Generation of Synthetic Tabular Healthcare Data Using Generative Adversarial Networks

Alireza Hossein Zadeh Nik[1,2], Michael A. Riegler[1,3(✉)], Pål Halvorsen[1,4], and Andrea M. Storås[1,4]

[1] SimulaMet, Oslo, Norway
michael@simula.no
[2] University of Stavanger, Stavanger, Norway
[3] University of Tromsø, Tromsø, Norway
[4] OsloMet, Oslo, Norway

Abstract. High-quality tabular data is a crucial requirement for developing data-driven applications, especially healthcare-related ones, because most of the data nowadays collected in this context is in tabular form. However, strict data protection laws complicates the access to medical datasets. Thus, synthetic data has become an ideal alternative for data scientists and healthcare professionals to circumvent such hurdles. Although many healthcare institutions still use the classical de-identification and anonymization techniques for generating synthetic data, deep learning-based generative models such as generative adversarial networks (GANs) have shown a remarkable performance in generating tabular datasets with complex structures. This paper examines the GANs' potential and applicability within the healthcare industry, which often faces serious challenges with insufficient training data and patient records sensitivity. We investigate several state-of-the-art GAN-based models proposed for tabular synthetic data generation. Healthcare datasets with different sizes, numbers of variables, column data types, feature distributions, and inter-variable correlations are examined. Moreover, a comprehensive evaluation framework is defined to evaluate the quality of the synthetic records and the viability of each model in preserving the patients' privacy. The results indicate that the proposed models can generate synthetic datasets that maintain the statistical characteristics, model compatibility and privacy of the original data. Moreover, synthetic tabular healthcare datasets can be a viable option in many data-driven applications. However, there is still room for further improvements in designing a perfect architecture for generating synthetic tabular data.

Keywords: Synthetic data generation · Deep learning · Medical data

D.-T. Dang-Nguyen et al. (Eds.): MMM 2023, LNCS 13833, pp. 434–446, 2023.
https://doi.org/10.1007/978-3-031-27077-2_34

1 Introduction

The use of machine learning (ML) in medicine has shown promising results to solve different tasks such as automatic detection of gastrointestinal diseases [1], radiology applications [2] and mental health [3]. However, most ML models are known to be 'data-hungry', meaning that they should be developed on a large amount of data in order to perform well. In the healthcare domain, access to large datasets can be challenging because of privacy protection regulations and lack of data, e.g., due to rare or neglected diseases [4,5]. Consequently, synthetic data generation is considered an attractive alternative in order to create vast amounts of data [6,7]. Synthetic data generation aims to synthesize new data through automated processes that preserve the underlying structure and statistical properties of the original sensitive data to prevent people's privacy from being compromised. In the medical field, synthetic health data can help medical practitioners to share data without any privacy violations and use the synthetic data in addition to the original health data itself [8]. Most of the time, synthetic data generation is focused on imaging data. Nevertheless, much of the healthcare data, including electronic healthcare record (EHR)s, are collected in tabular form or as time-series data (e.g., from biomedical sensors).

While several data generation methods have been proposed, e.g., variational auto-encoders [9] and probabilistic Bayesian networks [10], generative adversarial network (GAN)s have shown excellent results. A GAN consists of two competing models referred to as the generator and the discriminator [11]. The generator tries to learn how to generate synthetic data that resembles the original data. The discriminator, on the other hand, tries to distinguish synthetic samples generated by the generator from original samples. During training of the GAN, the generator and the discriminator compete against each other, which ideally leads to improved performance for both models [11]. After training, the generator can be applied to generate synthetic data. Although synthetic data generation has become very popular, the real effect of the generated data is not well understood and researched yet.

Consequently, this paper focuses on generation and evaluation of synthetic tabular healthcare data. We train different types of GANs developed for tabular data generation on healthcare datasets of different sizes and with different numbers of variables, column data types, feature distributions and inter-variable correlations. After training, the resulting GANs are used to generate synthetic tabular data. Moreover, a comprehensive evaluation framework is defined to evaluate the quality of the synthetic records and the viability of each model in preserving the patients' privacy. We evaluate the strengths and weaknesses of each model based on statistical similarity metrics, ML-based evaluation scores, and distance-based privacy metrics.

The rest of this paper is organized as follows: Sect. 2 presents related work. The method and data is outlined in Sect. 3, and the results are included in Sect. 4. Section 5 provides a discussion of the findings, and conclusions are drawn in Sect. 6.

