

Appendix

TabStruct: Measuring Structural Fidelity of Tabular Data

Table of Contents

A	Broader Impact Statement	24
B	Summary of Related Work	24
B.1	Conventional Evaluation Dimensions	24
B.2	Structural Fidelity of Tabular Data	25
B.3	Tabular Data Genertor	25
B.4	Evaluation Scope Comparison	25
C	Designs of Structural Fidelity Metrics	28
C.1	Conditional Independence (CI) Scores	28
C.1.1	Deriving CI Statements from a Causal Graph	28
C.1.2	Compute CI Scores on Tabular Data	29
C.2	Global Utility Score	29
C.2.1	Downstream Predictor Configurations	29
C.2.2	Pruning the Ensemble of Downstream Predictors	29
D	Rationales for Evaluation Framework Design	30
D.1	Structural Prior for Tabular Data	30
D.2	CPDAG-level Evaluation of Structural Fidelity	30
E	Reproducibility	31
E.1	Benchmark Datasets	31
E.1.1	SCM Datasets	31
E.1.2	Real-world Datasets	32
E.2	Data Processing	33
E.3	Implementations of Benchmark Generators	33
E.4	Hyperparameter Tuning for Downstream Predictors	37
E.5	Aggregation of Evaluation Results	37
E.6	Software and Computing Resources	37
F	TabStruct Open-source Benchmark Suite	38
G	Extended Analysis and Discussion	38
G.1	Extended Analysis on Validity of Global Utility	38
G.2	Extended Analysis on Structural Fidelity of Generators	38
G.3	Extended Analysis on Practicability of Global Utility	40
G.4	Practical Guidance	41
G.5	Future Work	41
H	Extended Experimental Results	42
H.1	Evaluation Results for SCM Datasets	42
H.1.1	Classification Datasets	42
H.1.2	Regression Datasets	43
H.2	Evaluation Results for Real-world Datasets	44
H.2.1	Classification Datasets	44
H.2.2	Regression Datasets	51

A Broader Impact Statement

This paper proposes integrating structural fidelity as a core evaluation dimension alongside conventional metrics for assessing tabular data generators. Specifically, we introduce *global utility*, a novel metric that evaluates the structural fidelity of synthetic tabular data without requiring access to the ground-truth causal structures. Furthermore, we present *TabStruct*, a comprehensive benchmark for tabular data generation that spans a wide evaluation scope – comprising 13 generators from nine distinct categories, evaluated on 29 datasets. Our benchmark results highlight that structural fidelity is an important yet previously underexplored evaluation dimension. It effectively captures whether generated data preserves the underlying causal structures present in real-world tabular datasets, serving as a valuable complement to existing evaluation dimensions.

This is particularly critical for tabular modalities, where visual inspection of data authenticity is not feasible, unlike in text or image domains [100]. The impact of our work extends to enabling broader machine learning applications in data-scarce domains. For instance, it can facilitate robust data analysis in clinical contexts where data collection is limited [61, 15, 67]. Enhancing the fidelity of synthetic data may promote the adoption of more advanced machine learning approaches and, ultimately, contribute to improving healthcare quality [67, 61].

B Summary of Related Work

As a supplement to Section 2, we provide a detailed summary of related work on tabular data generation. We begin by outlining the conventional evaluation dimensions for tabular generators (Appendix B.1). We then highlight the importance of assessing structural fidelity in the evaluation of such models (Appendix B.2). We further summarise existing tabular data generators (Appendix B.3). Finally, we present a comprehensive and quantitative comparison of the evaluation scope covered by TabStruct versus prior work, including both benchmarks and model studies (Appendix B.4).

B.1 Conventional Evaluation Dimensions

Density estimation assesses the discrepancy between the distributions of reference and synthetic data, considering both marginal (i.e., low-order) and joint (i.e., high-order) distributions [40, 48, 62, 83, 72]. A generator may achieve high performance on low-order metrics by sampling each feature independently, thereby ignoring inter-feature dependencies. While high-order metrics aim to measure sample-level similarity, they still fall short of explicitly revealing whether the synthetic data preserves the underlying causal structures present in the reference data.

Following prior studies [40, 80, 98], we evaluate density estimation using four metrics of two categories: (i) Low-order: *Shape* and *Trend* [93]. *Shape* measures the synthetic data’s ability to replicate each column’s marginal density. *Trend* assesses its capacity to capture correlations between different columns. (ii) High-order: α -precision and β -recall [4]. α -precision quantifies the similarity between the reference and synthetic data, and β -recall assesses the diversity of the synthetic data.

Privacy preservation evaluates the trade-off between the utility of synthetic data in downstream tasks and the risk of privacy leakage [61, 38, 88, 45, 99, 65, 66]. However, this dimension is often tailored to specific tasks (e.g., classification and regression), and as such, it does not directly evaluate the structural fidelity of tabular data. Consequently, privacy preservation alone cannot comprehensively assess a generator’s ability to capture the fundamental characteristics of tabular data, such as causal structures.

Following prior studies [61, 53, 101], we measure privacy preservation using two metrics: (i) *median Distance to Closest Record* (DCR) [101], where a higher DCR indicates that synthetic data is less likely to be directly copied from the reference data; (ii) δ -Presence [76]. We note that certain implementations of δ -Presence interpret smaller values as indicative of better privacy preservation; however, we adapt the implementation provided by Synthcity [76], wherein larger values correspond to improved privacy preservation.

ML efficacy measures the performance gap observed when replacing reference data with synthetic data in downstream tasks. This metric is inherently task-specific and can be heavily biased by the choice of predictive models and target variables. A useful parallel can be drawn from image generation: Mixup [75] enhances training data by interpolating between real samples, often improving downstream task performance. However, it simultaneously distorts the spatial structure of images, producing visually unrealistic outputs [70]. As illustrated in Figure 1, assessing the authenticity of synthetic tabular data is far more difficult than in image domains. Consequently, synthetic data that performs well in downstream tasks may still fail to preserve important causal structures of the

reference data. This example shows that ML efficacy, while useful for specific tasks, cannot serve as a holistic measure of a tabular data generator.

Following prior studies [94, 61, 79], we adopt the “train-on-synthetic, test-on-real” strategy for quantifying ML efficacy of synthetic data. To mitigate the bias from downstream models, we evaluate the utility with the performance of an ensemble of nine predictors (i.e., AutoGluon-full [27] and TabPFN [41]). Specifically, the downstream models include three standard baselines: Logistic Regression (LR) [22], KNN [30] and MLP [36]; five tree-based methods: Random Forest (RF) [13], Extra Trees [27], LightGBM [63], CatBoost [63], and XGBoost [18]; and a PFN method: TabPFN [41].

Furthermore, as noted in prior work [53], tuning downstream models does affect the relative rankings of tabular generators under ML efficacy. Therefore, to draw generalisable conclusions, we perform hyperparameter tuning for all nine predictors, and the technical details are provided in Appendix E.

B.2 Structural Fidelity of Tabular Data

As illustrated in Figure 1, one of the key desiderata for faithful synthetic tabular data is the preservation of causal structures present in real data. Prior work [89] primarily assesses structural fidelity using toy datasets, as existing metrics [16, 84] typically assume access to the ground-truth SCMs – a condition that is seldom satisfied and arguably infeasible for most real-world datasets [46, 34, 102, 71].

To bridge this gap, we introduce *global utility*, an SCM-free metric that quantifies how well a generator preserves the causal structure of real data. Global utility provides a complementary perspective to conventional metrics, enabling a more holistic assessment of synthetic tabular data.

B.3 Tabular Data Generator

The common paradigm for tabular data generation is to adapt Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs) [94]. For instance, TableGAN [76] employs a convolutional neural network to optimise the label quality, and TVAE [94] is a variant of VAE for tabular data. However, these methods learn the joint distribution and thus cannot preserve the stratification of the reference data [61]. CTGAN [94] refines the generation to be class-conditional. The recent ARF [92] is an adversarial variant of random forest for density estimation, and GOGGLE [58] enhances VAE by learning relational structure with a Graph Neural Network (GNN). Another emerging direction is the use of denoising diffusion models [53, 98, 80]. For instance, TabDDPM [53] demonstrates that diffusion models can approximate typical distributions of tabular data. In addition, several energy-based models have recently been proposed for tabular data generation, such as TabEBM [61] and NRGBoost [12]. These models aim to improve synthetic data quality by learning energy-based representations of the data distribution.

In a broader context, there is growing interest in adapting Large Language Models (LLMs) for tabular data generation [29, 79, 11]. For example, GReaT fine-tunes GPT-2 to generate realistic tabular data, while CLLM leverages the domain knowledge embedded in LLMs during generation. However, most state-of-the-art LLMs do not disclose their pretraining data, raising concerns about data contamination — i.e., whether the reference data (even the test data) has been included during pretraining [29, 61], which can undermine fair comparisons between tabular generators. To ensure fairness and reproducibility, TabStruct excludes models based on proprietary or undisclosed LLMs, such as GPT-4 [79]. We restrict our evaluation to models built on fully open-source LLMs, such as GReaT, thereby mitigating concerns related to data contamination. We would like to emphasise that, although TabStruct excludes certain LLM-based tabular generators to ensure fair and uncontaminated benchmarking, researchers and practitioners are encouraged to integrate their own LLM-based models.

We acknowledge that some models exist beyond those currently implemented in TabStruct. We note that TabStruct offers unified APIs that support up to nine distinct categories of tabular generators (one of the widest scopes to date shown in Table 4), enabling broad compatibility for most tabular generators. Therefore, beyond its current evaluation scope, TabStruct functions as a standardised and extensible benchmarking framework. It is designed to accommodate future methods, promoting continued development and evaluation within a consistent and reproducible environment.

B.4 Evaluation Scope Comparison

Table 3 and Table 4 present a comparative analysis of TabStruct against prior studies on the evaluation of tabular generative models. TabStruct considers four key evaluation dimensions: density estimation, privacy preservation, ML efficacy, and structural fidelity. In addition, it supports all nine categories of tabular generators, offering a more comprehensive and holistic overview of the current landscape of generative modelling for tabular data.

Table 3: **Comparison of considered tabular datasets between TabStruct and prior studies.** TabStruct introduces a novel benchmark designed for the holistic evaluation of tabular generative models, with particular emphasis on evaluating the underlying structure of tabular data. It offers a diverse suite of datasets spanning both classification and regression tasks, thereby supporting comprehensive and structure-aware evaluation across varied use cases.

Paper	Venue	Structural Fidelity	# Datasets	Mixed features	Classification		Regression		# Feature range		
					# Sample range	# Feature range	# Class range	# Datasets	Mixed features	# Sample range	# Feature range
Model studies											
CTGAN [94]	NeurIPS 2019	✗	5	✓	48,842-4,000,000	14-54	2-7	1	✓	39,644-39,644	48-48
TVAE [94]	NeurIPS 2019	✗	5	✓	48,842-4,000,000	14-54	2-7	1	✓	39,644-39,644	48-48
NFLOW [26]	NeurIPS 2019	✗	4	✓	130,065-2,075,259	6-43	2-2	✗	✗	✗	✗
ARF [92]	AISTATS 2023	✗	5	✓	48,842-4,000,000	14-54	2-7	✗	✗	✗	✗
GOGGLE [58]	ICLR 2023	✗	4	✓	569-581,012	12-168	2-7	✗	✗	✗	✗
GReaT [111]	ICLR 2023	✗	5	✓	954-101,766	6-47	2-3	1	✗	20,640-20,640	8-8
STaSy [98]	ICLR 2023	✗	13	✓	1,473-284,807	9-57	2-7	2	✓	39,644-43,824	12-48
TabDDPM [53]	ICML 2023	✗	10	✓	768-130,064	8-50	2-4	6	✓	1,338-197,080	8-51
CoDi [98]	ICML 2023	✗	12	✓	1,000-45,211	10-28	2-7	3	✓	740-1,036	6-21
TabSyn [98]	ICLR 2024	✗	4	✓	12,330-48,842	11-25	2-2	2	✓	39,644-43,824	12-48
CLLM [79]	ICML 2024	✗	7	✓	20-200	12-29	Unknown (private data)	✗	✗	✗	✗
TabEBM [61]	NeurIPS 2024	✗	8	✓	20-500	7-77	2-26	✗	✗	✗	✗
NRGBoost [12]	ICLR 2025	✗	3	✓	10,000-116,202	12-50	2-7	3	✓	835-9,146	8-9
TabDiff [80]	ICLR 2025	✗	5	✓	12,330-101,766	11-36	2-3	2	✓	39,644-43,824	12-48
Benchmarks											
Hansen et al. [40]	NeurIPS 2023	✗	11	✓	7608-71,090	7-26	2-2	✗	✗	✗	✗
Syntheticity [76]	NeurIPS 2023	✗	✗	Unknown	✗	✗	✗	✗	✗	✗	✗
SynMeter [25]	arXiv	✗	8	✓	1,941-48,842	11-31	2-7	4	✓	1,338-39,644	7-60
CauTabBench [89]	arXiv	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Livieris et al. [59]	IFIP 2024	✓	2	✓	5,456-20,757	18-24	3-5	✗	✗	✗	✗
SynthEval [59]	DMKD 2025	✗	1	✓	1,385-1,385	28-28	4	✗	✗	✗	✗
Kapur et al. [47]	arXiv	✗	2	✓	12,960-48,842	8-15	2-3	✗	✗	✗	✗
TabStruct (Ours)	–	✓	17	✓	846-100,000	6-145	2-100	12	✓	345-100,000	6-82

Table 4: **Comparison of considered tabular generative models between TabStruct and prior studies.** TabStruct encompasses nine distinct categories of tabular generators, enabling a comprehensive and systematic comparison across a broad spectrum of generative approaches.

Paper	Venue	# Generators	Interpolation	Bayesian Network	Tabular Generative Models							
					GAN	VAE	Normalising Flows	Tree	Diffusion	EBM	Language Model	
Model studies												
CTGAN [94]	NeurIPS 2019	7	✗	✓	✓	✓	✗	✗	✗	✗	✗	✗
TVAE [94]	NeurIPS 2019	7	✗	✓	✓	✓	✗	✗	✗	✗	✗	✗
NFLOW [26]	NeurIPS 2019	10	✗	✗	✓	✓	✓	✗	✗	✗	✗	✗
ARF [92]	AISTATS 2023	6	✗	✗	✓	✓	✗	✗	✗	✗	✗	✗
GOGGLE [58]	ICLR 2023	7	✗	✓	✓	✓	✓	✗	✗	✗	✗	✗
GReaT [11]	ICLR 2023	4	✗	✗	✓	✓	✗	✗	✗	✗	✗	✓
STaSy [98]	ICLR 2023	8	✗	✗	✓	✓	✓	✗	✗	✗	✗	✗
TabDDPM [53]	ICML 2023	6	✓	✗	✓	✓	✗	✓	✓	✗	✗	✗
CoDi [98]	ICML 2023	9	✗	✗	✓	✓	✓	✗	✓	✗	✗	✗
TabSyn [98]	ICLR 2024	9	✓	✗	✓	✓	✗	✗	✓	✗	✗	✓
CLLM [79]	ICML 2024	7	✓	✗	✓	✓	✓	✗	✓	✗	✗	✓
TabEBM [61]	NeurIPS 2024	9	✓	✗	✓	✓	✓	✓	✓	✓	✗	✗
NRGBoost [12]	ICLR 2025	6	✗	✗	✗	✓	✓	✓	✓	✓	✓	✗
TabDiff [80]	ICLR 2025	9	✗	✗	✗	✓	✗	✗	✓	✗	✗	✓
Benchmarks												
Hansen et al. [40]	NeurIPS 2023	5	✗	✓	✓	✓	✓	✓	✗	✓	✗	✗
Syntheticity [76]	NeurIPS 2023	6	✗	✗	✓	✓	✗	✗	✓	✗	✗	✗
SynMeter [25]	arXiv	8	✗	✗	✓	✓	✗	✗	✓	✗	✓	✓
CauTabBench [89]	arXiv	7	✗	✗	✓	✓	✗	✗	✓	✗	✓	✓
Livieris et al. [59]	IFIP 2024	5	✗	✗	✓	✓	✗	✗	✗	✗	✗	✗
SynthEval [59]	DMKD 2025	5	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗
Kapar et al. [47]	arXiv	6	✗	✗	✓	✓	✗	✗	✓	✓	✗	✗
TabStruct (Ours)	–	13	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

1100 C Designs of Structural Fidelity Metrics

1101 In this section, we detail the design and computation of structural fidelity metrics. We first detail the
 1102 computation of Conditional Independence scores (Appendix C.1), and then detail the computation of
 1103 the proposed global utility score (Appendix C.2).

1104 C.1 Conditional Independence (CI) Scores

1105 C.1.1 Deriving CI Statements from a Causal Graph

1106 **Goal.** For each pair of distinct variables (x_j, x_k) , our objective is to construct: (i) a family of
 1107 d -separation sets $S_{j,k}$ such that $x_j \perp\!\!\!\perp x_k \mid S_{j,k}$, and (ii) a family of d -connection sets $\hat{S}_{j,k}$ such that
 1108 $x_j \not\perp\!\!\!\perp x_k \mid \hat{S}_{j,k}$.

1109 **Notations.** We introduce the following notations, which will be used in the derivation of conditional
 1110 independence (CI) statements:

- 1111 • Let $\mathcal{G} := (\mathcal{X}, E)$ denote a directed acyclic graph (DAG), where the node set $\mathcal{X} :=$
 1112 $\{x_1, \dots, x_D, x_{D+1}\}$ consists of the variables introduced in Section 3.
- 1113 • An undirected path \mathcal{P} in \mathcal{G} is a sequence of distinct nodes $\langle v_1, \dots, v_L \rangle$ such that for each edge on
 1114 the path, $(v_\ell, v_{\ell+1}) \in E$ or $(v_{\ell+1}, v_\ell) \in E$, and each $v_\ell \in \mathcal{X}$.
- 1115 • A non-endpoint node v_ℓ on \mathcal{P} is a *collider* iff the adjacent edges on \mathcal{P} converge head-to-head at v_ℓ
 1116 (i.e. $\rightarrow v_\ell \leftarrow$ in the induced subpath).
- 1117 • For disjoint subsets $\{x_j\}, \{x_k\}, S \subseteq \mathcal{X}$, a path \mathcal{P} is said to be *blocked by* S if **either**: (i) \mathcal{P}
 1118 includes a non-collider that is in S , **or** (ii) \mathcal{P} includes a collider such that neither the collider nor
 1119 any of its descendants is in S .
- 1120 • The variables x_j and x_k are d -separated by $S_{j,k}$ (denoted $x_j \perp\!\!\!\perp x_k \mid S_{j,k}$) if every path between
 1121 x_j and x_k is blocked by $S_{j,k}$.

