2024 IEEE International Conference on Big Data (BigData) | 979-8-3503-6248-0/24/$31.00 ©2024 IEEE | DOI: 10.1109/BigData62323.2024.10825316

Enhancing Synthetic Data Generation for Class Imbalance and Privacy Preservation

Weijie Niu, Alberto Huertas, Jingjing Li, Burkhard Stiller
Communication Systems Group CSG, Department of Informatics, University of Zurich UZH, CH–8050 Zürich, Switzerland
[niu, huertas, stiller]@ifi.uzh.ch, jingjing.li@uzh.ch

***Abstract*—Synthetic data generation has emerged as a powerful solution to meet the demand for high-quality, diverse, and privacy-preserving data in many domains. Still, there is an open challenge when dealing with class imbalance and privacy preservation in synthetic tabular data generation. Thus, this study introduces two algorithms: balanced Tabular Generative Adversarial Network (b-TGAN) and balanced Tabular Principle Component Analysis (b-TPCA). While b-TGAN proactively tackles class imbalance by incorporating a re-balancing mechanism and leveraging an Autoencoder, b-TPCA offers a privacy-preserving solution by generating synthetic data using statistical properties. Through experiments on five datasets, this study demonstrates the effectiveness of b-TGAN in generating balanced data, particularly in improving the performance on minority classes. b-TPCA also shows promising results, achieving comparable ML utility to the baseline method while enhancing privacy preservation.**

***Index Terms*—Synthetic Data Generation, Generative Models, Class Imbalance, Privacy Preservation**

I. Introduction

With the advances in artificial intelligence, the performance of machine learning (ML) models is often tied to the volume and quality of data they are trained on. However, obtaining a large amount of real-world data faces many problems. The collection process can be time-consuming and expensive. In addition, privacy concerns, particularly in sensitive domains like healthcare and finance, severely limit data accessibility. In such a context, some regulations such as the General Data Protection Regulation (GDPR) [1] restrict data acquisition and publication. However, the conflict between the necessity for data and the requirement to protect privacy has created a pressing need for innovative solutions. Besides, real-world datasets often face the problems of imbalance and data scarcity in under-representative classes.

To meet the growing demand for data, synthetic data generation has emerged as a solution. Synthetic data refers to artificially generated data points that mimic the characteristics and statistical properties of real-world data [2]. Unlike real data collected from actual observations in the real world, synthetic data is created using algorithms or models. The applications of synthetic data are manifold, ranging from augmenting existing datasets to enhancing model generalization to safeguarding privacy in data-sharing scenarios and testing the robustness of ML systems [3].

However, generating high-quality synthetic data is a challenging task, especially considering both synthetic data quality for ML and privacy preservation [2]. Existing methods often lack the focus on distinguishing class imbalance and rebalancing the data [4]. In real-world scenarios, many datasets have class imbalance problems and ML models often need to learn features from minority classes to perform classification or prediction tasks. In such cases, obtaining a balanced dataset is crucial to improve the model effectiveness. However, existing data generation methods can only generate data that closely resembles the original distribution [4]. Additionally, non-probabilistic methods to generate synthetic data with limited contexts are lacking. The existing synthetic data generation methods based on deep learning are probabilistic and require large amounts of data exposure [5].

This work aims to overcome these limitations by introducing two algorithms: a balanced Tabular Generative Adversarial Network (b-TGAN) and a balanced Tabular Principle Component Analysis (b-TPCA) (publicly available at [6]). b-TGAN, an evolution of the Conditional Tabular GAN (CTGAN) framework [2], is specifically designed to address the issue of class imbalance in synthetic tabular data generation. By incorporating a re-balancing mechanism and leveraging the power of an autoencoder, b-TGAN can produce balanced and high-quality synthetic data that facilitates the development of fair and unbiased ML models. On the other hand, b-TPCA, inspired by the PCA dimensionality reduction algorithm, tackles the challenge of generating synthetic data with limited context. By requiring only basic statistical properties like mean, variance, and covariance as input, b-TPCA offers a privacy-preserving solution for tabular data generation. Through a series of experiments on diverse datasets, this work demonstrates the effectiveness of b-TGAN and b-TPCA in generating synthetic data that exhibits ML utility and privacy preservation. The results also show that there is still room for improvement in synthetic data generation, especially the trade-off between utility, imbalance and privacy preservation.