2 Background and Related Work

The development of deep learning-based tabular synthetic data generative models has been an active research area for the scientific community in recent years. Specifically, a plethora of publications has proposed GAN-based generative models for synthesizing tabular data. MedGAN, developed by Choi et al. [12], is one of the first architectures designed to generate discrete aggregated healthcare patient records. The authors proposed using a pre-trained auto-encoder to circumvent the problems of generating discrete values in GANs. While the generator learns the continuous latent codes in the training process, the pre-trained decoder, placed between the generator and discriminator, translates the generator's output to the original data format and passes it to the discriminator. TableGAN is one of the first GAN-based models developed to simultaneously generate tabular datasets containing both numerical and categorical columns [13]. The generator and discriminator in this tabular synthesizer are adopted based on deep convolutional neural networks to capture inter-variable dependencies between columns. Moreover, an auxiliary classifier is incorporated into the training process to increase the semantic integrity of the generated samples. Lei Xu et al. introduced tabular GAN (TGAN) [14] and conditional tabular GAN (CTGAN) [15] to create high-quality tabular datasets of different data types. In TGAN the generator is a recurrent neural network with long-short-term memory (LSTM), while the discriminator is a multilayer perceptron (MLP). On the other hand, CTGAN uses a conditional generator and a training-by-sampling technique to tackle the challenge of generating imbalanced data. Both CTGAN and TGAN models use a mode-specific normalization technique to deal with the complexity of generating multi-modal numerical columns. However, in contrast to the TGAN architecture, the generator in CTGAN is a fully connected neural network. Zhao et al. [16] adopted the core features of CTGAN and TableGAN models to handle the highly imbalanced categorical features and to improve generating skewed multi-modal and long-tailed continuous columns. They proposed conditional TableGAN (CTABGAN) to model tabular datasets of mixed types, including categorical and continuous features. Not only does CTABGAN use a novel conditional generator, but it also uses classification, information and generator losses in the training process.

Recently, the research community has focused on protecting the generative models against malicious attacks compromising the privacy and integrity of sensitive medical information. Several research papers, such as [17–20], proposed using differential privacy in the generation process to circumvent this. However, in complex use cases, it has been demonstrated that the quality of the synthetic records would decrease significantly when the noise is added in the generation process to ensure differential privacy constraints. For elaboration on GANs for tabular healthcare data generation, the interested reader is referred to [21].

Although many tabular synthesizers are proposed for healthcare generation tasks, most are designed for specific medical applications. For instance, many papers investigating the tabular data generation in the healthcare domain use the MIMIC III clinical database [22] to exclusively synthesize patients' ICD-9 codes (diagnostic codes). However, this paper intends to study the strengths and

weaknesses of the synthetic generative models in the healthcare domain that are not application-specific. In other words, we investigate the GAN-based models' capabilities of generating tabular healthcare datasets that are both representative to most medical applications and contain various data types.

3 Data and Method

In order to explore how different properties of tabular data are captured by the generative models, the models are trained on four different tabular data sets from the healthcare domain. The datasets are of various sizes and include different data types and distributions. The four datasets are the Epileptic Seizure, Thyroid and Diabetes datasets, as well as selected tables from the MIMIC III data. An overview of the datasets is provided in Table 1.

Table 1. The datasets applied in this work for synthetic tabular data generation. All target columns are binary categorical.

Dataset	#Train	#Test	Target Name	Explanation of Target Name (yes/no)
Epileptic Seizure	9,000	2,500	Y	Epileptic seizure
Thyroid	7,100	2,000	BinaryClass	Thyroid disease
Diabetes	70,000	19,000	Readmitted	Hospital readmission
MIMIC III	31,900	9,000	Hospital_Expire_Flag	Survived

The Epileptic Seizure dataset, originally from [23], includes electroencephalogram (EEG) recordings from 500 patients. A preprocessed version of the dataset available at Kaggle is applied [24]. Converting the EEG signals from time series to a tabular format results in a dataset consisting of $11,500$ rows and 178 columns. The column values are discrete and range from -1850 to 1750.

The Thyroid dataset from the UCI Machine Learning Repository [25] includes $9,172$ rows representing patient records, and 20 columns with information about the patients, including whether they are diagnosed with thyroid disease or not. The column indicating whether the patient is on antithyroid therapy or not is highly imbalanced, with the majority of patients not being on antithyroid therapy. Missing values are imputed using the SimpleImputer from the scikit-learn library [26].

