 ± 3.74	81.4 ± 3.25	60.38 ± 2.59	93.45 ± 0.4	92.60 ± 1.47	73.91 ± 4.63	72.80 ± 3.85	73.38 ± 6.58	61.12 ± 0.74	50.83 ± 0.57	54.44 ± 1.17	81.17 ± 7.73	34.39 ± 17.7	54.84 ± 16.77	55.09 ± 0.68	45.41 ± 0.41	47.85 ± 1.76	
DIDA	**93.07 ± 0.88**	**88.41 ± 1.46**	**79.73 ± 2.55**	77.19 ± 5.03	96.41 ± 1.05	95.76 ± 0.00	**93.45 ± 0.83**	**93.64 ± 1.01**	**89.22 ± 1.78**	**79.13 ± 0.30**	**74.27 ± 0.26**	**66.75 ± 0.59**	98.84 ± 1.11	**79.43 ± 15.8**	**28.91 ± 24.3**	**64.28 ± 0.49**	**58.10 ± 0.20**	**42.58 ± 4.06**	
SMOTE	91.87 ± 0.93	86.57 ± 1.78	78.79 ± 2.64	**78.56 ± 3.48**	**96.90 ± 0.76**	**95.76 ± 0.00**	88.79 ± 1.77	90.01 ± 2.14	75.97 ± 1.94	78.19 ± 0.68	73.58 ± 0.99	62.12 ± 1.31	**98.92 ± 0.99**	77.00 ± 8.76	8.16 ± 13.94	60.69 ± 0.51	54.40 ± 0.49	38.71 ± 1.44	
CTGAN	90.14 ± 1.17	83.27 ± 1.69	68.75 ± 5.19	73.92 ± 5.01	96.06 ± 1.25	95.76 ± 0.00	86.62 ± 3.52	89.58 ± 2.35	73.23 ± 2.07	77.95 ± 0.79	73.37 ± 1.01	62.25 ± 2.09	96.32 ± 2.99	55.29 ± 16.24	7.14 ± 18.90	60.87 ± 0.24	54.49 ± 0.22	39.69 ± 5.29	
SMOTE-CDNN	92.16 ± 1.07	86.79 ± 2.16	77.70 ± 3.13	76.06 ± 1.66	96.21 ± 0.52	95.76 ± 0.00	88.99 ± 2.11	90.63 ± 2.23	77.22 ± 0.96	78.14 ± 0.52	73.59 ± 0.83	62.16 ± 1.08	98.36 ± 1.78	74.39 ± 6.50	8.16 ± 13.94	60.78 ± 0.64	54.46 ± 0.73	38.67 ± 2.24	Extra Trees
TabDDPM	91.10 ± 1.57	85.27 ± 2.29	71.01 ± 3.76	66.68 ± 5.37	95.16 ± 0.88	95.76 ± 0.00	83.51 ± 3.87	86.94 ± 3.36	74.78 ± 2.16	78.08 ± 0.49	73.27 ± 0.77	62.09 ± 0.68	81.74 ± 3.65	11.81 ± 3.39	2.20 ± 5.81	60.85 ± 0.45	54.47 ± 0.47	38.61 ± 2.52	
MDLM	91.84 ± 0.87	86.53 ± 1.86	75.02 ± 2.08	75.31 ± 4.19	96.32 ± 0.91	95.76 ± 0.00	84.24 ± 2.35	87.26 ± 2.57	73.74 ± 1.65	77.20 ± 0.78	72.18 ± 1.22	57.04 ± 1.76	98.48 ± 0.81	65.28 ± 19.28	0.00 ± 0.00	60.47 ± 0.50	54.35 ± 0.6	36.2 ± 2.00	
TANDEM	91.46 ± 1.29	85.95 ± 2.26	74.68 ± 3.42	76.41 ± 2.45	96.61 ± 0.42	95.76 ± 0.00	85.17 ± 3.15	87.67 ± 2.63	74.51 ± 1.7	77.42 ± 1.08	72.43 ± 1.50	58.54 ± 3.24	97.81 ± 1.05	56.31 ± 15.56	0.00 ± 0.00	59.97 ± 0.75	53.64 ± 0.83	36.38 ± 3.06	
DIDA	**95.06 ± 1.00**	**93.04 ± 1.81**	**91.32 ± 0.86**	80.31 ± 3.4	96.92 ± 0.99	**96.46 ± 0.46**	**92.89 ± 0.52**	**93.32 ± 1.44**	**88.53 ± 1.37**	**80.17 ± 0.48**	**75.93 ± 0.43**	**68.07 ± 0.59**	**99.79 ± 0.57**	**98.15 ± 4.9**	**97.4 ± 4.44**	**68.40 ± 0.42**	**60.49 ± 0.37**	**48.85 ± 3.48**	
SMOTE	94.85 ± 0.96	92.42 ± 1.97	90.66 ± 1.84	78.95 ± 3.57	96.79 ± 1.17	96.43 ± 0.48	88.09 ± 1.92	88.49 ± 3.1	76.49 ± 2.08	79.75 ± 0.66	75.66 ± 0.83	64.60 ± 1.76	99.29 ± 1.08	94.64 ± 6.82	93.25 ± 7.4	67.06 ± 0.52	59.04 ± 0.56	47.39 ± 1.49	
CTGAN	94.62 ± 1.60	92.17 ± 2.53	87.90 ± 2.94	77.08 ± 3.93	96.37 ± 0.96	95.96 ± 0.26	87.65 ± 3.61	88.37 ± 3.8	74.79 ± 1.93	79.55 ± 0.75	75.48 ± 0.99	64.15 ± 1.92	98.78 ± 1	83.33 ± 16.26	69.68 ± 19.04	67.33 ± 0.32	59.15 ± 0.26	45.42 ± 4.52	
SMOTE-CDNN	94.81 ± 0.78	92.79 ± 1.58	89.95 ± 1.62	80.06 ± 3.09	97.00 ± 1.08	96.33 ± 0.25	87.35 ± 2.41	88.21 ± 3.35	76.62 ± 2.87	79.65 ± 0.51	75.50 ± 0.55	64.89 ± 1.59	99.33 ± 1.17	95.98 ± 6.88	91.34 ± 11.79	66.86 ± 0.40	58.56 ± 0.43	47.26 ± 1.94	Gradient Boosting
