A Comprehensive Guide to Synthetic Tabular Data Generation

Distilled from Recent Research

May 23, 2025

Overview

This document provides a comprehensive summary of synthetic tabular data generation, drawing insights from recent research. It covers the definition, core challenges, various generation methodologies, essential post-processing steps, evaluation criteria, practical applications, and future research directions in this rapidly evolving field.

Contents

1 Introduction to Synthetic Tabular Data Generation

Synthetic tabular data generation is the process of creating artificial datasets that mimic the statistical properties and structure of real-world tabular data. This field has gained prominence due to the increasing need to overcome limitations associated with real data, such as:

- **Data Scarcity:** Real-world datasets, especially in specialized domains, can be limited in size.
- **Privacy Concerns:** Strict regulations like GDPR and CCPA restrict the use and sharing of sensitive information.
- **Class Imbalance:** Datasets often have an imbalanced representation of different classes, which can bias machine learning models.
- **Missing Values:** Incomplete data is a common issue that complicates modeling efforts.

Generative models, including traditional statistical methods, Variational Autoencoders (VAEs), Generative Adversarial Networks (GANs), Diffusion Models, and Large Language Models (LLMs), are employed to learn the underlying distribution of real data and produce high-fidelity, privacy-preserving synthetic samples. The goal is to generate data that is not only statistically similar but also useful for downstream tasks like machine learning model training, data augmentation, and software testing.

Key Objective: To learn the statistical distribution of an existing tabular dataset (D_e) and produce a synthetic dataset (D_s) that closely approximates it, often represented as $D_s \leftarrow p_\theta(D_e)$.

2 Core Challenges in Tabular Data Synthesis

Generating high-quality synthetic tabular data is inherently more challenging than synthesizing other data types like images or text due to the unique characteristics of tabular data. These challenges can be broadly categorized:

2.1 Data Quality Issues of Original Data

- **Limited Data Size:** Small datasets, common in specialized fields like healthcare or finance, make it difficult for generative models to learn robust distributions.
- **Imbalanced Class Distribution:** Severe class imbalances can lead to models that poorly represent minority classes.
- **Missing Values:** The presence of missing entries necessitates models that can handle incomplete information effectively during the learning process.

2.2 Inherent Characteristics of Tabular Data

- **Heterogeneity:** Tabular data consists of mixed data types (numerical, categorical, ordinal, boolean, etc.), each with distinct statistical properties, making joint distribution modeling complex. Direct adaptation of models designed for homogeneous data (like images) is often insufficient.
- **Complex and Sparse Feature Dependencies:** Capturing the intricate correlations between various features (e.g., numerical-categorical dependencies) is crucial but difficult, especially since these relationships are often sparse.
- **Mixed-Type Single Features:** A single column might contain values of different data types (e.g., numerical and categorical labels mixed together), further complicating encoding and modeling.

2.3 Complex and Diverse Data Distributions

- **Non-Gaussian Distribution of Numerical Features:** Numerical features often do not follow Gaussian distributions, exhibiting skewness, heavy tails, or multiple modes.
- **Imbalanced Categories in Categorical Features:** Categorical columns frequently have dominant categories, which can lead to models overfitting these majority classes.
- **Varied Marginal Effects of Columns:** The meaning of a value can change based on the feature context (e.g., '10' could be an age or a rating), and preserving these semantic distinctions is vital.
- **Domain-Specific Constraints:** Real-world tabular data often has implicit or explicit constraints (e.g., age cannot be negative, certain medical values must be within realistic ranges) that models need to respect.

3 Methods for Generating Synthetic Tabular Data

Synthetic tabular data generation methods are broadly categorized into traditional methods, diffusion models, and LLM-based methods.

3.1 Traditional Generation Methods

These include early statistical and machine learning approaches, as well as VAEs and GANs.

3.1.1 Machine Learning-based Methods

Before deep generative models, techniques like Copulas, Gaussian Mixture Models (GMMs), and Bayesian Networks were used. Resampling methods like SMOTE (Synthetic Minority Over-sampling Technique) and its variants were also common for addressing class imbalance. However, these often struggle with high-dimensional data and complex distributions.