1122 **Procedures.** For each pair of variables (x_j, x_k) , we enumerate all subsets $S \subseteq \mathcal{X} \setminus \{x_j, x_k\}$ and
 1123 apply the d-separation test [89, 84] to the triple (x_j, x_k, S) . If the test returns true, we add S to the
 1124 set $S_{j,k}$. Once the d-separation sets are identified, we derive the corresponding d-connection sets by
 1125 selectively removing elements from the $S_{j,k}$ sets. The full procedure is detailed in Algorithm 1.

Algorithm 1 Derive complete CI statements

Input: DAG \mathcal{G} over nodes $\mathcal{X} = \{x_1, \dots, x_{D+1}\}$
Output: Full CI statements $\mathcal{C}_{\text{global}}$

```

 $\mathcal{C}_{\text{global}} \leftarrow \emptyset$  // initialise output
foreach unordered pair  $(j, k) \in \{(a, b) \mid 1 \leq a < b \leq D + 1\}$  do
     $S_{j,k} \leftarrow \emptyset$  // reset container
    foreach  $S \subseteq \mathcal{X} \setminus \{x_j, x_k\}$  do
        if  $d\text{-separation\_test}(x_j, x_k, S)$  then
             $S_{j,k} \leftarrow S_{j,k} \cup \{S\}$  // store separator
             $\mathcal{C}_{\text{global}} \leftarrow \mathcal{C}_{\text{global}} \cup \{(x_j \perp\!\!\!\perp x_k \mid S)\}$  // record conditional independence
        end
    end
    foreach  $S \in S_{j,k}$  do
        foreach  $v \in S$  do
             $\hat{S} \leftarrow S \setminus \{v\}$  // candidate d-connection set
            if not  $d\text{-separation\_test}(x_j, x_k, \hat{S})$  then
                 $\mathcal{C}_{\text{global}} \leftarrow \mathcal{C}_{\text{global}} \cup \{(x_j \not\perp\!\!\!\perp x_k \mid \hat{S})\}$  // record conditional dependence
            end
        end
    end
end
return  $\mathcal{C}_{\text{global}}$  // complete CI statements

```

1126 C.1.2 Compute CI Scores on Tabular Data

1127 We compute CI scores according to Equation (3), where the key step is to select an appropriate
1128 conditional independence test for different types of features. For categorical datasets (i.e., all variable
1129 are categorical), we employ the chi-square test of independence [64]. For numerical datasets (i.e., all
1130 variables are numerical), we use partial correlation based on the Pearson correlation coefficient [7]. For
1131 mixed datasets (i.e., mixed variable types), we utilise a residualisation-based conditional independence
1132 test [5, 56, 68]. We implement all conditional independence tests using pgmpy [6], an open-source
1133 Python library for causal and probabilistic inference. By default, the significance level is set to 0.01
1134 (i.e., the p-value is 0.01).

1135 C.2 Global Utility Score

1136 C.2.1 Downstream Predictor Configurations

1137 To compute the utility per feature as defined in Equation (4), we need to evaluate the performance of
1138 downstream predictors when predicting the variable x_j , which requires selecting an appropriate set
1139 of predictors. As discussed in Section 3, the utility per feature is inherently affected by the inductive
1140 biases of downstream models. For instance, KNN tends to perform better when the number of classes
1141 is large [44], whereas XGBoost often performs well on skewed target distributions [63]. To mitigate
1142 such biases, we employ an ensemble of nine predictors with distinct inductive biases. Specifically,
1143 we use the widely adopted “AutoGluon-full” [27], which includes eight predictors, and supplement it
1144 with the competitive TabPFN [41].

1145 Furthermore, as shown in prior work [53], tuning downstream predictors can impact the relative
1146 rankings of tabular data generators. To account for this, we allocate a time budget of one hour
1147 per feature for tuning the full ensemble. We refer to this configuration (i.e., using all nine tuned
1148 predictors) as “Full-tuned”.

1149 C.2.2 Pruning the Ensemble of Downstream Predictors

1150 In addition to the “Full-tuned” setup, we define three alternative configurations of downstream
1151 predictors. These four configurations are summarised below:

- 1152 • **Full-tuned:** A *tuned* ensemble of nine predictors: Logistic Regression (LR), KNN, MLP, Random
1153 Forest, Extra Trees, LightGBM, CatBoost, XGBoost, TabPFN;
- 1154 • **Light-tuned:** A *tuned* ensemble of eight predictors: Logistic Regression (LR), MLP, Random
1155 Forest, Extra Trees, LightGBM, CatBoost, XGBoost, TabPFN;
- 1156 • **Tiny-tuned:** A *tuned* ensemble of three predictors: KNN, XGBoost, TabPFN;
- 1157 • **Tiny-default:** An *untuned* ensemble of three predictors: KNN, XGBoost, TabPFN.

1158 An important observation is that tuning the downstream predictors does improve the absolute per-
1159 formance of the utility per feature. However, we find that *global utility* is more robust to the choice
1160 of downstream predictors than *local utility*. Specifically, when the ensemble is reduced from nine
1161 to three predictors, the relative rankings of tabular generators under global utility remain consistent,
1162 whereas the rankings under local utility fluctuate notably. For instance, under local utility, CTGAN
1163 ranks second with “Full-tuned”, but drops to 10th with “Tiny-default”.

1164 We attribute this robustness to the fairness inherent in the design of global utility – each variable
1165 is treated equally as a prediction target, thereby reducing the bias towards any specific decision
1166 boundary (i.e., downstream predictor). This design helps to mitigate the effect of predictor-specific
1167 biases. Full experimental results are provided in Appendix G.

1168 **Practical guidance for computing local and global utility.** For a comprehensive and fair evaluation,
1169 TabStruct reports all results under the “Full-tuned” configuration. For local utility, we strongly
1170 recommend using the “Full-tuned” configuration. Using a less robust setup may lead to unstable rank-
1171 ings and potentially misleading conclusions about generator performance. In contrast, Appendix G
1172 demonstrates that global utility remains consistent even under the “Tiny-default” configuration, as
1173 both “Full-tuned” and “Tiny-default” settings produce identical relative rankings across 13 tabular
1174 generators. Therefore, we recommend using “Tiny-default” when computing global utility for model
1175 selection, particularly in scenarios where computational efficiency is a priority.

D Rationales for Evaluation Framework Design

D.1 Structural Prior for Tabular Data

The underlying structure of tabular data has long been an open research question [51, 41, 69, 39, 91, 90, 103, 23, 17, 55]. For other modalities like textual data, it is natural to characterise their structure as autoregressive, guided by human knowledge [95]. Therefore, pretraining paradigms aligned with the autoregressive structure, such as next-token prediction [1], have proven successful in textual generative modelling. In contrast, heterogeneous tabular data does not naturally lend itself to human interpretation, making a structural prior for such data generally elusive.

Recent studies [41, 69] on tabular foundation predictors have begun to shed light on the underlying structure of tabular data. TabPFN [41] is a tabular foundation predictor pretrained on 100 million “synthetic” tabular datasets. These datasets are “synthetic” because they do not incorporate real-world semantics: they are produced with randomly constructed structural causal models (SCM). Remarkably, despite not being explicitly trained on any real-world dataset, TabPFN is able to outperform an ensemble of strong baseline predictors, which have been fine-tuned on each individual classification task. The exceptional performance of TabPFN suggests that the SCMs used to construct the pretraining datasets, despite lacking real-world semantics, effectively reflect the structural information encoded in real-world tabular data. However, it is important to note that this does not imply SCMs can fully capture the underlying structure of all tabular data, as no definitive theoretical guarantees have been made yet in the tabular domain. Instead, TabPFN demonstrates that the causal relationships between features, as modelled by SCMs, serve as an empirically effective structural prior for a substantial proportion of real-world tabular data.

As the success of LLMs primarily stems from their ability to leverage the autoregressive nature of textual data, we argue that a robust tabular data generation process should be able to capture the unique causal structures within the tabular data. More specifically, generating data aligned with the causal structures in reference data could provide valuable insights into the open research question of how to effectively leverage the structural information inherent in tabular data.

D.2 CPDAG-level Evaluation of Structural Fidelity

Prior studies [89, 84] typically evaluate the causal structure alignment at three different levels: (i) skeleton level, (ii) Markov equivalence class level, and (iii) causal graph level.

Skeleton level is limited in capacity. At the skeleton level, all causal directions are ignored, resulting in a loss of information about the causal relationships between features. For instance, the causal skeleton is unable to reflect encoded physical laws. Consider the physical system illustrated in Figure 1: the ground-truth causal path from ρ to F_{Earth} is $\rho \rightarrow m_A \rightarrow F_{\text{Earth}}$. This encodes a meaningful interpretation of physical law: given m_A , changing ρ should *not* affect the gravitational force acting on ball A. However, if all directions are removed from the causal path, the resulting skeleton allows for alternative paths, such as $\rho \rightarrow m_A \leftarrow F_{\text{Earth}}$, which share the same undirected structure but imply contradictory physical laws. In this case, the alternative path suggests that, given m_A , changing ρ *would* affect the gravitational force, which is incorrect. Therefore, we choose not to evaluate structural fidelity at the skeleton level due to its inability to capture reliable causal relationships across variables.

Causal graph level necessitates efficient and accurate causal discovery methods, which remains an open research question. At the causal graph level, structural fidelity is assessed by comparing the directed acyclic graphs (DAGs) of the reference and synthetic datasets, accounting for both the skeleton and the causal directions of edges. In principle, this level provides the most fine-grained evaluation of structural fidelity. However, current causal discovery methods struggle to recover accurate DAGs from observational tabular data [71]. Section 4 and Appendix H demonstrate such limitations – where Bayesian Network (BN) performs poorly in generating high-quality synthetic data – suggesting that existing causal discovery tools are inadequate for learning precise causal graphs.

This limitation is well-documented in the literature: recovering perfect DAGs from tabular data remains an unresolved problem for current algorithms [96, 46, 71]. This limitation further supports our argument that CauTabBench provides limited insights for real-world datasets. While CauTabBench attempts to evaluate structural fidelity by applying causal discovery methods to infer a “pseudo” causal graph from real-world data, the absence of ground-truth (GT) causal structures makes such evaluations unreliable. Without access to a known GT, it is impossible to assess the validity of the

inferred graphs. Moreover, the poor empirical performance of BN suggests that these pseudo causal graphs may not be accurate.

Moreover, evaluating at the DAG level requires running causal discovery algorithms on both the reference and synthetic datasets. Employing a specific causal discovery algorithm may introduce evaluation bias – analogous to how utility scores are affected by the choice of predictor models. To reduce this bias, one would need to ensemble multiple causal discovery methods. However, unlike downstream predictors, causal discovery algorithms are often computationally expensive. For instance, the DAGMA algorithm [8] takes over 24 hours to recover a causal graph from a dataset with more than 100 features on our machine (Intel(R) Xeon(R) CPU @ 2.20GHz, 64 cores), due to the exponential scaling of its computation cost with dimensionality.

CPDAG-level evaluation strikes a good balance between evaluation efficiency and validity. Unlike full DAG constructing via causal discovery, CPDAG-level evaluation does not require the orientation of all edges, making it a more tractable yet still meaningful metric of structural fidelity. It is important to note that even reference datasets do not guarantee CI scores of 1. This is analogous to ML efficacy, where even reference data cannot ensure perfect downstream utility (e.g., balanced accuracy = 1 or RMSE = 0). However, as shown in Section 4 and Appendix H, conditional independence (CI) tests generally provide valid and reliable evaluation results. Specifically, CI tests yield consistently high scores on reference datasets, indicating their ability to distinguish between high- and low-quality datasets and thus produce meaningful fidelity assessments. A CPDAG represents the Markov equivalence class of a DAG, preserving essential causal relationships while greatly reducing computational overhead. As illustrated in Figure 1, CPDAGs retain sufficient real-world semantics for practical use cases. Therefore, TabStruct evaluates structural fidelity at the CPDAG level, balancing semantic richness with computational feasibility.

E Reproducibility

E.1 Benchmark Datasets

E.1.1 SCM Datasets

To accurately quantify structural fidelity, the reference data should be paired with ground-truth causal structures. To this end, we construct benchmark SCM datasets using structural causal models (SCMs) that have been validated by human experts [78]. All six SCM datasets are publicly available, with further details provided in Table 5 and Table 6.

Table 5: Details of three SCM classification datasets from bnlearn [78].

Dataset	Domain	# Samples (N)	# Features (D)	N/D	# Numerical	# Categorical	# Classes	# Samples per class (Min)	# Samples per class (Max)
Hailfinder	Meteorology	100,000	56	1785.71	0	56	3	25,048	44,200
Insurance	Economics	100,000	27	3703.70	0	27	4	1,648	56,361
Sangiovese	Agriculture	100,000	15	6666.67	14	1	16	5,659	6,841

Table 6: Details of three SCM regression datasets from bnlearn [78].

Dataset	Domain	# Samples (N)	# Features (D)	N/D	# Numerical	# Categorical
Healthcare	Medicine	100,000	7	14285.71	4	3
MAGIC-IRRI	Life Science	100,000	64	1562.50	64	0
MEHRA	Meteorology	100,000	24	4166.67	20	4

Human validation ensures that the causal structures are realistic, thereby increasing the likelihood that TabStruct’s benchmark results can generalise to other real-world datasets where ground-truth SCMs are not available. We note that this is a core difference between TabStruct and prior studies [89, 41]: rather than relying on toy SCM datasets lacking real-world semantics, TabStruct introduces one of the first comprehensive benchmarks for tabular generative models, based on datasets with expert-validated causal structures, mixed feature types, and more than 10 features.

We outline the process of building the reference SCM datasets as follows. Firstly, we use ground-truth SCMs with realistic and expert-validated structures. Secondly, we perform prior sampling on these SCMs: root nodes are randomly initialised, and their values are propagated through the causal graph. A single sample is generated by recording the node values after propagation, with each propagation producing one sample. Thirdly, this process is repeated until sufficient samples are obtained. In TarStruct, we set $N_{\text{full}} = 100,000$. By following this procedure, we construct full datasets $\mathcal{D}_{\text{full}}$ with accessible and well-defined causal structures. The pseudocode is in Algorithm 2.

Algorithm 2 Constructing full SCM datasets

Input: Ground-truth structural causal model, $M = \langle \mathcal{X}, \mathcal{G}, \mathcal{F}, \mathcal{E} \rangle$, number of samples N_{full} (default to 100,000 samples)

Output: Full SCM dataset $\mathcal{D}_{\text{full}} = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N_{\text{full}}}$

```

Pre-processing  $\pi \leftarrow \text{TopologicalSort}(\mathcal{G})$     // topological order of the variables
 $\mathcal{D}_{\text{full}} \leftarrow \text{InitDataset}()$                 // Initialise an empty dataset

for  $i \leftarrow 1$  to  $N_{\text{full}}$  do
  for  $j \in \pi$  do
    if  $\text{pa}(\mathbf{x}_j) = \emptyset$  then
       $\mathbf{x}_j^{(i)} \leftarrow \text{Sample}(\epsilon_j)$                 // root node: random initialisation
    else
       $\mathbf{x}_j^{(i)} \leftarrow f_j(\{\mathbf{x}_k^{(i)} : \mathbf{x}_k \in \text{pa}(\mathbf{x}_j)\}, \epsilon_j)$     // propagate through SCM
    end
  end
  Append  $(\mathcal{D}_{\text{full}}, (\mathbf{x}_1^{(i)}, \dots, \mathbf{x}_{D+1}^{(i)}))$     // Add the new sample to the SCM dataset
end
return  $\mathcal{D}_{\text{full}}$ 

```

E.1.2 Real-world Datasets

To demonstrate the generalisability of the proposed global utility and TabStruct, we further select 23 challenging real-world datasets from the open-source TabZilla benchmark [63], the OpenML repository (<https://www.openml.org/search?type=data&sort=runs>), and the UCI repository (<https://archive.ics.uci.edu/datasets>). All datasets are publicly available, with further details provided in Table 7 and Table 8.

The dataset selection follows three main criteria: Firstly, the datasets are non-trivial, meaning that generative models cannot easily achieve evaluation results comparable to those obtained from the reference data. Secondly, the datasets originate from diverse domains. For example, “Credit-g” pertains to business applications, whereas “Plants” relates to biological studies. Thirdly, the datasets were not part of the meta-validation stage for TabPFN, reducing the likelihood that their causal structures were implicitly leaked during the development or pretraining of TabPFN.

Table 7: Details of 14 real-world classification datasets.

Dataset	Domain	Source	ID	# Samples (N)	# Features (D)	N/D	# Numerical	# Categorical	# Classes	# Samples per class (Min)	# Samples per class (Max)
Ada	Economics	OpenML	1043	4,562	48	95.04	47	1	2	1,132	3,430
Characters	Images	OpenML	1459	10,218	8	1277.25	7	1	10	600	1,416
Credit-g	Economics	OpenML	46378	1,000	21	47.62	7	14	2	300	700
Electricity	Economics	OpenML	151	45,312	9	5034.67	7	2	2	19,237	26,075
Higgs	Physics	OpenML	4532	98,050	29	3381.03	28	1	2	46,223	51,827
Jasmine	Life Science	OpenML	41143	2,984	145	20.58	8	137	2	1,492	1,492
Nomao	Economics	OpenML	45078	34,465	119	289.62	89	30	2	9,844	24,621
Phoneme	Language	OpenML	1489	5,404	6	900.67	5	1	2	1,586	3,818
Plants	Life Science	OpenML	1493	1,599	65	24.60	64	1	100	15	16
QSAR	Chemistry	OpenML	1494	1,055	42	25.12	41	1	2	356	699
SpeedDating	Social Science	OpenML	40536	8,378	121	69.24	59	62	2	1,380	6,998
Splice	Life Science	OpenML	46	3,190	61	52.30	0	61	3	767	1,655
Vehicle	Transportation	OpenML	54	846	19	44.53	18	1	4	199	218
Zemike	Images	OpenML	22	2,000	48	41.67	47	1	10	200	200

Table 8: Details of nine real-world regression datasets.