TabDDPM	94.53 ± 0.71	92.32 ± 1.34	89.60 ± 1.19	75.24 ± 3.84	96.64 ± 0.60	96.07 ± 0.19	84.42 ± 5.02	86.81 ± 4.22	76.12 ± 3.49	79.60 ± 0.50	75.29 ± 0.51	64.44 ± 0.95	85.08 ± 5.74	17.45 ± 7.93	9.8 ± 17.05	67.14 ± 0.25	58.72 ± 0.28	46.65 ± 1.53	
MDLM	94.87 ± 0.99	92.95 ± 1.52	89.03 ± 0.81	**80.63 ± 5.14**	**97.20 ± 1.11**	96.32 ± 0.54	84.55 ± 3.1	86.36 ± 3.92	71.49 ± 4.66	79.35 ± 0.63	74.56 ± 0.79	59.24 ± 2.50	98.96 ± 0.97	84.57 ± 15.09	67.27 ± 14.82	67.15 ± 0.26	58.81 ± 0.27	44.57 ± 1.99	
TANDEM	95.00 ± 0.83	92.62 ± 1.69	90.47 ± 1.16	77.16 ± 2.51	96.55 ± 0.62	96.15 ± 0.28	86.23 ± 1.97	87.21 ± 2.53	76.72 ± 2.56	78.62 ± 1.03	74.06 ± 1.27	58.43 ± 4.19	98.82 ± 1.27	73.1 ± 23.64	57.66 ± 27.44	66.73 ± 0.53	58.40 ± 0.80	44.38 ± 1.99	
DIDA	52.38 ± 1.30	37.21 ± 0.71	31.18 ± 2.85	55.65 ± 4.00	92.79 ± 0.63	94.79 ± 0.42	**74.02 ± 1.99**	73.08 ± 2.61	**74.56 ± 2.59**	**59.92 ± 0.73**	**50.85 ± 0.51**	49.69 ± 1.04	93.86 ± 7.04	52.83 ± 20.21	49.15 ± 19.77	**59.21 ± 0.38**	**50.47 ± 0.52**	45.01 ± 2.46	
SMOTE	52.08 ± 1.24	36.97 ± 0.43	33.90 ± 2.64	57.25 ± 5.31	93.05 ± 0.87	94.60 ± 0.37	71.89 ± 1.73	72.13 ± 1.73	71.16 ± 1.64	57.29 ± 0.9	48.83 ± 0.8	**50.35 ± 1.67**	94.87 ± 6.79	**55.28 ± 11.91**	**52.32 ± 11.91**	57.56 ± 0.15	48.74 ± 0.26	45.89 ± 0.67	
CTGAN	53.02 ± 2.85	37.46 ± 1.70	32.96 ± 5.25	57.06 ± 4.14	92.97 ± 0.67	**95.61 ± 0.21**	74.01 ± 2.35	73.47 ± 2.07	72.74 ± 3.73	57.76 ± 0.79	49.08 ± 0.75	49.98 ± 1.25	91.73 ± 10.08	49.42 ± 21.17	42.11 ± 16.51	57.29 ± 0.37	48.09 ± 0.33	45.06 ± 2.20	
SMOTE-CDNN	**53.53 ± 1.96**	**37.98 ± 1.25**	35.25 ± 1.65	**58.7 ± 6.76**	**93.28 ± 1.16**	94.67 ± 0.45	72.02 ± 3.81	71.62 ± 3.2	72.41 ± 3.14	56.72 ± 0.51	48.36 ± 0.49	49.40 ± 2.12	**95.01 ± 6.77**	54.31 ± 11.97	50.52 ± 13.52	57.51 ± 0.42	48.49 ± 0.36	46.71 ± 0.85	k-NN
TabDDPM	52.72 ± 2.61	37.14 ± 1.78	**36.62 ± 4.97**	55.53 ± 6.42	92.77 ± 1.14	92.57 ± 1.83	72.64 ± 2.42	73.07 ± 2.1	70.83 ± 1.6	58.05 ± 0.78	49.22 ± 0.38	49.69 ± 0.86	76.12 ± 12.37	14.31 ± 9.66	16.91 ± 6.66	57.58 ± 0.51	48.47 ± 0.52	**47.63 ± 1.81**	
MDLM	52.29 ± 0.92	37.13 ± 0.79	32.15 ± 2.83	57.23 ± 6.09	93.02 ± 0.99	95.57 ± 0.27	73.09 ± 4.11	**73.7 ± 3.31**	72.69 ± 3.31	57.83 ± 0.76	49.19 ± 0.46	48.37 ± 0.82	92.06 ± 5.83	43.56 ± 20.77	34.39 ± 24.82	57.33 ± 0.32	48.66 ± 0.49	45.26 ± 0.85	
TANDEM	52.86 ± 2.15	37.10 ± 1.58	33.87 ± 3.22	56.87 ± 4.06	92.95 ± 0.65	95.47 ± 0.34	71.22 ± 3.15	71.85 ± 3.02	71.33 ± 1.65	57.94 ± 1.21	49.19 ± 0.75	48.07 ± 1.65	94.54 ± 4.12	45.82 ± 8.72	43.57 ± 12.49	56.44 ± 0.68	47.36 ± 0.79	42.57 ± 2.38	
DIDA	**89.69 ± 0.98**	**83.14 ± 1.99**	**76.41 ± 1.25**	**78.73 ± 4.16**	**97.09 ± 1.04**	95.52 ± 0.55	91.38 ± 1.20	91.46 ± 1.85	**86.3 ± 0.86**	80.58 ± 0.57	76.62 ± 0.51	**68.35 ± 0.72**	97.41 ± 1.82	**73.45 ± 9.62**	**62.38 ± 13.34**	**61.19 ± 0.44**	**56.05 ± 0.41**	**38.66 ± 5.08**	

