Received 24 January 2025, accepted 11 March 2025, date of publication 14 March 2025, date of current version 24 March 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3551594

RESEARCH ARTICLE

Enhancing Tabular Data Generation With Dual-Scale Noise Modeling

XIAORONG ZHANG, FEI LI, AND XUTING HU
Anhui Technical College of Mechanical and Electrical Engineering, Wuhu 241002, China

Corresponding author: Xiaorong Zhang (0120190010@ahcme.edu.cn)

This work was supported in part by Anhui Provincial Key Natural Science Foundation Project: Research on Pavement Defect Detection Method Based on Deep Learning under Project 2023AH052699, and in part by Anhui Provincial Key Natural Science Foundation Project: High-Frequency Modeling and Application Research of Multiphase Peak Current Mode Buck Power Converter in Industrial Internet under Project 2022AH052364.

This work involved human subjects or animals in its research. The authors confirm that all human/animal subject research procedures and protocols are exempt from review board approval.

ABSTRACT The generation of synthetic tabular data plays a critical role in applications such as data enhancement, privacy preservation, and model validation. However, the heterogeneity of tabular data—comprising both continuous and categorical features with intricate interdependencies—presents significant challenges in generating high-quality data. This study presents the Dual-Scale Diffusion Probabilistic Method (DSDDPM), a novel framework developed to improve the Denoising Diffusion Probabilistic Model (DDPM) performance for tabular data generation. DSDDPM tackles the challenge of capturing both global dependencies and local details through the use of a dual-scale noise-handling mechanism. It models the global structure at the coarse scale and the regional patterns at the fine scale throughout the forward and backward diffusion processes. Experimental evaluations on several benchmark tabular datasets show that DSDDPM outperforms existing state-of-the-art methods. Specifically, DSDDPM improves machine learning efficiency by an average of 5-7% compared to other methods. DSDDPM exhibits the lowest correlation deviation across all datasets regarding global dependency modeling. Moreover, regarding detail consistency, DSDDPM shows the highest alignment of feature distributions between the generated and actual data, significantly outperforming other methods. These results position DSDDPM as a robust and scalable solution for generating high-quality synthetic tabular data, enhancing fidelity and versatility across various applications.

INDEX TERMS Synthetic data generation, tabular data, diffusion models, data enhancement, global and local dependencies.

I. INTRODUCTION

Tabular data is widely used in various fields, including finance, healthcare, and social sciences, and is often the basis for data-driven decision-making. However, generating high-quality synthetic tabular data still faces significant challenges due to the inherent complexity of tabular data. Unlike the relatively homogeneous structure of image or text data, tabular data usually consists of multiple heterogeneous features, including continuous and categorical variables, which have different distributions and interdependencies [1]. In addition, capturing complex relationships between columns—such as dependencies between numerical features or correlations between categorical variables-makes modeling tabular data even more complicated [2], [3].

Traditional synthetic data generation methods [4], [5], [6], [7], [8], [9], have made significant progress in improving the

The associate editor coordinating the review of this manuscript and approving it for publication was Sedat Akleylek.

fidelity and diversity of generated samples. However, these methods often struggle to handle the complex, multimodal distributions in tabular data and typically fail to preserve global structure (e.g., overall correlation between features) and local details (e.g., feature distribution). Moreover, these models often operate as "black boxes," making it difficult to interpret the generated data or ensure that it accurately reflects the distribution of the underlying data [10].

Denoising diffusion probabilistic models [11] have emerged in recent years as a promising alternative approach to data generation, showing great potential [12], [13]. These models have demonstrated the ability to generate high-quality samples by progressively optimizing noisy data through multiple steps. However, applying diffusion models to tabular data presents new challenges due to the heterogeneity of tabular features. Unlike images, where pixel dependencies are typically local and spatially coherent, tabular data exhibits complex inter-column dependencies, requiring the modeling of global and regional relationships [14], [15].

In this study, we propose a novel generative framework. This dual-scale diffusion method aims to overcome the challenges of DDPM in tabular data generation by introducing a dual-scale noise-handling mechanism. Our approach models the coarse-scale global structure and fine-scale local data patterns during forward and backward diffusion. In this way, DSDDPM can generate synthetic tabular data that captures both the overall distribution of the dataset and the complex dependencies between individual features. This dual-scale approach improves the quality of the generated data and enables the model to handle the heterogeneity of tabular data better.

We evaluate DSDDPM's performance on several standard tabular datasets and demonstrate that it outperforms existing state-of-the-art methods regarding the quality of generated samples. The experimental results show that DSDDPM achieves significant improvements in capturing global and local dependencies of tabular data, making it a robust and scalable solution for generating high-quality synthetic tabular data. The main contributions of this study are as follows:

- We propose a novel Dual-Scale Diffusion Probabilistic Model to improve tabular data generation.
- Our method captures both global dependencies and local details using a dual-scale noise-handling mechanism.
- Experimental results show that DSDDPM outperforms state-of-the-art methods in global correlation preservation, detail consistency, and machine learning efficiency.

The structure of this paper is organized as follows: Section II presents the background and related work, focusing on the limitations of existing approaches and the motivation for this study. Section III provides a detailed description of the proposed methodology, including the design principles of the model, the algorithm framework, and implementation details. Section IV presents the experimental setup, data descriptions, and results, followed by an analysis of the performance and advantages of the proposed method. Finally, Section V summarizes the main contributions of this study and outlines directions for future research.

II. RELATED WORK

The generation of synthetic tabular data has attracted significant attention due to its broad applications in data augmentation [16], privacy preservation [17], and model validation [18]. Early methods for tabular data generation primarily relied on statistical models, such as the Gaussian Mixture Model (GMM) [19] and Markov Chain Monte Carlo (MCMC) [20]. These methods focus on capturing the marginal distributions of features; however, they struggle to model the complex multidimensional dependencies between features, limiting their effectiveness in real-world applications.

Deep generative models have emerged as a promising alternative with the advancement of deep learning. Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs) have been widely adopted. GAN-based methods, such as CTGAN [9] and Tab-GAN [23], generate synthetic data via adversarial training, effectively simulating the distribution of the original dataset. These models perform well in scenarios with relatively simple feature distributions. However, they often struggle with datasets containing continuous and categorical variables, failing to capture their intricate dependencies. Additionally, GANs suffer from issues such as mode collapse and unstable training, mainly when applied to high-dimensional tabular data.

VAEs provide an alternative approach by learning probabilistic representations of tabular data through variational inference. Methods like TVAE [24] and MMD-VAE [25] offer better training stability compared to GANs. Nevertheless, they still face challenges in generating diverse and realistic samples, particularly when handling heterogeneous datasets with complex feature dependencies.

Denoising Diffusion Probabilistic Models (DDPMs) have recently demonstrated strong generative capabilities, particularly in image synthesis. The TabDDPM framework extends DDPM principles to tabular data, gradually introducing and removing noise to generate high-quality samples with distributions that closely match real-world datasets. However, despite its effectiveness, TabDDPM still struggles with modeling complex interdependencies among heterogeneous features [26]. Moreover, existing diffusion models for tabular data typically treat all features equally, which is suboptimal for datasets where feature distributions vary significantly.

Researchers have explored multimodal and hybrid approaches for tabular data generation to overcome these limitations. AutoDiff [27] combines a diffusion model with an autoencoder, enhancing its ability to handle feature heterogeneity and capture correlations more effectively. Similarly, TabMT [28] employs a transformer-based architecture to improve the modeling of feature relationships.

Transformer-based models such as TTVAE [41] leverage attention mechanisms and latent space interpolation to enhance representation learning.

In addition, concerns over data privacy have driven the development of privacy-preserving generative models. Conditional GANs (CGANs) have been explored to generate synthetic data while maintaining both privacy and utility [42]. Recent studies have quantified the trade-offs in SD generation for datasets containing numerical and categorical attributes. Despite these advances, existing methods still face two fundamental challenges:

- Capturing global dependencies, such as feature correlations across the dataset.
- reserving local details, including the fine-grained distribution of individual features.

