MargCTGAN: A “Marginally” Better CTGAN for the Low Sample Regime

Tejumade Afonja [1] **Dingfan Chen** [1] **Mario Fritz** [1]

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 state-of-the-art 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` models 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 decorrelated marginals, which results in a consistent improvement in downstream utility as well as statistical properties of the synthetic data.

1. Introduction

Tabular data, despite being the most widely used data type (Kaggle), 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 (Xu et al., 2019; Zhao et al., 2021), facilitating the safe sharing of personal records (Choi et al., 2017; Park et al., 2018; Mottini et al., 2018), and mitigating class imbalance (Engelmann & Lessmann, 2021). 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 (Alaa et al., 2022).

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 4). 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 state-of-the-art models. For instance, the popular `CTGAN` (Xu et al., 2019) model typically demonstrates high attribute fidelity but falls short in utility for low-data scenarios. Conversely, `TableGAN` (Park et al., 2018) 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.

2. Related Works

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 (Choi et al., 2017), continuous values (Mottini et al., 2018), and heterogeneous mixtures (Park et al., 2018; Xu & Veeramachaneni, 2018; Xu et al., 2019; Zhao et al., 2021). Notably, `TableGAN` (Park et al., 2018), `CTGAN` (Xu et al., 2019), and `TVAE` (Xu et al., 2019) stand out as state-of-the-art 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 (Chen et al., 2019; Ma et al., 2020; Yoon et al., 2018). Our work aims to fill this

[1] CISPA Helmholtz Center for Information Security, Saarbrücken, Germany. Correspondence to: Tejumade Afonja <tejumade.afonja@cispa.de>.

Workshop on Challenges in Deployable Generative AI at International Conference on Machine Learning (ICML), Honolulu, Hawaii, USA. 2023. Copyright 2023 by the author(s).

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.

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 (Theis et al., 2016). Recent studies have introduced a variety of metrics: prominent among these are downstream machine learning efficacy, unified metrics evaluation (Chundawat et al., 2022), and evaluations focusing on distinct aspects (Alaa et al., 2022; Dankar et al., 2022). 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. Method: `MargCTGAN`

`MargCTGAN` adheres to the standard Generative Adversarial Networks (GANs) paradigm (Goodfellow et al., 2014), 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 (Gulrajani et al., 2017) 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:

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

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` (Xu et al., 2019) framework, we adopt several key techniques to adapt GAN models for tabular data. Firstly, one-hot encoding is applied to pre-process 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 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 6), likely because the decorrelated feature representation supports the independent moment matching.

Formally,

$$\mathcal{L}_{\text{mean}} = \Big\| \underset{\boldsymbol{x}\sim p_{\text{data}}}{\mathbb{E}}\big[f\left(\boldsymbol{x}\right)\big] - \underset{\boldsymbol{z}\sim p_{\boldsymbol{z}}}{\mathbb{E}}\big[f\big(\mathcal{G}(\boldsymbol{z})\big)\big]\Big\|_2 \tag{1}$$

$$\mathcal{L}_{\text{std}} = \Big\| \underset{\boldsymbol{x}\sim p_{\text{data}}}{\mathbb{SD}}\big[f(\boldsymbol{x})\big] - \underset{\boldsymbol{z}\sim p_{\boldsymbol{z}}}{\mathbb{SD}}\big[f\big(\mathcal{G}(\boldsymbol{z})\big)\big]\Big\|_2 \tag{2}$$

$$\mathcal{L}_{\text{marg}} = \mathcal{L}_{\text{mean}} + \mathcal{L}_{\text{std}} \tag{3}$$

$$\mathcal{L}^{\mathcal{G}} = -\mathcal{L}_{\text{WGP}} + \mathcal{L}_{\text{cond}} + \mathcal{L}_{\text{marg}} \tag{4}$$

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.

4. Multi-Dimensional Evaluation Metrics

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

Application 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. A key prerequisite for accurately replicating the real data distribution is to ensure a match in the distribution of each individual column, i.e., aligning the marginals. The evaluation of this criterion essentially involves quantifying the disparity between two one-dimensional variables. Commonly used metrics for this purpose include the *Jensen-Shannon divergence*, *Wasserstein distance*, *column correlation*, and *histogram intersection*. While the divergence measures are directly computed for categorical attributes, numerical columns typically undergo a pre-processing discretization step via binning prior to divergence computation to ensure tractability.

