MargCTGAN: A "Marginally" Better CTGAN for the Low Sample Regime

Tejumade Afonja[0000−0003−0639−9668], Dingfan Chen[0000−0001−7279−6624], and Mario Fritz[0000−0001−8949−9896]

CISPA Helmholtz Center for Information Security
{tejumade.afonja,dingfan.chen,fritz}@cispa.de

Abstract. The potential of realistic and useful synthetic data is significant. However, current evaluation methods for synthetic tabular data generation predominantly focus on downstream task usefulness, often neglecting the importance of statistical properties. This oversight becomes particularly prominent in low sample scenarios, accompanied by a swift deterioration of these statistical measures. In this paper, we address this issue by conducting an evaluation of three popular synthetic tabular data generators based on their marginal distribution, column-pair correlation, joint distribution and downstream task utility performance across high to low sample regimes. The popular `CTGAN` model shows strong utility, but underperforms in low sample settings in terms of utility. To overcome this limitation, we propose `MargCTGAN` that adds feature matching of de-correlated marginals, which results in a consistent improvement in downstream utility as well as statistical properties of the synthetic data.

Keywords: Synthetic Data · GAN · Tabular Data · Evaluation Metrics.

1 Introduction

Tabular data, despite being the most widely used data type [12], presents substantial challenges ranging from data heterogeneity and quality measurement to imbalance and privacy concerns. Encouragingly, recent advancements in synthetic tabular data generators have shown considerable promise in tackling these issues. These models have shown effectiveness in handling heterogeneous data attributes [22,25], facilitating the safe sharing of personal records [5,19,18], and mitigating class imbalance [8]. Nonetheless, the evaluation of existing models predominantly focuses on downstream machine learning tasks and large datasets. This evaluation paradigm overlooks their utility in broader practical scenarios, especially the data-limited, low-resource settings, and fails to consider other crucial aspects of synthetic datasets including fidelity, diversity, and authenticity [1].

In response to these challenges, we introduce a comprehensive evaluation framework, integrating nine distinct metrics across four critical dimensions: downstream task utility, joint fidelity, preservation of attribute correlations, and alignment of marginals (Section 5). Our objective is to thoroughly evaluate the representative models using diverse metrics, aiming at a comprehensive

understanding of their quality and adaptability, particularly for scenarios that are underexplored in existing literature.

Our evaluation uncovers intriguing insights into the characteristics of three popular synthetic tabular data generators: `TableGAN` [19], `CTGAN` [22], `TVAE` [22]. For instance, `CTGAN` model typically demonstrates high attribute fidelity but falls short in utility for low-data scenarios. Conversely, `TableGAN` exhibits better utility but lacks performance in other dimensions. To capitalize on the strengths of both models, we propose `MargCTGAN` that improves upon `CTGAN` by introducing feature matching of decorrelated marginals in the principal component space. This approach consistently improves utility without compromising other fidelity measures, especially in the data-limited settings.

In summary, we make the following contributions:

- We conduct an extensive investigation into the performance of representative tabular data generators across various contexts, with a specific focus on the low-sample regime that is underexplored in existing literature. This investigation is carried out using a comprehensive evaluation framework, which assesses the models across four critical dimensions: downstream task utility, joint fidelity, column-pair fidelity, and marginal fidelity.
- Our comprehensive evaluation framework is released[1] as an open-source tool, with the aim of facilitating reproducible research, encouraging fair and extensive comparisons among methods, as well as providing a deeper understanding of the models' performance, quality, and fidelity.
- Prompted by the suboptimal performance of existing tabular generators in low-sample scenarios, we propose `MargCTGAN`. This model improves upon `CTGAN` by introducing a moment-matching loss within a decorrelated feature space, which effectively steers the generator towards capturing the statistical characteristics intrinsic to the data distribution.

2 Related Works

2.1 Tabular Data Generators

In recent years, deep generative models have seen significant advancements in their application to diverse forms of tabular data, including discrete attributes [5], continuous values [18], and heterogeneous mixtures [19,23,22,25]. Notably, `TableGAN` [19], `CTGAN` [22], and `TVAE` [22] stand out as the most popular benchmark models and will be the focus of our empirical evaluation. On the other hand, the issue of limited data availability remains underexplored in literature, despite several attempts to bypass such challenges by effectively combining multiple data sources [4,17,24]. Our work aims to fill this research gap by introducing a systematic evaluation across various scenarios, ranging from full-resource to data-limited cases. Additionally, we propose model improvements that designed to effectively capture the underlying data structure in low-sample settings.