Dataset	Domain	Source	ID	# Samples (N)	# Features (D)	N/D	# Numerical	# Categorical
Ailerons	Physics	OpenML	296	13,750	41	335.37	41	0
California	Economics	OpenML	43939	20,640	10	2064.00	9	1
Elevators	Physics	OpenML	216	16,599	19	873.63	19	0
H16	Economics	OpenML	574	22,784	17	1340.24	17	0
Liver	Medicine	OpenML	8	345	6	57.50	6	0
Sales	Economics	OpenML	42092	21,613	20	1080.65	18	2
Space	Demographics	OpenML	507	3,107	7	443.86	7	0
Superconductivity	Chemistry	UCI	464	21,263	82	259.30	82	0
Wine	Life Science	UCI	186	6,497	12	541.42	12	0

1285 E.2 Data Processing

1286 **Data splitting (Figure 4).** For each dataset of N samples, we first split it into train and test sets
 1287 (80% train and 20% test). We further split the train set into a training split (\mathcal{D}_{ref}) and a validation
 1288 split (90% training and 10% validation). For classification datasets, stratification is preserved during
 1289 data splitting. We repeat the splitting 10 times, summing up to 10 runs per dataset. All benchmark
 1290 generators are trained on \mathcal{D}_{ref} , and each generator produces a synthetic dataset with N_{ref} samples. For
 1291 classification, the synthetic data preserves the stratification of the reference data.

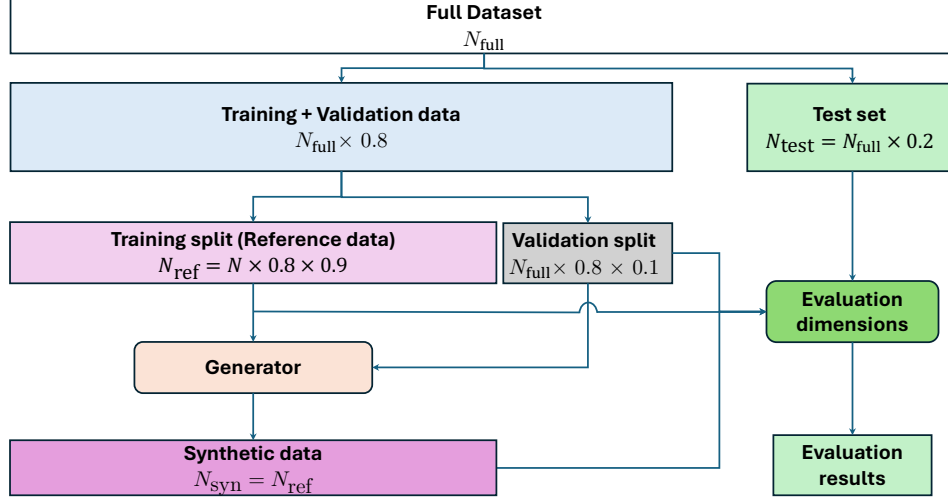


Figure 4: Data splitting strategies for benchmarking tabular data generators.

1292 **Feature preprocessing.** Following the procedures presented in prior work [63, 37], we perform
 1293 preprocessing in three steps. Firstly, we impute the missing values with the mean value for numerical
 1294 features and the mode value for categorical features. We then compute the required statistics with
 1295 training data and then transform the training split. For categorical features, we convert them into
 1296 one-hot encodings. For numerical features, we perform Z-score normalisation. We compute each
 1297 feature’s mean and standard deviation in the training data and then transform the training samples to
 1298 have a mean of zero and a variance of one for each feature. Finally, we apply the same transformation
 1299 to the validation and test data before conducting evaluations.

1300 E.3 Implementations of Benchmark Generators

1301 Following prior work [53, 40], we tune the parametrised generators to ensure a fair comparison.
 1302 Specifically, we use Optuna [3] to optimise each generator by minimising its average validation
 1303 loss across 10 repeated runs. Each generator is given at most two hours to complete a single repeat.
 1304 Importantly, to mitigate bias introduced by specific evaluation metrics, we tune each generator based
 1305 on its own objective function rather than any external metric. Different from prior work [25], this
 1306 approach ensures that each model is evaluated under conditions aligned with its intended optimisation
 1307 direction. The technical details and hyperparameter search space for each method are described below.

1308 **SMOTE** is an interpolation-based oversampling technique [15], which generates synthetic samples by
 1309 interpolating between existing minority class instances. We employ the open-source implementation
 1310 of SMOTE provided by Imbalanced-learn [54], where the number of nearest neighbours k can be
 1311 specified. Unless stated otherwise, we use the default setting of $k = 5$.

1312 **Bayesian Network (BN)** is a probabilistic graphical model used to represent and reason about the
 1313 dependence relationships between features [76, 40]. It consists of two main components: (i) a causal
 1314 discovery model to construct a directed acyclic graph (DAG), where features and the target serve as
 1315 nodes, and their dependencies are represented as edges; (ii) a parameter estimation mechanism to
 1316 quantify the dependence relationships. Following prior work [40], the causal discovery method is
 1317 selected from Hill Climbing Search [52], the Peter-Clark algorithm [52, 84], LiNGAM [81], LiM [97],
 1318 DAGMA [8], DCD [74], AutoCD [14], and Chow-Liu or Tree-augmented Naive Bayes [19, 31]. We

empirically find that AutoCD generally achieves the highest structural fidelity, and thus we build a parametrised BN with AutoCD and maximum likelihood estimation.

Table 9: Hyperparameter search space of BN.

Hyperparameter	Range
struct_learning_score	{"k2", "bdeu", "bic", "bds"}

TVAE is a variational autoencoder (VAE) designed for tabular data [94]. TVAЕ employs mode-specific normalisation to handle the complex distributions of numerical features. To address the class imbalance problem, TVAЕ conditions on specific categorical features during generation.

Table 10: Hyperparameter search space of TVAЕ.

Hyperparameter	Range
encoder_n_layers_hidden	[1, 5]
encoder_n_units_hidden	[50, 500]
encoder_nonlin	{relu, leaky_relu, tanh, elu}
n_units_embedding	[50, 500]
decoder_n_layers_hidden	[1, 5]
decoder_n_units_hidden	[50, 500]
decoder_nonlin	{relu, leaky_relu, tanh, elu}
n_iter	[100, 1000]
lr	$[10^{-4}, 10^{-3}]$ (log)
weight_decay	$[10^{-4}, 10^{-3}]$ (log)

GOGGLE is a VAE-based tabular data generator designed to model the dependence relationships between features [58]. GOGGLE proposes to learn an adjacency matrix to model the dependence relationships between features. However, TabStruct and prior benchmarks [61, 98, 80] all show that the downstream utility of GOGGLE is limited. We hypothesise that this stems from the challenge of learning accurate structures of tabular data. The inherent structure learning mechanism in GOGGLE fails to capture accurate conditional independence relationships between features, which could thus lead to low-quality synthetic data.

Table 11: Hyperparameter search space of GOGGLE.

Hyperparameter	Range
encoder_dim	[32, 128]
encoder_l	[1, 5]
decoder_dim	[32, 128]
decoder_arch	{gcn, het, sage}
n_iter	[100, 500]
learning_rate	$[10^{-4}, 5 \times 10^{-3}]$ (log)
weight_decay	$[10^{-4}, 10^{-3}]$ (log)
alpha	[0.0, 1.0]
beta	[0.0, 1.0]
iter_opt	{True, False}
threshold	[0.0, 1.0]

CTGAN is a conditional generative adversarial network (GAN) designed for tabular data [94]. CTGAN leverages PacGAN [57] framework to mitigate mode collapse. In addition, CTGAN employs the same mode-specific normalisation technique as TVAЕ.

NFlow is a normalisation flow model designed for tabular data generation [26]. NFlow incorporates neural splines as a drop-in replacement for affine or additive transformations in coupling and autoregressive layers, which assists in the modelling of tabular data.

Table 12: Hyperparameter search space of CTGAN.

Hyperparameter	Range
generator_n_layers_hidden	[1, 4]
generator_n_units_hidden	[50, 150]
generator_nonlin	{relu, leaky_relu, tanh, elu}
discriminator_n_layers_hidden	[1, 4]
discriminator_n_units_hidden	[50, 150]
discriminator_nonlin	{relu, leaky_relu, tanh, elu}
n_iter	[100, 1000]
discriminator_n_iter	[1, 5]
lr	$[10^{-4}, 10^{-3}]$ (log)
weight_decay	$[10^{-4}, 10^{-3}]$ (log)

Table 13: Hyperparameter search space of NFlow.

Hyperparameter	Range
n_layers_hidden	[1, 10]
n_units_hidden	[10, 100]
linear_transform_type	{lu, permutation, svd}
base_transform_type	{affine-coupling, quadratic-coupling, rq-coupling, affine-autoregressive, quadratic-autoregressive, rq-autoregressive}
dropout	[0.0, 0.2]
batch_norm	{False, True}
lr	$[2 \times 10^{-4}, 10^{-3}]$ (log)
n_iter	[100, 5000]

1337 **ARF** is a tree-based model for tabular data generation [92]. ARF employs a recursive adaptation
 1338 of unsupervised random forests for joint density estimation by iteratively refining synthetic data
 1339 distributions using adversarial training principles.

Table 14: Hyperparameter search space of ARF.

Hyperparameter	Range
num_trees	{10, 20, ..., 100}
delta	{0, 2, ..., 50}
max_iters	[1, 5]
early_stop	{True, False}
min_node_size	{2, 4, ..., 20}

1340 **TabDDPM** is a diffusion-based model for tabular data generation [53]. TabDDPM introduces two
 1341 core diffusion processes: (i) Gaussian noise for numerical features and (ii) multinomial diffusion with
 1342 categorical noise for categorical features. TabDDPM directly concatenates numerical and categorical
 1343 features as the input and output of the denoising function.

Table 15: Hyperparameter search space of TabDDPM.

Hyperparameter	Range
n_iter	$[10^3, 10^4]$
lr	$[10^{-5}, 10^{-1}]$ (log)
weight_decay	$[10^{-4}, 10^{-3}]$ (log)
num_timesteps	[10, 10^3]

1344 **TabSyn** is a diffusion-based model for tabular data generation [98]. It synthesises tabular data by
 1345 employing a diffusion model within the latent space of a variational autoencoder (VAE). TabSyn
 1346 supports a wide range of data types by mapping them into a unified representation space and explicitly
 1347 modelling inter-column dependencies.

Table 16: Hyperparameter search space of TabSyn.

Hyperparameter	Range
vae.num_epochs	[100, 1000]
vae.max_beta	$[10^{-3}, 10^{-2}]$ (log)
vae.min_beta	$[10^{-5}, 10^{-4}]$ (log)
vae.lambd	[0.1, 1.0]
vae.num_layers	[1, 4]
vae.d_token	[1, 8]
vae.n_head	[1, 4]
vae.factor	[1, 64]
vae.lr	$[10^{-4}, 10^{-2}]$ (log)
vae.wd	$[0, 10^{-2}]$ (log)
tabsyn.num_epochs	[100, 500]
tabsyn.lr	$[10^{-4}, 10^{-2}]$ (log)
tabsyn.wd	$[0, 10^{-2}]$ (log)

1348 **TabDiff** is a diffusion-based model for tabular data generation [80]. It introduces a joint diffusion
 1349 framework capable of capturing the mixed-type distributions inherent in tabular data within a single
 1350 model. In particular, TabDiff utilises a joint continuous-time diffusion process and leverages a
 1351 transformer architecture to handle both numerical and categorical variables.

Table 17: Hyperparameter search space of TabDiff.

Hyperparameter	Range
batch_size	{512, 1024, 2048, 4096, 8192}
c_lambda	[0.1, 10.0]
check_val_every	{10, 20, 30, 40, 50}
class_weight_schedule	{"constant", "anneal", "linear"}
d_lambda	[0.1, 10.0]
ema_decay	[0.9, 0.9999]
factor	[0.1, 0.99]
lr	$[10^{-5}, 10^{-2}]$ (log)
lr_scheduler	{"reduce_lr_on_plateau", "cosine", "none"}
reduce_lr_patience	{10, 30, 50, 70}
steps	{100, 200, 300, 500}
weight_decay	$[0, 10^{-2}]$ (log)

1352 **TabEBM** is an energy-based model for tabular data generation [61]. It transforms a pretrained tabular
 1353 predictor into a set of class-specific generators. While the original paper only provides TabEBM
 1354 implementation for classification tasks, we extend its applicability in TabStruct to regression task by
 1355 treating all reference samples as a single class, and then performing sampling over the energy surface.

Table 18: Hyperparameter search space of TabEBM.

Hyperparameter	Range
starting_point_noise_std	$[10^{-4}, 10^{-1}]$ (log)
sgld_step_size	$[10^{-3}, 10^{-1}]$ (log)
sgld_noise_std	$[10^{-4}, 10^{-1}]$ (log)
sgld_steps	{50, 100, 200, 500}

1356 **NRGBoost** is an energy-based model for tabular data generation [12]. It is trained by maximising
 1357 a local second-order approximation to the log-likelihood at each stage of the boosting process.
 1358 NRGBoost is shown to offer generally good discriminative performance and competitive sampling
 1359 performance compared to more specialised alternatives.

Table 19: Hyperparameter search space of NRGBoost.

Hyperparameter	Range
num_trees	{1, 5, 10, 20, 50}
shrinkage	[0.01, 0.3]
max_leaves	{32, 64, 128, 256, 512}
max_ratio_in_leaf	[1, 5]
num_model_samples	{10,000, 40,000, 80,000, 160,000}
p_refresh	[0.01, 0.3]
num_chains	{4, 8, 16, 32}
burn_in	{50, 100, 200, 500}

1360 **GReaT** leverages large language models (LLMs) to generate synthetic tabular data [11]. GReaT
 1361 converts each sample into a sentence and fine-tunes the language model to capture the sentence-level
 1362 distributions. Additionally, GReaT shuffles the order of features to mitigate the permutation variance
 1363 in sentence-level distributions.

Table 20: Hyperparameter search space of GReaT.

Hyperparameter	Range
n_iter	{100, 300, 500, 1000}
learning_rate	$[10^{-4}, 10^{-2}]$ (log)
weight_decay	$[10^{-5}, 10^{-2}]$ (log)

1364 E.4 Hyperparameter Tuning for Downstream Predictors

1365 As discussed in Appendix C.2, we employ AutoGluon’s built-in tuning functionality for training the
 1366 ensemble predictors. For each variable, the ensemble predictor is allocated one hour of tuning budget
 1367 per repeat, resulting in a total of 10 hours per variable for each dataset. We note that TabPFN is
 1368 not integrated into AutoGluon. However, the default configuration of TabPFN already demonstrates
 1369 competitive performance [41], and thus, we use its default hyperparameters in our experiments.

1370 E.5 Aggregation of Evaluation Results

1371 The reported results are averaged by default over 10 repeats. We aggregate results across all SCM
 1372 or real-world datasets because the findings are consistent across classification and regression tasks.
 1373 Specifically, we use the average distance to the minimum (ADTM) metric [37, 63, 41, 61, 44] via
 1374 affine renormalisation between the top-performing and worse-performing models.

1375 E.6 Software and Computing Resources

1376 **Software implementation.** (i) *For generators:* We implemented SMOTE with Imbalanced-learn [54],
 1377 an open-source Python library for imbalanced datasets with an MIT license. For TabSyn and
 1378 TabEBM, we used their open-source implementations with an Apache-2.0 license. For TabDiff and
 1379 NRGBoost, we used their open-source implementations with an MIT license. For other benchmark
 1380 generators, we used their open-source implementations in Synthcity [76], a library for generating
 1381 and evaluating synthetic tabular data with an Apache-2.0 license. (ii) *For downstream predictors:*
 1382 We implemented TabPFN with its open-source implementation ([https://github.com/automl/](https://github.com/automl/TabPFN)
 1383 TabPFN). We implemented the other eight downstream predictors (i.e., Logistic Regression, KNN,
 1384 MLP, Random Forest, Extra Trees, LightGBM, CatBoost, and XGBoost) with their open-source
 1385 implementation in scikit-learn [73] and AutoGluon [27], an open-source Python library under an
 1386 Apache-2.0 license. (iii) *For result analysis and visualisation:* All numerical plots and graphics have

been generated using Matplotlib 3.7 [43], a Python-based plotting library with a BSD license. The icons for evaluation dimensions in Figure 5 are from <https://icons8.com/>.

We ensure the consistency and reproducibility of experimental results by implementing a uniform pipeline using PyTorch Lightning, an open-source library under an Apache-2.0 license. We further fixed the random seeds for data loading and evaluation throughout the training and evaluation process. This ensured that all benchmark models in TabStruct were trained and evaluated on the same set of samples. The experimental environment settings, including library dependencies, are specified in the open-source library for reference and reproduction purposes.

Computing Resources. All the experiments were conducted on a machine equipped with an NVIDIA A100 GPU with 80GB memory and an Intel(R) Xeon(R) CPU (at 2.20GHz) with 64 cores. The operating system used was Ubuntu 20.04.5 LTS.

F TabStruct Open-source Benchmark Suite

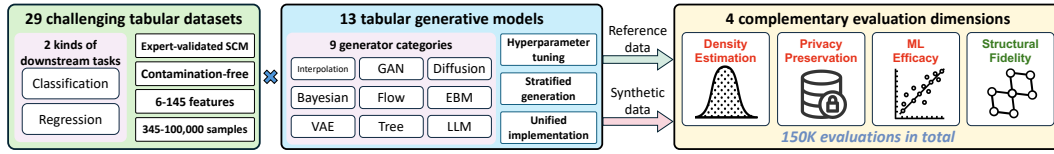


Figure 5: **Overview of the proposed evaluation framework.** TabStruct provides a comprehensive evaluation benchmark, including structural fidelity and conventional dimensions, for 13 representative tabular generative models on 29 challenging tabular datasets.

G Extended Analysis and Discussion

G.1 Extended Analysis on Validity of Global Utility

Global utility provides similar generator rankings as global CI. Figure 6 demonstrates that the rankings of generators under global utility closely align with those under global CI. Notably, the Top-3 methods are identical across both metrics: TabSyn, TabDDPM, and TabDiff. In contrast, when using local utility, the Top-3 methods shift to SMOTE, CTGAN, and TabDiff. This reveals a great discrepancy between the rankings produced by global CI and those from the local utility. In comparison, the proposed global utility yields rankings consistent with global CI, indicating that global utility is an effective proxy for global CI. Consequently, global utility serves as an informative metric for evaluating global structural fidelity.

The evaluation results are consistent across classification and regression datasets of different domains. In Table 21, we present per-dataset evaluation results for both local and global utility. SMOTE remains one of the most competitive methods for capturing local structure, and diffusion models consistently rank among the top-3 for modelling global data structure. These findings indicate that the proposed “utility per variable” metric is stable and provides a unified lens for interpreting evaluation results across both classification and regression datasets.