5. Experiments

Setup. We conducted evaluations on four benchmark tabular datasets: Adult (Kohavi & Becker, 1996), Census (Lane & Kohavi, 2010), News (Fernandes et al., 2015), and Texas. These datasets exhibit diverse properties in terms of size (spanning 30-110 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 “-1”), to emulate a range from low to high resource settings. In line with existing studies (Xu et al., 2019), 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. See details in Appendix A.

5.1. Correlation of Metrics

We begin by a thorough correlation analysis of various metrics discussed in Section 4, as illustrated in Figure 1. Our observations revealed that metrics within each dimension generally exhibit a high degree of correlation. This is particularly evident in the case of the marginal-based metrics, as demonstrated by Pearson coefficients ranging between 0.74 and 0.98. Consequently, any of these metrics could adequately represent their respective dimension. For the sake of clarity and computational efficiency, we specifically chose to employ the **efficacy test (machine learning efficacy)**, **closeness approximation (distance to closest record)**, **associations difference**, and **histogram intersection** as representative metrics for summarizing each vital dimension discussed in Section 4.

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

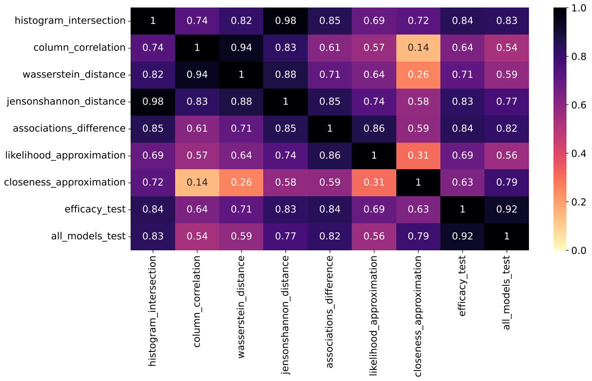


Figure 1: Pearson correlation coefficients (in absolute value) among different metrics across multiple experimental trials on all datasets.

5.2. Methods Comparison

Downstream Utility. The *efficacy score* measures the utility of synthetic data in downstream tasks, as illustrated in the left-most plot of Figure 2. In the best-case scenario (marked as “-1” 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 second subplot of Figure 2, measures the alignment between the real and synthetic joint

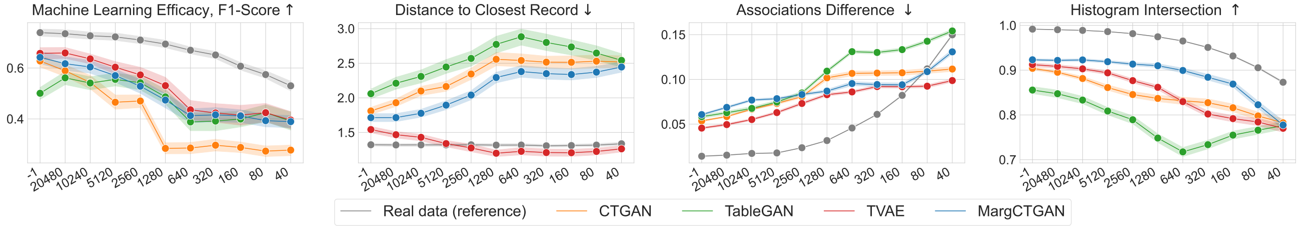


Figure 2: Averaged score across datasets. The **X-axis** represents the size of the training dataset, with "-1" indicating the full dataset size. **Real data (reference)** corresponds to the metrics directly measured on the real (train vs. test) data, serving as the reference (oracle score) for optimal performance.

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 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 (Chen et al., 2020). 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 (third 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.

Marginal Matching. The *histogram intersection* metric, depicted in the right-most 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.

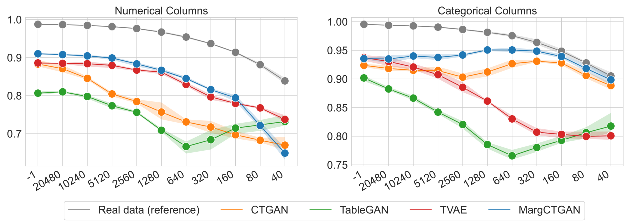


Figure 3: Histogram intersection score for numerical and categorical columns respectively, which is averaged across datasets.