[1] https://github.com/tejuafonja/margctgan/

2.2 Evaluation of Tabular Data Generators

The evaluation of generators, particularly for tabular data, is a challenging area due to its requirement for complex metrics, unlike simpler visual inspection for image data [21]. Recent studies have introduced a variety of metrics: prominent among these are downstream machine learning efficacy, unified metrics evaluation [6], and evaluations focusing on distinct aspects [1,7]. In this work, we consider comprehensive evaluation methods encompassing machine learning efficacy, statistical properties such as divergence on marginals, column correlations, and joint distance.

3 Background

We examine three leading tabular data generators with diverse architectures and preprocessing schemes. Here, we present an overview of each model.

3.1 Tabular GAN (`TableGAN`)

`TableGAN` is a GAN-based method for synthesizing tabular data [19]. It converts categorical columns into numerical representations using label encoding and applies min-max normalization to numerical columns. Each record is transformed into a 2D image represented as a square matrix, allowing the use of the DCGAN model [20] for generating synthetic data. The architecture consists of a generator network ($\mathcal{G}$), a discriminator network ($\mathcal{D}$), and a classifier network ($\mathcal{C}$). $\mathcal{G}$ generates synthetic data resembling real data, while $\mathcal{D}$ distinguishes between real and synthetic data. $\mathcal{C}$ helps generate data suitable for downstream tasks by predicting labels. Training follows standard GAN techniques [10], optimizing $\mathcal{G}$ and $\mathcal{D}$ using the GAN loss. Information loss ($\mathcal{L}^{\mathcal{G}}_{\text{info}}$) minimizes statistical differences between real and synthetic data. The final loss objective optimized by $\mathcal{G}$ is $\mathcal{L}^{\mathcal{G}} = \mathcal{L}^{\mathcal{G}}_{\text{orig}} + \mathcal{L}^{\mathcal{G}}_{\text{info}} + \mathcal{L}^{\mathcal{G}}_{\text{class}}$, where $\mathcal{L}^{\mathcal{G}}_{\text{orig}}$ minimizes $\log(1 - \mathcal{D}(\mathcal{G}(z)))$, and $\mathcal{L}^{\mathcal{G}}_{\text{info}}$ minimizes statistical properties with privacy control.

3.2 Conditional tabular GAN (`CTGAN`)

`CTGAN` [22] is a GAN-based model designed to tackle challenges in data synthesis, including generating multi-modal numerical columns and balancing categorical columns. It introduces novel preprocessing schemes: mode-specific normalization and training-by-sampling. Mode-specific normalization uses a variational Gaussian mixture model to estimate the number of modes in numerical columns and samples normalized values accordingly. Training-by-sampling addresses imbalanced categorical columns by conditioning the generator ($\mathcal{G}$) and discriminator ($\mathcal{D}$) on a condition vector, which resamples the categorical column during training iterations. `CTGAN` is based on the PacGAN [16] framework and uses fully connected neural networks for $\mathcal{G}$ and $\mathcal{D}$. It optimizes the Wasserstein loss with gradient penalty (WGP) [2] . The final loss objective optimized by $\mathcal{G}$ includes the Wasserstein loss and a generator condition loss.

3.3 Tabular VAE model (TVAE)

TVAE [22] is a variant of the variational autoencoder (VAE), a generative model based on encoder-decoder neural networks. It employs fully connected neural networks for the encoder (enc) and decoder (dec) networks. The encoder maps the input data to a latent code, which serves as a compressed representation. The decoder reconstructs the original data from this latent code. Both networks are trained simultaneously using the evidence lower-bound (ELBO) [13] objective, promoting effective latent space distribution learning and accurate data reconstruction. By training the encoder and decoder together, TVAE learns a compact representation capturing the data structure and generating realistic samples. Numerical columns undergo similar preprocessing as CTGAN, while categorical columns are one-hot encoded.

4 Method: MargCTGAN

MargCTGAN adheres to the standard Generative Adversarial Networks (GANs) paradigm [10], which involves training a generator $\mathcal{G}$ and a discriminator $\mathcal{D}$ in an adversarial manner. The training target is to enhance the discriminator's ability to distinguish between real and fake data, while simultaneously updating the generator to produce samples that are increasingly realistic. We adopt the WGAN-GP objective [11] in which the overall training process can be interpreted as optimizing the generator to minimize the Wasserstein distance between the distributions of the generated and real data:

$$\mathcal{L}_{\text{WGP}} = \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}}\big[\mathcal{D}\big(\mathcal{G}(\boldsymbol{z})\big)\big] - \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_{\text{data}}}\big[\mathcal{D}(\boldsymbol{x})\big] + \lambda\big(\|\nabla_{\hat{\boldsymbol{x}}}\mathcal{D}(\hat{\boldsymbol{x}})\|_2 - 1\big)^2 \tag{1}$$

where $\hat{\boldsymbol{x}}$ is constructed by interpolating real and generated samples and λ denotes the weight for the gradient penalty. The discriminator is trained to minimize $\mathcal{L}_{\text{WGP}}$, while the generator is trained to maximize it.