The Diabetes dataset [27] includes medical characteristics about diabetic patients that were admitted to hospitals in the United States. The columns representing diagnostic information (diag_1 - 3) exhibits multiple modes. The dataset also contains columns with discrete values with small and large value-ranges. Following preprocessing, the dataset has $89,053$ rows (inpatient encounters) and 29 columns.

The MIMIC III dataset [22] contains 26 tables including information such as diagnoses, prescribed medications and laboratory measurements for patients admitted to critical care units. For this work, we join seven of these tables in order to create a dataset containing $40,895$ rows and 14 columns. Each row

represents a patient and includes demographic and medical details during a stay at the intensive care unit at the hospital. The column representing the length of patient's stay is highly skewed to the right with some outliers that stayed in the hospital for a very long time. Patients above 90 years old did not have their true age registered. For these patients, we randomly assigned them ages between 90 and 100 years, with decreasing probability with increasing age.

Four different GAN architectures are trained for generation of tabular healthcare data: TGAN [14], CTGAN [15], CTABGAN [16] and Wasserstein GAN with gradient penalty (WGAN-GP) [28]. The WGAN-GP is used as a baseline model because this is a general GAN that has proven robust to the mode collapse problem [28]. Consequently, we want to explore whether the GAN architectures developed specifically for tabular data generation outperform the more general WGAN-GP. The code used to run all experiments including details regarding the model architectures and hyperparameters is available on GitHub[1].

All the experiments are conducted using Python 3.8 as the primary programming language. The proposed tabular data generation models are implemented using Tensorflow [29], except the CTABGAN model, which is built with Pytorch [30]. Furthermore, the experiments are conducted on the University of Stavanger's GPU cluster (Gorina6) on an Nvidia Tesla V100 machine equipped with 32 GB of memory. However, the evaluations and comparisons are conducted on a Desktop PC with specifications of AMD Ryzen 5 5600G with 8 GB of memory.

The models are evaluated based on their ability to generate realistic synthetic data and protect the privacy of the individuals. Two different methods are applied to evaluate the abilities of the GANs to generate realistic samples: Statistical resemblance and a ML model-based approach. For statistical resemblance, we include the marginal column distributions for the original and synthetic data. Ideally, the distributions should be similar. Regarding the ML model-based approach, ML models are trained to classify samples as real or synthetic. If the quality of the synthetic data is high, the classifiers will not be able to distinguish between real and synthetic data. Predictive ML models are also trained on either real or synthetic data and then the predictions on the same test set are compared. For realistic generated data, the predictive performance should not be too different between the models trained on real and fake data, respectively. To evaluate privacy protection, the Euclidean distances are calculated between the synthetic samples and their original counterparts. Consequently, the distribution of pairwise distances between each synthetic record and its nearest original neighbor is achieved. Ideally, the resulting distribution has a large mean and small standard deviation when evaluating the models through the lens of privacy. However, a large mean also indicates poor quality of the synthetic data, meaning that the distance-based privacy metric is inversely proportional to the ML-based evaluation scores. We therefore evaluate them simultaneously to find the overall best-performing model in terms of generating realistic samples while still preserving privacy.

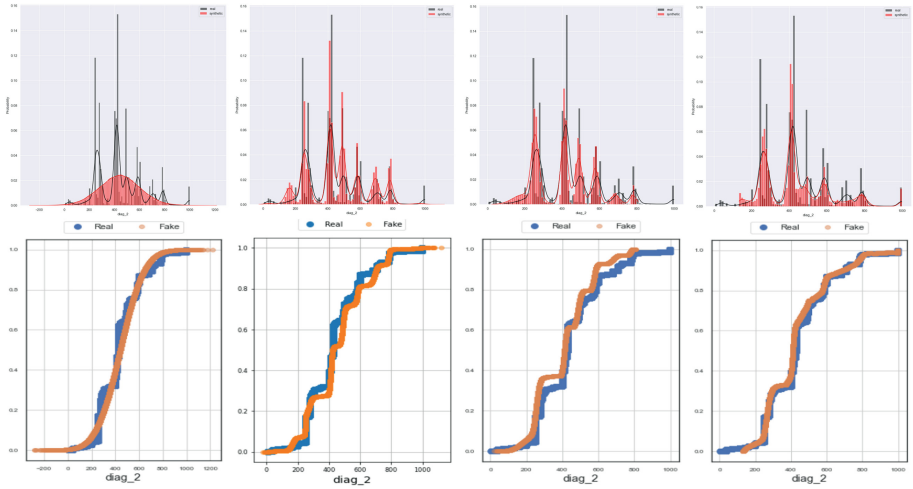
[1] https://github.com/ds-anik/Synthetic_Tabular_Healthcare_Data_Generation.