6. Conclusion

In conclusion, our comprehensive evaluation of state-of-the-art synthetic tabular data generators provided valuable insights, particularly regarding their performance in low-resource settings. By introducing the MargCTGAN model as an improvement over CTGAN, we addressed some of the limitations observed. Our results demonstrated that MargCTGAN outperformed CTGAN across multiple metrics, exhibiting stronger utility and marginal-based performance. These findings highlight the effectiveness of MargCTGAN in enhancing both the utility and marginal score of the generated synthetic data. Our code and setup is published to ensure reproducibility and increase impact[1].

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

Acknowledgement

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.

References

Alaa, A., Van Breugel, B., Saveliev, E. S., and 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.

Chen, D., Yu, N., Zhang, Y., and 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.

Chen, H., Jajodia, S., Liu, J., Park, N., Sokolov, V., and Subrahmanian, V. Faketables: Using gans to generate functional dependency preserving tables with bounded real data. In *IJCAI*, 2019.

Choi, E., Biswal, S., Malin, B., Duke, J., Stewart, W. F., and Sun, J. Generating multi-label discrete patient records using generative adversarial networks. In *Machine learning for healthcare conference*. PMLR, 2017.

Chundawat, V. S., Tarun, A. K., Mandal, M., Lahoti, M., and Narang, P. A universal metric for robust evaluation of synthetic tabular data. *IEEE Transactions on Artificial Intelligence*, pp. 1–11, 2022. doi: 10.1109/TAI.2022.3229289.

Dankar, F. K., Ibrahim, M. K., and Ismail, L. A multi-dimensional evaluation of synthetic data generators. *IEEE Access*, 10, 2022.

Engelmann, J. and Lessmann, S. Conditional wasserstein gan-based oversampling of tabular data for imbalanced learning. *Expert Systems with Applications*, 174, 2021.

Fernandes, K., Vinagre, P., and Cortez, P. A proactive intelligent decision support system for predicting the popularity of online news. In *Portuguese conference on artificial intelligence*. Springer, 2015.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K. (eds.), *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. Improved training of wasserstein gans. *Advances in neural information processing systems (NeurIPS)*, 30, 2017.

Kaggle. 2018 kaggle machine learning & data science survey.

Karlsson, A. and Sjöberg, T. Synthesis of tabular financial data using generative adversarial networks, 2020.

Kohavi, R. and Becker, B. Uci machine learning repository: adult data set. *UCI machine learning repository. Accessed: 2022-05-10*, 1996.

Lane, T. and Kohavi, R. Census-income (kdd) data set. *UCI machine learning repository. Accessed: 2022-05-10*, 2010.

Ma, C., Tschiatschek, S., Turner, R., Hernández-Lobato, J. M., and Zhang, C. Vaem: a deep generative model for heterogeneous mixed type data. *Advances in Neural Information Processing Systems (NeurIPS)*, 33, 2020.

Mottini, A., Lheritier, A., and Acuna-Agost, R. Airline passenger name record generation using generative adversarial networks. *arXiv preprint arXiv:1807.06657*, 2018.

Park, N., Mohammadi, M., Gorde, K., Jajodia, S., Park, H., and Kim, Y. Data synthesis based on generative adversarial networks. *Proceedings of the VLDB Endowment*, 2018. doi: 10.14778/3231751.3231757.

Stadler, T., Oprisanu, B., and Troncoso, C. Synthetic data–anonymisation groundhog day. In *31st USENIX Security Symposium (USENIX Security 22)*, 2022.

Theis, L., van den Oord, A., and Bethge, M. A note on the evaluation of generative models. In *International Conference on Learning Representations (ICLR 2016)*, 2016.

Xu, L. and Veeramachaneni, K. Synthesizing tabular data using generative adversarial networks. *arXiv preprint arXiv:1811.11264*, 2018.

Xu, L., Skoularidou, M., Cuesta-Infante, A., and Veeramachaneni, K. Modeling tabular data using conditional gan. *Advances in Neural Information Processing Systems (NeurIPS)*, 32, 2019.

Yoon, J., Jordon, J., and 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.