II. Related Work

Generative models for tabular synthetic data generation are an active and emerging research field in the ML community [2], [3], [7]–[10]. Tabular datasets are the most common data format in ML and store generally structured information. However, tabular data are often limited in size and privacy concerns [8], [9]. Unlike in NLP and vision problems, for which there are huge amounts of images and text data on the Internet, tabular datasets are often limited to specific topics [8], such as healthcare, finance, cybersecurity etc. Plus tabular

979-8-3503-6248-0/24/$31.00 ©2024 IEEE

datasets are more likely privacy-sensitive, which restraints their sharing and publication for downstream analysis tasks.

Therefore, this section dives into related studies focused on the specific tabular synthetic data generation. In this sense, the GAN architecture [11] was proposed in 2014 and it was initially applied to image generation or processing [12], later its application extended to tabular data generation fields and GAN-based methods took up the majority of the mainstream methods. Xu et al. [2] propose CTGAN as a synthetic tabular data generator to address issues such as the need to simultaneously model discrete and continuous columns, and multi-modal non-Gaussian values. CTAB-GAN [9] introduces the information loss and classification loss to the conditional GAN to solve long tail problems. Zhang et al. [4] design a GAN-based model called GANBLR which focuses on addressing the interpretation limitation in existing tabular GAN-based models. Zhao et al. [3] propose a CTAB-GAN+ model trained with DP stochastic gradient descent to impose strict privacy guarantees. Apart from the GAN-based model, diffusion model approaches also gain attention in tabular data generation, [8] introduces a diffusion model that can be applied to tabular datasets with mixed features. Marchev et al. [5] proposed a non-probabilistic data generation approach with Cholesky Decomposition.

In conclusion, the main challenge of existing solutions is that most methods are probabilistic and based on deep learning which requires a huge amount of data and they often lack the focus on class imbalance. The existing non-probabilistic methods, however, often overlook privacy preservability. These limitations underscore the need for enhancing data synthesis for class imbalance and privacy preservation.

III. Methodology

The essence of synthetic data generation lies in the approximation of the underlying probability distribution. The process of synthetic data generation can be mathematically formulated as a two-step procedure. The first step, as can be seen in (1), involves training a generative model, represented as a function G, to fit the hidden pattern of the real data, denoted as P_{data},

$$G^* = \arg\min_{G} D(P_{data}||P_G) \tag{1}$$

where G^* is the optimal generator that best approximates the real data distribution. $D(P_{data}||P_G)$ represents a divergence measure that quantifies the dissimilarity between the real data distribution P_{data} and P_G. Once the generator has learned the underlying distribution, the second step is to sample from the learned distribution to generate new synthetic data points as in (2),

$$x_{syn} \sim P_{G^*} \tag{2}$$

where x_{syn} denotes a synthetic data point generated by the model. The effectiveness of the synthetic data generation process relys on the ability of the generator to accurately capture the complexities and nuances of the real data distribution.

A. Balanced Tabular GAN (b-TGAN)

Class imbalance is a common issue in reality, which can lead to biased ML models. This study designed a balanced Tabular Generative Adversarial Network (b-TGAN) as an extension of the Conditional Tabular GAN framework for tackling the class imbalance problem. Fig. 1 presents the architecture of b-TGAN. b-TGAN first proactively identifies and then rectifies class imbalance in the original dataset before training the generator and discriminator. By employing techniques such as oversampling, the re-balancing mechanism ensures the model is exposed to a more balanced representation of data distribution, encouraging it to generate synthetic samples that reflect both majority and minority classes.