To address these issues, our proposed Dual-Scale Diffusion Probabilistic Model (DSDDPM) introduces a novel dual-scale noise processing mechanism that enables diffusion models to simultaneously model coarse-scale (global structure) and fine-scale (local details) dependencies. Unlike prior methods, DSDDPM explicitly processes global feature interactions in the forward diffusion process and refines local feature distributions during backward diffusion. As a result, our model can generate high-fidelity and diverse synthetic tabular data that better align with real datasets. This innovation effectively overcomes the limitations of existing techniques while leveraging the inherent strengths of diffusion modeling.

III. BACKGROUND

Diffusion Models (Diffusion Models) are a class of in-depth generative frameworks that realize data generation by gradually adding and removing noise. The core idea is to decompose the complex data distribution into a series of conditional distributions, thus simplifying the modeling difficulty of the generation process. Diffusion models consist of two stages: Forward Diffusion and Reverse Diffusion.

During the forward diffusion process, the raw data x_0 is gradually injected with noise and eventually transformed into a standard Gaussian distribution $q(x_T) = \mathcal{N}(0, \mathbf{I})$. This process can be defined as:

$$q(x_t \mid x_{t-1}) = \mathcal{N}\left(x_t; \sqrt{1-\beta_t}\, x_{t-1}, \beta_t \mathbf{I}\right). \tag{1}$$

β_t is a parameter that controls the amplitude of the noise. The whole process is modeled by a conditionally independent Markov chain, whose goal is to gradually transform the data distribution into a simple distribution that is easy to handle.

In the reverse generation process, the diffusion model gradually recovers data from Gaussian noise. The inverse process $p_\theta(x_t \mid x_{t-1})$ is usually parameterized by a neural network to a Gaussian distribution:

$$p_\theta(x_t \mid x_{t-1}) = \mathcal{N}\left(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)\right) \tag{2}$$

Of these, the $\mu_\theta(x_t, t)$ and $\Sigma_\theta(x_t, t)$ are the mean and variance of the neural network prediction, respectively. In order to optimize the generation process, the diffusion model is trained on the inverse process by means of a Variational Lower Bound (VLB):

$$\log q(x_0) \geq \mathbb{E}_{q(x_{1:T} \mid x_0)} \left[-\sum_{t=1}^{T} \mathrm{KL}\Big(q(x_{t-1} \mid x_t, x_0) \,||\, p_\theta(x_{t-1} \mid x_t)\Big)\right] \tag{3}$$

This training objective can be simplified by transforming the noise prediction task into minimizing the mean square error of the predicted noise concerning the proper noise:

$$L_t^{\text{simple}} = \mathbb{E}_{x_0, \epsilon, t} \|\epsilon - \epsilon_\theta(x_t, t)\|^2 \tag{4}$$

The Gaussian diffusion model is mainly used for the generation of continuous features whose forward and inverse processes are modeled with Gaussian distributions:

$$q(x_t \mid x_{t-1}) := \mathcal{N}(x_t; \sqrt{1-\beta_t}\, x_{t-1}, \beta_t \mathbf{I}) \tag{5}$$

$$q(x_T) := \mathcal{N}(x_T; 0, \mathbf{I}) \tag{6}$$

$$p_\theta(x_{t-1} \mid x_t) := \mathcal{N}\left(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)\right) \tag{7}$$

In the reverse generation process, the model predicts the denoising mean through a neural network $\mu_\theta(x_t, t)$:

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{\beta_t}{\sqrt{1-\overline{\alpha}_t}} \epsilon_\theta(x_t, t) \right) \tag{8}$$

Among them, $\alpha_t = 1 - \beta_t$, and $\overline{\alpha}_t = \prod_{i=1}^{t} \alpha_i$ is the cumulative scaling parameter. The simplified training objective remains to minimize the noise prediction error.

The polynomial diffusion model introduces a diffusion process based on the unconditional distribution to generate categorical features. The forward process is realized by injecting uniform noise into the one-hot encoding of the categorical features, which is defined as:

$$q(x_t \mid x_{t-1}) = \mathrm{Cat}(x_t; (1-\beta_t)x_{t-1} + \beta_t / K) \tag{9}$$

where K denotes the number of categories and Cat is the categorical distribution. The ultimate goal is to smooth the categorical distribution to a uniform distribution $q(x_T) = \mathrm{Cat}(x_T; 1/K)$. The inverse generation learns the conditional probability distribution through a neural network $p_\theta(x_t \mid x_{t-1})$ to restore the original categorical distribution gradually.

IV. DUAL-SCALE NOISE DDPM

This section introduces the DSDDPM framework, which is designed to better capture tabular data's global structure and local details. The core innovation lies in introducing a dual-scale noise-processing mechanism, which manages coarse-scale and fine-scale noise during the forward diffusion and backward denoising processes. This mechanism enhances the model's ability to operate at different scales, enabling the generation of higher-quality synthetic tabular data. Figure 1 illustrates the architecture of the DSDDPM model, where QT represents a quantile transformer.

Notably, although DSDDPM's primary design is intended for handling heterogeneous data types, its dual-scale noise weighting mechanism is inherently adaptive. For purely numerical or purely categorical data, the model can automatically focus on noise modeling at a single scale through dynamic weight adjustment (Section IV-E), where numerical data emphasizes coarse-scale noise and categorical data emphasizes fine-scale noise without modifying the architecture for single-type scenarios. Experimental validation demonstrates that this mechanism performs better even on single-type datasets (see Section V-D).

A. BASIC FRAMEWORK

DSDDPM models numerical and categorical data through Gaussian and polynomial diffusion, respectively. We divide the model architecture into coarse-scale and fine-scale noise processing and balance global information and local details in the generation process through a weighting mechanism.

- Coarse-scale noise diffusion: Captures the structure of the global distribution of the data by introducing a significant noise variance.
- Fine-scale noise diffusion: Preserves local details of the data through a minor noise variance.

The model controls global and local information balance by modeling coarse-scale and fine-scale noise separately and eventually by a weighted combination of the noise components. To model the inverse process, we have adopted the MLP architecture based on [34] with appropriate modifications better to suit the task requirements of the dual-scale diffusion model.

B. FORWARD PROCESSES: TWO-SCALE NOISE DIFFUSION

We introduce coarse-scale and fine-scale noise during forward diffusion for numerical and categorical features, respectively. At each time step t, we schedule the noise according to the noise scheduling β_t, adding noise to the data and dealing with coarse and fine scales separately.

1) COARSE-SCALE NOISE DIFFUSION OF NUMERICAL FEATURES

For numerical features, we use Gaussian diffusion, where we first introduce a significant noise variance to destroy the global structure of the data. The process can be expressed as:

$$q(x_{t,c} \mid x_{t-1,c}) = \mathcal{N}\left(x_{t,c}; \sqrt{1-\beta_t}\, x_{t-1,c}, \beta_t \mathbf{I}\right) \tag{10}$$

where $x_{t,c}$ denotes the numerical type of features at time step t, the β_t controls the amount of noise added at each step, and $\mathbf{I}$ is the unit matrix. In this way, coarse-scale noise diffusion helps capture the data's global structure.

2) FINE-SCALE NOISE DIFFUSION OF CATEGORY FEATURES

We use polynomial diffusion for categorical features, introducing noise with more minor variance to preserve local details in the data. Specifically, the fine-scale noise diffusion process is:

$$q(x_{t,f} \mid x_{t-1,f}) = \text{Cat}\left(x_{t,f}; (1-\beta_t)x_{t-1,f} + \frac{\beta_t}{K}\right) \tag{11}$$

where $x_{t,f}$ denotes the polynomial diffusion process of the category type feature, β_t is the noise dispatch, and K is the number of categories. Fine-scale noise diffusion helps the model preserve each category's nuances and ensures the generated data's statistical properties.