4 Results

This section presents the experimental results, starting with the column distributions for the generated datasets in Sect. 4.1, then providing the ML-based evaluation scores in Sect. 4.2 and finally presenting the metrics for privacy preservation in Sect. 4.3.

4.1 Column Distributions

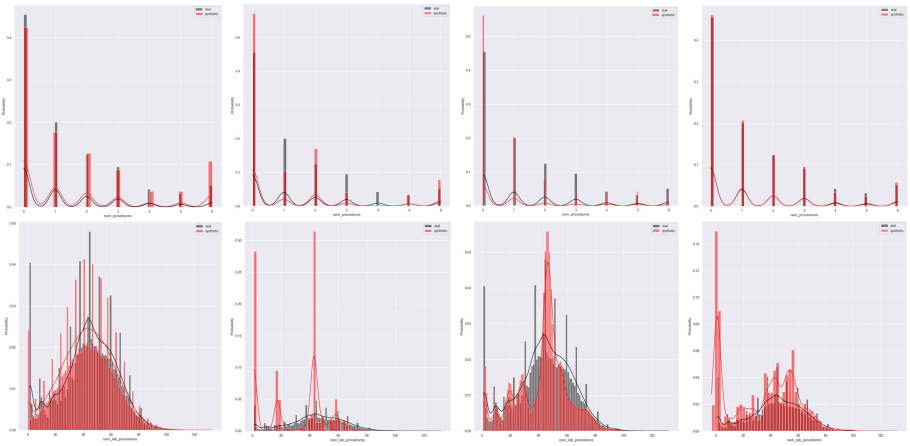
Figure 1 compares the marginal distributions (top row) and the cumulative distributions (bottom row) of the diag_2 numerical column in the Diabetes dataset. It is clear that the marginal distribution of the original data has a dominant peak at 450 and multiple lower peaks around it. The WGAN-GP implementation lacks any specific normalization to detect various modes in the numerical features and generates a simple normal distribution around the dominant peak, i.e. it faces mode-collapse. Although the Wasserstein GAN loss function was introduced to circumvent the mode-collapse issue, we observe that it is not applicable to detect complex multi-modal distributions as in our case. The other three models clearly excel WGAN-GP in capturing the modes of the diag_2 column, probably because they apply a mode-specific normalization technique for dealing with multi-modal continuous columns. Comparing the cumulative distributions of all the models demonstrate that the CTABGAN architecture outperforms the other architectures in generating skewed multi-modal numerical columns.



(a) WGAN (b) TGAN (c) CTGAN (d) CTABGAN

Fig. 1. The marginal and cumulative probability distributions for the multi-modal *diag_2* column in the Diabetes dataset.

Figure 2 shows the marginal probability distributions of two discrete numerical columns in the Diabetes dataset. One column has a small range of integer values and the other has a wide range. We observe that for small-range discrete numerical columns, the original and synthetic distribution tend to resemble perfectly. However, the models' performances significantly drop when generating integer columns with a wide range of values. While the marginal distribution of the original data resembles a simple Gaussian distribution, the probability distributions resulting from the TGAN, CTGAN, and CTABGAN consist of several modes. WGAN-GP, on the other hand is better at generating wide-ranging integer variables.



(a) WGAN-GP (b) TGAN (c) CTGAN (d) CTABGAN

Fig. 2. The marginal distributions of two integer columns with a small range (upper row) and a large range of values (lower row) in the Diabetes dataset.