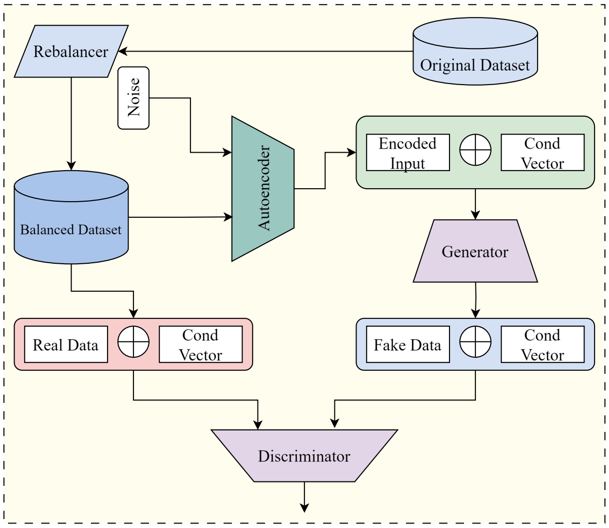

Fig. 1. The Architecture of b-TGAN

Another extension incorporated in b-TGAN is the Autoencoder-Guided input. b-TGAN leverages a pre-trained autoencoder component to enhance the quality and diversity of the generated synthetic data. The Autoencoder is first trained on the original dataset, and it learns a compressed latent representation of the data that captures the most salient characteristics, mapping the original data into the latent space. Then, during the GAN training process, instead of feeding purely random noise into the generator, b-TGAN uses the encoder from the pre-trained Autoencoder to transform the random noise into the latent space. This encoded noise, instilled with structural information from the real data, is then fed into the generator. This infusion of structured information derived from the real data provides the generator with a starting point that is already closer to the real data distribution. It helps the generator produce more realistic, diverse, and representative synthetic samples. Besides, the structured guidance from the encoded noise can also help prevent the model from getting stuck in a model collapse scenario where it produces only a limited set of repetitive samples.

B. Balanced Tabular Principal Component Analysis (b-TPCA) Generation Algorithm

The Balanced Tabular Principal Component Analysis (b-TPCA) generation algorithm is designed as a simple way to generate synthetic data based on specified statistical properties—namely, the mean vector, variance, and covariance matrix. This approach effectively reverses the traditional Principal Component Analysis (PCA) process, enabling the generation of data that preserves the essential characteristics of the original dataset. The b-TPCA algorithm considers the following configuration Mean vector: $\mu \in \mathbb{R}^d$, Covariance matrix: $\Sigma \in \mathbb{R}^{d \times d}$, and Variance: *Var* Derived from the diagonal elements of Σ. The goal is to generate synthetic data that approximates the distribution defined by these parameters. The algorithm begins with the eigen decomposition of the covariance matrix:

$$\Sigma = V \Lambda V^T \tag{3}$$

where Λ is the diagonal matrix of eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_d$, and V is the matrix of corresponding eigenvectors $v_1, v_2, \ldots, v_d$. Each eigenvector v_i corresponds to the eigenvalue λ_i in Λ.

The top n eigenvectors are selected corresponding to the largest eigenvalues to reduce dimensionality and focus on the most significant features. This forms the matrix V_n:

$$V_n = [v_1, v_2, \ldots, v_n] \tag{4}$$

The original mean vector is then transformed into the reduced space defined by the principal components. The variance in the reduced space is calculated as k:

$$\mu_{\text{new}} = \mu \cdot V_n \tag{5}$$

$$k = \text{Var} \cdot (V_n \circ V_n) \tag{6}$$

where $\circ$ denotes the Hadamard product. The transformed variance is then given by:

$$\sigma_{\text{new}} = \sqrt{k} \tag{7}$$

Synthetic samples are generated in the reduced space using the new mean and variance:

$$A \sim \mathcal{N}(\mu_{\text{new}}, \sigma^2_{\text{new}}) \tag{8}$$

where $\mathcal{N}$ represents a normal distribution with mean μ_{new} and variance σ^2_{new}.