Algorithm 1 Forward Diffusion Process for Numerical and Categorical Features

```
Input: $\eta_c, \eta_f$: Coarse and fine noise levels
        $T$: Total number of timesteps
        Input data $X$: Continuous features $x_{\text{num}}$ and
categorical features $x_{\text{cat}}$
Normalize continuous features $x_{\text{num}}$;
One-hot encode categorical features $x_{\text{cat}}$;
for $t = 1, 2, \ldots, T$ do
    For Continuous Features $x_{\text{num}}$:
        Add coarse noise:
          $x_{\text{num}}^{(t)} = x_{\text{num}}^{(t-1)} + \eta_c \cdot z_{\text{num}}, \quad z_{\text{num}} \sim \mathcal{N}(0, \mathbf{I})$;
    For Categorical Features $x_{\text{cat}}$:
        Add fine noise:
    $x_{\text{cat}}^{(t)} = x_{\text{cat}}^{(t-1)} + \eta_f \cdot z_{\text{cat}}, \quad z_{\text{cat}} \sim$ Categorical Uniform;
Output: Noisy data $x_{\text{num}}^{(t)}, x_{\text{cat}}^{(t)}$
```

C. INVERSE PROCESS: TWO-SCALE DENOISING

In the backpropagation process, neural networks denoise coarse-scale and fine-scale noise separately to recover the true distribution of the data. At each time step t, the model outputs the denoised signals for both coarse-scale and fine-scale.

1) COARSE-SCALE DENOISING

For coarse-scale denoising, the model learns the denoising process of the global structure. Specifically, the coarse-scale denoising process can be represented as:

$$p_\theta(x_{t,c} \mid x_{t-1}) = \mathcal{N}\left(x_{t-1,c}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)\right) \tag{12}$$

Fine-scale denoising focuses on the local structure of the data and is especially important in recovering category-based features. Fine-scale noise has less variance and usually retains more local details, so it aims to recover these nuances.

In the inverse process, fine-scale denoising helps to repair local relationships between features and ensure that the detailed parts of the data are consistent with the original data.

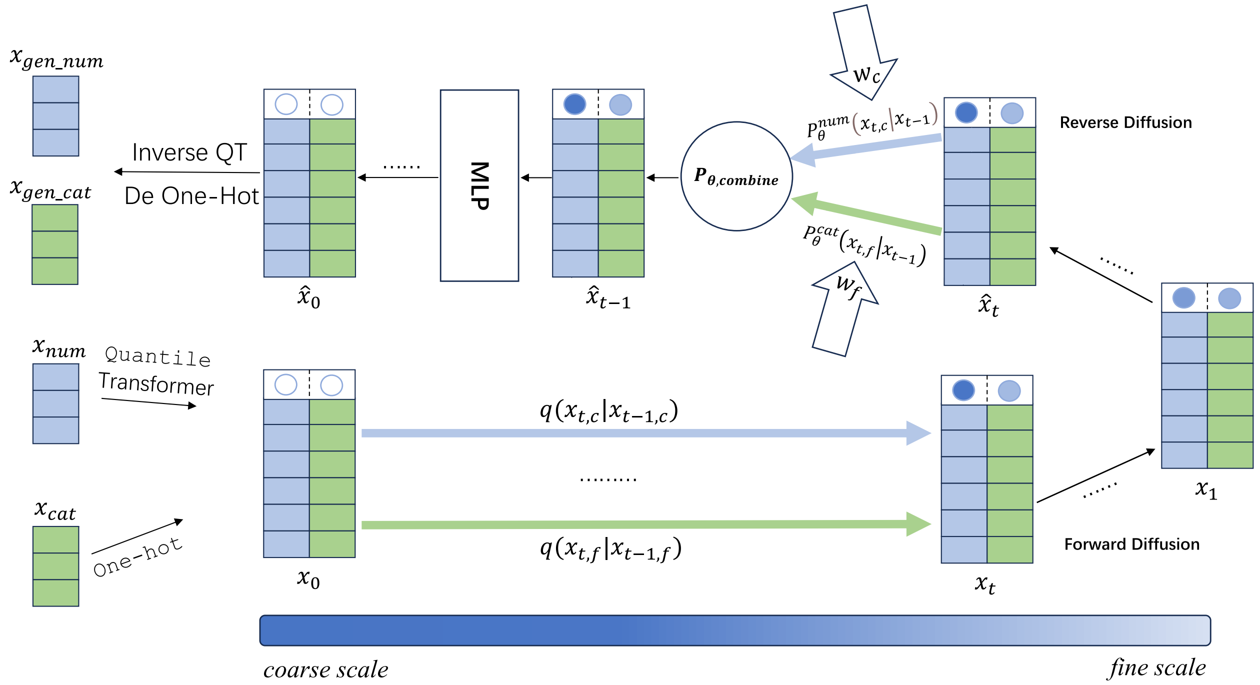


FIGURE 1. The architecture of the proposed DSDDPM model.

This process is usually learned by a separate neural network that removes fine-scale noise. The following equation can represent the process of fine-scale denoising:

$$p_\theta(x_{t,f} \mid x_{t-1}) = \text{Cat}\left(x_{t,f};\, \mu_\theta(x_t, t),\, \Sigma_\theta(x_t, t)\right) \tag{13}$$

Of these, $\mu_\theta(x_t, t)$ and $\Sigma_\theta(x_t, t)$ are the estimates of the neural network outputs, representing the denoising process of the category-based data. By removing fine-scale noise, the model can recover the nuances of each category feature.

D. NOISE-WEIGHTED COMBINATIONS

In order to control the balance between global and local information, we design a weighting mechanism that combines coarse- and fine-scale noise components. Specifically, at each time step t, we perform a weighted combination of the two noises to obtain the final denoised signal:

$$p_\theta^{\text{combine}}(x_t \mid x_{t-1}) = w_c p_\theta^{\text{num}}(x_{t,c} \mid x_{t-1}) + w_f p_\theta^{\text{cat}}(x_{t,f} \mid x_{t-1}) \tag{14}$$

Of which w_c and w_f are the weights of coarse-scale and fine-scale noise, respectively. Notice that $w_c + w_f = 1$, and the weights w_c and w_f can be dynamically adjusted according to the complexity of the data and the needs of the model. Intuitively, the model is more inclined to capture the correlation between the features when the global noise weights are large, while the model is more concerned with the details of the distribution of the features themselves when the local noise weights are large.

Algorithm 2 Reverse Denoising Process for Reconstructing Tabular Data

```
Input: Pretrained denoising models $f_c, f_f$
       Noisy data $x_{\text{num}}^{(T)}$, Noisy data $x_{\text{cat}}^{(T)}$
Initialize reverse process: Set $t = T$;
while $t > 0$ do
    For Continuous Features $x_{\text{num}}$:
        Restore global structure using coarse denoising model $f_c$:
            $x_{\text{num}}^{(t)} = f_c(x_{\text{num}}^{(t)})$;
    For Categorical Features $x_{\text{cat}}$:
        Refine local details using fine denoising model $f_f$:
            $x_{\text{cat}}^{(t)} = f_f(x_{\text{cat}}, t)$;
    Update $t = t - 1$;
Output: Generated data $x_{\text{num}}^{(0)}, x_{\text{cat}}^{(0)}$
```

E. ADAPTIVE WEIGHT ADJUSTMENT MECHANISM

We designed a dynamic weight adjustment mechanism to enable the model to adaptively adjust the proportion of global and local information. This mechanism automatically adjusts the generation effect by monitoring the contribution of coarse- and fine-scale noise in each batch to the w_c and w_f values. Specifically, we dynamically calculate the weights for each time step t by using the following equations:

$$w_c(t) = \frac{\lambda_c \cdot \mathcal{L}_c(t)}{\lambda_c \cdot \mathcal{L}_c(t) + \lambda_f \cdot \mathcal{L}_f(t)} \tag{15}$$

$$w_f(t) = 1 - w_c(t) \tag{16}$$

Of these, the $\mathcal{L}_c(t)$ and $\mathcal{L}_f(t)$ are coarse-scale and fine-scale error terms, respectively. In this way, the model is able to dynamically adjust the scale of the noise according to the different characteristics of the dataset.