Zhao, Z., Kunar, A., Birke, R., and Chen, L. Y. Ctab-gan: Effective table data synthesizing. In *Asian Conference on Machine Learning*. PMLR, 2021.

A. Implementation Details

All the code was implemented in the Python and all experiments are conducted on a single Titan RTX GPU.

A.1. Metrics

Application Utility. For the *machine learning efficacy* and *all models test* metrics, we implemented them using the SDMetrics package [2]. We trained logistic regression, decision tree classifier, and multilayer perceptron models for the classification tasks, and linear regression, decision tree regressor, and multilayer perceptron models for the regression tasks. We kept the default hyperparameter settings. For the classification models, we standardized the numerical columns during training, while for the regression models, no normalization scheme was applied as it yielded more stable results without it. The categorical columns for both regression and classification tasks were one-hot encoded. Each classification model was evaluated using the F1-score while each regression model is evaluated with R^2-score. The R^2-score has a range of $[-\infty, 1]$, so we normalized it to $[0, 1]$ using the normalization scheme from the SDMetrics package.

Joint Fidelity. We applied a min-max normalization scheme to all numerical columns to constrain their value range between 0 and 1. Additionally, we one-hot encoded the categorical columns. We used the scikit-learn nearest-neighbor implementation [3] and chose the number of nearest neighbors in the range of $[1, 2, 3, \ldots, 9]$ while using the Euclidean distance. 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. We use the default hyperparameters from the package.

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. We applied a min-max normalization scheme to all numerical columns to constrain their value range between 0 and 1, and label-encoded the categorical columns. The marginal metrics were applied to each column in the dataset. To speed up the calculation, we parallelized the process with joblib [6]. The *histogram intersection*, *Wasserstein distance*, and *Jenson-Shannon distance* used the same binning strategy for the numerical columns. We assumed a uniform grid between 0 and 1 and computed the bin width $\triangle_i$ for the real data (train and test) using a pre-defined bin sizes $\in \{25, 50, 100\}$. We also used the real data (train and test) to determine the number of categories for each categorical column. The categorical columns has $\triangle_i = 1$. The normalized count for each bin was calculated for the synthetic and real data. We used the SciPy package [7] to calculate *Wasserstein distance* by passing the grids as empirical values and the normalized bin values as weights, and to calculate *Jenson-Shannon distance* with the base=2 setting. We used the dython package to compute *column correlation* metric. As for the *histogram intersection* metric, we implemented it ourselves. Unless otherwise specified, the hyperparameter settings were taken from the packages.

A.2. Generative Models

We used the open-source implementation provided by the authors for `CTGAN` and `TVAE` (Xu et al., 2019) and the SDGym package[8] for `TableGAN`. For the `MargCTGAN`, we modified the `CTGAN` model and follow the default hyperparameter setting. We use the same number of training epochs and batch size for all models. The data generators were trained on each

[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

[4] https://github.com/Baukebrenninkmeijer/table-evaluator

[5] http://shakedzy.xyz/dython/modules/nominal/

[6] https://joblib.readthedocs.io/en/latest/

[7] https://docs.scipy.org/

[8] https://github.com/sdv-dev/SDGym/tree/41ffcc1982e2c5081271ce0e138792550c2a3480/sdgym/synthesizers

subset of the datasets, and after training, the synthetic datasets were sampled. To ensure robustness, we repeated this process three times using different random seeds.

To evaluate the performance of the generators, we sampled five synthetic datasets with 20,000 samples from each data synthesizer. We fitted utility and joint-based metrics on the synthetic datasets and then evaluated the fitted model against the real test dataset. For the column-pair-based and marginal-based metrics, we first computed statistics for the synthetic datasets and real test dataset separately and then compared the statistics for evaluation. For the real data reference (always shown as gray lines in the figures), we run the same evaluation process as above but replace the synthetic set by the real training set.

B. Datasets

Adult [9]**.** The UCI Adult dataset (Kohavi & Becker, 1996) is from the 1994 U.S. Census and contains 14 attributes with a total sample size of 48,000. The associated task is a binary classification to determine if an individual earns more than 50,000 dollars a year.

Census [10]**.** The UCI Census-income dataset (Lane & Kohavi, 2010) is a weighted dataset of the 1994-5 U.S. Census conducted by the U.S. Census Bureau. It includes over 299,000 samples and 40 attributes with an associated binary classification task of determining whether a person earns more than 50,000 dollars a year.