4.2 Machine Learning-based Evaluation

To compare the inferential ability of the original and synthetic datasets, we train a set of predictive models on both the real and fake datasets and compare their predictive capability using the real data. Since all the chosen datasets include a categorical target column, we use the Macro-F1 score to evaluate the predictive capabilities of the models. The F1 score is the harmonic mean of the sensitivity and precision, taking both of them into account. The value ranges from 0 to 1, where 1 is best. Macro-F1 calculates the F1 score for each category and compute unweighted mean of the F1 scores. Macro-F1 is used instead of accuracy due to the imbalanced nature of many categorical features across the investigated datasets. The goal is to verify if the same insights are derived from real and fake datasets when trained on an equally tuned ML model, not picking the best

classifier. Thus, we exclude hyper-parameter tuning for each predictive model and compare the GANs based on the average Macro-F1 scores of the classifiers. In addition to comparing the inferential abilities on original and synthetic data, we train logistic regression (LR) and support vector machine (SVM) classifiers on the labelled original and fake datasets to evaluate whether the models are able to distinguish original and fake samples. The normalized area under the receiver operating characteristic (AUROC) score is used for model evaluation. If the synthetic data is inseparable from the original one, the unnormalized AUROC score would be 0.5, indicating that the classifier is guessing randomly and unable to distinguish the real and fake classes. However, since most of the evaluation metrics in our setting are in the range of 0 to 1, we normalize the classification result to 1 minus the average AUROC score. A normalized AUROC score of 1 is best and means that the real and synthetic data are inseparable. A synthetic dataset with low Macro-F1 difference and high normalized AUROC is considered ideal. Tables 2 and 3 show the absolute difference in Macro-F1 scores of the decision tree (DT), random forest (RF), LR and MLP classifiers trained on the original and synthetic datasets, respectively. WG, TG, CT and CTAB stand for WGAN-GP, TGAN, CTGAN and CTABGAN respectively. The normalized AUROC scores for the LR and SVM classifiers are reported in the last two rows of both tables.

Table 2. The difference of the Macro-F1 classification scores and the ML detection scores in the Diabetes and MIMIC III datasets. The best results for each dataset are highlighted in bold. Abbreviations: ΔF1 = difference in Macro-F1 classification score.

	Diabetes				MIMIC III			
	WG	TG	CT	CTAB	WG	TG	CT	CTAB
ΔF1-DT	0.083	0.056	0.030	**0.016**	0.189	0.068	0.052	**0.031**
ΔF1-RF	0.122	0.082	0.044	**0.023**	0.120	0.030	0.020	**0.010**
ΔF1-LR	0.129	0.050	0.046	**0.009**	0.094	**0.001**	0.015	0.007
ΔF1-MLP	0.143	0.087	0.044	**0.013**	**0.015**	0.035	0.039	0.028
ΔF1-average	0.119	0.068	0.041	**0.015**	0.104	0.033	0.031	**0.019**
AUROC-LR	0.330	0.520	0.540	**0.700**	0.220	0.620	0.730	**0.790**
AUROC-SVM	0.110	0.290	0.330	**0.560**	0.110	0.390	0.480	**0.540**

If we average the Macro-F1 differences across all four classifiers in the Diabetes dataset, we find that the CTABGAN model has the lowest average score of 0.015, followed by CTGAN and TGAN models with average scores of 0.041 and 0.068. This pattern is repeated in the MIMIC III and Thyroid datasets, with the CTABGAN model outperforming others in terms of the difference in Macro-F1 classification scores followed by the CTGAN, TGAN, and WGAN-GP. The reason why CTABGAN is the best performing model in these datasets is due to the modified conditional GAN architecture and an additional information loss term

in the optimization process. Although the Diabetes, MIMIC III, and Thyroid datasets follow a similar pattern regarding the average Macro-F1 scores, there is a larger gap between the CTABGAN and CTGAN in the Diabetes dataset compared to the ones in the other two datasets. This can be related to the multi-modal nature of the numerical columns in the Diabetes dataset and how the CTABGAN model successfully generates this type of numerical distribution, benefiting from an extended condition vector in its architecture.

However, in the Epileptic dataset, the WGAN-GP outperforms other models regarding the average Macro-F1 differences. WGAN-GP is the best performing model with an average score of 0.085, followed by CTABGAN, TGAN, and CTGAN, with average scores of 0.18, 0.21, and 0.23, respectively. The Epilpetic dataset includes 178 integer columns with wide ranges. Similar to our interpretation in Sect. 4.1, we find that although the mode-specific normalization approach in the CTGAN, TGAN, and CTABGAN is well-suited for numerical columns with complex distributions, it may prevent the model from reaching an ideal optimum in the smaller datasets with the discrete numerical variables (integers).

Table 3. The difference of the Macro-F1 classification scores and the ML detection scores in the Thyroid and Epileptic datasets. The best results for each dataset are highlighted in bold. Abbreviations: ΔF1 = difference in Macro-F1 classification score.