F. LOSS FUNCTION

To train the dual-scale TabDDPM, we design a loss function that combines coarse-scale, fine-scale, and weighted combining errors. The final training loss is:

$$\begin{aligned}\mathcal{L} = \lambda_c \|\mathcal{E}_c - \hat{\epsilon}_c(x_t, t)\|^2 \\ + \lambda_f \|\mathcal{E}_f - \hat{\epsilon}_f(x_t, t)\|^2 \\ + \lambda_{\text{combined}} \|\mathcal{E}_{\text{combined}} - \hat{\epsilon}(x_t, t)\|^2\end{aligned} \tag{17}$$

Among them, λ_c, λ_f, and $\lambda_{\text{combined}}$ control the weights of the coarse-scale, fine-scale, and weighted combination error terms, respectively. These weights allow the model to optimize the generation process by adjusting the importance of individual loss terms according to the complexity of the data.

G. BISCALE GENERATION IN THE SAMPLING PROCESS

During sampling, the model generates new samples by backpropagation. At each time step, the model updates the data using a weighted combination of coarse-scale and fine-scale noise:

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_{\text{combined}}\right) + \sigma_t z \tag{18}$$

where α_t is the scaling factor at each step, and z is the standard normally distributed random noise. The model preserves global structure and local details through this process when generating new samples.

H. TIME COMPLEXITY ANALYSIS

Let $D = D_{\text{num}} + D_{\text{cat}}$ denote the total feature dimension, where D_{num} and D_{cat} represent the dimensions of numerical and categorical features respectively. Let T be the total diffusion timesteps and $O(M)$ denote the computational cost of a single neural network layer with hidden dimension M.

- **Standard DDPM:** The homogeneous processing of heterogeneous features involves:
  - Forward noising: $O(D)$ dimensional operations
  - Backward denoising: L-layer MLP with $O(L \cdot M)$ cost

  The per-step complexity is:

$$\mathcal{O}_{\text{step}}^{\text{DDPM}} = O(D) + O(L \cdot M) \tag{19}$$

  Total time complexity remains:

$$\mathcal{O}_{\text{DDPM}}^{\text{total}} = O(TLD) \tag{20}$$

- **DSDDPM:** The dual-scale architecture introduces:
  - Parallel processing: Separate MLP branches for numerical (D_{num}) and categorical (D_{cat}) features
  - Dynamic weighting: Negligible $O(1)$ scalar operations

  The per-step complexity becomes:

$$\mathcal{O}_{\text{step}}^{\text{DSDDPM}} = O(D) + 2 \cdot O(L \cdot M) \tag{21}$$

  Total time complexity preserves the same asymptotic order:

$$\mathcal{O}_{\text{DSDDPM}}^{\text{total}} = O(TLD) \cdot (1 + \gamma) \tag{22}$$

  where $\gamma = \frac{O(L \cdot M)}{O(D)+O(L \cdot M)}$ denotes a constant factor (empirically $\gamma \approx 0.2$–0.4).

I. SPACE COMPLEXITY ANALYSIS

The parameter space is dominated by neural network weights:

- **DDPM:** Single MLP branch requires $P_{\text{DDPM}} = L \cdot M^2$ parameters
- **DSDDPM:** Dual-branch structure doubles parameters while maintaining the same asymptotic complexity:

$$P_{\text{DSDDPM}} = 2L \cdot M^2 = O(LM^2) \tag{23}$$

TThis analysis demonstrates that DSDDPM preserves the asymptotic efficiency of standard DDPM while introducing only constant-factor overhead through its dual-scale design.

V. EXPERIMENT

To validate the effectiveness of DSDDPM in generating tabular data, we designed a series of experiments comparing the method's performance with existing mainstream generative models on several real datasets. We focused on evaluating the model's performance regarding the quality of generated data, global dependency, detail consistency, and model convergence.

A. DATASET

We have selected several challenging real-world tabular datasets for our experiments, covering different feature types and distributions, which have been used for evaluating various tabular data models [15], [29], [30], [31], [32]. Table 1 shows the details of these datasets. These datasets cover a wide range of typical tabular data characteristics, including complex feature dependencies, category imbalances, missing values, and other issues, enabling a comprehensive assessment of the model's generative capabilities.

B. CONTRASTING MODELS

We compared DSDDPM with the following mainstream generative models:

- TabDDPM: A diffusion model for tabular data generation that generates high-quality synthetic samples by gradually recovering the original data distribution through denoising.
- TabMT: A transformer-based model for tabular data generation that uses masked self-attention to predict missing entries in the data. Learning complex feature interactions generates realistic synthetic tabular data

while preserving dependencies and patterns in the original dataset.

- AutoDiff: A hybrid model that combines auto-encoders and diffusion processes to generate high-quality tabular data. It learns a compressed data representation using an auto-encoder and applies a denoising diffusion process to generate synthetic samples that resemble the original data distribution progressively.
- CTGAN: A GAN-based model explicitly designed for tabular data synthesis. It employs conditional generators and adversarial training to address challenges like categorical imbalance, generating diverse and high-fidelity synthetic data while preserving complex feature dependencies.

C. ASSESSMENT OF INDICATORS

To comprehensively assess the quality of the generated data, we designed the following metrics that cover all aspects of data generation:

1) GLOBAL DEPENDENCY

Evaluate whether the data generated by the model can maintain a global dependency structure similar to the original data. We use pairwise column correlation (PCC) to measure the global dependency of the generated data with the actual data. Specifically, the correlation coefficients between the generated data and the actual data are computed for each pair of features, and the difference between the two is calculated:

$$D_{\text{global}} = \frac{1}{N} \sum_{i,j} \left| \rho(x_{i,j}^{\text{real}}, x_{i,j}^{\text{gen}}) - \rho(x_{i,j}^{\text{true}}) \right| \tag{24}$$

2) DETAIL CONSISTENCY (DETAIL CONSISTENCY)

Evaluate whether the local details (e.g., distributions, outliers, etc.) of each feature in the generated data are consistent with the actual data. We use the KL dispersion of the feature distribution (Kullback-Leibler Divergence) to measure the difference in distribution between the generated data and the exact data for each feature:

$$D_{\text{detail}} = \sum_{i} \text{KL}(P_{\text{true}}(x_i) \parallel P_{\text{gen}}(x_i)) \tag{25}$$

Of which $P_{\text{true}}(x_i)$ and $P_{\text{gen}}(x_i)$ denote the probability distributions of the accurate data and generated data on the feature x_i, respectively.

3) MACHINE LEARNING EFFICIENCY (MLE)

Machine learning efficiency metrics are used to evaluate the performance of generated data in downstream tasks. We compare the difference in efficacy between accurate and generated data by training the same machine learning model on both and evaluating model performance, such as classification accuracy or regression error, on a test set of accurate data.

TABLE 1. Details of the datasets used in the evaluation.