News [11]**.** The UCI Online News Popularity dataset (Fernandes et al., 2015) summarizes a heterogeneous set of features about articles published by Mashable over a two-year period. It includes over 39,000 instances with 58 attributes whose goal is to predict the number of shares on social networks (we dropped the non-predictive attributes). It is a regression task, but it can be transformed into a classification task. As executed in Fernandes et al. (2015), we assume a binary classification task, where an article is considered “popular” if the number of shares is higher than a fixed decision threshold t, else it is “unpopular”. $t = 1400$.

Texas [12]**.** The Texas Hospital Discharge dataset is a large public use data file provided by the Texas Department of State Health Services [13]. We used the preprocessed version from Stadler et al. (2022), which consists of 100,000 records uniformly selected from a pre-processed file containing patient data from 2013. We retain 18 attributes and assume a binary classification task by predicting only minor and major mortality risk. The final size of the dataset was therefore reduced to 75,159.

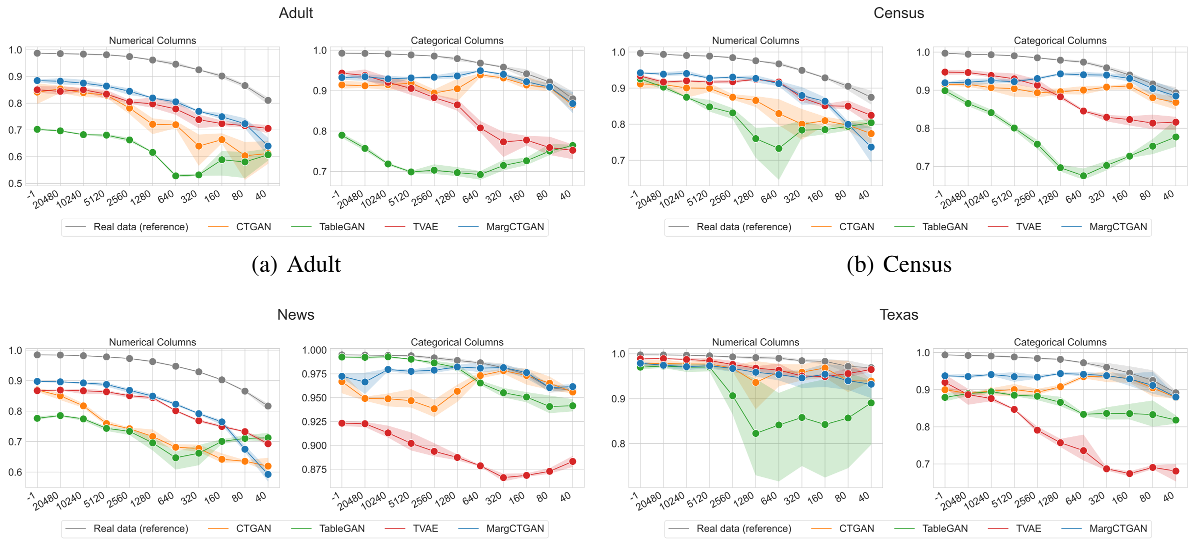
C. Extended Results

C.1. 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 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 4(a), 4(b), 4(c), 4(d) further shows the breakdown across the different datasets.

[9] https://archive.ics.uci.edu/ml/datasets/adult
[10] https://archive.ics.uci.edu/ml/datasets/US+Census+Data+%281990%29
[11] https://archive.ics.uci.edu/ml/datasets/online+news+popularity
[12] https://github.com/spring-epfl/synthetic_data_release/blob/master/data/texas.csv
[13] https://www.dshs.texas.gov/thcic/



(a) Adult

(b) Census

(c) News

(d) Texas

Figure 4: Histogram intersection score on Adult 4(a), Census 4(b), News 4(c), and Texas 4(d) dataset, respectively.