Finally, the synthetic data is generated by the transformation back into the original feature space:

$$\text{Synthetic Data} = A \cdot V_n^T \tag{9}$$

This re-projection ensures that the generated synthetic data retains the statistical properties of the original dataset, thereby enabling the creation of realistic and useful synthetic datasets.

The b-TPCA algorithm as described in **Algorithm 1**, provides a robust method for generating synthetic data, particularly when working with high-dimensional datasets where privacy concerns or limited data availability require the generation of new samples that approximate the original data statistical characteristics.

Algorithm 1 b-TPCA Data Generation

```
Input:
  • Mean vector $\mu$
  • Variance Var
  • Covariance Matrix $\Sigma$
Output: Synthetic data sample
Compute eigenvalues $\lambda$ and eigenvectors $v$ of $\Sigma$
Select matrix $V_n$ containing eigenvectors corresponding to
  $n$ largest eigenvalues
Transform mean: $\mu_{\text{new}} \leftarrow \mu \cdot V_n$
Compute transformed variance: $k \leftarrow \text{Var} \cdot (V_n \circ V_n)$
Compute new variance: $\sigma_{\text{new}} \leftarrow \sqrt{k}$
Generate synthetic data sample: $A \sim \mathcal{N}(\mu_{\text{new}}, \sigma^2_{\text{new}})$
Transform synthetic data back to original space:
return $A \cdot V_n^T$
```

IV. Experiments and Results

To evaluate the two developed synthetic data generation algorithms, a series of experiments are conducted on five distinct real-world public tabular datasets from diverse fields: Adult [13], Covertype [14], Intrusion [15], Credit [16], and MIMIC [17]. These five datasets vary in terms of dataset size, dimensionality, class number and imbalance rate, which covers a variety of scenarios to evaluate the efficiency of the proposed algorithms thoroughly. The size of datasets varies from $\sim$ 2k to $\sim$ 580k. The MIMIC-III dataset is the only one with balanced classes among all datasets. TABLE I summarizes the information about used datasets. The synthetic datasets generated by these algorithms are assessed based on two key evaluation standards: ML utility and privacy preservability.

TABLE I
Summary of Datasets

Dataset	Size	Features	Field	Classes	Im-Bal
Adult [13]	48,842	14	Sociology	2	25:75
Covertype [14]	581,012	54	Forestry	7	High
Intrusion [15]	98,805	41	Cybersecurity	5	High
Credit [16]	284,807	30	Finance	2	High
MIMIC-III [17]	2,018	50	Healthcare	2	Bal

Note: In this paper, we define a dataset as 'highly imbalanced' when one or more classes represent less than 5% of the total data. This designation, 'high', signifies a significant disparity in the number of instances across different classes. "Bal" means the dataset is balanced across different classes.

A. Machine Learning Utility

The ML utility is evaluated by training two widely used and arguably stable ML classifiers on the synthetic datasets. The evaluation pipeline is shown in Fig. 2. The original dataset is first split into a training (80%) and test datasets (20%). The training dataset is then fed into the synthetic data generator which fits the data distribution and samples new data from this distribution. This forms a synthetic dataset which yields the real original distribution but contains no one-to-one matching relationship between the original data samples and the synthetic data samples. Support Vector Machine (SVM) and Random Forest classifiers are chosen as the downstream prediction models. These two classifiers are then trained on the synthetic datasets output by the data synthesizers and tested on the held-out test dataset, to evaluate the ML utility of the synthetic data generators. Precision, recall, accuracy and weighted F1-score are chosen as evaluation metrics. This work selects CTGAN as a deep learning based data generation algorithm to make a comparison with b-TGAN, and Cholesky Decomposition generation as a non-probabilistic method to serve as a baseline for b-TPCA generation algorithm.