G.2 Extended Analysis on Structural Fidelity of Generators

Autoregressive models remain limited in learning tabular data structure. Table 2 shows that the autoregressive model GReaT, even with the help of large language models, fails to outperform even the simple baselines like SMOTE and TVAE. Although token-wise likelihood training is a well-established approach for sequential modalities like text and time series, its underlying assumptions misalign with the permutation-invariant nature of tabular data. An autoregressive generator needs to linearise the feature set and then factorise the joint distribution as $\prod_{j=1}^d p(\mathbf{x}_{\pi(j)} \mid \mathbf{x}_{\pi(<j)})$, where π denotes a chosen ordering of features. Any fixed ordering π can introduce directional bias: feature dependencies that run *against* the ordering are modelled only indirectly, via long chains of conditionals. This could weaken the estimation of $p(\mathbf{x}_j \mid \mathcal{X} \setminus \{\mathbf{x}_j\})$ when j appears early in the sequence. While GReaT attempts to mitigate this issue by randomising π when fine-tuning large

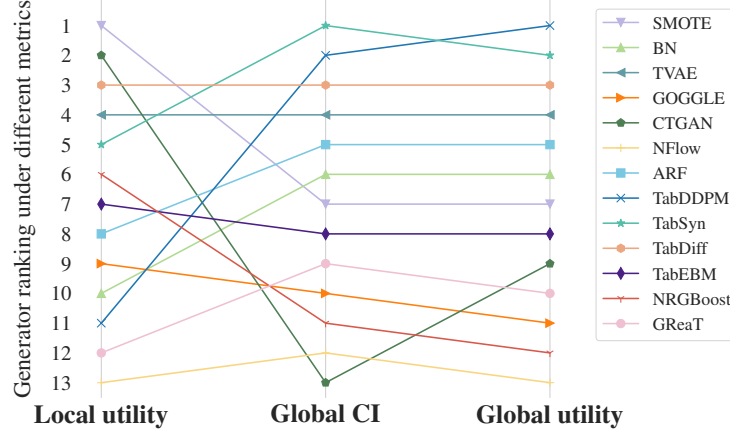


Figure 6: **Rank comparison of 13 tabular data generators across three evaluation metrics on six SCM datasets.** Compared to local utility, global CI and global utility rankings are relatively consistent, suggesting that global utility can serve as an effective proxy for global CI.

Table 21: **Top-3 tabular generators across the TabStruct benchmark suite.** For each dataset, we report the Top-3 tabular generators in terms of both local and global utility. For visualisation, we abbreviate ‘‘Classification’’ as ‘‘Class.’’, and ‘‘Regression’’ as ‘‘Reg.’’. The results indicate that while SMOTE remains a simple yet effective approach for ML efficacy, diffusion models demonstrate stronger capability in capturing the holistic structure of tabular data.

Dataset	# Samples (<i>N</i>)	# Features (<i>D</i>)	<i>N/D</i>	Local utility			Global utility			
				1st	2nd	3rd	1st	2nd	3rd	
SCM datasets										
Class.	Hailfinder	100,000	56	1785.71	SMOTE	CTGAN	NRGBoost	TabDDPM	TabSyn	TabDiff
	Insurance	100,000	27	3703.70	SMOTE	TabEBM	TVAE	TabDDPM	TabDiff	TabSyn
	Sangiovese	100,000	15	6666.67	SMOTE	CTGAN	TabEBM	TabDDPM	TabSyn	TVAE
Reg.	Healthcare	100,000	7	14285.71	SMOTE	TabDiff	TabSyn	BN	ARF	TabDDPM
	MAGIC-IRRI	100,000	64	1562.50	SMOTE	TVAE	TabSyn	TVAE	TabDDPM	TabSyn
	MEHRA	100,000	24	4166.67	SMOTE	TabSyn	GOGGLE	TabDDPM	TabDiff	TabSyn
Real-world datasets										
Class.	Ada	4,562	48	95.04	SMOTE	TabEBM	TabDiff	TVAE	TabDDPM	ARF
	Characters	10,218	8	1277.25	SMOTE	TabEBM	ARF	TabDDPM	TabSyn	TabDiff
	Credit-g	1,000	21	47.62	SMOTE	TabEBM	TabDiff	TabSyn	TabDiff	TabDDPM
	Electricity	45,312	9	5034.67	SMOTE	TabEBM	TabDiff	TabDDPM	TabDiff	ARF
	Higgs	98,050	29	3381.03	SMOTE	CTGAN	TabEBM	TabDDPM	TabSyn	TabDiff
	Jasmine	2,984	145	20.58	SMOTE	TVAE	TabSyn	TabSyn	TabDiff	TabDDPM
	Nomao	34,465	119	289.62	SMOTE	CTGAN	TVAE	TabDiff	TVAE	TabDDPM
	Phoneme	5,404	6	900.67	SMOTE	TabEBM	NRGBoost	TabDDPM	TabSyn	TabDiff
	Plants	1,599	65	24.60	SMOTE	TabEBM	NRGBoost	TabDDPM	TabSyn	TabDiff
	QSAR	1,055	42	25.12	SMOTE	TabEBM	NRGBoost	TabSyn	TabDDPM	TabDiff
	SpeedDating	8,378	121	69.24	SMOTE	TabEBM	TVAE	TabDDPM	TabSyn	TabDiff
	Splice	3,190	61	52.30	SMOTE	TVAE	CTGAN	TabSyn	TabDiff	TabDDPM
	Vehicle	846	19	44.53	SMOTE	TabEBM	TabSyn	TabSyn	TabDDPM	TabDiff
	Zernike	2,000	48	41.67	SMOTE	TabEBM	TVAE	TabSyn	TabDDPM	TabDiff
Reg.	Ailerons	13,750	41	335.37	SMOTE	TabDiff	TabSyn	TabDiff	TabDDPM	TabSyn
	California	20,640	10	2064.00	SMOTE	TabSyn	TabDiff	TabDDPM	TabSyn	TabDiff
	Elevators	16,599	19	873.63	SMOTE	TabDiff	TabSyn	TabDDPM	TabDiff	TabSyn
	H16	22,784	17	1340.24	SMOTE	TabDiff	CTGAN	BN	TabDDPM	TabDiff
	Liver	345	6	57.50	TabDiff	TabSyn	SMOTE	ARF	TabDiff	TabSyn
	Sales	21,613	20	1080.65	SMOTE	TabDiff	TabSyn	TabDiff	TabSyn	TVAE
	Space	3,107	7	443.86	SMOTE	TabSyn	TabDiff	BN	TabDDPM	TabSyn
	Superconductivity	21,263	82	259.30	SMOTE	TabDiff	TabSyn	BN	TabDiff	TabSyn
	Wine	6,497	12	541.42	SMOTE	TabSyn	TabDiff	TabDiff	TabSyn	TabDDPM

language models, such randomisation expands the state space substantially, making optimisation intractable for wide tables (e.g., we observe that GRaT often fail to converge on datasets with more than 100 features). More importantly, randomising π does not resolve the fundamental misalignment. For instance, if the random ordering π happens to reverse the topological order encoded by the ground-truth causal structure, the autoregressive model is forced to learn spurious conditional independence across features, thereby harming the learned global structure.

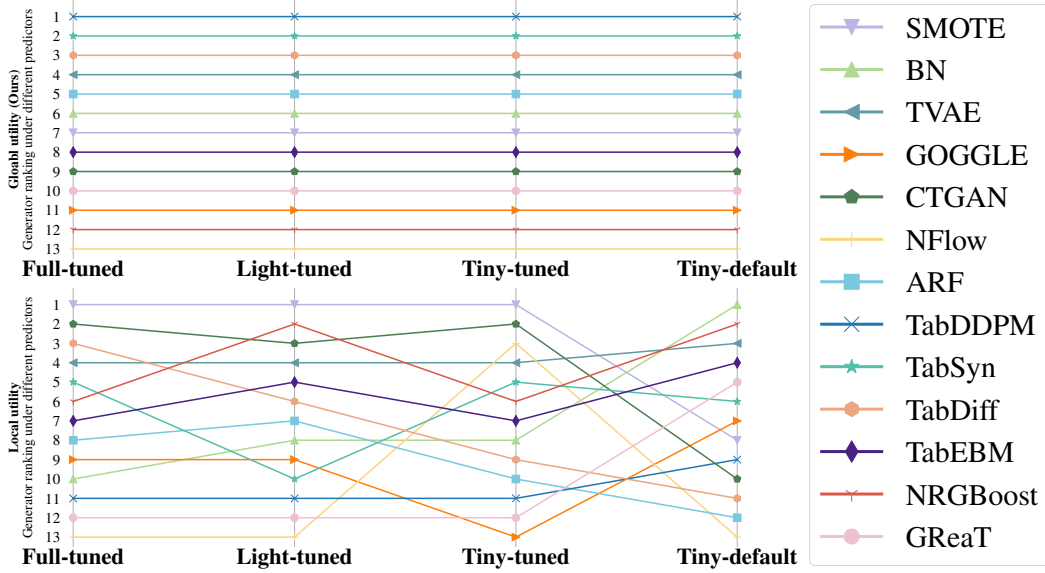


Figure 7: **Comparison of ranking stability between global utility and local utility on 23 real-world datasets, evaluated using different downstream predictors.** The proposed global utility produces consistent generator rankings across downstream predictors. In contrast, local utility necessitates a large set of tuned downstream predictors (i.e., Full-tuned) to yield meaningful rankings. As a result, global utility can achieve high computational efficiency with only a small ensemble of default predictors (i.e., Tiny-default in Figure 3).

G.3 Extended Analysis on Practicability of Global Utility

Global utility remains stable across different downstream predictors. Figure 7 shows that the relative rankings of tabular generators are consistent even when the number of downstream predictors is reduced from nine to three. In contrast, local utility is far more sensitive to the choice of predictors: its rankings fluctuate greatly even when simply reducing from nine to eight predictors. The instability of local utility stems from its bias towards the prediction target, which may introduce unfair bias towards specific types of predictors. For example, KNN tends to perform better when the number of classes is large [44], while XGBoost typically favours skewed target distributions [63]. Since local utility evaluates performance on a single feature, such biases are amplified, yielding unstable rankings even after ensembling different predictors. In contrast, global utility aggregates performance across all features, diluting predictor-specific biases and producing more robust generator rankings.

Global utility is stable regardless of hyperparameter tuning. Figure 7 shows that global utility provides consistent rankings of tabular generators regardless of whether downstream predictors are tuned. We note that this does not imply that tuning is unnecessary. In line with prior work [53, 63, 25], we also observe that tuning improves absolute performance. However, tuning has a negligible effect on the *relative* rankings under global utility. In contrast, local utility necessitates tuning to guarantee reliable results. Such robustness further reflects the core rationale of global utility: by not focusing on a single feature, it avoids introducing feature-specific biases and is therefore less susceptible to variation caused by tuning for a particular downstream prediction target. Such robustness further supports the rationale for using global utility as a stable and unbiased evaluation metric.

Global utility can be more computationally efficient than local utility. In practical settings, we are often interested in identifying the most promising generator before fine-tuning it for optimal performance. In other words, we care more about relative rankings than absolute scores during the early model selection phase. In such cases, global utility can offer an efficient and informative evaluation with as few as three downstream predictors (i.e., Tiny-default). However, local utility typically necessitates a more extensive suite of predictors and hyperparameter tuning. Figure 3 (right) quantifies this trade-off: for every 1,000 samples, getting ranking generators under global utility only takes 0.64s, while local utility may require up to 1.21s to reach a comparably robust conclusion on the relative rankings of different tabular generators.

G.4 Practical Guidance

Conventional evaluation dimensions are insufficient. On SCM datasets, Figure 2 (left) shows that none of the existing evaluation metrics exhibit a strong correlation with global CI. More notably, on real-world datasets, SMOTE and BN outperform diffusion models by a clear margin in density estimation. However, diffusion models are the top-performing generators when it comes to capturing the global structure of tabular data. In other words, the low-fidelity synthetic data generated by SMOTE and BN are often deemed high-quality when assessed only using conventional evaluation dimensions. This observation aligns with our motivating example in Figure 1. These findings underscore the necessity of incorporating structural fidelity as a core evaluation dimension in tabular generative modelling.

Evaluation dimensions are complementary, not interchangeable. Table 2 shows that no single metric is fully indicative of all other metrics. Therefore, researchers and practitioners should select evaluation dimensions that align with the specific objectives of their tasks, rather than relying on a single dimension. If the objective is leakage-free data sharing, the privacy preservation and ML efficacy should be prioritised over density estimation and structural fidelity. Conversely, when the aim is to model a real-world physical system like Figure 1, global structural fidelity should take precedence, because it promotes realistic inter-feature relationships, instead of being distorted towards a single prediction target.

SMOTE is a simple yet effective method for ML efficacy. In Table 21, we provide per-dataset guidance for selecting appropriate tabular generators based on ML efficacy. Surprisingly, SMOTE achieves the highest local utility on 28 out of 29 datasets. Despite this strong performance, Table 4 shows that it has been largely overlooked in prior studies [80, 12, 94]. We strongly encourage researchers and practitioners to consider SMOTE as a robust baseline in scenarios where ML efficacy is the primary objective and other dimensions, such as privacy, are less critical. For instance, in data augmentation tasks, SMOTE can serve as an effective baseline to compare against.

G.5 Future Work

Theoretical justifications for causal modelling of tabular data. Bridging the gap between empirical metrics on real-world tabular datasets and structural causal models (SCMs) remains a major theoretical challenge in causal machine learning [71, 89, 96]. A promising direction for future research lies in developing theoretical underpinnings for the proposed global utility metric. Currently, the proposed global utility serves as an empirically effective metric for structural fidelity, grounded in its correlation with conditional independence (CI) scores. A more rigorous formalisation could help enhance its interpretability in relation to specific causal relationships, and potentially inspire new paradigms for evaluating tabular generators.

Efficient and accurate causal discovery in real-world scenarios. A promising direction for future work is the development of more effective causal discovery algorithms for real-world tabular data. In practical scenarios, ground-truth causal graphs are seldom available, and despite progress in constraint-based, score-based, and hybrid approaches, reliably recovering even partial or probabilistic SCMs remains a challenge – particularly in high-dimensional settings [97, 46, 71]. Nevertheless, incorporating such approximated structures as priors or regularisers in the global utility computation could enhance both its scalability and its fidelity to causal semantics. This would not only enable structural fidelity evaluation on more complex datasets but also improve the robustness of global utility by reducing the influence of spurious statistical associations.

Structure-aware tabular data generation. Beyond evaluation, another important avenue for future work is the design of structure-aware tabular data generators that are explicitly optimised for structural fidelity. These models could embed inductive biases or incorporate regularisation objectives that encourage alignment with the conditional independence structure observed in the reference data. This would mark a shift away from conventional likelihood-driven generation toward structure-informed tabular data generation, enabling the generation of data that better complies with domain-specific constraints (e.g., scientific laws in Figure 1).

Extension to dynamic and temporal data modalities. While TabStruct already offers broad coverage of static tabular datasets (Appendix B), a promising direction for future work is to extend the framework to support temporal and event-based data, where causal relationships may change over time. Many real-world domains – such as healthcare, finance, and operations research – exhibit longitudinal structures that challenge the assumptions of static SCMs [10]. Adapting global utility to reflect time-dependent causal structures would broaden TabStruct’s applicability.

1520 H.2 Evaluation Results for Real-world Datasets

1521 H.2.1 Classification Datasets

Table 28: **Raw benchmark results of 13 tabular generators on “Ada” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Density Estimation				Privacy Preservation		ML Efficacy	Structural Fidelity
	Shape \uparrow	Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.24 \pm 0.01	0.99 \pm 0.00	0.89 \pm 0.01	0.76 \pm 0.01	0.04 \pm 0.03	0.00 \pm 0.00	1.01 \pm 0.05	0.47 \pm 0.38
BN	0.31 \pm 0.07	0.97 \pm 0.02	0.83 \pm 0.12	0.25 \pm 0.11	0.25 \pm 0.07	0.16 \pm 0.27	0.86 \pm 0.13	0.36 \pm 0.28
TVAE	0.23 \pm 0.01	0.98 \pm 0.00	0.70 \pm 0.03	0.22 \pm 0.01	0.28 \pm 0.02	0.05 \pm 0.06	0.96 \pm 0.09	0.77 \pm 0.13
GOGGLE	0.36 \pm 0.02	0.97 \pm 0.02	0.78 \pm 0.12	0.31 \pm 0.08	0.21 \pm 0.04	0.26 \pm 0.30	0.88 \pm 0.12	0.36 \pm 0.27
CTGAN	0.22 \pm 0.01	0.97 \pm 0.00	0.90 \pm 0.05	0.15 \pm 0.03	0.22 \pm 0.03	0.03 \pm 0.01	0.95 \pm 0.11	0.29 \pm 0.26
NFlow	0.23 \pm 0.02	0.97 \pm 0.00	0.87 \pm 0.07	0.07 \pm 0.02	0.20 \pm 0.10	0.02 \pm 0.01	0.87 \pm 0.12	0.53 \pm 0.21
ARF	0.24 \pm 0.01	0.98 \pm 0.00	0.96 \pm 0.00	0.24 \pm 0.01	0.26 \pm 0.05	0.00 \pm 0.00	0.94 \pm 0.09	0.73 \pm 0.17
TabDDPM	0.35 \pm 0.02	0.93 \pm 0.06	0.50 \pm 0.41	0.21 \pm 0.19	0.10 \pm 0.09	0.22 \pm 0.38	0.88 \pm 0.12	0.74 \pm 0.16
TabSyn	0.29 \pm 0.07	0.91 \pm 0.12	0.51 \pm 0.43	0.19 \pm 0.20	0.22 \pm 0.11	0.32 \pm 0.50	0.98 \pm 0.07	0.73 \pm 0.16
TabDiff	0.44 \pm 0.09	0.96 \pm 0.03	0.80 \pm 0.15	0.19 \pm 0.20	0.33 \pm 0.16	0.21 \pm 0.30	0.98 \pm 0.07	0.70 \pm 0.18
TabEBM	0.43 \pm 0.08	0.98 \pm 0.00	0.93 \pm 0.04	0.27 \pm 0.13	0.24 \pm 0.07	0.01 \pm 0.00	0.98 \pm 0.07	0.36 \pm 0.27
NRGBoost	0.30 \pm 0.06	0.96 \pm 0.03	0.45 \pm 0.47	0.19 \pm 0.20	0.29 \pm 0.13	2.07 \pm 2.80	0.98 \pm 0.07	0.20 \pm 0.21
GReaT	0.36 \pm 0.02	0.97 \pm 0.02	0.78 \pm 0.12	0.31 \pm 0.08	0.21 \pm 0.04	0.26 \pm 0.30	0.88 \pm 0.12	0.36 \pm 0.27