C.2. Relative Error

We additionally provide the **relative error** w.r.t. *real data (reference)* for each metric (averaged across datasets) in Table 2, 3, 4, and 5.

subset	-1	20480	10240	5120	2560	1280	640	320	160	80	40
TableGAN	32.24	23.61	25.55	23.13	23.65	29.90	42.05	39.89	34.19	26.03	26.21
TVAE	11.03	10.29	12.49	16.50	19.30	23.54	34.96	34.99	31.65	26.08	25.40
CTGAN	15.07	19.81	25.25	35.55	33.75	59.07	57.35	54.45	52.57	52.38	47.66
MargCTGAN	13.14	16.04	16.89	21.05	25.56	31.72	38.36	36.14	32.13	31.55	26.57

Table 2: Averaged relative error of *machine learning efficacy*. The lower the better.

subset	-1	20480	10240	5120	2560	1280	640	320	160	80	40
TableGAN	-55.76	-67.57	-74.93	-85.25	-94.39	-110.51	-118.69	-114.20	-108.63	-100.92	-90.18
TVAE	-16.59	-11.23	-8.28	-1.41	3.44	8.92	6.89	7.49	7.95	6.99	5.49
CTGAN	-36.92	-46.24	-58.97	-63.89	-77.39	-94.15	-92.71	-92.43	-91.71	-92.02	-88.41
MargCTGAN	-29.65	-29.96	-34.60	-43.51	-54.38	-74.16	-80.52	-79.75	-78.25	-80.00	-82.99

Table 3: Averaged relative error of *closeness approximation*. Generally, a lower value is considered better. However, an excessively low value may suggest memorization.

subset	-1	20480	10240	5120	2560	1280	640	320	160	80	40
TableGAN	-312.05	-308.51	-291.45	-319.78	-258.71	-245.16	-186.87	-113.52	-62.23	-27.60	-2.90
TVAE	-221.95	-222.38	-219.46	-253.43	-209.10	-161.99	-88.33	-51.00	-11.69	17.34	34.10
CTGAN	-278.87	-280.84	-286.25	-311.11	-241.34	-221.51	-133.71	-75.76	-30.71	2.29	25.52
MargCTGAN	-328.64	-347.45	-345.54	-340.73	-251.48	-174.78	-108.87	-54.71	-14.36	2.76	12.69

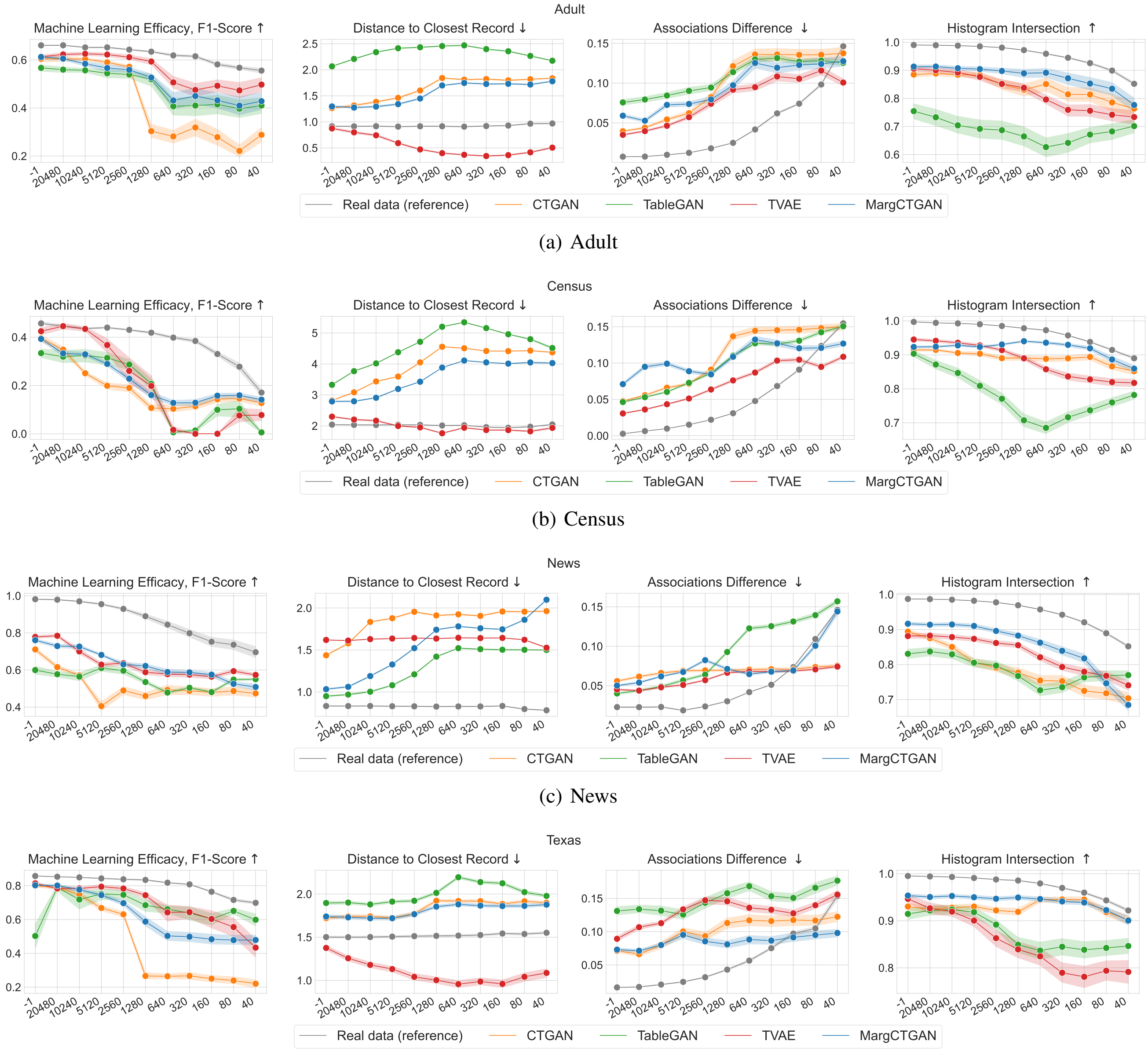
Table 4: Averaged relative error of *associations difference*. The lower the better.