	Thyroid				Epileptic			
	WG	TG	CT	CTAB	WG	TG	CT	CTAB
ΔF1-DT	0.300	0.240	0.200	**0.100**	**0.070**	0.210	0.210	0.180
ΔF1-RF	0.260	0.220	0.190	**0.100**	**0.050**	0.280	0.250	0.230
ΔF1-LR	0.140	0.090	**0.050**	**0.050**	**0.010**	**0.010**	0.080	0.030
ΔF1-MLP	0.240	0.170	0.120	**0.080**	**0.210**	0.350	0.390	0.280
ΔF1-average	0.230	0.180	0.140	**0.080**	**0.085**	0.210	0.230	0.180
AUROC-LR	0.530	0.700	0.700	**0.780**	**0.770**	0.330	0.450	0.730
AUROC-SVM	0.380	0.600	0.530	**0.620**	**0.620**	0.250	0.380	0.590

The LR and SVM classifiers' normalized AUROC scores follow the same pattern as the average Macro-F1 differences. Due to the auxiliary classifier in the CTABGAN's architecture and the classification loss term in its optimization process, it is much harder to distinguish the synthetic data generated from the CTABGAN model from the original data. Consequently, the normalized AUROC scores of the CTABGAN model outperform the scores from the other models for the Diabetes, MIMIC III, and Thyroid datasets. Again, WGAN-GP's score exceeds the scores of the other models for the Epileptic dataset, followed by CTABGAN, CTGAN, and TGAN.

Overall, the CTABGAN model achieves the best ML cross-testing scores for the Diabetes, MIMIC III, and Thyroid datasets. This is because of its modified

conditional GAN architecture and an additional information loss term in its optimization process. However, WGAN-GP outperforms the other architectures for the Epileptic dataset as the normalization techniques in TGAN, CTGAN, and CTABGAN are less suitable for the wide-ranging integer columns in this dataset. The same pattern is observed for the normalized AUROC scores: The CTABGAN model is the best-performing model at generating synthetic records that are indistinguishable from the original ones, except from the Epileptic dataset where WGAN-GP is ranked highest.

4.3 Preserved Privacy

We also evaluate the GAN's potential to preserve the privacy of sensitive data. This evaluation category is especially important in the healthcare domain, where patients share sensitive and private information. Suppose a patient's confidential data is to be re-identified by accessing the synthetic data. In that case, the patient's sensitive information is undoubtedly leaked in the synthetic dataset, and the GAN simply replicates the original records when generating new ones.

Table 4. The distribution of Euclidean distances between synthetic records and their closest original counterparts. The format is $mean \pm std$.

Model	Diabetes	MIMIC III	Thyroid	Epileptic
WGAN-GP	3.10 ± 0.46	1.37 ± 1.14	1.88 ± 0.97	7.55 ± 8.45
TGAN	2.69 ± 0.61	0.93 ± 0.80	1.39 ± 1.05	7.81 ± 9.26
CTGAN	2.71 ± 0.64	0.97 ± 0.90	1.34 ± 1.08	9.10 ± 9.18
CTABGAN	3.02 ± 0.46	1.11 ± 0.84	1.76 ± 1.06	8.08 ± 9.27

Table 4 shows the mean and standard deviation of the distance to closest record distributions. In the Diabetes, MIMIC III, and Thyroid datasets, the WGAN-GP model maintains the largest distance between the original and synthetic data (lowest privacy risk). This verifies the results in Sect. 4.2, as the WGAN-GP model was the worst-performing regarding ML utilities. Interestingly, in contrast to the results for ML utilities, we observe that the CTABGAN is the second best-performing model in all the datasets. Although the CTABGAN model outperforms the CTGAN and TGAN models in the Diabetes, MIMIC III, and Thyroid datasets, it preserves the privacy more when generating synthetic records.

In the Epileptic dataset, the CTGAN and WGAN-GP models are the best and worst models regarding privacy preservation with means of 9.10 and 7.55, respectively. This verifies the results for ML utilities as the WGAN-GP is the best-performing and CTGAN is the worst-performing one in the Epileptic dataset.