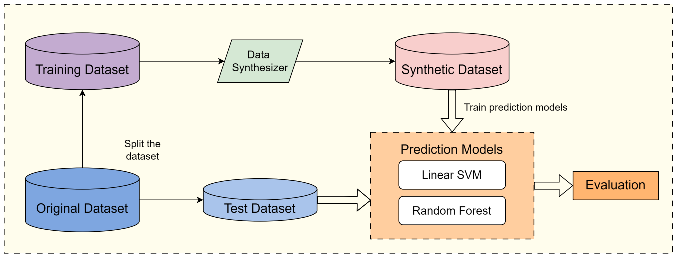

Fig. 2. The Pipeline of ML Utility Evaluation

TABLE II and TABLE III presents the results of SVM and Random Forest classifiers when are trained on the synthetic dataset generated by default CTGAN and b-TGAN, and evaluated on test data from original datasets. The experimental results highlight the capacity of b-TGAN to address the common challenge of class imbalance in tabular datasets. The observed improvement in F1-scores, particularly for minority classes in the Adult and Covertype datasets, underscores its ability to generate synthetic data that represents the underlying class distribution. Take the Adult dataset as an instance, the SVM F1-score of the minority class increases from 0.30 to 0.49 after using b-TGAN generator instead of CTGAN, while the random forest F1-score of the same class is 0.60 using b-TGAN compared to 0.47 using default CTGAN (see TABLE III). The successful application of b-TGAN to the highly imbalanced Credit dataset, where default CTGAN failed to generate any samples for the minority class, further demonstrates its robustness. This capability is particularly valuable in domains such as fraud detection or rare disease diagnosis, where the identification of minority classes is important. Although the SVM and Random Forest algorithm can make better classification of minority classes using b-TGAN synthetic datasets, these classes still occupy a relatively small proportion in the test set since this work uses a held-out test dataset which preserves the original class distribution. Thus the weighted average F1-score is lower, which suggests that there is still room for optimization in the b-TGAN algorithm.

The performance of b-TPCA reveals its potential in tabular data generation. PCA is primarily a dimensionality reduction technique, but its understanding of the underlying data structure can be leveraged for data generation in certain scenarios. Despite operating with limited information, relying solely on mean, variance, and covariance, b-TPCA can achieve comparable or superior ML utility to the baseline Cholesky Decomposition method as in TABLE II and III. This is particularly evident in the Covertype dataset, where b-TPCA-generated data led to improved weighted average F1-scores. The ability to generate useful synthetic data without access to the full dataset is crucial in scenarios where data privacy is a major concern, such as healthcare or finance.

TABLE II
Performance Comparison of Synthetic Data Generation Methods based on SVM classifier

Dataset	Method	F1*	Precision	Recall	F1-score
Adult	CTGAN	0.30	0.77	0.79	0.74
	b-TGAN	0.49	0.77	0.60	0.63
	Cholesky	0.35	0.79	0.80	0.75
	b-TPCA	0.29	0.81	0.80	0.74
Covertype	CTGAN	0.14	0.67	0.62	0.63
	b-TGAN	0.40	0.60	0.56	0.57
	Cholesky	0.07	0.41	0.31	0.31
	b-TPCA	0.11	0.45	0.37	0.38
Intrusion	CTGAN	0.99	0.99	0.99	0.9
	b-TGAN	0.95	0.97	0.98	0.97
	Cholesky	0.87	0.98	0.91	0.94
	b-TPCA	0.88	0.97	0.74	0.83
Credit	CTGAN	-	-	-	-
	b-TGAN	0.35	1.00	1.00	1.00
	Cholesky	0.29	1.00	1.00	1.00
	b-TGAN	0.50	1.00	1.00	1.00
MIMIC-III	CTGAN	-	0.27	0.52	0.35
	b-TGAN	-	0.23	0.48	0.31
	Cholesky	-	0.48	0.31	0.48
	b-TPCA	-	0.51	0.49	0.40

Note: F1* represents the F1-score for the minority classes. Other metrics are averagely weighted.