Following the CTGAN [22] framework, we adopt several key techniques to adapt GAN models for tabular data. Firstly, one-hot encoding is applied to preprocess categorical attributes, paired with the Gumbel-softmax function serving as the network output activation function, thereby ensuring differentiability. Secondly, for numerical attributes, we apply a technique known as *mode-specific normalization* in the pre-processing phase, enabling an accurate reflection of the multi-modality in the values distribution. Lastly, we employ the *training-by-sampling* strategy during the training process, which effectively balances the occurrences of different classes in the categorical columns to match their real distribution. This strategy introduces an additional loss term on the generator, which we denote as $\mathcal{L}_{\text{cond}}$.

While CTGAN generally demonstrates promising utility for training downstream machine learning classifiers, it often falls short in capturing low-level distribution statistics, particularly in low-sample scenarios (See Figure 2). Drawing inspiration from TableGAN , we propose a moment matching loss that proactively encourages

the generator to learn and mirror the first and second-order data statistics. Notably, unlike `TableGAN` which attempts to match statistics on the features extracted by the discriminator, we compute the first and second moments after conducting the Principal Component Analysis (PCA) on the data. Specifically, the transform is performed while maintaining the original data dimensionality, i.e., we simply decorrelate without down-projection. Intriguingly, this straightforward technique proves effective (Figure 4), likely because the decorrelated feature representation supports the independent moment matching. Formally,

$$\begin{aligned}
\mathcal{L}_{\text{mean}} &= \Big\| \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_{\text{data}}} \big[f(\boldsymbol{x})\big] - \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}} \big[f\big(\mathcal{G}(\boldsymbol{z})\big)\big] \Big\|_2 \\
\mathcal{L}_{\text{std}} &= \Big\| \mathop{\mathbb{SD}}_{\boldsymbol{x}\sim p_{\text{data}}} \big[f(\boldsymbol{x})\big] - \mathop{\mathbb{SD}}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}} \big[f\big(\mathcal{G}(\boldsymbol{z})\big)\big] \Big\|_2 \\
\mathcal{L}_{\text{marg}} &= \mathcal{L}_{\text{mean}} + \mathcal{L}_{\text{std}} \\
\mathcal{L}^{\mathcal{G}} &= -\mathcal{L}_{\text{WGP}} + \mathcal{L}_{\text{cond}} + \mathcal{L}_{\text{marg}}
\end{aligned} \tag{2}$$

where $f(\cdot)$ denotes the PCA transformation function. $\mathcal{L}_{\text{mean}}$ targets the mean, while $\mathcal{L}_{\text{std}}$ focuses on the standard deviation. The total training losses for the generator and discriminator are $\mathcal{L}^{\mathcal{G}}$ and $\mathcal{L}_{\text{WGP}}$, respectively.

5 Multi-Dimensional Evaluation Metrics

We present a comprehensive evaluation that accesses tabular data generators performance across four critical dimensions: downstream task utility, joint fidelity, column-pair fidelity, and marginal fidelity. The implementation details can be found in Section 6.

Downstream Task Utility. This dimension focuses on the efficacy of synthetic data as a substitute for real data in specific tasks. This effectiveness is typically quantified by *machine learning efficacy* that evaluates the performance (e.g., F1-score or accuracy) on a distinct real test dataset when training predictive ML models on synthetic data. In situations where knowledge of the target downstream task is unavailable, an alternate methodology known as *dimension-wise prediction* (or *all-models test*) may be employed. This methodology considers each column as a potential target variable for the task and reports the mean performance across all cases.

Joint Fidelity. This category aims to quantify the similarity between the overall *joint* distributions of real and synthetic data. While an exact measurement is always intractable, the most commonly used approximation is the *distance to closest record*. This computes the Euclidean distance between each synthetic data sample and its nearest neighbors in the real test dataset, intending to assess the possibility of each synthetic sample being real. Conversely, the *likelihood approximation* computes the distance between each real test sample and its

closest synthetic sample. This mirrors the probability of each real sample being potentially generated by the model, thereby encapsulating a concept of data likelihood.

Column-Pair Fidelity. This dimension investigates the preservation of feature interactions, specifically focusing on the direction and strength of correlations between pairs of columns in the synthetic dataset as compared to the real dataset. A commonly used metric for this purpose is the *association difference*, also referred to as the *pairwise correlation difference*. This measure quantifies the discrepancy between the correlation matrices of the real and synthetic datasets, where the correlation matrix encapsulates the pairwise correlation of columns within the data.