Table 29: **Raw benchmark results of 13 tabular generators on “Characters” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Density Estimation				Privacy Preservation		ML Efficacy	Structural Fidelity
	Shape \uparrow	Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.83 \pm 0.00	0.99 \pm 0.00	0.99 \pm 0.01	0.41 \pm 0.03	0.06 \pm 0.01	0.00 \pm 0.00	0.97 \pm 0.05	0.40 \pm 0.42
BN	0.85 \pm 0.02	0.93 \pm 0.00	0.98 \pm 0.00	0.01 \pm 0.00	0.32 \pm 0.02	0.01 \pm 0.00	0.30 \pm 0.13	0.05 \pm 0.05
TVAE	0.82 \pm 0.02	0.91 \pm 0.01	0.95 \pm 0.01	0.04 \pm 0.00	0.31 \pm 0.01	0.01 \pm 0.00	0.77 \pm 0.14	0.41 \pm 0.35
GOGGLE	0.85 \pm 0.01	0.93 \pm 0.01	0.96 \pm 0.01	0.19 \pm 0.05	0.23 \pm 0.02	0.01 \pm 0.00	0.40 \pm 0.23	0.19 \pm 0.19
CTGAN	0.80 \pm 0.02	0.93 \pm 0.01	0.94 \pm 0.03	0.02 \pm 0.00	0.30 \pm 0.03	0.02 \pm 0.01	0.75 \pm 0.26	0.07 \pm 0.07
NFlow	0.82 \pm 0.02	0.88 \pm 0.01	0.94 \pm 0.04	0.00 \pm 0.00	0.41 \pm 0.03	0.02 \pm 0.01	0.20 \pm 0.03	0.02 \pm 0.02
ARF	0.85 \pm 0.00	0.89 \pm 0.01	0.99 \pm 0.00	0.11 \pm 0.00	0.11 \pm 0.02	0.00 \pm 0.00	0.82 \pm 0.03	0.49 \pm 0.04
TabDDPM	0.84 \pm 0.01	0.95 \pm 0.02	0.98 \pm 0.01	0.17 \pm 0.06	0.16 \pm 0.08	0.01 \pm 0.00	0.46 \pm 0.30	0.76 \pm 0.27
TabSyn	0.84 \pm 0.02	0.92 \pm 0.02	0.95 \pm 0.03	0.14 \pm 0.09	0.20 \pm 0.02	0.01 \pm 0.00	0.79 \pm 0.22	0.69 \pm 0.32
TabDiff	0.86 \pm 0.01	0.90 \pm 0.04	0.96 \pm 0.01	0.12 \pm 0.12	0.23 \pm 0.07	0.01 \pm 0.00	0.80 \pm 0.21	0.66 \pm 0.36
TabEBM	0.88 \pm 0.03	0.95 \pm 0.02	0.98 \pm 0.01	0.16 \pm 0.07	0.31 \pm 0.10	0.01 \pm 0.00	0.88 \pm 0.15	0.25 \pm 0.27
NRGBoost	0.84 \pm 0.02	0.92 \pm 0.01	0.97 \pm 0.01	0.12 \pm 0.12	0.28 \pm 0.07	0.01 \pm 0.00	0.77 \pm 0.24	0.11 \pm 0.16
GReaT	0.79 \pm 0.07	0.89 \pm 0.04	0.89 \pm 0.09	0.12 \pm 0.12	0.31 \pm 0.11	0.04 \pm 0.03	0.29 \pm 0.21	0.10 \pm 0.17

Table 30: **Raw benchmark results of 13 tabular generators on “Credit-g” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Density Estimation				Privacy Preservation		ML Efficacy	Structural Fidelity
	Shape \uparrow	Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.95 \pm 0.00	0.90 \pm 0.01	0.87 \pm 0.02	0.79 \pm 0.02	0.19 \pm 0.02	0.00 \pm 0.00	1.17 \pm 0.30	0.42 \pm 0.30
BN	0.97 \pm 0.00	0.95 \pm 0.00	0.97 \pm 0.01	0.68 \pm 0.02	0.06 \pm 0.00	0.00 \pm 0.00	0.92 \pm 0.09	0.39 \pm 0.28
TVAE	0.93 \pm 0.01	0.86 \pm 0.02	0.80 \pm 0.04	0.48 \pm 0.02	0.55 \pm 0.04	0.02 \pm 0.01	1.11 \pm 0.34	0.47 \pm 0.14
GOGGLE	0.79 \pm 0.17	0.67 \pm 0.25	0.55 \pm 0.42	0.35 \pm 0.24	0.36 \pm 0.06	0.37 \pm 0.49	1.05 \pm 0.36	0.33 \pm 0.21
CTGAN	0.80 \pm 0.06	0.72 \pm 0.09	0.83 \pm 0.12	0.27 \pm 0.07	0.50 \pm 0.05	0.21 \pm 0.16	1.13 \pm 0.32	0.21 \pm 0.10
NFlow	0.90 \pm 0.01	0.84 \pm 0.01	0.84 \pm 0.08	0.27 \pm 0.04	0.50 \pm 0.07	0.02 \pm 0.01	0.85 \pm 0.04	0.24 \pm 0.11
ARF	0.97 \pm 0.00	0.86 \pm 0.01	0.98 \pm 0.01	0.45 \pm 0.03	0.53 \pm 0.05	0.00 \pm 0.00	0.87 \pm 0.04	0.43 \pm 0.06
TabDDPM	0.75 \pm 0.19	0.63 \pm 0.25	0.45 \pm 0.47	0.28 \pm 0.29	0.17 \pm 0.17	0.15 \pm 0.13	0.91 \pm 0.09	0.55 \pm 0.29
TabSyn	0.86 \pm 0.08	0.76 \pm 0.13	0.67 \pm 0.24	0.41 \pm 0.15	0.44 \pm 0.12	0.03 \pm 0.02	1.15 \pm 0.31	0.64 \pm 0.17
TabDiff	0.90 \pm 0.03	0.74 \pm 0.14	0.91 \pm 0.04	0.40 \pm 0.17	0.43 \pm 0.11	0.02 \pm 0.02	1.15 \pm 0.31	0.62 \pm 0.19
TabEBM	0.92 \pm 0.01	0.84 \pm 0.03	0.93 \pm 0.04	0.50 \pm 0.06	0.38 \pm 0.05	0.02 \pm 0.02	1.15 \pm 0.31	0.33 \pm 0.22
NRGBoost	0.87 \pm 0.07	0.80 \pm 0.08	0.75 \pm 0.15	0.34 \pm 0.23	0.38 \pm 0.06	0.03 \pm 0.02	1.15 \pm 0.31	0.25 \pm 0.16
GReaT	0.88 \pm 0.06	0.80 \pm 0.08	0.64 \pm 0.27	0.43 \pm 0.14	0.42 \pm 0.09	0.10 \pm 0.09	0.91 \pm 0.09	0.26 \pm 0.16

Table 31: **Raw benchmark results of 13 tabular generators on “Electricity” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Density Estimation				Privacy Preservation		ML Efficacy	Structural Fidelity
	Shape \uparrow	Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.86 \pm 0.00	0.99 \pm 0.00	0.99 \pm 0.00	0.78 \pm 0.00	0.01 \pm 0.00	0.00 \pm 0.01	0.98 \pm 0.02	0.41 \pm 0.41
BN	0.93 \pm 0.00	0.97 \pm 0.00	0.98 \pm 0.00	0.21 \pm 0.00	0.05 \pm 0.03	0.01 \pm 0.01	0.75 \pm 0.11	0.23 \pm 0.25
TVAE	0.89 \pm 0.01	0.92 \pm 0.03	0.96 \pm 0.02	0.20 \pm 0.00	0.09 \pm 0.03	0.21 \pm 0.23	0.90 \pm 0.06	0.56 \pm 0.28
GOGGLE	0.86 \pm 0.03	0.91 \pm 0.02	0.93 \pm 0.04	0.33 \pm 0.07	0.07 \pm 0.02	0.43 \pm 0.76	0.76 \pm 0.12	0.27 \pm 0.26
CTGAN	0.86 \pm 0.01	0.92 \pm 0.01	0.96 \pm 0.03	0.19 \pm 0.01	0.02 \pm 0.00	0.01 \pm 0.01	0.92 \pm 0.08	0.21 \pm 0.24
NFlow	0.84 \pm 0.02	0.85 \pm 0.02	0.91 \pm 0.05	0.09 \pm 0.02	0.14 \pm 0.06	0.03 \pm 0.01	0.75 \pm 0.03	0.25 \pm 0.22
ARF	0.86 \pm 0.00	0.81 \pm 0.04	0.95 \pm 0.00	0.26 \pm 0.01	0.02 \pm 0.01	0.00 \pm 0.00	0.90 \pm 0.01	0.62 \pm 0.13
TabDDPM	0.87 \pm 0.02	0.95 \pm 0.03	0.98 \pm 0.01	0.32 \pm 0.07	0.03 \pm 0.02	0.03 \pm 0.03	0.77 \pm 0.13	0.76 \pm 0.23
TabSyn	0.54 \pm 0.37	0.73 \pm 0.20	0.48 \pm 0.51	0.20 \pm 0.21	0.12 \pm 0.14	4.77 \pm 9.27	0.86 \pm 0.18	0.60 \pm 0.42
TabDiff	0.88 \pm 0.01	0.88 \pm 0.05	0.96 \pm 0.01	0.23 \pm 0.17	0.06 \pm 0.04	0.03 \pm 0.03	0.93 \pm 0.07	0.75 \pm 0.24
TabEBM	0.90 \pm 0.01	0.92 \pm 0.03	0.97 \pm 0.01	0.21 \pm 0.19	0.21 \pm 0.18	0.03 \pm 0.03	0.93 \pm 0.07	0.26 \pm 0.26
NRGBoost	0.87 \pm 0.02	0.92 \pm 0.01	0.97 \pm 0.00	0.22 \pm 0.18	0.04 \pm 0.01	0.03 \pm 0.03	0.92 \pm 0.08	0.22 \pm 0.21
GReaT	0.86 \pm 0.03	0.91 \pm 0.02	0.93 \pm 0.04	0.33 \pm 0.07	0.07 \pm 0.02	0.43 \pm 0.76	0.76 \pm 0.12	0.27 \pm 0.26

Table 32: **Raw benchmark results of 13 tabular generators on “Higgs” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Shape \uparrow	Density Estimation			Privacy Preservation		ML Efficacy	Structural Fidelity
		Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.90 \pm 0.00	0.99 \pm 0.00	0.76 \pm 0.00	0.82 \pm 0.00	0.12 \pm 0.03	0.00 \pm 0.00	0.99 \pm 0.01	0.45 \pm 0.39
BN	0.92 \pm 0.00	0.99 \pm 0.00	0.98 \pm 0.01	0.29 \pm 0.00	0.06 \pm 0.01	0.00 \pm 0.00	0.85 \pm 0.09	0.24 \pm 0.17
TVAE	0.86 \pm 0.00	0.97 \pm 0.00	0.92 \pm 0.01	0.37 \pm 0.01	0.25 \pm 0.04	0.12 \pm 0.14	0.93 \pm 0.04	0.63 \pm 0.22
GOGGLE	0.90 \pm 0.01	0.97 \pm 0.00	0.90 \pm 0.01	0.40 \pm 0.07	0.13 \pm 0.02	0.12 \pm 0.19	0.84 \pm 0.08	0.29 \pm 0.22
CTGAN	0.85 \pm 0.01	0.97 \pm 0.00	0.95 \pm 0.02	0.32 \pm 0.02	0.11 \pm 0.02	0.09 \pm 0.08	0.95 \pm 0.05	0.23 \pm 0.17
NFlow	0.83 \pm 0.01	0.95 \pm 0.00	0.87 \pm 0.08	0.23 \pm 0.02	0.18 \pm 0.05	0.69 \pm 1.20	0.77 \pm 0.04	0.16 \pm 0.09
ARF	0.88 \pm 0.00	0.95 \pm 0.00	0.91 \pm 0.00	0.26 \pm 0.00	0.13 \pm 0.02	0.00 \pm 0.00	0.89 \pm 0.01	0.50 \pm 0.06
TabDDPM	0.92 \pm 0.03	0.97 \pm 0.00	0.93 \pm 0.03	0.42 \pm 0.05	0.06 \pm 0.06	0.10 \pm 0.19	0.85 \pm 0.08	0.80 \pm 0.22
TabSyn	0.91 \pm 0.02	0.97 \pm 0.00	0.94 \pm 0.03	0.38 \pm 0.10	0.14 \pm 0.04	0.10 \pm 0.19	0.95 \pm 0.05	0.76 \pm 0.24
TabDiff	0.89 \pm 0.00	0.96 \pm 0.02	0.85 \pm 0.07	0.27 \pm 0.22	0.20 \pm 0.10	0.11 \pm 0.19	0.95 \pm 0.06	0.70 \pm 0.31
TabEBM	0.91 \pm 0.02	0.97 \pm 0.00	0.93 \pm 0.03	0.25 \pm 0.24	0.26 \pm 0.15	0.10 \pm 0.19	0.95 \pm 0.05	0.22 \pm 0.17
NRGBoost	0.91 \pm 0.01	0.97 \pm 0.01	0.78 \pm 0.13	0.25 \pm 0.23	0.11 \pm 0.02	0.12 \pm 0.19	0.95 \pm 0.06	0.18 \pm 0.18
GReaT	0.90 \pm 0.01	0.97 \pm 0.00	0.90 \pm 0.01	0.40 \pm 0.07	0.13 \pm 0.02	0.12 \pm 0.19	0.84 \pm 0.08	0.29 \pm 0.22

Table 33: **Raw benchmark results of 13 tabular generators on “Jasmine” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Shape \uparrow	Density Estimation			Privacy Preservation		ML Efficacy	Structural Fidelity
		Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.98 \pm 0.00	0.98 \pm 0.00	0.86 \pm 0.01	0.82 \pm 0.01	0.06 \pm 0.01	0.00 \pm 0.00	0.98 \pm 0.02	0.46 \pm 0.18
BN	0.96 \pm 0.03	0.94 \pm 0.06	0.84 \pm 0.13	0.36 \pm 0.07	0.39 \pm 0.11	0.13 \pm 0.20	0.91 \pm 0.06	0.42 \pm 0.13
TVAE	0.96 \pm 0.00	0.94 \pm 0.01	0.83 \pm 0.02	0.28 \pm 0.02	0.49 \pm 0.04	0.14 \pm 0.05	0.97 \pm 0.02	0.47 \pm 0.10
GOGGLE	0.95 \pm 0.03	0.91 \pm 0.04	0.79 \pm 0.10	0.34 \pm 0.07	0.31 \pm 0.04	0.18 \pm 0.18	0.90 \pm 0.06	0.40 \pm 0.11
CTGAN	0.94 \pm 0.03	0.92 \pm 0.04	0.84 \pm 0.15	0.18 \pm 0.08	0.36 \pm 0.07	0.04 \pm 0.05	0.96 \pm 0.04	0.36 \pm 0.08
NFlow	0.95 \pm 0.01	0.91 \pm 0.01	0.77 \pm 0.05	0.01 \pm 0.00	0.31 \pm 0.05	0.03 \pm 0.01	0.85 \pm 0.04	0.31 \pm 0.04
ARF	0.99 \pm 0.00	0.90 \pm 0.00	0.93 \pm 0.01	0.21 \pm 0.02	0.37 \pm 0.06	0.00 \pm 0.00	0.94 \pm 0.02	0.46 \pm 0.05
TabDDPM	0.81 \pm 0.17	0.72 \pm 0.24	0.44 \pm 0.46	0.21 \pm 0.22	0.40 \pm 0.13	1.24 \pm 1.45	0.90 \pm 0.06	0.59 \pm 0.16
TabSyn	0.91 \pm 0.07	0.83 \pm 0.14	0.58 \pm 0.37	0.21 \pm 0.22	0.40 \pm 0.13	0.07 \pm 0.15	0.97 \pm 0.03	0.61 \pm 0.14
TabDiff	0.93 \pm 0.06	0.87 \pm 0.10	0.63 \pm 0.40	0.24 \pm 0.18	0.37 \pm 0.10	0.39 \pm 0.69	0.97 \pm 0.03	0.61 \pm 0.14
TabEBM	0.98 \pm 0.01	0.96 \pm 0.02	0.92 \pm 0.06	0.41 \pm 0.01	0.37 \pm 0.09	0.03 \pm 0.04	0.97 \pm 0.03	0.42 \pm 0.13
NRGBoost	0.96 \pm 0.01	0.94 \pm 0.01	0.87 \pm 0.03	0.21 \pm 0.22	0.23 \pm 0.07	0.09 \pm 0.08	0.97 \pm 0.03	0.36 \pm 0.12
GReaT	0.95 \pm 0.03	0.91 \pm 0.04	0.79 \pm 0.10	0.34 \pm 0.07	0.31 \pm 0.04	0.18 \pm 0.18	0.90 \pm 0.06	0.40 \pm 0.11

Table 34: **Raw benchmark results of 13 tabular generators on “Nomao” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Shape \uparrow	Density Estimation			Privacy Preservation		ML Efficacy	Structural Fidelity
		Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.70 \pm 0.00	0.99 \pm 0.00	0.98 \pm 0.00	0.77 \pm 0.01	0.03 \pm 0.01	0.00 \pm 0.00	0.99 \pm 0.01	0.40 \pm 0.38
BN	0.77 \pm 0.00	0.93 \pm 0.00	0.95 \pm 0.01	0.21 \pm 0.01	0.15 \pm 0.02	0.00 \pm 0.00	0.72 \pm 0.19	0.38 \pm 0.36
TVAE	0.73 \pm 0.01	0.88 \pm 0.01	0.89 \pm 0.01	0.13 \pm 0.00	0.05 \pm 0.01	0.06 \pm 0.02	0.96 \pm 0.00	0.61 \pm 0.17
GOGGLE	0.73 \pm 0.03	0.85 \pm 0.05	0.84 \pm 0.07	0.25 \pm 0.07	0.11 \pm 0.03	1.58 \pm 1.11	0.72 \pm 0.19	0.26 \pm 0.23
CTGAN	0.68 \pm 0.01	0.89 \pm 0.01	0.92 \pm 0.02	0.02 \pm 0.00	0.06 \pm 0.00	0.07 \pm 0.06	0.96 \pm 0.04	0.19 \pm 0.16
NFlow	0.70 \pm 0.01	0.81 \pm 0.01	0.61 \pm 0.06	0.00 \pm 0.00	0.20 \pm 0.16	5.97 \pm 2.37	0.55 \pm 0.03	0.05 \pm 0.05
ARF	0.74 \pm 0.01	0.76 \pm 0.05	0.95 \pm 0.03	0.07 \pm 0.10	0.04 \pm 0.01	0.15 \pm 0.38	0.96 \pm 0.01	0.53 \pm 0.05
TabDDPM	0.64 \pm 0.13	0.75 \pm 0.16	0.45 \pm 0.47	0.16 \pm 0.17	0.18 \pm 0.13	2.77 \pm 2.45	0.72 \pm 0.19	0.60 \pm 0.34
TabSyn	0.58 \pm 0.20	0.72 \pm 0.18	0.75 \pm 0.26	0.16 \pm 0.17	0.12 \pm 0.11	7.12 \pm 12.67	0.95 \pm 0.06	0.60 \pm 0.34
TabDiff	0.74 \pm 0.03	0.78 \pm 0.12	0.80 \pm 0.10	0.16 \pm 0.17	0.10 \pm 0.07	0.48 \pm 0.54	0.95 \pm 0.06	0.68 \pm 0.22
TabEBM	0.74 \pm 0.03	0.84 \pm 0.05	0.94 \pm 0.05	0.18 \pm 0.15	0.26 \pm 0.19	0.47 \pm 0.54	0.95 \pm 0.06	0.30 \pm 0.27
NRGBoost	0.73 \pm 0.03	0.86 \pm 0.04	0.78 \pm 0.12	0.16 \pm 0.17	0.09 \pm 0.01	1.92 \pm 2.41	0.95 \pm 0.06	0.17 \pm 0.22
GReaT	0.73 \pm 0.03	0.85 \pm 0.05	0.84 \pm 0.07	0.25 \pm 0.07	0.11 \pm 0.03	1.58 \pm 1.11	0.72 \pm 0.19	0.26 \pm 0.23