5 Discussion

From the results in Sect. 4.1, we observe that CTABGAN outperforms the other GANs when generating data that are distributed in a similar way as the original data. This is also true regarding skewed multi-modal numerical columns. Although both CTGAN and CTABGAN use a conditional generator and the training-by-sampling technique to handle generating imbalanced categorical features, it is observed that the CTGAN model slightly outperforms the CTABGAN architecture when dealing with highly imbalanced categorical variables. This is because of the modification of the CTABGAN's conditional generator to capture skewed multi-modal distributions more effectively compared to the CTGAN model. Besides the one-hot encoded representations for categorical columns, the extended conditional vector in CTABGAN includes the mode of the numerical columns, increasing its performance in capturing modes with less weight. In Fig. 1, it is clear that WGAN-GP faces mode collapse, generating a simple normal distribution. Although the Wasserstein GAN loss function was introduced to circumvent the mode-collapse issue, we observe that it is not applicable to detect complex multi-modal distributions, as in our case.

One of the few areas where there is room for further improvement is the generation of discrete numerical columns, as none of the proposed models makes any distinction between the continuous and discrete numerical features. This is confirmed in Fig. 2 for the column with wide-ranging values in the Diabetes dataset and is also observed in Sect. 4.2, where the models struggle with the wide-ranging, discrete numerical columns in the Epileptic dataset. TGAN, CTGAN, and CTABGAN produces multi-modal distributions even though the original data shows a single mode. This occurs due to the mode-specific normalization implemented in the mentioned GANs. Although this technique is well-suited for continuous columns with complex distributions, it may prevent the model from reaching an ideal optimum for the discrete numerical variables. The exception is the WGAN-GP architecture, which might explain why the WGAN-GP model nicely captures the wide-ranging discrete column in the Diabetes dataset and is ranked highest for the ML-based metrics for the Epileptic dataset. However, WGAN-GP shows poor performance in capturing multimodal distributions as well as for the ML-based metrics for the other datasets.

When considering the comparison of column distributions, ML-based evaluation scores and privacy preserving metrics simultaneously, the CTABGAN model outperforms the other models in generating realistic samples while also preserving the privacy of the original samples in all datasets.

6 Conclusion

In conclusion, the obtained visual and quantitative results demonstrate that synthetic healthcare data can be a reliable substitute for original data. Of the tested architectures, CTABGAN seems to be most promising for generating realistic synthetic tabular healthcare data without leaking individuals' sensitive information. However, for datasets containing columns of wide-ranging integer values,

a vanilla WGAN-GP might be more appropriate. Generating data using GANs eliminates the need for traditional anonymization and obfuscation techniques which are too risky and negatively impact the data utilities. For future work, further developments in the models' architectures can potentially improve the performance of synthetic data generation. Moreover, there is much room for the proposed data generating models to improve training convergence in small-sized datasets and generate discrete numerical columns. Development of generative models that also handle other types of data such as free text, ordinal values and time series should be explored. Finally, we only investigate the strengths and weaknesses of the GAN-based models in generating healthcare tabular datasets. However, it would be interesting to test other generative models like variational autoencoders, Gaussian copula and Bayesian networks in the medical use cases.

References

1. Tavanapong, W., Oh, J., Riegler, M., Khaleel, M.I., Mitta, B., De Groen, P.C.: Artificial intelligence for colonoscopy: past, present, and future, IEEE Journal of Biomedical and Health Informatics
2. Choy, G.: Current applications and future impact of machine learning in radiology. Radiology **288**(2), 318 (2018)
3. Shatte, A.B., Hutchinson, D.M., Teague, S.J.: Machine learning in mental health: a scoping review of methods and applications. Psychol. Med. **49**(9), 1426–1448 (2019)
4. van de Sande, D., et al.: Developing, implementing and governing artificial intelligence in medicine: a step-by-step approach to prevent an artificial intelligence winter, BMJ Health & Care Informatics 29 (1)
5. Rajkomar, A., Dean, J., Kohane, I.: Machine learning in medicine. N. Engl. J. Med. **380**(14), 1347–1358 (2019)
6. Thambawita, V., et al.: DeepSynthBody: the beginning of the end for data deficiency in medicine. In: 2021 International Conference on Applied Artificial Intelligence (ICAPAI), pp. 1–8. IEEE (2021)
7. Goncalves, A., Ray, P., Soper, B., Stevens, J., Coyle, L., Sales, A.P.: Generation and evaluation of synthetic patient data. BMC Med. Res. Methodol. **20**(1), 1–40 (2020)
8. Rashidian, S., et al.: SMOOTH-GAN: towards sharp and smooth synthetic EHR data generation. In: Michalowski, M., Moskovitch, R. (eds.) AIME 2020. LNCS (LNAI), vol. 12299, pp. 37–48. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59137-3_4
9. Kingma, D.P., Welling, M.: Auto-encoding variational bayes (2013)
10. Gogoshin, G., Branciamore, S., Rodin, A.S.: Synthetic data generation with probabilistic Bayesian networks. Math. Biosci. Eng. MBE **18**(6), 8603 (2021)
11. Goodfellow, I., et al.: 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)
12. Choi, E., Biswal, S., Malin, B., Duke, J., Stewart, W.F., Sun, J.: Generating multi-label discrete patient records using generative adversarial networks. In: Doshi-Velez, F., Fackler, J., Kale, D., Ranganath, R., Wallace, B., Wiens, J., (Eds.), Proceedings of the 2nd Machine Learning for Healthcare Conference, vol. 68 of Proceedings of Machine Learning Research, pp. 286–305. PMLR (2017)