Marginal Fidelity. Accurately replicating the real data distribution requires aligning the marginals, ensuring a match in the distribution of each individual column. Evaluating this criterion involves quantifying the disparity between two one-dimensional variables. Commonly used metrics for this purpose include the *Jensen-Shannon divergence*, *Wasserstein distance*, and *column correlation*. Additionally, we propose *histogram intersection*, widely used in various other fields. The metric calculates the sum of the minimum probabilities between the synthetic and real data distributions, expressed as $HI(p,q) = \sum_i \min(p_i, q_i)$. A perfect match between p and q yields $HI(p,q) = 1$, while $HI(p,q) = 0$ indicates no overlap between the two distributions. For numerical columns, discretization via binning is typically performed prior to calculating divergence measures to ensure tractability.

6 Implementation Details

6.1 Metrics

Downstream Task Utility. For the *machine learning efficacy* and *all models test* metrics, we used the SDMetrics package [2] with logistic regression, decision tree classifier, and multilayer perceptron models for classification tasks, and linear regression, decision tree regressor, and multilayer perceptron models for regression tasks. We standardized numerical columns for classification models and performed one-hot encoding for categorical columns. F1-score was used for classification models, and R^2-score was normalized to $[0, 1]$ for regression models.

Joint Fidelity. Numerical columns were min-max normalized to range between 0 and 1, and categorical columns were one-hot encoded. We used the scikit-learn nearest-neighbor implementation [3] with Euclidean distance and different

[2] https://docs.sdv.dev/sdmetrics/metrics/metrics-in-beta/ml-efficacy-single-table/binary-classification

[3] https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html

Table 1: Summary of Datasets. **Col** refers to number of columns. **N/B/M** correspond to the number of numerical, binary, and multi-class categorical columns, respectively.

Dataset	Train/Test Size	Col	N/B/M	Task
Adult	34118/14622	15	7/2/6	classification
Census	199523/99762	41	7/3/31	classification
News	31644/8000	60	45/15/0	classification
Texas	60127/15032	18	7/1/10	classification

numbers of nearest neighbors ($[1, 2, 3, \ldots, 9]$). For the *likelihood approximation*, we calculated the distance between each of 5000 random test samples to its closest synthetic sample and report the average over real test samples. For the *distance to closest record* metric, we compute the distance of each sample in the synthetic set to its nearest neighbor in a set of 5000 random test samples and report the average over the synthetic samples.

Column-Pair Fidelity. The *associations difference* metric was inspired by the "plot correlation difference" function from the tabular-evaluator package [4] and implemented using the dython package [5]. We used Pearson correlation coefficient for numerical columns, Cramer's V for categorical columns, and the Correlation Ratio for numerical-categorical columns. The range of Cramer's V and Correlation Ratio is between 0 and 1, while Pearson correlation coefficient ranges from -1 to 1. We calculated the absolute difference between the association matrices of the synthetic data and the real data, reporting the mean absolute difference.

Marginal Fidelity. Numerical columns were min-max normalized to range between 0 and 1, and categorical columns were one-hot encoded. The marginal metrics were applied to each column in the dataset. Binning followed a uniform grid between 0 and 1, using bin widths sizes of 25, 50, or 100 for the real data. For the *histogram intersection*, *Wasserstein distance*, and *Jenson-Shannon distance* metrics, the same binning strategy was used for numerical columns. The SciPy package [6] was used to calculate *Wasserstein distance* and *Jenson-Shannon distance* with the base=2 setting. The Dython package was used to compute the *column correlation* metric and provided an implementation for the *histogram intersection* metric, which was not available in prior work.

7 Experiments

Setup. We conducted evaluations on four benchmark tabular datasets: Adult [14], Census [15], News [9], and Texas. These datasets exhibit diverse properties in

[4] https://github.com/Baukebrenninkmeijer/table-evaluator
[5] http://shakedzy.xyz/dython/modules/nominal/
[6] https://docs.scipy.org/

terms of size (spanning 30-199 thousand samples), column heterogeneity, and distinct characteristics (Refer to Table 1 and Appendix B for details). Our investigation spans a geometric progression of sample sizes, extending from 40 to the full dataset size (notated as "all"), to emulate a range from low to high resource settings. In line with existing studies [22], models were trained for 300 epochs. The evaluations were conducted on a separate test set that was never used during the whole training process of the tabular data generators. To account for potential randomness, experiments were conducted over three different random seeds for model training and repeated across five trials for generating synthetic datasets. All the code was implemented in the Python and all experiments are conducted on a single Titan RTX GPU. See details in Appendix A.

7.1 Correlation of Metrics

In order to evaluate the tabular data generators comprehensively across various dimensions, we conducted a thorough correlation analysis of the metrics discussed in Section 5, as illustrated in Figure 1. Our analysis revealed a significant degree of correlation among metrics within each dimension. This was particularly evident for the marginal-based metrics, with Pearson coefficients ranging from 0.74 to 0.98. Consequently, any of these metrics could effectively represent their respective dimension. To ensure clarity and computational efficiency, we specifically selected the **efficacy test (machine learning efficacy)**, **closeness approximation (distance to closest record)**, **associations difference**, and **histogram intersection** as representative metrics for summarizing each crucial dimension discussed in Section 5.