Table 35: **Raw benchmark results of 13 tabular generators on “Phoneme” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Shape \uparrow	Density Estimation			Privacy Preservation		ML Efficacy	Structural Fidelity
		Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.96 \pm 0.00	0.95 \pm 0.01	0.99 \pm 0.00	0.74 \pm 0.01	0.08 \pm 0.01	0.00 \pm 0.00	1.00 \pm 0.04	0.44 \pm 0.41
BN	0.97 \pm 0.00	0.98 \pm 0.00	0.98 \pm 0.00	0.46 \pm 0.01	0.12 \pm 0.01	0.00 \pm 0.00	0.82 \pm 0.17	0.42 \pm 0.38
TVAE	0.91 \pm 0.00	0.86 \pm 0.01	0.94 \pm 0.01	0.13 \pm 0.01	0.17 \pm 0.01	0.01 \pm 0.00	0.93 \pm 0.07	0.52 \pm 0.29
GOGGLE	0.88 \pm 0.14	0.89 \pm 0.08	0.93 \pm 0.08	0.30 \pm 0.12	0.14 \pm 0.03	0.37 \pm 0.94	0.79 \pm 0.15	0.27 \pm 0.24
CTGAN	0.80 \pm 0.07	0.79 \pm 0.04	0.89 \pm 0.09	0.07 \pm 0.01	0.19 \pm 0.04	0.45 \pm 0.74	0.90 \pm 0.13	0.11 \pm 0.12
NFlow	0.90 \pm 0.02	0.90 \pm 0.01	0.94 \pm 0.04	0.09 \pm 0.01	0.16 \pm 0.02	0.02 \pm 0.01	0.80 \pm 0.04	0.22 \pm 0.13
ARF	0.95 \pm 0.00	0.91 \pm 0.02	0.99 \pm 0.00	0.22 \pm 0.01	0.11 \pm 0.02	0.00 \pm 0.00	0.91 \pm 0.01	0.67 \pm 0.05
TabDDPM	0.94 \pm 0.02	0.95 \pm 0.03	0.97 \pm 0.02	0.31 \pm 0.08	0.10 \pm 0.03	0.03 \pm 0.05	0.79 \pm 0.15	0.81 \pm 0.20
TabSyn	0.90 \pm 0.03	0.87 \pm 0.04	0.95 \pm 0.01	0.25 \pm 0.14	0.16 \pm 0.04	0.11 \pm 0.13	0.88 \pm 0.18	0.71 \pm 0.30
TabDiff	0.92 \pm 0.01	0.91 \pm 0.01	0.96 \pm 0.02	0.22 \pm 0.18	0.18 \pm 0.07	0.03 \pm 0.05	0.93 \pm 0.10	0.69 \pm 0.33
TabEBM	0.94 \pm 0.02	0.92 \pm 0.02	0.97 \pm 0.02	0.29 \pm 0.11	0.24 \pm 0.13	0.03 \pm 0.05	0.97 \pm 0.06	0.31 \pm 0.27
NRGBoost	0.94 \pm 0.01	0.93 \pm 0.02	0.97 \pm 0.01	0.23 \pm 0.17	0.14 \pm 0.03	0.03 \pm 0.05	0.96 \pm 0.07	0.21 \pm 0.20
GReaT	0.90 \pm 0.03	0.89 \pm 0.03	0.86 \pm 0.10	0.22 \pm 0.18	0.19 \pm 0.08	0.12 \pm 0.21	0.71 \pm 0.13	0.17 \pm 0.21

Table 36: **Raw benchmark results of 13 tabular generators on “Plants” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Shape \uparrow	Density Estimation			Privacy Preservation		ML Efficacy	Structural Fidelity
		Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.84 \pm 0.00	0.98 \pm 0.01	0.88 \pm 0.00	0.82 \pm 0.01	0.14 \pm 0.02	0.00 \pm 0.00	1.00 \pm 0.02	0.45 \pm 0.01
BN	0.87 \pm 0.00	0.97 \pm 0.00	0.97 \pm 0.01	0.26 \pm 0.00	0.15 \pm 0.02	0.00 \pm 0.00	0.89 \pm 0.08	0.29 \pm 0.11
TVAE	0.82 \pm 0.02	0.92 \pm 0.01	0.80 \pm 0.07	0.20 \pm 0.07	0.25 \pm 0.03	0.08 \pm 0.06	0.95 \pm 0.04	0.52 \pm 0.11
GOGGLE	0.84 \pm 0.06	0.94 \pm 0.03	0.88 \pm 0.05	0.37 \pm 0.06	0.20 \pm 0.03	0.18 \pm 0.16	0.88 \pm 0.08	0.28 \pm 0.01
CTGAN	0.79 \pm 0.05	0.91 \pm 0.02	0.89 \pm 0.06	0.24 \pm 0.08	0.20 \pm 0.04	0.20 \pm 0.22	0.95 \pm 0.06	0.19 \pm 0.07
NFlow	0.82 \pm 0.02	0.94 \pm 0.01	0.87 \pm 0.06	0.24 \pm 0.05	0.22 \pm 0.04	0.26 \pm 0.38	0.86 \pm 0.03	0.30 \pm 0.09
ARF	0.86 \pm 0.01	0.94 \pm 0.01	0.90 \pm 0.02	0.29 \pm 0.04	0.19 \pm 0.03	0.02 \pm 0.01	0.93 \pm 0.01	0.57 \pm 0.05
TabDDPM	0.82 \pm 0.08	0.95 \pm 0.02	0.78 \pm 0.16	0.33 \pm 0.09	0.19 \pm 0.08	0.10 \pm 0.08	0.88 \pm 0.08	0.74 \pm 0.11
TabSyn	0.85 \pm 0.03	0.94 \pm 0.02	0.91 \pm 0.03	0.31 \pm 0.06	0.18 \pm 0.05	0.08 \pm 0.05	0.94 \pm 0.08	0.74 \pm 0.03
TabDiff	0.85 \pm 0.02	0.93 \pm 0.02	0.86 \pm 0.05	0.25 \pm 0.03	0.27 \pm 0.11	0.09 \pm 0.06	0.96 \pm 0.05	0.68 \pm 0.03
TabEBM	0.87 \pm 0.02	0.96 \pm 0.01	0.94 \pm 0.03	0.35 \pm 0.14	0.30 \pm 0.08	0.05 \pm 0.05	0.97 \pm 0.04	0.28 \pm 0.06
NRGBoost	0.86 \pm 0.02	0.96 \pm 0.01	0.89 \pm 0.07	0.27 \pm 0.05	0.19 \pm 0.03	0.05 \pm 0.05	0.97 \pm 0.04	0.22 \pm 0.05
GReaT	0.85 \pm 0.02	0.94 \pm 0.01	0.86 \pm 0.05	0.34 \pm 0.11	0.21 \pm 0.05	0.10 \pm 0.04	0.85 \pm 0.07	0.24 \pm 0.06

Table 37: **Raw benchmark results of 13 tabular generators on “QSAR” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Shape \uparrow	Density Estimation			Privacy Preservation		ML Efficacy	Structural Fidelity
		Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.76 \pm 0.00	0.96 \pm 0.01	0.95 \pm 0.01	0.76 \pm 0.01	0.08 \pm 0.01	0.01 \pm 0.01	0.93 \pm 0.11	0.41 \pm 0.03
BN	0.77 \pm 0.01	0.98 \pm 0.00	0.98 \pm 0.00	0.54 \pm 0.01	0.12 \pm 0.00	0.00 \pm 0.00	0.78 \pm 0.20	0.40 \pm 0.03
TVAE	0.71 \pm 0.01	0.89 \pm 0.00	0.83 \pm 0.03	0.09 \pm 0.01	0.15 \pm 0.02	0.10 \pm 0.04	0.85 \pm 0.17	0.47 \pm 0.08
GOGGLE	0.74 \pm 0.08	0.92 \pm 0.04	0.87 \pm 0.08	0.30 \pm 0.00	0.14 \pm 0.00	0.24 \pm 0.18	0.70 \pm 0.12	0.26 \pm 0.02
CTGAN	0.64 \pm 0.04	0.86 \pm 0.02	0.89 \pm 0.01	0.04 \pm 0.01	0.15 \pm 0.03	0.27 \pm 0.25	0.85 \pm 0.18	0.12 \pm 0.01
NFlow	0.72 \pm 0.01	0.91 \pm 0.01	0.85 \pm 0.07	0.05 \pm 0.01	0.16 \pm 0.00	0.10 \pm 0.07	0.66 \pm 0.03	0.19 \pm 0.03
ARF	0.77 \pm 0.00	0.93 \pm 0.01	0.96 \pm 0.01	0.15 \pm 0.01	0.13 \pm 0.03	0.01 \pm 0.01	0.75 \pm 0.01	0.55 \pm 0.07
TabDDPM	0.71 \pm 0.08	0.93 \pm 0.03	0.71 \pm 0.24	0.25 \pm 0.09	0.08 \pm 0.02	0.22 \pm 0.26	0.70 \pm 0.12	0.70 \pm 0.16
TabSyn	0.75 \pm 0.02	0.92 \pm 0.02	0.92 \pm 0.03	0.26 \pm 0.02	0.15 \pm 0.01	0.08 \pm 0.04	0.87 \pm 0.18	0.73 \pm 0.02
TabDiff	0.76 \pm 0.01	0.92 \pm 0.01	0.90 \pm 0.03	0.20 \pm 0.02	0.22 \pm 0.06	0.07 \pm 0.05	0.89 \pm 0.14	0.67 \pm 0.03
TabEBM	0.81 \pm 0.04	0.94 \pm 0.01	0.95 \pm 0.03	0.32 \pm 0.05	0.27 \pm 0.04	0.04 \pm 0.01	0.91 \pm 0.12	0.30 \pm 0.01
NRGBoost	0.76 \pm 0.03	0.93 \pm 0.00	0.77 \pm 0.16	0.21 \pm 0.03	0.17 \pm 0.03	0.04 \pm 0.02	0.90 \pm 0.12	0.19 \pm 0.03
GReaT	0.71 \pm 0.06	0.91 \pm 0.03	0.76 \pm 0.15	0.20 \pm 0.02	0.17 \pm 0.02	0.17 \pm 0.07	0.66 \pm 0.11	0.18 \pm 0.02

Table 38: **Raw benchmark results of 13 tabular generators on “SpeedDating” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Density Estimation				Privacy Preservation		ML Efficacy	Structural Fidelity
	Shape \uparrow	Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.95 \pm 0.00	0.94 \pm 0.01	0.96 \pm 0.01	0.78 \pm 0.01	0.14 \pm 0.03	0.00 \pm 0.00	1.02 \pm 0.03	0.44 \pm 0.01
BN	0.95 \pm 0.00	0.95 \pm 0.00	0.97 \pm 0.00	0.28 \pm 0.01	0.34 \pm 0.02	0.01 \pm 0.00	0.83 \pm 0.10	0.36 \pm 0.09
TVAE	0.90 \pm 0.00	0.86 \pm 0.00	0.91 \pm 0.02	0.07 \pm 0.00	0.34 \pm 0.03	0.01 \pm 0.00	0.98 \pm 0.05	0.52 \pm 0.00
GOGGLE	0.88 \pm 0.00	0.87 \pm 0.02	0.87 \pm 0.07	0.28 \pm 0.03	0.25 \pm 0.03	0.24 \pm 0.18	0.83 \pm 0.11	0.25 \pm 0.03
CTGAN	0.85 \pm 0.04	0.84 \pm 0.02	0.92 \pm 0.05	0.04 \pm 0.01	0.26 \pm 0.03	0.23 \pm 0.31	0.93 \pm 0.11	0.13 \pm 0.02
NFlow	0.86 \pm 0.01	0.84 \pm 0.01	0.75 \pm 0.04	0.05 \pm 0.00	0.25 \pm 0.03	0.15 \pm 0.19	0.81 \pm 0.03	0.13 \pm 0.08
ARF	0.92 \pm 0.02	0.89 \pm 0.03	0.90 \pm 0.03	0.24 \pm 0.03	0.23 \pm 0.03	0.06 \pm 0.06	0.92 \pm 0.01	0.56 \pm 0.06
TabDDPM	0.86 \pm 0.08	0.81 \pm 0.13	0.71 \pm 0.24	0.24 \pm 0.11	0.23 \pm 0.04	0.31 \pm 0.40	0.84 \pm 0.10	0.72 \pm 0.14
TabSyn	0.86 \pm 0.05	0.83 \pm 0.07	0.86 \pm 0.08	0.21 \pm 0.06	0.27 \pm 0.06	0.09 \pm 0.03	0.93 \pm 0.12	0.69 \pm 0.04
TabDiff	0.89 \pm 0.04	0.86 \pm 0.05	0.88 \pm 0.06	0.19 \pm 0.04	0.30 \pm 0.09	0.14 \pm 0.15	0.96 \pm 0.08	0.67 \pm 0.03
TabEBM	0.93 \pm 0.01	0.91 \pm 0.01	0.95 \pm 0.03	0.27 \pm 0.02	0.33 \pm 0.13	0.02 \pm 0.01	0.98 \pm 0.06	0.30 \pm 0.01
NRGBoost	0.91 \pm 0.03	0.90 \pm 0.02	0.87 \pm 0.06	0.19 \pm 0.05	0.22 \pm 0.03	0.04 \pm 0.02	0.97 \pm 0.07	0.18 \pm 0.05
GReaT	0.89 \pm 0.01	0.87 \pm 0.02	0.84 \pm 0.04	0.24 \pm 0.03	0.27 \pm 0.06	0.12 \pm 0.00	0.80 \pm 0.10	0.20 \pm 0.04

Table 39: **Raw benchmark results of 13 tabular generators on “Splice” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Density Estimation				Privacy Preservation		ML Efficacy	Structural Fidelity
	Shape \uparrow	Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.98 \pm 0.00	0.97 \pm 0.00	0.98 \pm 0.00	0.88 \pm 0.01	0.00 \pm 0.00	0.00 \pm 0.00	1.01 \pm 0.04	0.47 \pm 0.34
BN	0.99 \pm 0.00	0.95 \pm 0.00	0.93 \pm 0.01	0.34 \pm 0.01	0.64 \pm 0.05	0.04 \pm 0.04	0.69 \pm 0.27	0.19 \pm 0.06
TVAE	0.95 \pm 0.00	0.92 \pm 0.00	0.89 \pm 0.01	0.56 \pm 0.01	0.83 \pm 0.04	0.03 \pm 0.02	1.00 \pm 0.04	0.48 \pm 0.23
GOGGLE	0.94 \pm 0.02	0.90 \pm 0.03	0.83 \pm 0.08	0.49 \pm 0.07	0.50 \pm 0.03	0.08 \pm 0.08	0.64 \pm 0.22	0.30 \pm 0.16
CTGAN	0.94 \pm 0.01	0.90 \pm 0.01	0.94 \pm 0.04	0.43 \pm 0.01	0.67 \pm 0.03	0.03 \pm 0.05	0.99 \pm 0.05	0.24 \pm 0.10
NFlow	0.85 \pm 0.01	0.77 \pm 0.01	0.70 \pm 0.15	0.21 \pm 0.04	0.64 \pm 0.07	0.04 \pm 0.03	0.47 \pm 0.07	0.31 \pm 0.20
ARF	0.99 \pm 0.00	0.95 \pm 0.00	0.91 \pm 0.01	0.45 \pm 0.01	0.91 \pm 0.02	0.01 \pm 0.00	0.78 \pm 0.07	0.21 \pm 0.05
TabDDPM	0.95 \pm 0.01	0.91 \pm 0.02	0.70 \pm 0.21	0.38 \pm 0.18	0.61 \pm 0.09	0.06 \pm 0.06	0.54 \pm 0.20	0.60 \pm 0.28
TabSyn	0.85 \pm 0.11	0.77 \pm 0.17	0.54 \pm 0.39	0.32 \pm 0.25	0.57 \pm 0.06	0.35 \pm 0.63	0.93 \pm 0.11	0.66 \pm 0.24
TabDiff	0.87 \pm 0.09	0.80 \pm 0.13	0.56 \pm 0.38	0.31 \pm 0.25	0.58 \pm 0.07	0.39 \pm 0.47	0.88 \pm 0.18	0.65 \pm 0.25
TabEBM	0.96 \pm 0.01	0.93 \pm 0.01	0.94 \pm 0.05	0.59 \pm 0.04	0.26 \pm 0.28	0.01 \pm 0.01	0.98 \pm 0.07	0.35 \pm 0.23
NRGBoost	0.92 \pm 0.04	0.88 \pm 0.05	0.82 \pm 0.10	0.38 \pm 0.18	0.28 \pm 0.26	0.04 \pm 0.03	0.97 \pm 0.07	0.31 \pm 0.20
GReaT	0.94 \pm 0.02	0.90 \pm 0.03	0.83 \pm 0.08	0.49 \pm 0.07	0.50 \pm 0.03	0.08 \pm 0.08	0.64 \pm 0.22	0.30 \pm 0.16