3.1.2 VAE-based Methods

Variational Autoencoders (VAEs) learn a latent representation of data by optimizing an Evidence Lower Bound (ELBO). Models like TVAE adapt VAEs for tabular data, while GOGGLE uses Graph Neural Networks (GNNs) within a VAE framework to model feature dependencies. VAEs can sometimes produce blurry or less sharp outputs compared to other deep generative models.

3.1.3 GAN-based Methods

Generative Adversarial Networks (GANs) use a generator and a discriminator in a min-max game to produce realistic samples.

- **Traditional GANs for Tabular Data:** Early adaptations like TGAN used LSTMs for sequential feature generation. MedGAN focused on EHR data, and TableGAN incorporated CNNs. These models made strides but sometimes faced challenges with mixed data types and mode collapse.
- **Conditional GANs (CGANs):** These extend GANs by allowing generation conditioned on specific attributes or classes (e.g., CTGAN, CTAB-GAN). This is useful for imbalanced data and targeted generation.
- **Differentially Private GANs (DP-GANs):** To address privacy, methods like DPGAN and PATE-GAN incorporate differential privacy mechanisms, often by adding noise to gradients or using ensembles of discriminators. CTAB-GAN+ uses Rényi Differential Privacy (RDP) for tighter privacy bounds.

3.2 Diffusion Model Methods

Diffusion models have emerged as powerful alternatives, addressing issues like training instability and mode collapse seen in GANs. They learn to reverse a gradual noising process.

3.2.1 DDPM-based Methods (Denoising Diffusion Probabilistic Models)

These models define a forward diffusion (noising) process and a reverse denoising process.

- **Generic Tabular DDPMs:** TabDDPM was a pioneering work extending DDPMs to tabular data, often using separate diffusion processes for numerical (Gaussian diffusion) and categorical features (multinomial diffusion). CoDi uses two co-evolving diffusion models that condition on each other. AutoDiff maps heterogeneous features to a continuous latent space using an autoencoder before applying diffusion. TABDIFF proposes a joint continuous-time diffusion process with feature-wise learnable noise schedules.
- **Domain-Specific DDPMs:** These models are tailored for specific domains. For healthcare, TabDDPM-EHR, DPM-EHR, and FLEXGEN-EHR (handles missing modalities) generate Electronic Health Records (EHRs). In finance, EntTabDiff generates data conditioned on entities, and FinDiff and Imb-FinDiff (addresses class imbalance) focus on financial data.

3.2.2 Score-based Generative Models (SGMs)

These models learn the score function (gradient of the log-density) of the data distribution and use it to define and reverse a continuous-time diffusion trajectory via Stochastic Differential Equations (SDEs).

- SOS was an early SGM for tabular data oversampling.
- STaSy uses self-paced learning and fine-tuning for stable training.
- MissDiff trains diffusion models on data with missing values by masking the regression loss.
- Forest-VP/Forest-Flow employs XGBoost instead of neural networks to estimate scores, naturally handling missing data.
- TABSYN operates in a VAE-learned latent space.

Diffusion models, while powerful, can be computationally intensive, especially for high-dimensional data and during sampling.

3.3 LLM-based Methods

Large Language Models (LLMs) are increasingly explored for tabular data generation due to their ability to understand context, semantics, and incorporate implicit knowledge.

3.3.1 Prompt-based Methods

These methods leverage the in-context learning (ICL) capabilities of LLMs, guiding generation through carefully designed prompts that include task descriptions, column meanings, examples, and generation instructions, without fine-tuning the LLM itself.

- EPIC uses balanced, grouped samples and specific formatting in prompts to generate data, even for imbalanced datasets.
- CLLM focuses on data augmentation in low-data scenarios using LLM priors and a curation mechanism.
- LITO is an oversampling framework that progressively masks and imputes features to synthesize minority samples.
- GReaT converts tabular data rows into natural language sentences, fine-tunes an LLM, and then uses it for generation, allowing arbitrary conditioning. (Note: GReaT involves fine-tuning, but its prompting aspect for generation is also significant).