subset	-1	20480	10240	5120	2560	1280	640	320	160	80	40
`TableGAN`	13.72	14.44	15.74	17.98	19.59	23.21	25.66	22.84	18.98	15.40	11.10
`TVAE`	7.99	8.27	8.68	9.34	10.69	11.58	14.03	15.64	15.03	13.38	11.78
`CTGAN`	8.83	9.60	10.87	12.67	13.85	14.13	13.83	12.98	12.42	11.89	10.38
`MargCTGAN`	6.90	6.92	6.67	6.82	6.93	6.63	6.82	7.01	6.74	9.13	10.94

Table 5: Averaged relative error of *histogram intersection*. The lower the better.

C.3. Metrics Results for Different Datasets

Figure 5(a), 5(b), 5(c), and 5(d) show the results of metric evaluation on each dataset respectively.



(a) Adult

(b) Census

(c) News

(d) Texas

Figure 5: Metric evaluation results on each dataset, which is supplementary to Figure 2 in the main paper.

D. Ablation

D.1. 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 6, 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.

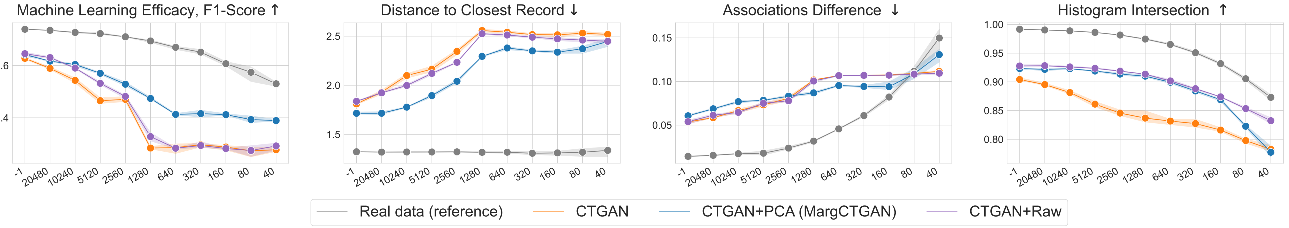
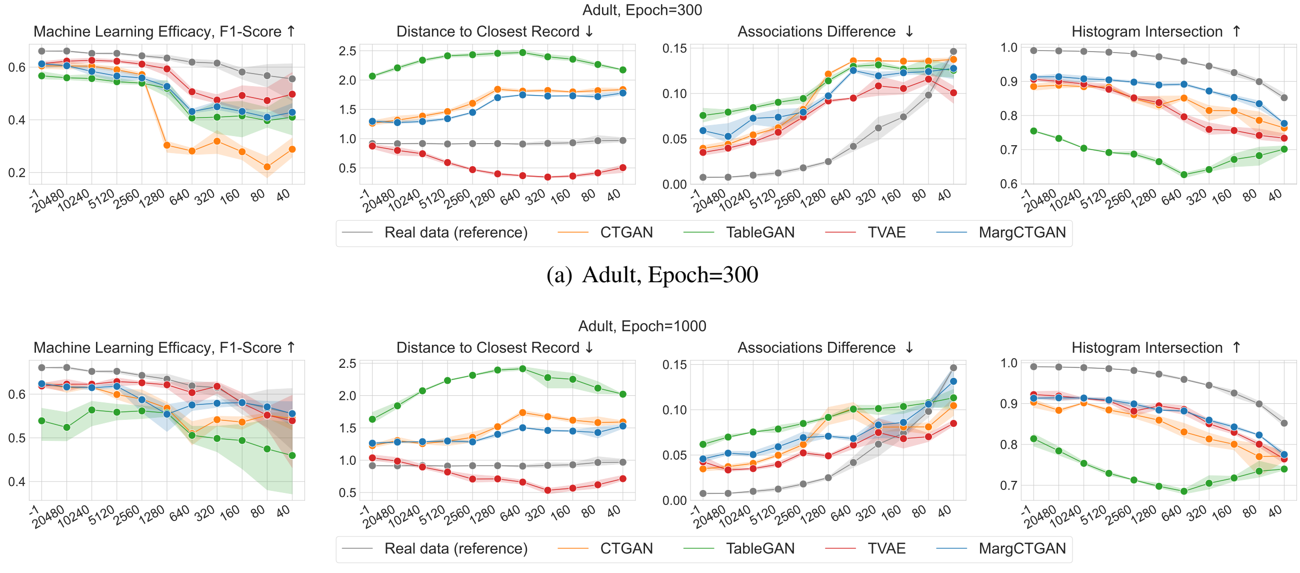


Figure 6: Comparison between CTGAN trained with PCA loss objective (MargCTGAN) and CTGAN trained with raw moment matching loss objective.

D.2. Training for Longer Epoch

Previous work (Karlsson & Sjöberg, 2020) has shown that the performance of CTGAN generally improves with further training, and we validate this finding in our experiments. Nevertheless, we limited the training epochs to 300 to align with the specified hyperparameters by the authors, considering the common usage of these models as off-the-shelf solutions without extensive tuning. Despite this constraint, our MargCTGAN model consistently outperforms CTGAN as shown in Figure 7(a), and 7(b). This highlights the enhanced performance of MargCTGAN in comparison to CTGAN in synthetic data generation tasks.



(a) Adult, Epoch=300

(b) Adult, Epoch=1000

Figure 7: Comparison showing performance of Adult dataset trained for 300 (top) and 1000 (bottom) epochs.

E. Limitations