However, the mixed results observed for b-TPCA across different datasets, with underperformance on the Intrusion and MIMIC-III datasets, suggest that the specific characteristics and complexities of the data distribution might influence its effectiveness. Since this study uses the normal distribution to generate data in the principal component space as in Equation (8) in Section III-B, this may not be always the case for some datasets. The potential of b-TPCA in privacy-preserving ML is clear, but its limitations highlight the need for continued research and development in this area.

B. Privacy Preservability

This work evaluates the privacy preservability of these two developed data generation algorithms. In this context, one shares the synthetic datasets for downstream tasks but without revealing the data points from the original datasets. Distance to Closest Record (DCR) is used as the privacy metric as in [3], [8], [9], which measures the distance between real and synthetic data points. Specifically, the DCR is calculated as the average Euclidean distance between each record in the synthetic dataset and its closest corresponding record in the

TABLE III
Performance Comparison of Synthetic Data Generation Methods based on Random Forest classifier

Dataset	Method	F1*	Precision	Recall	F1-score
Adult	CTGAN	0.47	0.81	0.82	0.79
	b-TGAN	0.60	0.81	0.77	0.78
	Cholesky	0.44	0.80	0.81	0.78
	b-TPCA	0.47	0.82	0.82	0.79
Covertype	CTGAN	0.19	0.66	0.62	0.63
	b-TGAN	0.25	0.58	0.48	0.49
	Cholesky	0.07	0.41	0.31	0.31
	b-TPCA	0.11	0.45	0.37	0.38
Intrusion	CTGAN	0.97	0.99	0.99	0.99
	b-TGAN	0.94	0.98	0.97	0.98
	Cholesky	0.87	0.98	0.95	0.96
	b-TPCA	0.86	0.96	0.79	0.87
Credit	CTGAN	-	-	-	-
	b-TGAN	0.10	1.00	0.98	0.99
	Cholesky	0.17	1.00	1.00	1.00
	b-TGAN	0.00	1.00	1.00	1.00
MIMIC-III	CTGAN	-	0.42	0.49	0.38
	b-TGAN	-	0.53	0.48	0.33
	Cholesky	-	0.23	0.48	0.31
	b-TPCA	-	0.51	0.49	0.40

Note: F1* represents the F1-score for the minority classes. Other metrics are averagely weighted.

TABLE IV
Distance to Closest Record (DCR)

	CTGAN	b-TGAN	Cholesky	b-TPCA
Adult	850.77	598.01	1036.72	9739.23
Covertype	191.55	219.22	315.46	369.71
Intrusion	801.49	210.54	240.21	303.64
Credit	46.96	73.47	180.27	1831.96
MIMIC	3132.40	2998.10	6227.97	13008.39

original dataset. Mathematically, for a synthetic dataset S and an original datset O, the DCR is calculated as (10)

$$DCR(S, O) = \frac{1}{|S|} \sum_{x \in S} \min_{y \in O} d(x, y) \tag{10}$$

Larger DCR values indicate that the synthetic data points are distant from the real data points rather than copies of real data, which prevents privacy leakage from attacks such as membership inference attacks. The datasets generated by b-TGAN consistently exhibit lower DCR values compared to the default CTGAN. This suggests that while b-TGAN performs well at generating balanced synthetic data, it might increase the risk of privacy leakage. The possible reason is the incorporation of encoded representations from the original data, which while beneficial for model training and imbalanced data utility, appears to bring the synthetic data closer to the real data, making it easier to identify individuals. In contrast, the b-TPCA methods consistently outperforms the baseline Cholesky decomposition in terms of DCR across all datasets. This indicates that b-TPCA generates data points that are significantly more distant from the original data, thereby enhancing privacy preservation against potential attacks.