Method	Alzheimers Disease Data			Chronic Kidney Disease data			Heart			Healthcare Synthetic Data			Teen Mental Health Dataset			Urban Street Food Vendor Survival Dataset			Model
	AUC-ROC (Mean±Std)	AUC-PR (Mean±Std)	F1 (Mean±Std)	AUC-ROC (Mean±Std)	AUC-PR (Mean±Std)	F1 (Mean±Std)	AUC-ROC (Mean±Std)	AUC-PR (Mean±Std)	F1 (Mean±Std)	AUC-ROC (Mean±Std)	AUC-PR (Mean±Std)	F1 (Mean±Std)	AUC-ROC (Mean±Std)	AUC-PR (Mean±Std)	F1 (Mean±Std)	AUC-ROC (Mean±Std)	AUC-PR (Mean±Std)	F1 (Mean±Std)	
SMOTE	89.35 ± 0.74	82.63 ± 1.35	75.83 ± 1.08	77.61 ± 3.38	96.86 ± 1.04	94.92 ± 0.56	91.49 ± 1.02	91.23 ± 1.84	70.05 ± 6.16	**80.65 ± 0.63**	76.74 ± 0.60	64.31 ± 1.87	**97.57 ± 1.84**	67.84 ± 16.12	60.64 ± 13.14	60.79 ± 0.53	54.96 ± 0.57	26.25 ± 3.66	Logistic Regression
CTGAN	88.59 ± 0.84	81.90 ± 1.39	72.29 ± 3.47	72.91 ± 6.06	96.15 ± 1.17	**95.80 ± 0.35**	91.12 ± 1.57	90.92 ± 2.04	64.52 ± 10.23	80.07 ± 0.53	76.02 ± 0.66	63.29 ± 3.58	94.98 ± 2.18	53.8 ± 10.05	47.86 ± 13.31	59.56 ± 0.30	53.77 ± 0.27	27.92 ± 10.31	
SMOTE-CDNN	89.25 ± 0.68	82.50 ± 1.68	75.81 ± 1.75	77.59 ± 3.68	96.84 ± 1.12	94.99 ± 0.53	**91.59 ± 0.95**	**91.51 ± 1.71**	70.73 ± 4.62	80.65 ± 0.64	**76.76 ± 0.59**	64.50 ± 1.57	97.46 ± 1.62	65.15 ± 14.58	58.78 ± 15.14	60.75 ± 0.66	54.93 ± 0.77	26.93 ± 4.44	
TabDDPM	86.81 ± 1.44	79.29 ± 1.42	72.32 ± 2.22	69.19 ± 5.34	95.77 ± 0.96	95.41 ± 0.60	89.17 ± 3.12	89.81 ± 3.53	82.66 ± 2.06	80.04 ± 0.57	76.00 ± 0.54	65.71 ± 1.05	62.56 ± 7.83	7.55 ± 6.37	5.21 ± 5.30	59.03 ± 1.05	53.80 ± 1.16	32.88 ± 7.50	
MDLM	89.00 ± 0.67	82.37 ± 1.03	75.59 ± 1.28	77.37 ± 5.81	96.81 ± 1.44	95.42 ± 0.32	89.22 ± 1.90	88.75 ± 2.87	72.7 ± 3.34	79.74 ± 0.62	75.72 ± 0.62	53.81 ± 3.00	96.4 ± 2.82	52.19 ± 25.67	42.94 ± 27.05	60.42 ± 0.62	54.74 ± 0.58	23.39 ± 3.21	
TANDEM	87.64 ± 0.92	79.79 ± 1.50	72.55 ± 2.62	75.85 ± 3.98	96.56 ± 1.15	95.32 ± 0.58	89.17 ± 1.64	88.93 ± 2.28	76.64 ± 5.34	77.02 ± 1.80	72.72 ± 1.72	52.24 ± 7.34	95.33 ± 3.92	48.8 ± 20.41	39.49 ± 24.50	57.90 ± 1.05	51.58 ± 1.95	22.45 ± 5.66	
DIDA	**88.57 ± 1.36**	**82.58 ± 2.09**	**71.79 ± 4.66**	**74.64 ± 3.82**	**96.69 ± 0.78**	93.92 ± 1.11	91.09 ± 1.39	**92.16 ± 1.43**	**85.76 ± 1.95**	80.44 ± 0.53	76.49 ± 0.47	**67.45 ± 3.34**	**98.22 ± 1.32**	**73.65 ± 16.87**	**60.95 ± 16.38**	**58.38 ± 2.26**	**53.45 ± 2.75**	36.9 ± 19.81	Neural Network