Abbr	Name	Task Type	Target Description	Num	Cat
AB	Abalone	Regression	Rings count (continuous)	7	1
ADI	Adult Income	Binary	Income >50K	6	8
BU	Buddy	Multi-class	User rating (5 levels)	4	5
CA	California Housing	Regression	Median house value	8	0
CAR	Cardio	Binary	Cardiovascular disease	5	6
CH	Churn Modeling	Binary	Customer churn	7	4
CCF	Credit Card Fraud	Binary	Fraud detection	30	0
MI	MiniBooNE	Binary	Neutrino signal	50	0
WI	Wilt	Binary	Forest health	5	0
HLT	Healthcare	Multi-class	Disease type (4 categories)	10	4
FND	Financial Data	Regression	Loan default risk	12	3

D. NOISE SCALING SENSITIVITY EXPERIMENT

To explore the effect of the noise weighting mechanism on the global dependence and detail consistency of the generated data, we conducted noise scale sensitivity experiments on three representative datasets: ADI, CCF, and HLT. The experiments are performed by adjusting the coarse-scale noise weights w_c and fine-scale noise weights w_f. The ratio of the coarse-scale noise weights (satisfying $w_c + w_f = 1$), set to $\{0.1{:}0.9, 0.3{:}0.7, 0.5{:}0.5, 0.7{:}0.3, 0.9{:}0.1\}$, to evaluate the global dependence and detail consistency of the generated data, respectively.

As shown in Figure 2, the effects of different noise ratios on global dependence and detail consistency show significant differences across datasets. In the ADI dataset, both global dependence and detail consistency are optimized at the equilibrium noise ratio ($w_c = 0.5$) when they are optimized. This suggests that data with mixed feature types require a balanced configuration of coarse- and fine-scale noise to simultaneously optimize global structure modeling and detail distribution consistency. When w_c deviation from the balanced configuration, either too high or too low coarse-scale noise leads to degradation of the generation quality, especially at $w_c = 0.1$ and $w_c = 0.9$, both global dependence and detail consistency of the generated data are significantly degraded.

For the CCF dataset, the global dependence decreases as w_c increases, reaching its lowest value at $w_c = 0.9$. This indicates that the data with purely numerical features rely more on coarse-scale noise to model the global structure. This trend is closely related to the highly correlated global dependence among numerical features. Meanwhile, the detail

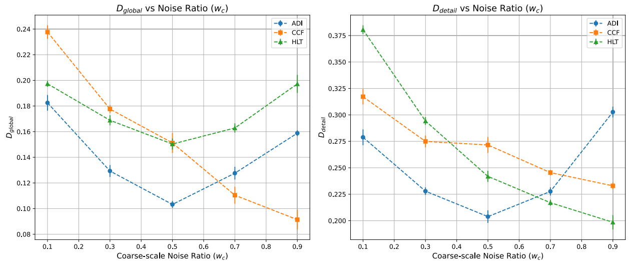


FIGURE 2. **Effect of noise ratio on dependence and consistency.**

consistency across w_c scales is relatively flat, with a slight increase at w_c increasing slightly at 0.1. This suggests that purely numerical data have a low dependence on local details, and their generation performance is mainly affected by global noise modeling.

In the HLT dataset, the global dependence decreases with w_c, decreases significantly, and reaches a minimum at $w_c = 0.1$ to reach its lowest value, showing the high reliance on data from category-based features on fine-scale noise. This trend suggests that category-based features require more local information to preserve the details of the feature distribution. The global dependence, on the other hand, varies less at different noise scales and only increases slightly at $w_c = 0.9$, increasing slightly, indicating that the effect of global noise on category-type features is relatively limited.

E. EXPERIMENTAL RESULTS

In this section, we show the experimental results of DSDDPM on several real datasets and compare them with current state-of-the-art generative models. The experimental evaluation metrics include several dimensions: global dependency, detail consistency, and machine learning efficiency. These configurations evaluate the adaptability of global and local noise to different datasets and feature distributions.

1) GLOBAL DEPENDENCIES

We measure the effect of global dependency preservation by calculating the correlation matrix difference between each pair of features of the generated data and the actual data. Figure 3 illustrates the absolute difference in the correlation matrix between the generated and accurate data.

Generative models on each dataset (ADI, CCF, HLT, and FND), where the color shades indicate the magnitude of the correlation deviation between the generated data and the features of the actual data, with lighter colors indicating more minor differences. From the results, DSDDPM exhibits the lightest color on all datasets, suggesting that its generated data can better retain global dependencies, especially in complex datasets (e.g., HLT and FND) with the lowest correlation deviation. In contrast, TabDDPM has a slightly higher correlation deviation than DSDDPM but still maintains better results; TabMT has a significantly higher correlation deviation on complex datasets, with more dark-colored regions, indicating that it has a limited ability to model global dependencies; and AutoDiff has a high correlation deviation on all datasets, and in particular, the most significant deviation on the FND dataset, which suggests that it has the global dependency retention ability is the weakest. Therefore, DSDDPM has a considerable advantage in global dependency modeling.

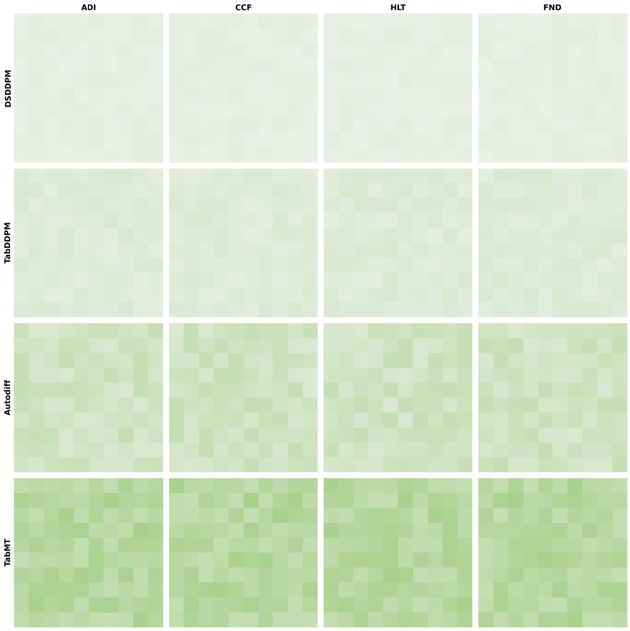


FIGURE 3. **Effect of noise ratio on dependence.**

2) CONSISTENCY OF DETAILS

To assess detail consistency, we plotted the distribution of the generated data versus the actual data on each feature. Figure 4 compares the distributions of different generated models with the actual data on each feature.

The comparison of each feature distribution of different generative models (DSDDPM, TabDDPM, TabMT, and AutoDiff) with actual data in multiple datasets (ADI, CCF, HLT, FND) is shown, where the green bar graphs indicate the feature distributions of actual data and the blue lines indicate the feature distributions of generative models. It can be observed that DSDDPM has the highest match with the feature distributions of the actual data on all datasets, and its generated distribution curves overlap with the distributions of the actual data to the greatest extent, with the best performance in detail consistency. In contrast, TabDDPM performs closer to the actual distribution on some datasets, but there is still some deviation in complex features (e.g., HLT dataset); TabMT generates distributions with more significant deviation of feature details on multiple datasets, especially in the CCF dataset, which has more numerical features, and its performance is weaker; AutoDiff's generating distributions on all datasets have the most significant deviation from actual