7.2 Performance Comparison

Downstream Task Utility. The *efficacy score* measures the utility of synthetic data in downstream tasks, as illustrated in the top-left plot of Figure 2. In the best-case scenario (marked as "all" in x-axis), the performances of `CTGAN`, `TVAE`, and `MargCTGAN` are comparable. Performance generally degrades in low-sample settings, with the most significant drop around the size of 640. This decline is particularly notable for `CTGAN`, which exhibits a relative error up to 57%. While `TVAE` generally outperforms the other models across varying sample sizes, our `MargCTGAN` performs robustly, demonstrating particular advantages in low-sample settings. Notably, `MargCTGAN` consistently outperforms its backbone model, `CTGAN`, across all settings.

Joint Fidelity & Memorization. The *distance to the closest record* metric, depicted in the bottom-left subplot of Figure 2, measures the alignment between the real and synthetic joint distribution and simultaneously illustrates the memorization effects of the generators. Striking a balance is crucial as over-memorization might compromise privacy. `TableGAN` consistently maintains the most substantial distance from the real data reference, aligning with its design objective of privacy

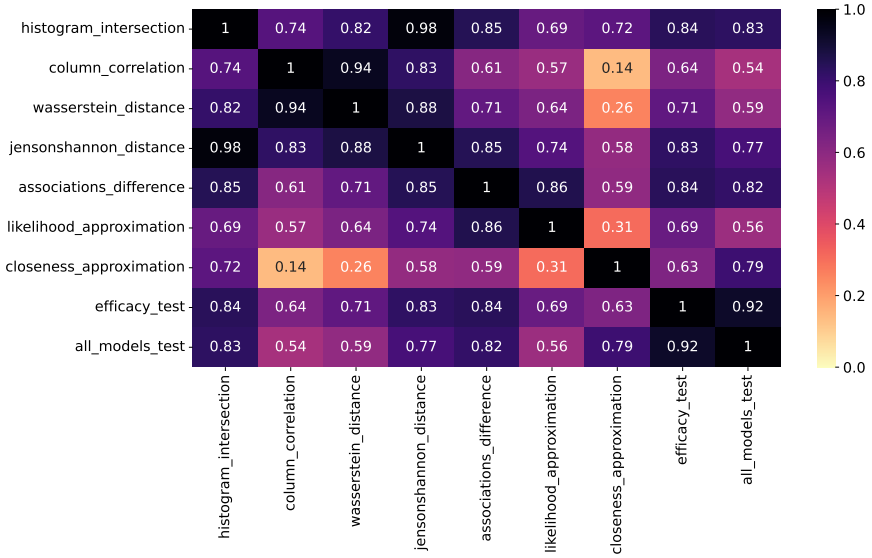

Fig. 1: Pearson correlation coefficients (in absolute value) among different metrics across multiple experimental trials on all datasets.

preservation. Conversely, `TVAE` displays the closest proximity, even exceeding the real reference, indicating a potential overfitting risk and privacy leakage. This may be attributed to its use of reconstruction loss in its training objective [3]. As the training size reduces, the distance between the synthetic and real data first increases then decreases, potentially signifying the generator's shift from generalization to memorization. While both `CTGAN` and `MargCTGAN` maintain a moderate distance from real data, our `MargCTGAN` generally demonstrates a closer proximity to the reference, presenting an appropriate balance between alignment and privacy protection.

Pairwise Correlation. The *association difference* metric (bottom-right subplot in Figure 2) quantifies the disparity between the correlation matrices of the real and synthetic data. As expected, this disparity increases as the sample size decreases, a trend also seen in the real data reference. This could be attributed to data diversity, where different smaller subsets might not retain the same statistical characteristics while the sampling randomness is accounted for in our repeated experiments. Among all models, `TableGAN` exhibits the largest associations difference score, particularly in the low-sample regime, indicating challenges in capturing associations with limited training samples. Both `MargCTGAN` and `TVAE` display similar behavior, with our `MargCTGAN` following the trend of real data reference more precisely, specifically in low-sample settings.