SMOTE	86.94 ± 1.22	79.82 ± 1.95	71.06 ± 2.69	71.44 ± 4.04	96.10 ± 0.84	94.10 ± 0.98	91.36 ± 0.7	91.86 ± 1.72	74.24 ± 8.18	**80.52 ± 0.6**	**76.65 ± 0.57**	60.12 ± 10.46	97.89 ± 1.27	63.63 ± 18.23	58.65 ± 11.47	57.21 ± 2.08	52.49 ± 1.77	34.53 ± 15.64	
CTGAN	85.44 ± 1.51	77.60 ± 2.63	66.59 ± 3.94	66.97 ± 7.13	95.29 ± 1.71	**94.55 ± 1.34**	91.25 ± 0.82	91.8 ± 1.37	68.8 ± 10.77	79.78 ± 0.48	75.80 ± 0.53	50.99 ± 13.83	97.00 ± 1.88	60.21 ± 17.8	40.47 ± 24.82	55.41 ± 4.99	50.48 ± 5.83	31.58 ± 23.39	
SMOTE-CDNN	86.85 ± 1.6	79.49 ± 2.38	69.69 ± 3.04	71.21 ± 6.48	95.89 ± 1.54	94.17 ± 1.23	**91.42 ± 1.13**	91.91 ± 1.96	77.49 ± 6.39	80.48 ± 0.69	76.58 ± 0.63	56.67 ± 10.9	97.42 ± 1.52	58.34 ± 20.58	47.35 ± 22.4	56.87 ± 1.94	52.7 ± 1.60	34.46 ± 12.41	
TabDDPM	84.10 ± 1.95	75.80 ± 2.76	65.84 ± 4.97	65.05 ± 6.03	94.97 ± 1.38	93.40 ± 2.28	89.21 ± 2.68	90.17 ± 3.12	82.35 ± 4.84	79.78 ± 0.55	75.78 ± 0.47	62.73 ± 9.60	86.59 ± 5.80	26.7 ± 15.98	24.34 ± 8.16	55.33 ± 2.90	51.03 ± 3.11	**38.13 ± 20.72**	
MDLM	86.04 ± 1.17	78.50 ± 2.08	66.89 ± 3.5	72.18 ± 6.65	96.19 ± 1.28	93.52 ± 1.91	89.75 ± 1.44	90.07 ± 2.35	78.13 ± 4.92	79.55 ± 0.48	75.48 ± 0.55	47.68 ± 20.33	97.52 ± 2.16	58.37 ± 27.81	43.31 ± 22.34	55.59 ± 3.55	50.89 ± 4.34	37.21 ± 26.2	
TANDEM	85.19 ± 1.63	76.21 ± 2.66	68.29 ± 8.69	71.98 ± 4.31	96.25 ± 0.96	93.38 ± 2.05	89.75 ± 1.21	89.69 ± 1.89	82.89 ± 3.42	77.84 ± 1.35	73.73 ± 1.15	57.65 ± 14.31	96.21 ± 2.85	51.68 ± 15.12	34.54 ± 12.41	53.87 ± 3.46	48.46 ± 4.22	36.98 ± 18.00	
DIDA	**95.12 ± 1.17**	**91.83 ± 2.28**	88.81 ± 2.46	**78.85 ± 2.98**	96.63 ± 0.83	95.76 ± 0.09	**92.42 ± 1.47**	**91.67 ± 1.99**	**88.17 ± 1.75**	**78.74 ± 0.55**	**73.69 ± 0.65**	**66.42 ± 1.04**	98.53 ± 2.43	88.19 ± 14.52	**49.25 ± 32.32**	**65.62 ± 0.95**	**58.4 ± 0.63**	**46.32 ± 3.3**	Random Forest
SMOTE	94.93 ± 1.12	91.62 ± 1.97	**88.99 ± 0.92**	77.98 ± 3.2	96.69 ± 0.93	95.87 ± 0.12	88.32 ± 2.05	89.38 ± 2.43	77.41 ± 1.37	78.06 ± 0.68	73.50 ± 1.05	63.19 ± 1.49	99.23 ± 0.87	88.83 ± 8.49	46.26 ± 13.53	64.17 ± 0.98	57.09 ± 0.66	43.55 ± 1.25	
CTGAN	94.65 ± 1.03	90.86 ± 1.99	85.51 ± 3.04	75.01 ± 1.9	96.25 ± 0.4	95.76 ± 0.00	87.60 ± 2.59	89.20 ± 2.08	74.32 ± 2.05	77.85 ± 0.77	73.38 ± 1.05	62.77 ± 1.90	96.65 ± 2.76	67.63 ± 22.37	19.39 ± 19.65	64.63 ± 0.69	57.11 ± 0.61	43.20 ± 4.06	