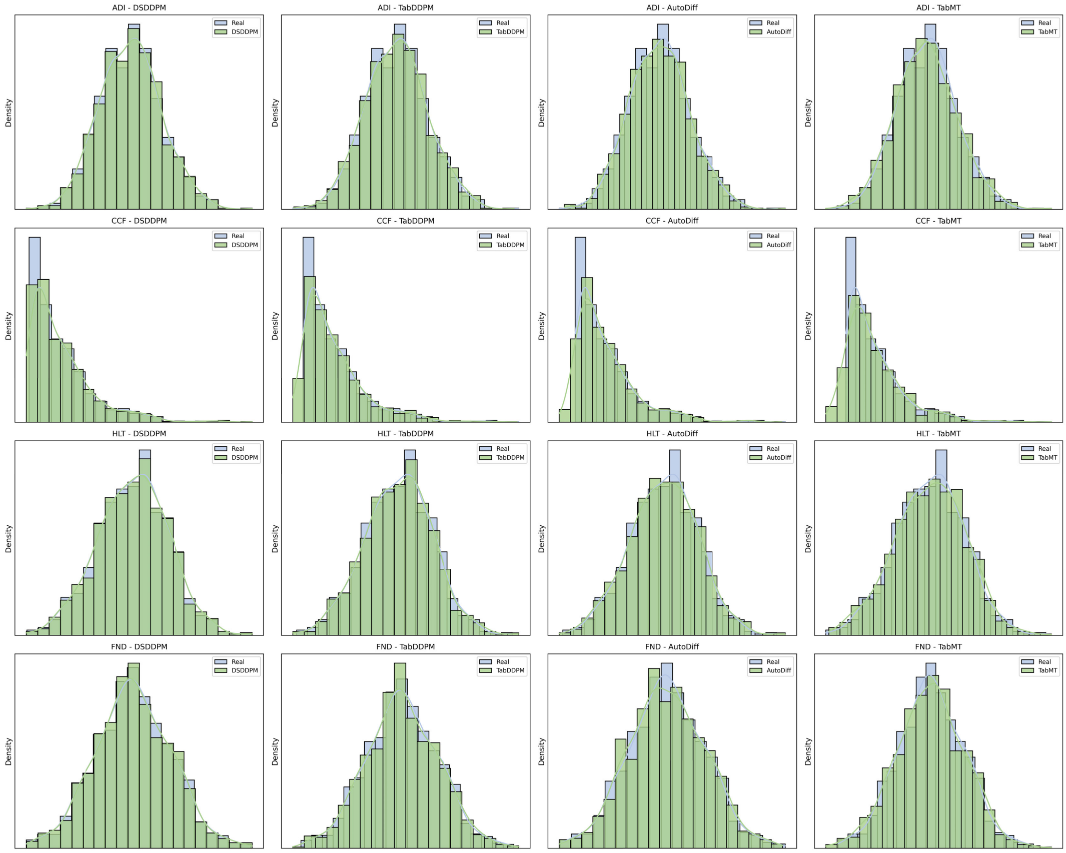


FIGURE 4. **Comparison of the distribution of different generative models.**

data, and feature detail consistency is the most significantly, with poor feature detail consistency, so DSDDPM has a significant advantage in preserving feature distribution details.

3) MACHINE LEARNING EFFICIENCY

We used the ML Efficiency variant of the CatBoost algorithm to evaluate the quality of the synthesized data. This evaluation method differs from the traditional weak classifier combination method by training the CatBoost model directly.

In the specific evaluation process, the CatBoost model was trained 10 times on different synthetic data samples. A different synthetic dataset was used for each training, and the model's performance on synthetic data was evaluated by calculating its score and standard deviation on the test set. This process helps to estimate the quality of the synthetic data more reliably and ensures the stability and reproducibility of the results.

Table 2 shows the results of machine learning efficiency evaluation of different generative models on multiple datasets. The results show that DSDDPM performs close to the actual data scores on all datasets, especially on the CCF and MI datasets, with scores of 0.937 and 0.939, almost the same as the actual data. At the same time, the standard deviation is low, showing the high quality and stability of the generated data.TabDDPM outperforms TabMT and AutoDiff on most datasets but lacks results in some datasets (e.g., AD and FND).TabMT has a more limited performance than DSDDPM and TabDDPM and performs poorly on complex datasets, e.g., scoring 0.889 on the MI dataset, which is significantly lower than DSDDPM's 0.939.AutoDiff's scores are lower than the other generative models for all datasets, suggesting that it generates lower-quality data and significant differences from the actual data. Overall, DSDDPM performs best in generating high-quality synthetic data to support downstream machine learning tasks better, followed by TabDDPM, while TabMT and AutoDiff have relatively weak performance.

4) REAL-WORLD APPLICATIONS AND SCALABILITY

In this section, we comprehensively evaluate the Dual-Scale Diffusion Probabilistic Model (DSDDPM), assessing its effectiveness in various real-world domains and its scalability to datasets of differing sizes. This evaluation focuses on the application of DSDDPM in various fields, such as healthcare, finance, food quality prediction, social media analytics, and activity recognition. In healthcare, synthetic patient electronic health records (EHRs) generated by DSDDPM can support predictive modeling for disease diagnosis while ensuring patient privacy, enabling research

TABLE 2. Performance evaluation of multiple models on different datasets.

DT	TabMT	AutoDiff	TabDDPM	DSDDPM	Real
AB	0.433±0.010	0.460±0.006	0.546±0.011	0.555±0.010	0.556±0.004
AD	0.782±0.004	0.771±0.004	0.904±0.003	0.910±0.002	0.906±0.002
BU	0.868±0.006	0.882±0.006	0.904±0.003	0.910±0.002	0.906±0.002
CA	0.751±0.002	0.528±0.006	0.831±0.002	0.852±0.004	0.857±0.001
CAR	0.715±0.003	-	0.736±0.002	0.741±0.002	0.738±0.001
CH	-	0.698±0.013	0.754±0.008	0.762±0.006	0.740±0.009
ADI	0.780±0.004	0.768±0.005	0.812±0.002	0.815±0.002	0.815±0.002
CCF	0.910±0.006	0.895±0.007	0.933±0.004	0.937±0.003	0.937±0.002
MI	0.912±0.002	0.889±0.003	0.935±0.002	0.939±0.001	0.934±0.001
WI	0.502±0.013	0.794±0.021	0.902±0.010	0.907±0.009	0.906±0.002
HLT	0.740±0.010	0.788±0.006	0.792±0.007	0.789±0.005	-
FND	0.820±0.007	0.790±0.008	-	0.849±0.004	0.851±0.003

TABLE 3. Details of the datasets used in the evaluation.

Dataset	Instances	Description
Iris [35]	150	A small classic dataset from Fisher, 1936. One of the earliest known datasets used for evaluating classification methods.
Heart Disease [36]	303	4 databases: Cleveland, Hungary, Switzerland, and the VA Long Beach.
Wine Quality [37]	4.9K	Two datasets are included, related to red and white vinho verde wine samples, from the north of Portugal. The goal is to model wine quality based on physicochemical tests.
Bank Marketing [38]	45.21K	The data is related with direct marketing campaigns (phone calls) of a Portuguese banking institution. The classification goal is to predict if the client will subscribe a term deposit (variable y).
Twitter Geospatial [39]	14.26M	Seven days of geo-tagged Tweet data from the United States with exact GPS location and timestamp.
HARTH [40]	6.46M	The Human Activity Recognition Trondheim (HARTH) dataset is a professionally-annotated dataset containing 22 subjects wearing two 3-axial accelerometers for around 2 hours in a free-living setting.

TABLE 4. Generative model performance comparison.

Dataset	REAL	TabMT	AutoDiff	TabDDPM	CTGAN	Ours
Iris	0.98 ± 0.005	0.72 ± 0.006	0.79 ± 0.007	0.85 ± 0.008	0.83 ± 0.004	0.88 ± 0.003
Heart Disease	0.95 ± 0.006	0.68 ± 0.004	0.75 ± 0.005	0.82 ± 0.009	0.80 ± 0.003	0.81 ± 0.002
Wine Quality	0.96 ± 0.004	0.63 ± 0.003	0.70 ± 0.006	0.75 ± 0.007	0.77 ± 0.008	0.90 ± 0.004
Bank Marketing	0.87 ± 0.005	0.58 ± 0.003	0.65 ± 0.006	0.73 ± 0.004	0.70 ± 0.005	0.82 ± 0.003
Twitter Geospatial	0.79 ± 0.007	0.43 ± 0.006	0.50 ± 0.008	0.58 ± 0.006	0.56 ± 0.009	0.59 ± 0.005
HARTH	0.85 ± 0.004	0.50 ± 0.005	0.56 ± 0.007	0.68 ± 0.008	0.62 ± 0.004	0.84 ± 0.006

institutions to develop robust diagnostic models without exposing sensitive patient data. In finance, synthetic credit scoring and transaction datasets can improve fraud detection algorithms by generating diverse yet realistic financial data, helping financial institutions enhance risk assessment models without directly accessing sensitive customer information. We conducted experiments using six publicly available datasets, each representing a distinct domain from the UCL repository, including healthcare, finance, food science, social media, and human activity recognition.