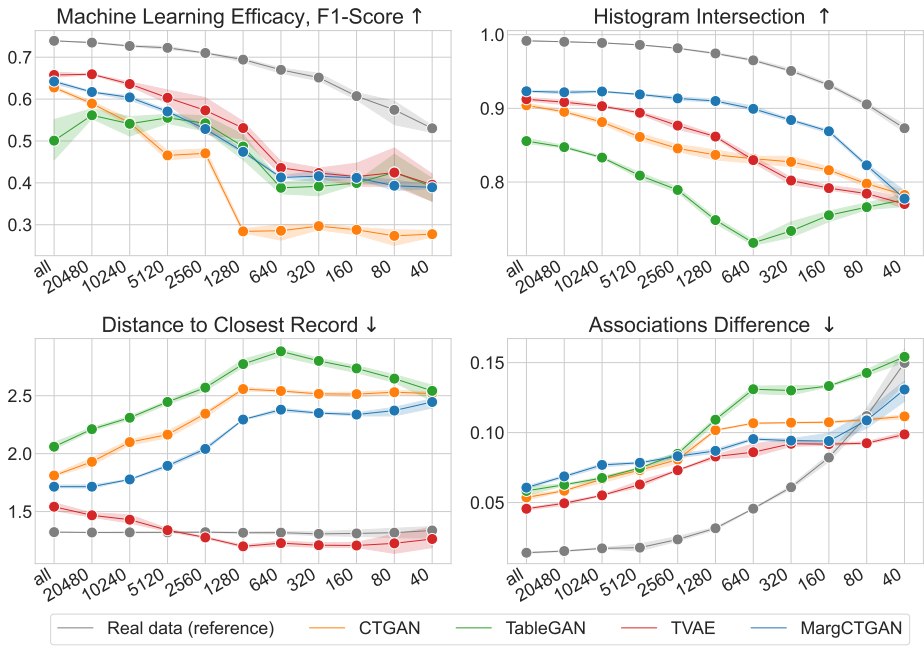

Fig. 2: Averaged score across datasets. The **X-axis** represents the size of the training dataset, with "all" indicating the full dataset size. **Real data (reference)** corresponds to the metrics directly measured on the real (train vs. test) data, serving as the reference (oracle score) for optimal performance.

Marginal Matching. The *histogram intersection* metric, depicted in the top-right subfigure in Figure 2, assessing the overlap of real and synthetic marginal distributions. Our moment matching objective within `MargCTGAN` explicitly encourages such coverage of low-level statistics, leading to consistent superior performance of `MargCTGAN` across various settings. A more detailed analysis, presented in Figure 3, reveals performance differences across numerical and categorical columns. Here, `TVAE` demonstrates good performance with numerical attributes but exhibits limitations in handling categorical ones, whereas `CTGAN` excels in handling categorical columns, possibly due to its training-by-sampling approach, but falls short with the numerical ones. Notably, `MargCTGAN` balances both aspects, outperforming `CTGAN` in numerical columns while matching its performance in categorical ones. Moreover, while most models show decreased performance in low-resource settings, `TableGAN` exhibits improvement, potentially due to its similar moment matching approach to ours, thereby further validating our design choice.

Insights into Performance and Behavior of Histogram Intersection Metric. Figure 3 highlighted that `TVAE` excels in datasets with numerical columns but struggles with categorical columns, resulting in subpar marginal performance due

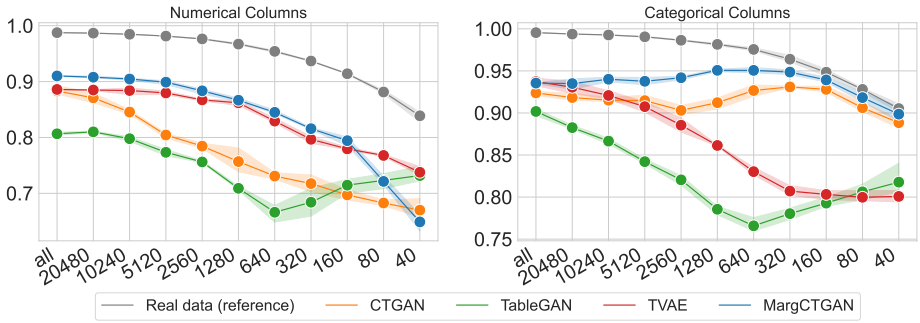

Fig. 3: Histogram intersection score for numerical and categorical columns respectively, which is averaged across datasets. The **X-axis** represents the size of the training dataset, with "all" indicating the full dataset size. **Real data (reference)** corresponds to the metrics directly measured on the real (train vs. test) data, serving as the reference (oracle score) for optimal performance.

to its inability to accurately reproduce different categories. Surprisingly, despite this limitation, the synthetic datasets generated by `TVAE` showed high utility in downstream tasks, suggesting that a good marginal distribution is not always a prerequisite for usefulness. On the other hand, `CTGAN` demonstrated superior performance in capturing associations within categorical columns, showcasing the effectiveness of its training-by-sampling approach. Notably, `MargCTGAN` achieved similar performance to `CTGAN` in categorical columns while outperforming it in numerical columns, aligning more closely with `TVAE` in this aspect.