Table 40: **Raw benchmark results of 13 tabular generators on “Vehicle” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Density Estimation				Privacy Preservation		ML Efficacy	Structural Fidelity
	Shape \uparrow	Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.95 \pm 0.00	0.97 \pm 0.01	0.96 \pm 0.01	0.86 \pm 0.01	0.12 \pm 0.04	0.01 \pm 0.01	0.98 \pm 0.04	0.41 \pm 0.37
BN	0.94 \pm 0.00	0.96 \pm 0.00	0.97 \pm 0.01	0.33 \pm 0.03	0.17 \pm 0.02	0.01 \pm 0.01	0.64 \pm 0.24	0.28 \pm 0.24
TVAE	0.83 \pm 0.01	0.86 \pm 0.00	0.77 \pm 0.01	0.08 \pm 0.01	0.39 \pm 0.08	0.16 \pm 0.09	0.84 \pm 0.10	0.39 \pm 0.35
GOGGLE	0.89 \pm 0.01	0.91 \pm 0.01	0.88 \pm 0.03	0.30 \pm 0.05	0.22 \pm 0.04	0.07 \pm 0.02	0.59 \pm 0.19	0.23 \pm 0.18
CTGAN	0.78 \pm 0.02	0.90 \pm 0.01	0.82 \pm 0.05	0.02 \pm 0.01	0.24 \pm 0.06	0.13 \pm 0.13	0.82 \pm 0.19	0.08 \pm 0.05
NFlow	0.88 \pm 0.01	0.85 \pm 0.01	0.89 \pm 0.02	0.00 \pm 0.00	0.24 \pm 0.03	0.13 \pm 0.07	0.46 \pm 0.06	0.09 \pm 0.05
ARF	0.94 \pm 0.00	0.93 \pm 0.00	0.96 \pm 0.01	0.16 \pm 0.02	0.17 \pm 0.03	0.01 \pm 0.00	0.84 \pm 0.04	0.43 \pm 0.05
TabDDPM	0.85 \pm 0.06	0.90 \pm 0.02	0.77 \pm 0.15	0.28 \pm 0.07	0.14 \pm 0.05	0.04 \pm 0.03	0.62 \pm 0.23	0.72 \pm 0.28
TabSyn	0.88 \pm 0.02	0.93 \pm 0.01	0.92 \pm 0.03	0.27 \pm 0.09	0.21 \pm 0.04	0.04 \pm 0.03	0.88 \pm 0.12	0.74 \pm 0.26
TabDiff	0.88 \pm 0.03	0.87 \pm 0.06	0.84 \pm 0.08	0.18 \pm 0.19	0.31 \pm 0.13	0.07 \pm 0.05	0.84 \pm 0.19	0.62 \pm 0.40
TabEBM	0.91 \pm 0.01	0.94 \pm 0.02	0.93 \pm 0.03	0.40 \pm 0.05	0.36 \pm 0.18	0.04 \pm 0.03	0.93 \pm 0.09	0.28 \pm 0.25
NRGBoost	0.91 \pm 0.00	0.88 \pm 0.05	0.88 \pm 0.03	0.18 \pm 0.18	0.26 \pm 0.08	0.12 \pm 0.10	0.88 \pm 0.13	0.15 \pm 0.16
GReaT	0.85 \pm 0.06	0.87 \pm 0.06	0.75 \pm 0.17	0.18 \pm 0.18	0.27 \pm 0.09	0.20 \pm 0.17	0.49 \pm 0.19	0.15 \pm 0.16

Table 41: **Raw benchmark results of 13 tabular generators on “Zernike” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Density Estimation				Privacy Preservation		ML Efficacy	Structural Fidelity
	Shape \uparrow	Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.97 \pm 0.00	0.98 \pm 0.00	0.90 \pm 0.01	0.90 \pm 0.01	0.20 \pm 0.03	0.00 \pm 0.00	0.98 \pm 0.03	0.31 \pm 0.32
BN	0.97 \pm 0.00	0.98 \pm 0.00	0.96 \pm 0.01	0.72 \pm 0.01	0.18 \pm 0.02	0.00 \pm 0.00	0.54 \pm 0.42	0.31 \pm 0.31
TVAE	0.87 \pm 0.00	0.93 \pm 0.00	0.76 \pm 0.02	0.03 \pm 0.01	0.47 \pm 0.03	0.31 \pm 0.20	0.90 \pm 0.06	0.38 \pm 0.37
GOGGLE	0.90 \pm 0.02	0.94 \pm 0.01	0.79 \pm 0.06	0.31 \pm 0.07	0.35 \pm 0.03	0.18 \pm 0.06	0.42 \pm 0.30	0.18 \pm 0.18
CTGAN	0.81 \pm 0.02	0.95 \pm 0.00	0.65 \pm 0.07	0.00 \pm 0.00	0.40 \pm 0.05	0.03 \pm 0.05	0.82 \pm 0.19	0.06 \pm 0.06
NFlow	0.90 \pm 0.01	0.87 \pm 0.00	0.77 \pm 0.02	0.00 \pm 0.00	0.41 \pm 0.03	0.80 \pm 0.20	0.14 \pm 0.03	0.01 \pm 0.01
ARF	0.96 \pm 0.00	0.94 \pm 0.00	0.87 \pm 0.01	0.01 \pm 0.00	0.40 \pm 0.04	0.01 \pm 0.00	0.77 \pm 0.04	0.21 \pm 0.01
TabDDPM	0.68 \pm 0.26	0.92 \pm 0.03	0.44 \pm 0.43	0.19 \pm 0.20	0.51 \pm 0.20	0.21 \pm 0.11	0.30 \pm 0.28	0.62 \pm 0.41
TabSyn	0.92 \pm 0.01	0.96 \pm 0.01	0.83 \pm 0.03	0.24 \pm 0.14	0.36 \pm 0.05	0.09 \pm 0.09	0.84 \pm 0.17	0.70 \pm 0.30
TabDiff	0.91 \pm 0.01	0.92 \pm 0.03	0.77 \pm 0.08	0.19 \pm 0.20	0.40 \pm 0.09	0.11 \pm 0.07	0.82 \pm 0.21	0.61 \pm 0.42
TabEBM	0.94 \pm 0.02	0.96 \pm 0.02	0.91 \pm 0.07	0.40 \pm 0.03	0.37 \pm 0.06	0.08 \pm 0.09	0.92 \pm 0.10	0.23 \pm 0.23
NRGBoost	0.95 \pm 0.02	0.94 \pm 0.01	0.89 \pm 0.05	0.19 \pm 0.20	0.35 \pm 0.05	0.16 \pm 0.14	0.90 \pm 0.11	0.13 \pm 0.16
GReaT	0.84 \pm 0.09	0.91 \pm 0.04	0.55 \pm 0.31	0.19 \pm 0.20	0.29 \pm 0.05	0.56 \pm 0.53	0.29 \pm 0.28	0.11 \pm 0.17

Table 42: **Raw benchmark results of 13 tabular generators on “Ailerons” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Density Estimation				Privacy Preservation		ML Efficacy	Structural Fidelity
	Shape \uparrow	Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.71 \pm 0.03	0.99 \pm 0.00	0.90 \pm 0.01	0.85 \pm 0.00	0.01 \pm 0.00	0.00 \pm 0.00	0.88 \pm 0.20	0.35 \pm 0.36
BN	0.75 \pm 0.03	0.96 \pm 0.00	0.94 \pm 0.00	0.13 \pm 0.00	0.05 \pm 0.02	0.00 \pm 0.00	0.56 \pm 0.31	0.51 \pm 0.18
TVAE	0.70 \pm 0.03	0.96 \pm 0.00	0.86 \pm 0.02	0.28 \pm 0.01	0.02 \pm 0.00	0.18 \pm 0.26	0.76 \pm 0.23	0.46 \pm 0.20
GOGGLE	0.57 \pm 0.20	0.90 \pm 0.06	0.53 \pm 0.37	0.19 \pm 0.19	0.05 \pm 0.04	1.61 \pm 2.36	0.80 \pm 0.29	0.18 \pm 0.20
CTGAN	0.68 \pm 0.02	0.96 \pm 0.00	0.91 \pm 0.05	0.09 \pm 0.02	0.02 \pm 0.00	0.06 \pm 0.06	0.75 \pm 0.38	0.13 \pm 0.12
NFlow	0.68 \pm 0.03	0.89 \pm 0.01	0.63 \pm 0.06	0.00 \pm 0.00	0.12 \pm 0.10	0.94 \pm 0.75	0.51 \pm 0.33	0.05 \pm 0.07
ARF	0.73 \pm 0.02	0.98 \pm 0.00	0.95 \pm 0.01	0.22 \pm 0.01	0.03 \pm 0.00	0.00 \pm 0.00	0.64 \pm 0.26	0.58 \pm 0.15
TabDDPM	0.72 \pm 0.04	0.94 \pm 0.02	0.84 \pm 0.05	0.30 \pm 0.07	0.02 \pm 0.02	0.09 \pm 0.10	0.52 \pm 0.33	0.69 \pm 0.24
TabSyn	0.52 \pm 0.26	0.90 \pm 0.07	0.60 \pm 0.33	0.18 \pm 0.19	0.17 \pm 0.21	3.08 \pm 4.43	0.81 \pm 0.27	0.64 \pm 0.32
TabDiff	0.76 \pm 0.02	0.97 \pm 0.02	0.87 \pm 0.16	0.22 \pm 0.16	0.07 \pm 0.12	0.09 \pm 0.10	0.87 \pm 0.20	0.71 \pm 0.23
TabEBM	0.76 \pm 0.02	0.96 \pm 0.00	0.93 \pm 0.05	0.22 \pm 0.15	0.07 \pm 0.04	0.09 \pm 0.10	0.57 \pm 0.29	0.43 \pm 0.03
NRGBoost	0.68 \pm 0.08	0.92 \pm 0.04	0.53 \pm 0.37	0.18 \pm 0.19	0.22 \pm 0.22	0.39 \pm 0.48	0.79 \pm 0.29	0.18 \pm 0.20
GReaT	0.67 \pm 0.10	0.94 \pm 0.03	0.67 \pm 0.22	0.20 \pm 0.18	0.07 \pm 0.05	0.97 \pm 1.33	0.49 \pm 0.33	0.20 \pm 0.20

Table 43: **Raw benchmark results of 13 tabular generators on “California” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Density Estimation				Privacy Preservation		ML Efficacy	Structural Fidelity
	Shape \uparrow	Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.98 \pm 0.00	0.99 \pm 0.00	0.99 \pm 0.00	0.78 \pm 0.00	0.03 \pm 0.02	0.00 \pm 0.00	0.96 \pm 0.05	0.44 \pm 0.44
BN	0.98 \pm 0.00	0.97 \pm 0.01	0.98 \pm 0.00	0.44 \pm 0.00	0.04 \pm 0.02	0.00 \pm 0.00	0.73 \pm 0.27	0.72 \pm 0.14
TVAE	0.94 \pm 0.01	0.91 \pm 0.01	0.97 \pm 0.01	0.23 \pm 0.01	0.07 \pm 0.02	0.10 \pm 0.09	0.81 \pm 0.12	0.53 \pm 0.24
GOGGLE	0.71 \pm 0.26	0.83 \pm 0.10	0.72 \pm 0.26	0.21 \pm 0.22	0.08 \pm 0.03	2.75 \pm 3.91	0.80 \pm 0.25	0.14 \pm 0.22
CTGAN	0.91 \pm 0.01	0.93 \pm 0.00	0.96 \pm 0.02	0.18 \pm 0.02	0.03 \pm 0.01	0.15 \pm 0.12	0.84 \pm 0.17	0.16 \pm 0.16
NFlow	0.89 \pm 0.02	0.86 \pm 0.01	0.90 \pm 0.04	0.08 \pm 0.03	0.12 \pm 0.05	0.33 \pm 0.38	0.45 \pm 0.10	0.06 \pm 0.10
ARF	0.97 \pm 0.00	0.87 \pm 0.01	0.99 \pm 0.00	0.26 \pm 0.01	0.05 \pm 0.01	0.00 \pm 0.00	0.69 \pm 0.24	0.68 \pm 0.16
TabDDPM	0.93 \pm 0.03	0.94 \pm 0.01	0.94 \pm 0.04	0.42 \pm 0.00	0.04 \pm 0.02	0.04 \pm 0.04	0.60 \pm 0.23	0.79 \pm 0.19
TabSyn	0.95 \pm 0.01	0.94 \pm 0.01	0.92 \pm 0.07	0.40 \pm 0.03	0.06 \pm 0.02	0.39 \pm 0.54	0.88 \pm 0.13	0.78 \pm 0.20
TabDiff	0.94 \pm 0.02	0.90 \pm 0.04	0.96 \pm 0.02	0.28 \pm 0.16	0.09 \pm 0.04	0.04 \pm 0.04	0.88 \pm 0.13	0.75 \pm 0.22
TabEBM	0.93 \pm 0.02	0.92 \pm 0.01	0.97 \pm 0.01	0.24 \pm 0.19	0.11 \pm 0.05	0.04 \pm 0.04	0.62 \pm 0.18	0.47 \pm 0.05
NRGBoost	0.93 \pm 0.02	0.89 \pm 0.05	0.94 \pm 0.07	0.23 \pm 0.20	0.05 \pm 0.01	0.05 \pm 0.04	0.77 \pm 0.30	0.15 \pm 0.21
GReaT	0.88 \pm 0.08	0.88 \pm 0.05	0.87 \pm 0.12	0.22 \pm 0.21	0.12 \pm 0.07	0.10 \pm 0.06	0.49 \pm 0.20	0.16 \pm 0.21

Table 44: **Raw benchmark results of 13 tabular generators on “Elevators” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Shape \uparrow	Density Estimation			Privacy Preservation		ML Efficacy	Structural Fidelity
		Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.85 \pm 0.01	0.99 \pm 0.00	0.95 \pm 0.00	0.81 \pm 0.00	0.02 \pm 0.01	0.00 \pm 0.00	0.92 \pm 0.06	0.39 \pm 0.06
BN	0.87 \pm 0.01	0.96 \pm 0.00	0.96 \pm 0.00	0.28 \pm 0.00	0.05 \pm 0.00	0.00 \pm 0.00	0.64 \pm 0.12	0.61 \pm 0.14
TVAE	0.82 \pm 0.02	0.94 \pm 0.00	0.91 \pm 0.01	0.25 \pm 0.01	0.05 \pm 0.01	0.14 \pm 0.06	0.78 \pm 0.04	0.50 \pm 0.05
GOGGLE	0.64 \pm 0.09	0.87 \pm 0.05	0.63 \pm 0.13	0.20 \pm 0.02	0.06 \pm 0.02	2.18 \pm 0.81	0.80 \pm 0.00	0.16 \pm 0.03
CTGAN	0.79 \pm 0.02	0.94 \pm 0.00	0.94 \pm 0.04	0.14 \pm 0.02	0.02 \pm 0.00	0.11 \pm 0.07	0.79 \pm 0.06	0.14 \pm 0.02
NFlow	0.78 \pm 0.02	0.88 \pm 0.01	0.77 \pm 0.05	0.04 \pm 0.01	0.12 \pm 0.01	0.64 \pm 0.43	0.48 \pm 0.04	0.05 \pm 0.01
ARF	0.85 \pm 0.01	0.93 \pm 0.01	0.97 \pm 0.00	0.24 \pm 0.01	0.04 \pm 0.01	0.00 \pm 0.00	0.67 \pm 0.03	0.63 \pm 0.07
TabDDPM	0.82 \pm 0.04	0.94 \pm 0.00	0.89 \pm 0.04	0.36 \pm 0.04	0.03 \pm 0.02	0.07 \pm 0.03	0.56 \pm 0.05	0.74 \pm 0.07
TabSyn	0.74 \pm 0.14	0.92 \pm 0.03	0.76 \pm 0.20	0.29 \pm 0.11	0.11 \pm 0.07	1.73 \pm 1.91	0.84 \pm 0.05	0.71 \pm 0.10
TabDiff	0.85 \pm 0.02	0.93 \pm 0.03	0.91 \pm 0.07	0.25 \pm 0.04	0.08 \pm 0.01	0.06 \pm 0.03	0.87 \pm 0.01	0.73 \pm 0.02
TabEBM	0.84 \pm 0.02	0.94 \pm 0.01	0.95 \pm 0.03	0.23 \pm 0.01	0.09 \pm 0.03	0.07 \pm 0.03	0.59 \pm 0.03	0.45 \pm 0.03
NRGBoost	0.81 \pm 0.05	0.91 \pm 0.03	0.74 \pm 0.22	0.21 \pm 0.03	0.14 \pm 0.11	0.22 \pm 0.25	0.78 \pm 0.02	0.17 \pm 0.02
GReaT	0.78 \pm 0.09	0.91 \pm 0.04	0.77 \pm 0.14	0.21 \pm 0.02	0.10 \pm 0.04	0.54 \pm 0.62	0.49 \pm 0.00	0.18 \pm 0.03

Table 45: **Raw benchmark results of 13 tabular generators on “H16” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Shape \uparrow	Density Estimation			Privacy Preservation		ML Efficacy	Structural Fidelity
		Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.88 \pm 0.01	0.99 \pm 0.00	0.95 \pm 0.00	0.83 \pm 0.00	0.05 \pm 0.02	0.00 \pm 0.01	0.98 \pm 0.02	0.45 \pm 0.43
BN	0.89 \pm 0.01	0.99 \pm 0.00	0.99 \pm 0.00	0.61 \pm 0.01	0.03 \pm 0.01	0.00 \pm 0.00	0.80 \pm 0.23	0.80 \pm 0.11
TVAE	0.85 \pm 0.01	0.98 \pm 0.00	0.94 \pm 0.02	0.29 \pm 0.01	0.10 \pm 0.01	0.35 \pm 0.50	0.86 \pm 0.09	0.62 \pm 0.22
GOGGLE	0.68 \pm 0.21	0.95 \pm 0.03	0.61 \pm 0.37	0.23 \pm 0.24	0.08 \pm 0.03	6.50 \pm 7.67	0.86 \pm 0.17	0.20 \pm 0.21
CTGAN	0.81 \pm 0.02	0.97 \pm 0.00	0.97 \pm 0.01	0.22 \pm 0.03	0.05 \pm 0.02	0.07 \pm 0.09	0.87 \pm 0.13	0.20 \pm 0.19
NFlow	0.83 \pm 0.02	0.94 \pm 0.00	0.86 \pm 0.08	0.07 \pm 0.01	0.11 \pm 0.05	0.07 \pm 0.05	0.57 \pm 0.11	0.10 \pm 0.14
ARF	0.90 \pm 0.00	0.98 \pm 0.00	0.94 \pm 0.00	0.19 \pm 0.01	0.06 \pm 0.03	0.00 \pm 0.00	0.74 \pm 0.20	0.70 \pm 0.16
TabDDPM	0.83 \pm 0.06	0.96 \pm 0.02	0.89 \pm 0.07	0.40 \pm 0.06	0.04 \pm 0.02	0.05 \pm 0.07	0.65 \pm 0.17	0.77 \pm 0.20
TabSyn	0.69 \pm 0.20	0.95 \pm 0.05	0.78 \pm 0.23	0.24 \pm 0.23	0.10 \pm 0.06	3.88 \pm 8.74	0.85 \pm 0.18	0.67 \pm 0.33
TabDiff	0.85 \pm 0.04	0.96 \pm 0.02	0.89 \pm 0.07	0.24 \pm 0.23	0.20 \pm 0.17	0.13 \pm 0.12	0.87 \pm 0.16	0.71 \pm 0.28
TabEBM	0.87 \pm 0.01	0.97 \pm 0.01	0.96 \pm 0.02	0.26 \pm 0.21	0.16 \pm 0.12	0.06 \pm 0.08	0.64 \pm 0.21	0.50 \pm 0.25
NRGBoost	0.87 \pm 0.01	0.95 \pm 0.03	0.94 \pm 0.01	0.24 \pm 0.23	0.07 \pm 0.02	0.23 \pm 0.32	0.82 \pm 0.23	0.17 \pm 0.22
GReaT	0.77 \pm 0.12	0.95 \pm 0.03	0.88 \pm 0.08	0.25 \pm 0.22	0.12 \pm 0.07	0.72 \pm 0.83	0.58 \pm 0.18	0.21 \pm 0.21