The evaluation of *associations difference* uncovered insightful observation regarding the performance of MargCTGAN in low-resource settings. As the sample size decreased to 40, the performance of MargCTGAN exhibited degradation, while CTGAN demonstrated better results for this metric. A similar trend was observed for the marginal-based metric. This behavior can be attributed to the inherent characteristics of the PCA-moment matching loss method employed by MargCTGAN. The effective number of PCA components used in the transformation is determined by the minimum value between the size of the data and the size of the feature. In other words, the method can accurately capture features up to the limit set by the smaller value of these two factors. Consequently, in scenarios where the sample size is smaller than the number of features, the PCA-moment matching approach becomes less competitive (due to the rank-deficiency during the matrix computation). This limitation arises from the method's inability to effectively match and reproduce the characteristics of all features due to the constrained number of components available. Consequently, the performance of MargCTGAN deteriorates in terms of both the associations difference metric and the marginal-based metric. Supporting evidence for this limitation can be found in the experiment conducted with raw features (see Appendix D.1). It is crucial to consider this constraint when selecting an appropriate model for synthetic data generation. In scenarios where the sample size is significantly smaller than the number of features, alternative models such as CTGAN+Raw may offer better performance in capturing associations and reproducing marginal distributions. Understanding the strengths and weaknesses of different models under varying resource constraints empowers researchers and practitioners to make informed decisions and select the most suitable synthetic data generation approach for their specific requirements.