3.3.2 Fine-Tuning Methods

To improve alignment with tabular data structure and constraints, LLMs are often fine-tuned on specific tabular datasets.

- GReaT fine-tunes autoregressive LLMs on serialized table data.
- TabMT uses a BERT-based masked transformer for generation.
- TAPTAP employs tabular data pre-training to enhance prediction and generation.
- HARMONIC uses an instruction fine-tuning dataset based on k-nearest neighbors to help LLMs learn inter-row relationships.
- AIGT uses metadata (descriptions, schemas) as prompts during fine-tuning and generation.
- DP-LLMTGen focuses on differentially private tabular data synthesis through a two-stage fine-tuning process with a novel loss function.

The "hallucination problem" in LLMs can lead to flawed or unrealistic samples, necessitating post-processing.

4 Post-Processing of Synthetic Data

Generated synthetic data, especially from LLMs, may contain unrealistic or incorrect instances (e.g., negative age, violations of common sense). Post-processing aims to enhance data quality and ensure alignment with human knowledge or domain-specific constraints.

4.1 Sample Enhancement

This involves modifying feature values or filtering out unreasonable samples.

4.1.1 Sample Filtering

Methods design criteria based on learning dynamics (confidence scores, impact functions) to filter out low-quality or problematic samples. LITO uses self-authentication with rejection sampling, where the LLM validates its own generated samples.

4.1.2 Sample Correction

This approach incorporates domain knowledge, often as constraints or rules, to correct samples.

- C-DGM / LL: Adds a layer to the generative model to restrict output to a space defined by linear inequalities, guaranteeing constraint satisfaction.
- DGM+DRL / DRL: Extends this to handle more complex constraints (e.g., disjunctions of linear inequalities) using Quantifier-Free Linear Real Arithmetic (QFLRA) formulas, allowing for non-convex output spaces.

4.2 Label Enhancement

This aims to correct potential annotation errors in synthetic samples.

- **Manual Re-labeling:** Actively selecting low-confidence samples for human re-annotation, though costly.
- **Proxy Model Approach:** Using automated models to refine labels. TAPTAP and AIGT use a discriminative model trained on original data to generate pseudo-labels for synthetic features. Pred-LLM prompts an LLM to generate labels after generating features.

5 Evaluating Synthetic Tabular Data

Evaluation is crucial and is typically multifaceted, focusing on data availability/utility and privacy.

5.1 Data Availability / Utility

This assesses the usefulness and realism of the synthetic data.

5.1.1 Machine Learning (ML) Efficiency / Utility

The "Train on Synthetic, Test on Real" (TSTR) protocol is common. A model trained on synthetic data is evaluated on a real test set, and its performance (e.g., accuracy, F1-score, RMSE) is compared to a model trained on real data. Good utility means comparable performance.

5.1.2 Fidelity

Fidelity measures how well synthetic data preserves the statistical properties of the original data.

- **Column-wise (Marginal) Fidelity:** Compares individual feature distributions. Metrics include Kolmogorov-Smirnov (KS) test or Wasserstein distance for numerical features, and Total Variation Distance (TVD) or Jensen-Shannon Divergence (JSD) for categorical features.
- **Pairwise Column Correlation:** Assesses if relationships between pairs of features are preserved. Metrics include differences in Pearson correlation matrices (DPCM) for numerical-numerical pairs, or contingency table-based similarities for categorical pairs.
- **Joint Distribution Fidelity:** Evaluates similarity of the entire joint distribution, e.g., using likelihood fitness scores, β-Recall, or Coverage score.

5.1.3 Alignment

Alignment assesses if synthetic data adheres to known domain-specific knowledge or constraints. Metrics include:

- **Constraint Violation Rate (CVR):** Percentage of samples violating at least one constraint.
- **Constraint Violation Coverage (CVC):** Percentage of constraints violated by at least one sample.
- **Sample-wise Constraint Violation Coverage (sCVC):** Average percentage of samples violating each constraint.