Table 46: **Raw benchmark results of 13 tabular generators on “Liver” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Density Estimation				Privacy Preservation		ML Efficacy	Structural Fidelity
	Shape \uparrow	Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.90 \pm 0.02	0.97 \pm 0.01	0.90 \pm 0.03	0.81 \pm 0.01	0.11 \pm 0.01	0.01 \pm 0.01	1.00 \pm 0.05	0.43 \pm 0.29
BN	0.91 \pm 0.01	0.96 \pm 0.01	0.94 \pm 0.02	0.54 \pm 0.03	0.23 \pm 0.05	0.01 \pm 0.01	0.91 \pm 0.11	0.76 \pm 0.14
TVAE	0.77 \pm 0.01	0.91 \pm 0.01	0.50 \pm 0.05	0.47 \pm 0.03	0.18 \pm 0.04	0.12 \pm 0.05	0.97 \pm 0.07	0.72 \pm 0.13
GOGGLE	0.65 \pm 0.20	0.90 \pm 0.04	0.77 \pm 0.09	0.37 \pm 0.19	0.18 \pm 0.03	0.17 \pm 0.12	0.94 \pm 0.13	0.28 \pm 0.21
CTGAN	0.49 \pm 0.06	0.87 \pm 0.03	0.61 \pm 0.17	0.16 \pm 0.06	0.29 \pm 0.08	0.37 \pm 0.27	0.95 \pm 0.11	0.19 \pm 0.13
NFlow	0.88 \pm 0.01	0.92 \pm 0.02	0.93 \pm 0.04	0.47 \pm 0.05	0.14 \pm 0.02	0.02 \pm 0.01	0.86 \pm 0.10	0.37 \pm 0.25
ARF	0.90 \pm 0.01	0.96 \pm 0.01	0.88 \pm 0.05	0.48 \pm 0.04	0.18 \pm 0.05	0.01 \pm 0.01	0.93 \pm 0.13	0.81 \pm 0.08
TabDDPM	0.84 \pm 0.01	0.93 \pm 0.02	0.88 \pm 0.06	0.54 \pm 0.02	0.13 \pm 0.04	0.06 \pm 0.05	0.87 \pm 0.10	0.77 \pm 0.14
TabSyn	0.86 \pm 0.03	0.95 \pm 0.01	0.89 \pm 0.07	0.55 \pm 0.02	0.17 \pm 0.03	0.05 \pm 0.05	1.00 \pm 0.05	0.80 \pm 0.13
TabDiff	0.86 \pm 0.03	0.96 \pm 0.02	0.87 \pm 0.06	0.49 \pm 0.09	0.20 \pm 0.05	0.05 \pm 0.05	1.00 \pm 0.05	0.81 \pm 0.13
TabEBM	0.86 \pm 0.03	0.94 \pm 0.01	0.89 \pm 0.07	0.65 \pm 0.10	0.13 \pm 0.04	0.06 \pm 0.05	0.93 \pm 0.05	0.61 \pm 0.05
NRGBoost	0.85 \pm 0.02	0.91 \pm 0.03	0.88 \pm 0.07	0.52 \pm 0.05	0.15 \pm 0.03	0.06 \pm 0.05	0.98 \pm 0.06	0.37 \pm 0.23
GReaT	0.78 \pm 0.05	0.93 \pm 0.02	0.81 \pm 0.05	0.46 \pm 0.11	0.17 \pm 0.02	0.08 \pm 0.04	0.87 \pm 0.11	0.36 \pm 0.23

Table 47: **Raw benchmark results of 13 tabular generators on “Sales” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Density Estimation				Privacy Preservation		ML Efficacy	Structural Fidelity
	Shape \uparrow	Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.80 \pm 0.00	0.96 \pm 0.01	0.97 \pm 0.01	0.79 \pm 0.00	0.03 \pm 0.01	0.00 \pm 0.00	0.97 \pm 0.04	0.39 \pm 0.38
BN	0.79 \pm 0.01	0.89 \pm 0.00	0.94 \pm 0.00	0.29 \pm 0.00	0.22 \pm 0.02	0.01 \pm 0.00	0.58 \pm 0.28	0.59 \pm 0.24
TVAE	0.73 \pm 0.01	0.87 \pm 0.00	0.91 \pm 0.02	0.27 \pm 0.01	0.25 \pm 0.03	0.03 \pm 0.01	0.81 \pm 0.12	0.62 \pm 0.22
GOGGLE	0.57 \pm 0.24	0.81 \pm 0.11	0.68 \pm 0.29	0.24 \pm 0.25	0.18 \pm 0.08	12.17 \pm 17.42	0.80 \pm 0.25	0.22 \pm 0.20
CTGAN	0.71 \pm 0.01	0.89 \pm 0.01	0.95 \pm 0.04	0.26 \pm 0.02	0.11 \pm 0.01	0.04 \pm 0.08	0.83 \pm 0.17	0.25 \pm 0.25
NFlow	0.73 \pm 0.01	0.84 \pm 0.01	0.87 \pm 0.11	0.24 \pm 0.02	0.17 \pm 0.05	0.07 \pm 0.06	0.43 \pm 0.19	0.14 \pm 0.20
ARF	0.75 \pm 0.04	0.89 \pm 0.02	0.85 \pm 0.09	0.38 \pm 0.10	0.16 \pm 0.03	1.38 \pm 1.66	0.57 \pm 0.28	0.62 \pm 0.20
TabDDPM	0.66 \pm 0.14	0.84 \pm 0.08	0.47 \pm 0.49	0.24 \pm 0.25	0.24 \pm 0.12	3.40 \pm 6.24	0.42 \pm 0.28	0.60 \pm 0.40
TabSyn	0.78 \pm 0.02	0.91 \pm 0.00	0.96 \pm 0.03	0.34 \pm 0.14	0.16 \pm 0.04	0.02 \pm 0.02	0.90 \pm 0.10	0.78 \pm 0.20
TabDiff	0.78 \pm 0.02	0.91 \pm 0.00	0.96 \pm 0.03	0.33 \pm 0.15	0.15 \pm 0.03	0.02 \pm 0.02	0.90 \pm 0.10	0.79 \pm 0.20
TabEBM	0.77 \pm 0.03	0.89 \pm 0.02	0.91 \pm 0.09	0.31 \pm 0.19	0.13 \pm 0.03	0.64 \pm 1.37	0.66 \pm 0.12	0.50 \pm 0.03
NRGBoost	0.70 \pm 0.09	0.84 \pm 0.07	0.56 \pm 0.40	0.27 \pm 0.21	0.23 \pm 0.10	0.15 \pm 0.24	0.72 \pm 0.37	0.18 \pm 0.19
GReaT	0.75 \pm 0.04	0.89 \pm 0.02	0.85 \pm 0.09	0.38 \pm 0.10	0.16 \pm 0.03	1.38 \pm 1.66	0.51 \pm 0.25	0.27 \pm 0.25

Table 48: **Raw benchmark results of 13 tabular generators on “Space” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Density Estimation				Privacy Preservation		ML Efficacy	Structural Fidelity
	Shape \uparrow	Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.98 \pm 0.00	0.99 \pm 0.00	0.98 \pm 0.01	0.78 \pm 0.01	0.08 \pm 0.04	0.00 \pm 0.00	0.96 \pm 0.04	0.42 \pm 0.41
BN	0.98 \pm 0.00	0.99 \pm 0.01	0.97 \pm 0.01	0.57 \pm 0.01	0.14 \pm 0.03	0.00 \pm 0.00	0.79 \pm 0.19	0.92 \pm 0.05
TVAE	0.87 \pm 0.01	0.90 \pm 0.01	0.85 \pm 0.02	0.11 \pm 0.01	0.20 \pm 0.02	0.23 \pm 0.18	0.75 \pm 0.15	0.40 \pm 0.36
GOGGLE	0.72 \pm 0.21	0.88 \pm 0.08	0.72 \pm 0.24	0.21 \pm 0.22	0.14 \pm 0.04	1.91 \pm 2.37	0.82 \pm 0.21	0.15 \pm 0.22
CTGAN	0.77 \pm 0.05	0.93 \pm 0.02	0.77 \pm 0.10	0.05 \pm 0.02	0.20 \pm 0.06	0.21 \pm 0.23	0.80 \pm 0.22	0.08 \pm 0.09
NFlow	0.89 \pm 0.03	0.89 \pm 0.02	0.91 \pm 0.05	0.09 \pm 0.02	0.15 \pm 0.03	0.04 \pm 0.04	0.57 \pm 0.09	0.11 \pm 0.12
ARF	0.97 \pm 0.00	0.98 \pm 0.00	0.98 \pm 0.01	0.32 \pm 0.01	0.10 \pm 0.01	0.00 \pm 0.00	0.73 \pm 0.15	0.73 \pm 0.17
TabDDPM	0.91 \pm 0.01	0.96 \pm 0.01	0.94 \pm 0.03	0.38 \pm 0.04	0.09 \pm 0.04	0.05 \pm 0.05	0.65 \pm 0.15	0.80 \pm 0.22
TabSyn	0.93 \pm 0.02	0.97 \pm 0.02	0.94 \pm 0.03	0.35 \pm 0.07	0.15 \pm 0.03	0.04 \pm 0.05	0.89 \pm 0.11	0.79 \pm 0.22
TabDiff	0.94 \pm 0.03	0.97 \pm 0.02	0.94 \pm 0.03	0.34 \pm 0.08	0.13 \pm 0.02	0.04 \pm 0.05	0.89 \pm 0.12	0.78 \pm 0.22
TabEBM	0.94 \pm 0.02	0.95 \pm 0.01	0.94 \pm 0.03	0.32 \pm 0.10	0.14 \pm 0.02	0.04 \pm 0.05	0.67 \pm 0.12	0.47 \pm 0.05
NRGBoost	0.93 \pm 0.01	0.87 \pm 0.09	0.90 \pm 0.02	0.21 \pm 0.22	0.20 \pm 0.09	0.40 \pm 0.40	0.84 \pm 0.19	0.16 \pm 0.21
GReaT	0.89 \pm 0.03	0.91 \pm 0.04	0.82 \pm 0.11	0.27 \pm 0.16	0.15 \pm 0.03	0.12 \pm 0.06	0.57 \pm 0.14	0.17 \pm 0.21

Table 49: **Raw benchmark results of 13 tabular generators on “Superconductivity” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Density Estimation				Privacy Preservation		ML Efficacy	Structural Fidelity
	Shape \uparrow	Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.95 \pm 0.00	1.00 \pm 0.00	0.99 \pm 0.00	0.45 \pm 0.00	0.01 \pm 0.00	0.00 \pm 0.00	0.97 \pm 0.03	0.41 \pm 0.42
BN	0.96 \pm 0.00	0.99 \pm 0.00	0.99 \pm 0.00	0.16 \pm 0.00	0.07 \pm 0.02	0.00 \pm 0.00	0.72 \pm 0.31	0.85 \pm 0.09
TVAE	0.89 \pm 0.00	0.94 \pm 0.00	0.91 \pm 0.01	0.00 \pm 0.00	0.35 \pm 0.01	0.04 \pm 0.01	0.73 \pm 0.16	0.42 \pm 0.34
GOGGLE	0.86 \pm 0.14	0.94 \pm 0.04	0.82 \pm 0.15	0.17 \pm 0.08	0.26 \pm 0.08	2.10 \pm 3.52	0.82 \pm 0.22	0.20 \pm 0.22
CTGAN	0.86 \pm 0.02	0.95 \pm 0.00	0.85 \pm 0.04	0.00 \pm 0.00	0.38 \pm 0.02	0.17 \pm 0.20	0.76 \pm 0.24	0.05 \pm 0.05
NFlow	0.87 \pm 0.01	0.84 \pm 0.01	0.63 \pm 0.02	0.00 \pm 0.00	0.50 \pm 0.03	4.94 \pm 3.48	0.32 \pm 0.08	0.01 \pm 0.01
ARF	0.95 \pm 0.00	0.99 \pm 0.00	0.96 \pm 0.00	0.02 \pm 0.00	0.18 \pm 0.01	0.00 \pm 0.00	0.64 \pm 0.26	0.54 \pm 0.27
TabDDPM	0.66 \pm 0.28	0.90 \pm 0.06	0.45 \pm 0.48	0.12 \pm 0.12	0.14 \pm 0.10	2.46 \pm 3.84	0.40 \pm 0.21	0.62 \pm 0.42
TabSyn	0.91 \pm 0.03	0.97 \pm 0.01	0.91 \pm 0.04	0.12 \pm 0.12	0.23 \pm 0.04	0.33 \pm 0.51	0.85 \pm 0.16	0.73 \pm 0.28
TabDiff	0.93 \pm 0.01	0.97 \pm 0.02	0.93 \pm 0.03	0.13 \pm 0.11	0.24 \pm 0.01	0.33 \pm 0.51	0.85 \pm 0.16	0.75 \pm 0.25
TabEBM	0.92 \pm 0.00	0.97 \pm 0.01	0.93 \pm 0.02	0.12 \pm 0.12	0.18 \pm 0.06	0.33 \pm 0.51	0.47 \pm 0.27	0.37 \pm 0.10
NRGBoost	0.93 \pm 0.01	0.89 \pm 0.07	0.75 \pm 0.17	0.12 \pm 0.12	0.31 \pm 0.08	4.29 \pm 4.34	0.74 \pm 0.32	0.12 \pm 0.21
GReaT	0.90 \pm 0.03	0.95 \pm 0.01	0.86 \pm 0.06	0.19 \pm 0.05	0.23 \pm 0.01	1.15 \pm 0.74	0.48 \pm 0.21	0.22 \pm 0.22

Table 50: **Raw benchmark results of 13 tabular generators on “Wine” dataset.** We report the normalised mean \pm std metric values across datasets. We highlight the **First**, **Second** and **Third** best performances for each metric. For visualisation, we abbreviate “conditional independence” as “CI”. SMOTE generally achieves the highest performance in capturing local structure (i.e., local utility and local CI), while diffusion models typically excel at capturing global structure (i.e., global CI and global utility)

Generator	Density Estimation				Privacy Preservation		ML Efficacy	Structural Fidelity
	Shape \uparrow	Trend \uparrow	α -precision \uparrow	β -recall \uparrow	DCR \uparrow	δ -Presence \uparrow	Local utility \uparrow	Global utility \uparrow
\mathcal{D}_{ref}	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00	0.00 \pm 0.00	0.00 \pm 0.00	1.00 \pm 0.00	1.00 \pm 0.00
SMOTE	0.97 \pm 0.00	0.99 \pm 0.00	0.94 \pm 0.01	0.67 \pm 0.01	0.08 \pm 0.03	0.00 \pm 0.00	0.98 \pm 0.02	0.44 \pm 0.44
BN	0.97 \pm 0.00	0.93 \pm 0.00	0.96 \pm 0.01	0.18 \pm 0.01	0.22 \pm 0.02	0.01 \pm 0.00	0.78 \pm 0.11	0.49 \pm 0.30
TVAE	0.89 \pm 0.01	0.95 \pm 0.01	0.78 \pm 0.05	0.18 \pm 0.02	0.23 \pm 0.04	0.07 \pm 0.10	0.88 \pm 0.07	0.48 \pm 0.31
GOGGLE	0.72 \pm 0.23	0.92 \pm 0.04	0.63 \pm 0.32	0.18 \pm 0.18	0.26 \pm 0.14	1.49 \pm 1.82	0.87 \pm 0.17	0.12 \pm 0.19
CTGAN	0.88 \pm 0.01	0.97 \pm 0.00	0.95 \pm 0.01	0.16 \pm 0.02	0.13 \pm 0.04	0.03 \pm 0.05	0.92 \pm 0.08	0.17 \pm 0.17
NFlow	0.89 \pm 0.01	0.91 \pm 0.00	0.92 \pm 0.04	0.10 \pm 0.02	0.16 \pm 0.05	0.04 \pm 0.01	0.70 \pm 0.08	0.08 \pm 0.12
ARF	0.96 \pm 0.00	0.98 \pm 0.00	0.97 \pm 0.01	0.22 \pm 0.02	0.17 \pm 0.03	0.00 \pm 0.00	0.81 \pm 0.13	0.66 \pm 0.21
TabDDPM	0.93 \pm 0.01	0.97 \pm 0.01	0.93 \pm 0.01	0.29 \pm 0.07	0.09 \pm 0.05	0.02 \pm 0.01	0.76 \pm 0.12	0.75 \pm 0.25
TabSyn	0.93 \pm 0.01	0.97 \pm 0.01	0.95 \pm 0.02	0.28 \pm 0.09	0.16 \pm 0.03	0.01 \pm 0.02	0.93 \pm 0.07	0.76 \pm 0.24
TabDiff	0.94 \pm 0.01	0.98 \pm 0.01	0.96 \pm 0.03	0.27 \pm 0.09	0.16 \pm 0.04	0.01 \pm 0.02	0.93 \pm 0.08	0.76 \pm 0.24
TabEBM	0.94 \pm 0.00	0.97 \pm 0.00	0.95 \pm 0.02	0.26 \pm 0.10	0.17 \pm 0.05	0.01 \pm 0.02	0.80 \pm 0.07	0.45 \pm 0.03
NRGBoost	0.94 \pm 0.00	0.93 \pm 0.04	0.91 \pm 0.02	0.20 \pm 0.17	0.13 \pm 0.02	0.02 \pm 0.01	0.90 \pm 0.11	0.14 \pm 0.18
GReaT	0.86 \pm 0.08	0.92 \pm 0.04	0.71 \pm 0.23	0.21 \pm 0.15	0.19 \pm 0.07	0.36 \pm 0.45	0.71 \pm 0.12	0.13 \pm 0.18