SMOTE-CDNN	95.05 ± 1.00	91.10 ± 2.55	88.88 ± 1.52	77.59 ± 2.9	96.49 ± 0.75	95.87 ± 0.14	88.04 ± 2.57	89.23 ± 2.91	78.01 ± 1.04	77.83 ± 0.52	73.20 ± 1.01	63.12 ± 1.18	98.50 ± 2.47	88.39 ± 11.95	24.83 ± 26.65	64.34 ± 0.94	56.85 ± 0.53	43.69 ± 1.83	
TabDDPM	94.05 ± 0.99	90.46 ± 1.37	83.04 ± 2.71	71.6 ± 7.09	96.08 ± 1.25	95.76 ± 0.00	86.35 ± 2.98	88.42 ± 3.28	78.30 ± 1.22	78.22 ± 0.44	73.55 ± 0.73	63.19 ± 0.65	89.45 ± 3.60	15.57 ± 2.99	0.00 ± 0.00	64.93 ± 0.64	57.19 ± 0.35	43.30 ± 0.79	
MDLM	94.64 ± 1.07	91.57 ± 1.91	86.55 ± 2.21	77.73 ± 3.75	96.67 ± 1.04	**95.87 ± 0.07**	86.14 ± 2.74	87.46 ± 2.78	77.56 ± 2.23	77.54 ± 0.40	72.50 ± 0.74	59.71 ± 0.73	97.96 ± 1.71	76.14 ± 18.34	4.08 ± 10.80	64.53 ± 0.85	57.19 ± 0.33	41.65 ± 1.71	
TANDEM	94.94 ± 1.24	91.53 ± 2.10	87.55 ± 1.34	78.02 ± 3.63	**96.83 ± 0.86**	95.80 ± 0.07	86.73 ± 2.58	87.73 ± 2.77	78.76 ± 1.68	77.37 ± 1.00	72.51 ± 1.52	59.58 ± 2.54	**99.25 ± 1.12**	**88.84 ± 15.31**	20.75 ± 28.13	64.59 ± 0.72	56.90 ± 0.41	40.28 ± 1.83	
DIDA	94.59 ± 1.19	91.53 ± 2.8	91.35 ± 2.13	78.05 ± 4.87	**96.72 ± 1.04**	96.00 ± 0.45	**92.76 ± 0.99**	**92.51 ± 1.96**	**87.4 ± 1.35**	**79.05 ± 0.77**	74.12 ± 1.58	**66.72 ± 1.16**	**99.71 ± 0.77**	**98.03 ± 5.22**	**94.81 ± 4.86**	**66.88 ± 0.57**	**58.89 ± 0.67**	45.22 ± 5.01	XGBoost
SMOTE	94.68 ± 0.80	91.46 ± 1.54	**91.83 ± 1.23**	76.41 ± 3.89	96.43 ± 0.89	96.07 ± 0.24	84.88 ± 3.36	84.41 ± 4.16	76.84 ± 3.68	78.72 ± 0.65	74.25 ± 0.85	63.11 ± 1.86	99.58 ± 0.74	96.31 ± 6.32	93.25 ± 7.40	65.04 ± 0.83	56.87 ± 0.72	44.11 ± 3.07	
CTGAN	**94.77 ± 1.02**	91.95 ± 3.12	88.01 ± 2.65	73.73 ± 5.99	96.01 ± 1.15	95.96 ± 0.30	84.48 ± 5.26	86.30 ± 4.40	74.81 ± 4.19	78.66 ± 0.65	**74.53 ± 1.18**	62.93 ± 1.60	98.42 ± 1.18	69.86 ± 18.17	56.58 ± 14.76	65.75 ± 0.59	57.50 ± 0.60	43.03 ± 4.63	
SMOTE-CDNN	94.43 ± 0.64	91.1 ± 2.05	91.65 ± 1.15	78.16 ± 2.25	96.19 ± 1.01	**96.22 ± 0.26**	86.45 ± 2.63	88.24 ± 2.30	77.85 ± 2.03	78.59 ± 0.61	74.28 ± 0.65	63.54 ± 1.55	99.34 ± 1.14	95.97 ± 6.88	88.75 ± 10.31	65.00 ± 0.52	56.57 ± 0.50	45.03 ± 2.96	
TabDDPM	94.52 ± 0.84	**92.77 ± 1.51**	90.22 ± 0.86	73.79 ± 4.70	96.32 ± 0.94	95.92 ± 0.33	84.15 ± 4.98	85.90 ± 5.58	76.37 ± 2.93	78.51 ± 0.68	73.90 ± 0.78	63.05 ± 0.88	93.46 ± 3.77	44.2 ± 16.85	39.18 ± 16.61	65.32 ± 0.53	56.98 ± 0.53	**46.74 ± 2.36**	