It is important to note that all models experienced degraded performance in low-resource settings. However, interestingly, `TableGAN` exhibited improved performance in such scenarios. This improvement may be attributed to its information loss, which share similar idea of our moment matching objective. Figures 5(a), 5(b), 5(c), 5(d) in Appendix C further shows the breakdown across the different datasets.

Ablation of Moment Matching in Raw Data Space. We conducted an additional ablation study to investigate the effect of the moment matching technique with and without applying PCA in `MargCTGAN`. As shown in Figure 4, while both moment matching without PCA (`CTGAN+Raw`) and with PCA (`MargCTGAN`) performs generally better than the baseline `CTGAN`, the PCA adopted in our `MargCTGAN` does provide additional notable improvement consistently across different metrics considered in our study.

8 Discussion

While our `MargCTGAN` shows consistent improvements over a broad range of settings, we discuss additional observations and limitations below. The ablation study examined the effect of the moment matching technique with and without

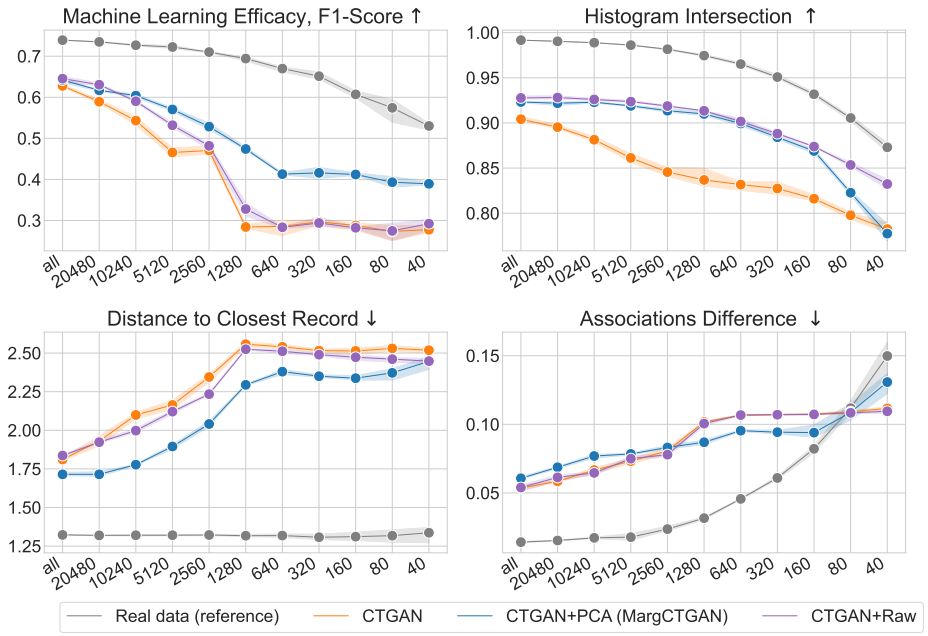

Fig. 4: Comparison between `CTGAN` trained with PCA loss objective (`MargCTGAN`) and `CTGAN` trained with raw moment matching loss objective. The **X-axis** represents the size of the training dataset, with "all" indicating the full dataset size. **Real data (reference)** corresponds to the metrics directly measured on the real (train vs. test) data, serving as the reference (oracle score) for optimal performance.

applying PCA in `MargCTGAN`. Both approaches outperformed the baseline `CTGAN`, but the PCA-moment matching in `MargCTGAN` provided notable improvement across different metrics. However, in extremely low-sample scenarios, `MargCTGAN` showed a performance drop compared to `CTGAN` in terms of capturing associations and reproducing marginal distributions. We attribute this to the rank-deficiency issue in the PCA-moment matching approach when the sample size is smaller than the number of features. In such cases, models like `CTGAN` with the raw feature moment matching (`CTGAN+Raw`) method may be more suitable. Understanding the strengths and weaknesses of different models under varying resource constraints helps in selecting the appropriate synthetic data generation approach.

9 Conclusion

In conclusion, our comprehensive evaluation of three popular tabular data generators across different dataset sizes underscores the importance of developing models that excel in low-sample regimes. Consequently, we propose `MargCTGAN`, an adaptation of the popular `CTGAN` model, which consistently exhibits performance improvements in various dataset sizes and setups. This further emphasizes

the significance of incorporating statistical moment matching techniques in the optimization process to enhance the model's learning capabilities. To ensure the impact and reproducibility of our work, we release our code and setup[7]. We hope that the availability of our evaluation framework will contribute to the advancement of the current state of evaluating tabular data generators and facilitate future research in this evolving field.