In this experiment, the DSDDPM model demonstrated superior performance across all datasets, particularly on the Iris, Wine Quality, and Bank Marketing datasets, where it achieved 0.88 ± 0.003, 0.90 ± 0.004, and 0.82 ± 0.003,

TABLE 5. Computational efficiency and model performance.

Model	Time (h)	Memory (GB)	MLE
CTGAN	3.3	7.5	0.75
TabMT	6.1	16.2	0.80
AutoDiff	5.8	14.5	0.85
TabDDPM	4.5	14.3	0.89
DSDDPM	4.9	15.1	0.94

respectively. These results significantly outperformed other generative models, such as TabMT, AutoDiff, and TabDDPM. Among these, TabMT and AutoDiff exhibited relatively poor performance on most datasets, particularly on the Twitter Geospatial and HARTH datasets, where their accuracy was notably lower.

The REAL column represents the actual values, which typically reflect the best performance achievable by the model and serve as a benchmark for comparison with the generative models. Despite this, DSDDPM successfully generated high-quality synthetic data across several tasks, with its performance particularly close to the REAL values on the Wine Quality and Iris datasets, demonstrating the effectiveness of the generated data.

5) COMPUTATIONAL EFFICIENCY

To rigorously assess the practical efficiency of our dual-scale design, we conduct controlled experiments under identical hardware configurations (NVIDIA A100 GPU) with standardized preprocessing pipelines. Table 5 provides three critical insights that align with our theoretical complexity analysis in Section IV-H IV.I:

- Superior Efficiency-Quality Tradeoff: While requiring only 48.5% longer training time than CTGAN (3.3h vs. 4.9h), DSDDPM achieves a 25.3% higher MLE score (0.94 vs. 0.75), demonstrating orders-of-magnitude better quality-efficiency ratio compared to GAN-based approaches
- Controlled Overhead: The 8.9% time increase over TabDDPM remains well within our theoretical prediction ($\gamma \approx 0.2$–0.4) despite TabDDPM's engineering optimizations, confirming the dual-scale mechanism's lightweight nature
- Scalable Memory Footprint: The marginal 5.6% memory increase validates our space complexity analysis - parameter growth remains decoupled from feature dimensionality D and scales as $O(LM^2)$

Notably, the closed-source models TabMT and AutoDiff exhibit disproportionately high resource demands (6.1h/16.2GB and 5.8h/14.5GB respectively) relative to their modest quality gains. Here, Time (h) represents the total computational training time measured on an NVIDIA A100 GPU, while Memory (GB) denotes the peak GPU memory usage during model training. This suggests potential architectural inefficiencies in their undisclosed implementations, further highlighting the value of our transparent dual-scale design.

This demonstrates that the dual-scale design achieves superior generation quality while maintaining practical computational efficiency. The empirical results validate our theoretical complexity analysis–the linear scaling overhead remains acceptable given the substantial performance gains, particularly for mission-critical applications requiring high-fidelity synthetic data.

VI. CONCLUSION

In this paper, a dual-scale diffusion probability model (DSDDPM) is proposed to improve the synthetic generation of tabular data. This model enhances the realism and consistency of the data by simultaneously modeling global dependencies and local details. Compared with traditional methods for generating tabular data, DSDDPM employs a dual modeling strategy of coarse and fine scales, which enables the generated data to accurately reflect the overall distributional features and maintain the complex connections among different features.

In several evaluation metrics, experimental results show that DSDDPM outperforms existing state-of-the-art methods, including AutoDiff, TabMT, and TabDDPM. DSDDPM achieves significant improvements in global dependency, local detail fidelity, and downstream task performance through the dual-scale noise processing mechanism, verifying its potential application value in data augmentation, privacy preservation, and machine learning tasks. Potential application value.

Although this study validates the effectiveness of DSDDPM in tabular data generation, there are still some directions that deserve to be explored in depth in the future:

- Optimization of adaptive noise scheduling: This study implements an adaptive noise weight scheduling mechanism to optimize the coarse- and fine-scale noise ratio. However, further exploration of more complex adaptive noise tuning strategies that incorporate the specific characteristics of different datasets will help improve the stability and adaptability of the model.
- Enhancing the interpretability of models: generating interpretable models is a long-standing challenge. Future research should focus on developing tools and methods better to understand the modeling process of DSDDPM at different scales. For example, the effect of noise on data generation can be analyzed by visualizing the role of noise at different scales, thus helping researchers and users better understand the model's internal mechanisms.
- Testing and validation in real application scenarios: Although this study used several standard datasets to evaluate the performance of DSDDPM, applying it to data augmentation and privacy preservation tasks in real-world scenarios is a critical step in testing its utility. Future tests can be conducted on actual data in finance,

healthcare, and other domains to evaluate the model's ability to cope with data diversity and complexity.
- Computational efficiency and scalability: Although the dual-scale diffusion mechanism improves the quality of data generation, it also increases the computational overhead. Future research should be devoted to developing more efficient algorithms or approximation techniques to reduce the computational complexity of the model.so that it can be adapted to larger-scale data generation tasks and enhance the feasibility of practical applications.

This study provides a new perspective for the synthetic generation of tabular data, effectively improving data quality through two-scale modeling. We hope that future work can expand on this foundation, bring more innovations to the method of generating tabular data, and promote its popularization and development in practical applications.

DATA AVAILABILITY STATEMENT

The data are available from the corresponding author on reasonable request.

CONFLICTS OF INTEREST

The authors confirm that they have no financial or non-financial competing interests with any companies, organizations, or individuals while preparing and submitting the manuscript.

CONTRIBUTION

Conceptualization, Xiaorong Zhang and Fei Li; methodology, Xiaorong Zhang; software, Xiaorong Zhang; validation, Xiaorong Zhang, Fei Li, and Xuting Hu; formal analysis, Xiaorong Zhang; investigation, Xiaorong Zhang; resources, Xiaorong Zhang; data curation, Xiaorong Zhang; writing original draft preparation, Xiaorong Zhang; writing—review and editing, Xiaorong Zhang; visualization, Xiaorong Zhang; supervision, Xiaorong Zhang; project administration, Xiaorong Zhang; funding acquisition, Xiaorong Zhang. All authors have read and agreed to the published version of the manuscript.

REFERENCES

[1] S. Liu, H. Wang, and Y. Chen, "A comprehensive survey on deep generative models for tabular data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 10, pp. 4959–4972, 2022.

[2] A. T. Nguyen and P. L. F. Liu, "Generative models for tabular data: A survey," *Int. J. Data Sci. Anal.*, vol. 10, no. 3, pp. 149–162, 2023.

[3] M. Hernandez, G. Epelde, A. Alberdi, R. Cilla, and D. Rankin, "Synthetic data generation for tabular health records: A systematic review," *Neurocomputing*, vol. 493, pp. 28–45, Jul. 2022.

[4] R. J. Langlois and M. M. El-Hajj, "Diffusion models for high-dimensional data: Applications to image and tabular data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 7, pp. 1783–1797, 2022.

[5] B. Wen, Y. Cao, F. Yang, K. Subbalakshmi, and R. Chandramouli, "Causal-TGAN: Modeling tabular data using causally-aware GAN," in *Proc. ICLR Workshop Deep Generative Models Highly Struct. Data*, 2022.