MDLM	94.24 ± 1.14	91.93 ± 1.67	89.82 ± 2.05	**78.23 ± 4.51**	96.55 ± 1.19	96.08 ± 0.45	81.98 ± 5.26	83.41 ± 4.93	73.19 ± 7.21	78.15 ± 0.60	73.24 ± 0.96	58.98 ± 2.12	99.45 ± 0.54	90.14 ± 8.68	80.43 ± 11.55	65.58 ± 0.42	57.30 ± 0.51	42.22 ± 2.73	
TANDEM	94.48 ± 1.09	92.19 ± 1.74	90.14 ± 1.93	74.98 ± 2.31	96.09 ± 0.78	95.89 ± 0.40	84.36 ± 3.21	85.36 ± 3.63	73.87 ± 3.60	77.96 ± 0.94	73.40 ± 1.34	59.45 ± 2.64	99.68 ± 0.11	91.38 ± 2.40	65.32 ± 23.50	64.94 ± 0.76	56.38 ± 0.84	43.25 ± 3.3	
DIDA	**83.72 ± 1.13**	**74.38 ± 1.68**	**69.06 ± 2.24**	58.69 ± 7.44	93.86 ± 1.21	94.55 ± 0.54	**87.79 ± 3.25**	**88.54 ± 2.80**	**83.48 ± 3.53**	**78.25 ± 0.66**	**73.34 ± 0.56**	**66.79 ± 0.88**	**92.82 ± 8.56**	**50.39 ± 31.85**	**35.63 ± 29.86**	**60.08 ± 1.39**	**53.91 ± 1.42**	**45.68 ± 2.99**	TabNet
SMOTE	79.12 ± 2.41	68.74 ± 2.60	62.49 ± 3.57	**58.72 ± 2.72**	**93.94 ± 0.96**	94.53 ± 0.68	65.63 ± 21.05	69.77 ± 17.78	66.32 ± 6.19	75.09 ± 0.9	69.19 ± 1.46	61.18 ± 2.48	85.65 ± 9.1	35.42 ± 24.04	22.26 ± 30.58	57.53 ± 0.64	50.96 ± 0.58	43.44 ± 2.34	
CTGAN	75.88 ± 3.05	66.22 ± 4.24	59.72 ± 2.81	56.57 ± 3.52	93.62 ± 1.24	**94.68 ± 0.93**	61.84 ± 19.75	65.30 ± 18.02	67.79 ± 7.74	75.45 ± 0.95	69.53 ± 1.10	60.19 ± 3.61	73.31 ± 8.05	16.19 ± 10.16	8.11 ± 8.41	57.14 ± 0.44	49.68 ± 0.82	41.27 ± 3.45	
SMOTE-CDNN	78.41 ± 1.84	67.65 ± 2.63	60.97 ± 2.62	56.45 ± 4.34	93.24 ± 0.63	94.77 ± 0.63	64.30 ± 23.16	69.80 ± 20.03	69.84 ± 7.45	74.5 ± 1.41	67.56 ± 2.24	60.72 ± 3.11	81.34 ± 9.54	35.44 ± 11.35	28.77 ± 14.28	56.97 ± 0.38	50.13 ± 1.03	44.17 ± 2.73	
TabDDPM	75.59 ± 4.22	64.69 ± 4.71	61.67 ± 3.69	55.10 ± 4.47	93.12 ± 0.96	93.21 ± 3.77	52.14 ± 12.80	59.26 ± 12.41	58.51 ± 12.89	76.00 ± 0.55	69.78 ± 0.78	61.59 ± 1.24	59.58 ± 6.4	13.16 ± 11.05	10.93 ± 8.14	55.45 ± 0.93	48.91 ± 0.77	44.71 ± 3.56	
MDLM	76.96 ± 1.00	65.05 ± 2.48	62.48 ± 2.45	55.34 ± 4.28	93.50 ± 0.82	94.05 ± 1.32	68.18 ± 16.02	72.52 ± 13.30	67.52 ± 4.31	73.37 ± 1.32	66.43 ± 1.53	52.44 ± 3.61	86.31 ± 12.45	35.54 ± 21.33	35.52 ± 24.22	56.08 ± 0.92	49.32 ± 1.67	42.02 ± 2.45	
TANDEM	77.85 ± 3.05	67.19 ± 3.71	62.32 ± 3.34	57.58 ± 4.33	93.52 ± 1.00	93.77 ± 1.86	57.90 ± 20.37	64.08 ± 18.12	64.95 ± 6.09	75.61 ± 1.54	69.86 ± 1.47	58.03 ± 3.03	74.52 ± 13.24	19.53 ± 16.32	16.73 ± 23.35	56.08 ± 0.87	49.00 ± 1.23	41.24 ± 2.69	