5.2 Privacy

Privacy metrics quantify the risk of re-identifying original records or inferring sensitive information from synthetic data.

- **Distance to Closest Record (DCR):** Minimum distance from a synthetic sample to any real sample. Low DCR may indicate memorization.
- **Attribute Inference Attack:** Assesses if an adversary with partial knowledge can infer missing sensitive attributes from the synthetic data. High accuracy implies poor privacy.
- **Membership Inference Attack:** Determines if an attacker can identify whether a specific individual's data was in the training set used to generate the synthetic data. High success rate suggests overfitting and privacy risk.
- For methods using differential privacy (DP), evaluation involves reporting results under different privacy budgets (ϵ).

6 Practical Applications

6.1 Improving Tabular Data Availability

- **Handling Class Imbalance:** Generative models like CGANs, CTAB-GAN, and CTGAN can synthesize minority class samples. SOS uses score-based models for style-transfer oversampling. LLMs like EPIC and LITO can be prompted to generate minority samples.
- **Missing Value Imputation:** While traditional methods like MICE exist, generative models, including diffusion models (e.g., MissDiff, CSDI, TabDiff) and fine-tuned LLMs, can implicitly or explicitly impute missing values. Forest-Diffusion uses XGBoost which inherently handles missing data for generation/imputation.

6.2 Privacy Protection

Synthetic data allows for data analysis and model development without exposing sensitive real data, crucial in domains like healthcare and finance.

- **Healthcare (EHRs):** Models like medGAN, TabDDPM-EHR, MedDiff, DPM-EHR, EHRDiff, and FLEXGEN-EHR generate synthetic EHRs.
- **Finance:** FinDiff, EntTabDiff, and Imb-FinDiff are designed for financial data synthesis. DP-Fed-FinDiff integrates differential privacy and federated learning for financial data. SiloFuse generates data from feature-partitioned silos using distributed latent diffusion. FedTabDiff uses federated learning for privacy-preserving tabular diffusion.

7 Future Directions and Open Challenges

Despite progress, several areas require further research:

- **Trade-off Between Data Utility/Availability and Privacy:** Finding the right balance remains a key challenge. Many models excel in one aspect but compromise on the other.
- **Alignment with Human Knowledge and Domain Constraints:** Ensuring synthetic data is not just statistically similar but also semantically correct and compliant with real-world logic is crucial, especially with the rise of neuro-symbolic AI. Current methods for enforcing complex constraints are still developing.
- **Enhanced Interpretability and Fairness:** Generative models are often black boxes and can inherit or amplify biases from training data. Developing interpretable and fair generation methods is vital. FairTabDDPM is an example addressing fairness.
- **Scalability and Efficiency:** Diffusion models, in particular, can be computationally intensive for large, high-dimensional datasets and slow at sampling. Efficient training and sampling techniques are needed.
- **Standardized Evaluation Metrics and Benchmarks:** Consistent and comprehensive evaluation is complicated by the diversity of data and tasks. More standardized benchmarks are needed, especially for domain-specific quality aspects.
- **Modeling Complex Dependencies and Heterogeneity:** Accurately capturing dependencies between mixed-type features and handling very high-dimensional sparse data remain ongoing challenges.
- **Hybrid Models:** Combining strengths of different architectures (e.g., VAEs/GANs with diffusion models or LLMs) could lead to more robust solutions. AutoDiff (Autoencoder + Diffusion) is one such example.
- **Cross-Modality Integration:** Exploring how tabular diffusion models can be combined with models for other data types (image, text, time-series) for cross-modal applications.
- **Advanced Post-Processing:** Developing more sophisticated post-processing techniques to refine synthetic data, correct errors, and ensure logical consistency beyond current methods.
- **Handling Small and Imbalanced Datasets:** While some methods address this (e.g., SOS, LITO), robust generation from very limited or extremely imbalanced data is still a significant hurdle.

Concluding Remarks

Synthetic tabular data generation is a dynamic and critical field. Addressing the outlined challenges and exploring future directions will be key to unlocking the full potential of synthetic data across various domains, enabling data-driven innovation while respecting privacy and ensuring data quality.