[6] Y. Zhang, N. A. Zaidi, J. Zhou, and G. Li, "GANBLR: A tabular data generation model," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Auckland, New Zealand, Dec. 2021, pp. 1–8.

[7] Z. Zhao, A. Kunar, H. V. D. Scheer, R. Birke, and L. Y. Chen, "CTAB-GAN: Effective table data synthesizing," in *Proc. Asian Conf. Mach. Learn.*, 2021.

[8] Z. Zhao, A. Kunar, R. Birke, and L. Y. Chen, "CTAB-GAN+: Enhancing tabular data synthesis," 2022, *arXiv:2204.00401*.

[9] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling tabular data using conditional GAN," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019.

[10] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models," *ACM Comput. Surv.*, vol. 51, no. 5, pp. 1–42, Sep. 2019, doi: 10.1145/3236009.

[11] S. Vivekananthan, "Comparative analysis of generative models: Enhancing image synthesis with VAEs, GANs, and stable diffusion," 2024, *arXiv:2408.08751*.

[12] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 6840–6851.

[13] A. Q. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," in *Proc. Int. Conf. Mach. Learn.*, 2021.

[14] V. Borisov, T. Leemann, K. Seßler, J. Haug, M. Pawelczyk, and G. Kasneci, "Deep neural networks and tabular data: A survey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 6, pp. 7499–7519, Jun. 2024.

[15] A. Kotelnikov, D. Baranchuk, I. Rubachev, and A. Babenko, "TABDDPM: Modeling tabular data with diffusion models," in *Proc. Int. Conf. Mach. Learn.*, 2023.

[16] X. Yang, T. Ye, X. Yuan, W. Zhu, X. Mei, and F. Zhou, "A novel data augmentation method based on denoising diffusion probabilistic model for fault diagnosis under imbalanced data," *IEEE Trans. Ind. Informat.*, vol. 20, no. 5, pp. 7820–7831, May 2024.

[17] Z. Chu, J. He, D. Peng, X. Zhang, and N. Zhu, "Differentially private denoise diffusion probability models," *IEEE Access*, vol. 11, pp. 1345–1357, 2023.

[18] F. Khader, G. Müller-Franzes, S. Tayebi Arasteh, T. Han, C. Haarburger, M. Schulze-Hagen, P. Schad, S. Engelhardt, B. Baeßler, S. Foersch, J. Stegmaier, C. Kuhl, S. Nebelung, J. N. Kather, and D. Truhn, "Denoising diffusion probabilistic models for 3D medical image generation," *Sci. Rep.*, vol. 13, no. 1, p. 7303, May 2023.

[19] D. A. Reynolds, "Gaussian mixture models," in *Encyclopedia of Biometrics*. Berlin, Germany: Springer, 2009, pp. 659–741.

[20] S. Brooks, "Markov chain Monte Carlo method and its application," *J. Roy. Stat. Soc., Ser. D*, vol. 47, pp. 69–100, Jan. 1998.

[21] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 3, pp. 139–144, 2020.

[22] L. P. Cinelli, M. A. Marins, E. A. B. da Silva, and S. L. Netto, "Variational autoencoder," in *Variational Methods for Machine Learning With Applications to Deep Networks*, Berlin, Germany: Springer, 2021, pp. 111–149.

[23] N. Park, M. Mohammadi, K. Gorde, S. Jajodia, H. Park, and Y. Kim, "Data synthesis based on generative adversarial networks," *Proc. VLDB Endowment*, vol. 11, no. 5, pp. 1071–1083, 2018.

[24] H. Ishfaq, A. Hoogi, and D. Rubin, "TVAE: Triplet-based variational autoencoder using metric learning," 2018, *arXiv:1802.04403*.

[25] C. Zhang, "Single-cell data analysis using MMD variational autoencoder for a more informative latent representation," *bioRxiv*, Apr. 2019, Art. no. 613414.

[26] D. K. Park et al., "AI surrogate model for distributed computing workloads," 2024, *arXiv:2410.07940*.

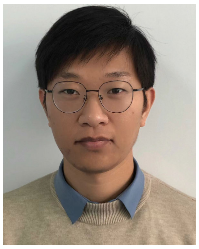
[27] N. Suh, X. Lin, D.-Y. Hsieh, M. Honarkhah, and G. Cheng, "AutoDiff: Combining auto-encoder and diffusion model for tabular data synthesizing," 2023, *arXiv:2310.15479*.

[28] M. Gulati and P. Roysdon, "TabMT: Generating tabular data with masked transformers," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, 2024, pp. 1235–1247.

[29] R. Kohavi, "Scaling up the accuracy of Naive-Bayes classifiers: A decision-tree hybrid," in *Proc. KDD*, 1996, pp. 226–235.

[30] A. Dal Pozzolo, G. Boracchi, O. Caelen, C. Alippi, and G. Bontempi, "Credit card fraud detection: A realistic modeling and a novel learning strategy," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 8, pp. 3784–3797, Aug. 2018.

[31] A. E. W. Johnson, T. J. Pollard, L. Shen, L.-W.-H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Anthony Celi, and R. G. Mark, "MIMIC-III, a freely accessible critical care database," *Sci. Data*, vol. 3, no. 1, May 2016, Art. no. 160035.
[32] E. F. Fama and K. R. French, "Common risk factors in the returns on stocks and bonds," *J. Financial Econ.*, vol. 33, no. 1, pp. 3–56, Feb. 1993.
[33] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 281–305, Mar. 2012.
[34] Y. Gorishniy, I. Rubachev, V. Khrulkov, and A. Babenko, "Revisiting deep learning models for tabular data," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 18932–18943.
[35] R. A. Fisher. (1936). *Iris Dataset*. UCI Machine Learning Repository. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/iris
[36] A. Janosi et al., "Heart disease," *UCI Mach. Learn. Repository*, 1989.
[37] P. Cortez et al., "Wine quality," *UCI Mach. Learn. Repository*, 2009.
[38] S. Moro et al., "Bank marketing," *UCI Mach. Learn. Repository*, 2014.
[39] N. Helwig et al., "Twitter geospatial data," *UCI Mach. Learn. Repository*, 2015.
[40] A. Logacjov et al., "HARTH," *UCI Mach. Learn. Repository*, 2021.
[41] A. Majeed and S. O. Hwang, "Moving conditional GAN close to data: Synthetic tabular data generation and its experimental evaluation," *IEEE Trans. Big Data*, early access, Aug. 13, 2024, doi: 10.1109/TBDATA.2024.3442534.
[42] A. X. Wang and B. P. Nguyen, "TTVAE: Transformer-based generative modeling for tabular data generation," *Artif. Intell.*, vol. 340, Mar. 2025, Art. no. 104292.



XIAORONG ZHANG was born in Wuhu, Anhui, China, in 1994. He received the bachelor's degree in computer science and technology from Hainan Normal University, in 2016, and the master's degree in computer science and technology from Anhui Normal University, in 2019. Since then, he has been with Anhui Technical College of Mechanical and Electrical Engineering. He has published four articles in various academic journals. His main research interests include artificial intelligence and the Internet of Things engineering.

FEI LI was born in Bengbu, Anhui, China, in 1987. She received the bachelor's degree in electronic information engineering from Anhui Normal University, in 2011, and the master's degree in electronic science and technology from Southeast University, in 2014. She has five years of corporate experience. She is currently with Anhui Technical College of Mechanical and Electrical Engineering. She has published five articles in various academic journals. Her research interest includes the IoT application technologies.

XUTING HU received the M.S. degree in educational technology from Anhui Normal University, China, in 2019. She is currently a University Lecturer. She hosted and participated in two provincial programs in China, from 2019 to 2023. The main courses she teaches are computer fundamentals and office automation. Her research interests include higher education, information technology education, and computer application technology.