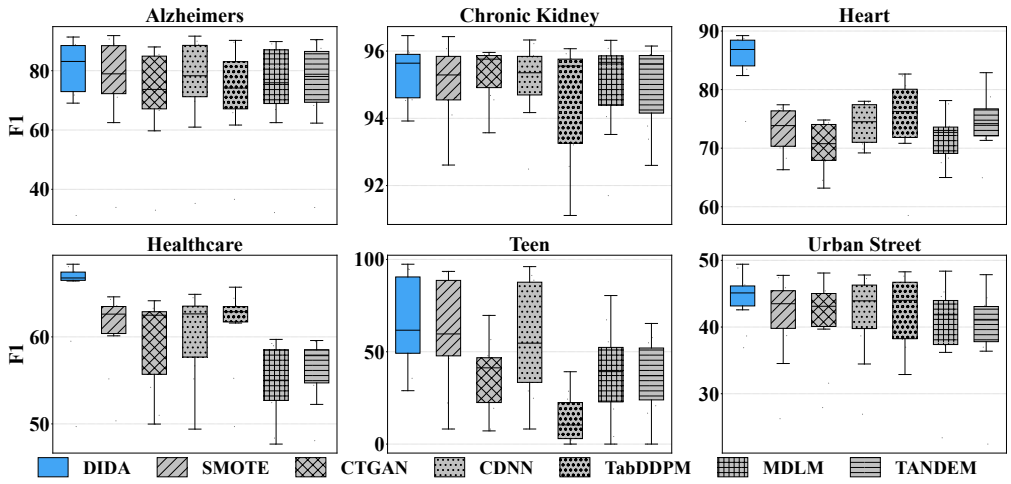

Figure 8: F1 box plots across datasets and methods: DIDA's median outperforms the baselines with tighter spread, and where a baseline matches DIDA's median its spread is several times wider, the same instability signature seen in the AUC-ROC panel.

Table 13: Ablation study results on the Heart dataset (mean ± std over 7 seeds).

Method	AUC-ROC	AUC-PR	F1 Score
−Feature	86.44 ± 8.99	86.63 ± 9.02	83.08 ± 6.30
−Sample	87.81 ± 6.08	87.23 ± 6.60	84.38 ± 4.91
−N1	87.46 ± 6.84	87.51 ± 7.45	83.41 ± 5.42
−N2	87.42 ± 6.83	87.56 ± 7.41	83.23 ± 5.46
−N3	85.67 ± 4.83	86.27 ± 4.96	79.57 ± 4.45
−Table	87.03 ± 8.99	86.90 ± 8.96	83.74 ± 6.40
DIDA (Full)	**88.83 ± 1.60**	**88.63 ± 1.89**	**85.46 ± 1.99**

The complete matrix confirms what Table 1 compresses: DIDA is best in 40 of the 60 AUC-ROC scenarios and 42 of the 60 F1 scenarios, and its advantage concentrates in the tree-ensemble and instance-based classifiers, where the semantic consistency of the augmented samples matters most, while the deep models remain the noisiest column. The per-scenario standard deviations, mostly within two points across the seven seeds, indicate that these gains are a stable property of the augmentation rather than an artifact of a favorable seed.

The box plots (Figures 5, 8, 9) make the same point visually: DIDA's medians sit above the baselines' and its whiskers are tighter across nearly every dataset-method combination, and where a baseline matches DIDA's median its spread is several times wider, the visual signature of the instability documented in Table 12. The F1 and AUC-PR panels agree with the AUC-ROC panel, so the advantage is not an artifact of any single threshold-free metric.