Acknowledgements

This work was partially funded by ELSA – European Lighthouse on Secure and Safe AI funded by the European Union under grant agreement No. 101070617. Additionally, this work is supported by the Helmholtz Association within the project "Protecting Genetic Data with Synthetic Cohorts from Deep Generative Models (PRO-GENE-GEN)" (ZT-I-PF-5-23) and Bundesministeriums für Bildung und Forschung "PriSyn" (16KISAO29K). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or European Commission. Neither the European Union nor the European Commission can be held responsible for them. Moreover, Dingfan Chen was partially supported by Qualcomm Innovation Fellowship Europe. We also thank the team from synthetic data vault project for open-sourcing their code.

References

1. Alaa, A., Van Breugel, B., Saveliev, E.S., van der Schaar, M.: How faithful is your synthetic data? sample-level metrics for evaluating and auditing generative models. In: International Conference on Machine Learning (ICML). PMLR (2022)
2. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein gan (2017)
3. Chen, D., Yu, N., Zhang, Y., Fritz, M.: Gan-leaks: A taxonomy of membership inference attacks against generative models. In: Proceedings of the 2020 ACM SIGSAC conference on computer and communications security (ACM CCS) (2020)
4. Chen, H., Jajodia, S., Liu, J., Park, N., Sokolov, V., Subrahmanian, V.: Faketables: Using gans to generate functional dependency preserving tables with bounded real data. In: IJCAI (2019)
5. Choi, E., Biswal, S., Malin, B., Duke, J., Stewart, W.F., Sun, J.: Generating multi-label discrete patient records using generative adversarial networks. In: Machine learning for healthcare conference. PMLR (2017)
6. Chundawat, V.S., Tarun, A.K., Mandal, M., Lahoti, M., Narang, P.: A universal metric for robust evaluation of synthetic tabular data. IEEE Transactions on Artificial Intelligence p. 1–11 (2022). https://doi.org/10.1109/TAI.2022.3229289
7. Dankar, F.K., Ibrahim, M.K., Ismail, L.: A multi-dimensional evaluation of synthetic data generators. IEEE Access **10** (2022)
8. Engelmann, J., Lessmann, S.: Conditional wasserstein gan-based oversampling of tabular data for imbalanced learning. Expert Systems with Applications **174** (2021)
9. Fernandes, K., Vinagre, P., Cortez, P.: A proactive intelligent decision support system for predicting the popularity of online news. In: Portuguese conference on artificial intelligence. Springer (2015)

[7] https://github.com/tejuafonja/margctgan/

10. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., Weinberger, K. (eds.) Advances in Neural Information Processing Systems. vol. 27. Curran Associates, Inc. (2014)
11. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of wasserstein gans. Advances in neural information processing systems (NeurIPS) **30** (2017)
12. Kaggle: 2018 kaggle machine learning & data science survey
13. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
14. Kohavi, R., Becker, B.: Uci machine learning repository: adult data set. UCI machine learning repository. Accessed: 2022-05-10 (1996)
15. Lane, T., Kohavi, R.: Census-income (kdd) data set. UCI machine learning repository. Accessed: 2022-05-10 (2010)
16. Lin, Z., Khetan, A., Fanti, G., Oh, S.: Pacgan: The power of two samples in generative adversarial networks (2018)
17. Ma, C., Tschiatschek, S., Turner, R., Hernández-Lobato, J.M., Zhang, C.: Vaem: a deep generative model for heterogeneous mixed type data. Advances in Neural Information Processing Systems (NeurIPS) **33** (2020)
18. Mottini, A., Lheritier, A., Acuna-Agost, R.: Airline passenger name record generation using generative adversarial networks. arXiv preprint arXiv:1807.06657 (2018)
19. Park, N., Mohammadi, M., Gorde, K., Jajodia, S., Park, H., Kim, Y.: Data synthesis based on generative adversarial networks. Proceedings of the VLDB Endowment (2018). https://doi.org/10.14778/3231751.3231757
20. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. In: Bengio, Y., LeCun, Y. (eds.) 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings (2016)
21. Theis, L., van den Oord, A., Bethge, M.: A note on the evaluation of generative models. In: International Conference on Learning Representations (ICLR 2016) (2016)
22. Xu, L., Skoularidou, M., Cuesta-Infante, A., Veeramachaneni, K.: Modeling tabular data using conditional gan. Advances in Neural Information Processing Systems (NeurIPS) **32** (2019)
23. Xu, L., Veeramachaneni, K.: Synthesizing tabular data using generative adversarial networks. arXiv preprint arXiv:1811.11264 (2018)
24. Yoon, J., Jordon, J., Schaar, M.: Radialgan: Leveraging multiple datasets to improve target-specific predictive models using generative adversarial networks. In: International Conference on Machine Learning (ICML). PMLR (2018)
25. Zhao, Z., Kunar, A., Birke, R., Chen, L.Y.: Ctab-gan: Effective table data synthesizing. In: Asian Conference on Machine Learning. PMLR (2021)