13. Park, N., Mohammadi, M., Gorde, K., Jajodia, S., Park, H., Kim, Y.: Data synthesis based on generative adversarial networks. Proceedings of the VLDB Endowment **11**(10), 1071–1083 (2018)
14. Xu, L., Veeramachaneni, K.: Synthesizing tabular data using generative adversarial networks (2018)
15. Xu, L., Skoularidou, M., Cuesta-Infante, A., Veeramachaneni, K.: Modeling tabular data using conditional GAN. In: Wallach, H., Larochelle, H., Beygelzimer, A., Alché-Buc, F.D., Fox, E., Garnett, R., (Eds.), Advances in Neural Information Processing Systems, vol. 32, Curran Associates Inc., (2019)
16. Zhao, Z., Kunar, A., Birke, R., Chen, L.Y.: CTAB-GAN: effective table data synthesizing. In: Balasubramanian, V.N., Tsang, I., (Eds.), Proceedings of The 13th Asian Conference on Machine Learning, vol. 157 of Proceedings of Machine Learning Research, pp. 97–112. PMLR (2021)
17. Xie, L., Lin, K., Wang, S., Wang, F., Zhou, J.: Differentially private generative adversarial network. arXiv preprint arXiv:1802.06739
18. Torkzadehmahani, R., Kairouz, P., Paten, B.: DP-CGAN: differentially private synthetic data and label generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (2019)
19. Torfi, A., Fox, E.A., Reddy, C.K.: Differentially private synthetic medical data generation using convolutional GANs. Inf. Sci. **586**, 485–500 (2022)
20. Jordon, J., Yoon, J., Van Der Schaar, M.: PATE-GAN: generating synthetic data with differential privacy guarantees. In: International Conference on Learning Representations (2018)
21. Coutinho-Almeida, J., Rodrigues, P.P., Cruz-Correia, R.J.: GANs for tabular healthcare data generation: a review on utility and privacy. In: Soares, C., Torgo, L. (eds.) DS 2021. LNCS (LNAI), vol. 12986, pp. 282–291. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-88942-5_22
22. Johnson, A.E., et al.: MIMIC-III, a freely accessible critical care database. Scientific Data 3 (160035)
23. Andrzejak, R.G., Lehnertz, K., Mormann, F., Rieke, C., David, P., Elger, C.E.: Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: dependence on recording region and brain state. Phys. Rev. E 64 (061907)
24. Harun-Ur-Rashid, Supriya, Epileptic seizure recognition (2018)
25. Dua, D., Graff, C.: UCI machine learning repository (2017)
26. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
27. Strack, B., et al.: Impact of HbA1c measurement on hospital readmission rates: analysis of 70,000 clinical database patient records, BioMed Research International (2014)
28. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C., Improved training of wasserstein GANs. In: Guyon, I., et al. (Eds.), Advances in Neural Information Processing Systems, vol. 30, Curran Associates Inc., (2017)
29. Abadi, M.: TensorFlow: large-scale machine learning on heterogeneous systems. Software available from tensorflow.org (2015)
30. Paszke, A., et al.: PyTorch: an imperative style, high-performance deep learning library. In: Wallach, H., Larochelle, H., Beygelzimer, A., d' Alché-Buc, F., Fox, E., Garnett, R., (Eds.), Advances in Neural Information Processing Systems 32, Curran Associates Inc., pp. 8024–8035 (2019)