F.7.3 Ablation Study Results

Table 13 reports the full ablation study results on the Heart dataset (mean ± std over 7 seeds), complementing Figure 6 in Section 5.

Table 13 sharpens the picture with the per-metric numbers. Removing the dynamic count regulation (−N3) is the single most damaging ablation, with AUC-ROC falling 3.16 points and F1 falling 5.89, and the fact that −N1 and −N2 degrade far less shows that the loss is not the retention itself but the loss of differentiation: a fixed retention treats every sample alike, while DIDA's value-dependent count spends more capacity on high-contribution cells. The two joint-drive ablations are asymmetric, with dropping feature importance (−Feature, −2.39 AUC-ROC) hurting more than dropping sample contribution (−Sample, −1.02), consistent with the feature-level coordinate being

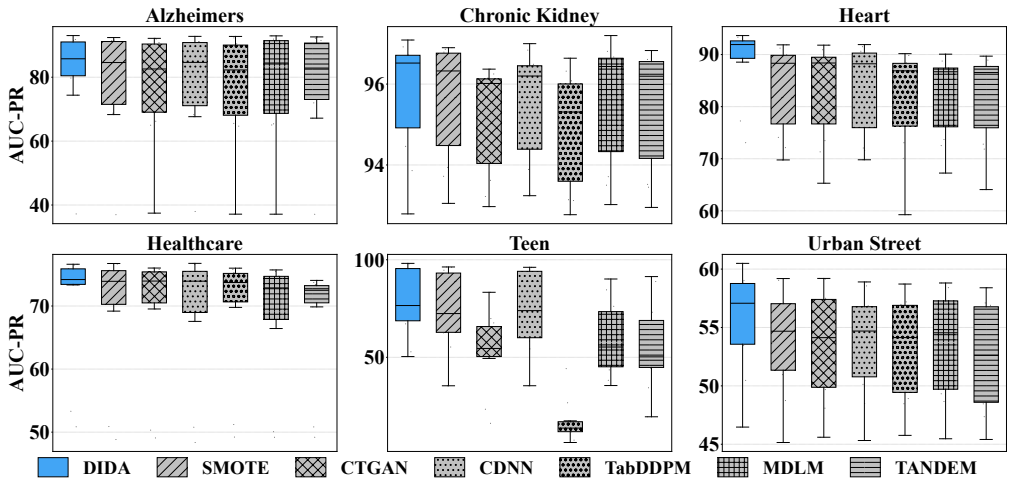

Figure 9: AUC-PR box plots across datasets and methods: DIDA's median and spread dominate the baselines, agreeing with the AUC-ROC and F1 panels and confirming the advantage is not an artifact of a single threshold-free metric.

the framework's dominant lever. The full model also carries the tightest standard deviation across the table, confirming that the modules stabilize one another rather than trading variance for accuracy.

F.7.4 Derived Properties

Beyond the raw numbers, four properties recur across the experiments and distinguish DIDA from a plain hyperparameter-tuning win, and each is stated so it can be falsified on new datasets.

Value-aware placement, not volume, is the lever. Uniform augmentation is not a substitute for value-aware augmentation. SMOTE, which perturbs every minority sample identically, trails DIDA by 1.94 points of average AUC-ROC at the same target sample count (Table 1), and removing only the dynamic count regulation ($-$N3, retention fixed at 3) costs 3.16 AUC-ROC and 5.89 F1 (Table 13), even though it alters no value coordinate. This is the empirical form of the claim that clean data is not always optimal. A classifier trained on the raw, unaugmented distribution leaves scarce minority regions under-sampled, and the remedy is synthetic points whose placement is graded by data value, not merely more synthetic points.

The three modules are complementary, not additive. The ablation losses in Table 13 do not add up linearly, which is evidence of synergy, not independence: each removal strikes a different failure mode (drift, over-augmentation, categorical collapse) instead of shaving off a fixed, predictable margin, and the full model's cross-seed standard deviation (± 1.60) is 3–5$\times$ tighter than any single ablation ($\pm 4.83 \sim \pm 8.99$), a stabilizing effect no single module reproduces.

Cross-seed variance is a first-class evaluation axis. Table 13 shows that the value-differentiated model achieves not only the best mean but the smallest variance. The single-coordinate ablations ($-$Feature ± 8.99, $-$Table ± 8.99) produce highly inconsistent samples across random conditions, whereas the joint two-coordinate score anchors augmentation to the value space and acts as a regularizer. Reporting means alone would hide this difference, so augmentation methods should be compared on the mean-variance frontier, not the mean alone.

Gains concentrate where the model is most sensitive to feature distortion. Downstream, tree ensembles, whose axis-aligned splits are the most easily corrupted by out-of-distribution synthetic features, gain the most (random forest 0.9345, extra trees 0.9370, XGBoost 0.9264 ROC), while deep models respond heterogeneously (MLP robust, TabNet benefiting from coverage). DIDA's advantage thus traces to the interaction between augmentation fidelity and model inductive bias, not to a uniform lift across all learners.