V. Conclusion

This work presents b-TGAN and b-TPCA, two algorithms tackling the critical limitations of class imbalance and privacy preservability in synthetic tabular data generation. b-TGAN addresses class imbalance, a common issue in real-world datasets, improving performance in minority classes, while b-TPCA offers a privacy-preserving alternative, generating synthetic data with limited context in statistics while maintaining ML utility. A series of experimental results on five datasets among various fields validated the effectiveness of both algorithms, especially the capability of b-TGAN to address minority classes and the privacy preservability of b-TPCA without much trade-off in data quality and utility. Future work is planned to optimize their performance on complex datasets and explore in-depth the trade-off between utility, imbalance, and privacy preservation. The insights gained from this study pave the way for the development of more robust and privacy-aware synthetic data generation techniques.

Acknowledgment

This work was partially supported by (a) the University of Zürich UZH, Switzerland, and (b) the Horizon Europe Framework Program's project AISym4MED, Grant Agreement No.101095387, funded by the Swiss State Secretariat for Education, Research, and Innovation SERI, under Contract No.22.00622.

References

[1] P. Voigt and A. Von dem Bussche, "The eu general data protection regulation (gdpr)," *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, vol. 10, no. 3152676, pp. 10–5555, 2017.

[2] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling tabular data using conditional gan," *Advances in neural information processing systems*, vol. 32, 2019.

[3] Z. Zhao, A. Kunar, R. Birke, H. Van der Scheer, and L. Y. Chen, "Ctab-gan+: Enhancing tabular data synthesis," *Frontiers in big Data*, vol. 6, p. 1296508, 2024.

[4] Y. Zhang, N. A. Zaidi, J. Zhou, and G. Li, "Ganblr: a tabular data generation model," in *2021 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2021, pp. 181–190.

[5] A. Marchev, Jr and V. Marchev, "Automated algorithm for multi-variate data synthesis with cholesky decomposition," in *Proceedings of the 7th International Conference on Algorithms, Computing and Systems*, 2023, pp. 1–6.

[6] J. Li, "Implementation of b-tgan and b-tpca," https://github.com/Jingjing-Li/dataset-generation-augmentation, 2024, [Version 1.2].

[7] Y. Gorishniy, I. Rubachev, V. Khrulkov, and A. Babenko, "Revisiting deep learning models for tabular data," *Advances in Neural Information Processing Systems*, vol. 34, pp. 18 932–18 943, 2021.

[8] A. Kotelnikov, D. Baranchuk, I. Rubachev, and A. Babenko, "Tabddpm: Modelling tabular data with diffusion models," in *International Conference on Machine Learning*. PMLR, 2023, pp. 17 564–17 579.

[9] Z. Zhao, A. Kunar, R. Birke, and L. Y. Chen, "Ctab-gan: Effective table data synthesizing," in *Asian Conference on Machine Learning*. PMLR, 2021, pp. 97–112.

[10] B. Wen, Y. Cao, F. Yang, K. Subbalakshmi, and R. Chandramouli, "Causal-tgan: Modeling tabular data using causally-aware gan," in *ICLR Workshop on Deep Generative Models for Highly Structured Data*, 2022.

[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” *Advances in neural information processing systems*, vol. 27, 2014.

[12] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, “Analyzing and improving the image quality of stylegan,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8110–8119.

[13] B. Becker and R. Kohavi, “Adult,” UCI Machine Learning Repository, 1996, DOI: https://doi.org/10.24432/C5XW20.

[14] J. Blackard, “Covertype,” UCI Machine Learning Repository, 1998, DOI: https://doi.org/10.24432/C50K5N.

[15] “Network Intrusion Detection,” Kaggle Datasets, 2018, https://www.kaggle.com/datasets/sampadab17/network-intrusion-detection.

[16] “Credit Card Fraud Detection,” Kaggle Datasets, 2017, https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud.

[17] A. E. W. Johnson, D. J. Stone, L. A. Celi, and T. J. Pollard, “The mimic code repository: enabling reproducibility in critical care research,” *Journal of the American Medical Informatics Association*, vol. 25, no. 1, pp. 32–39, 2018.