A Hyperparameter Tuning Framework for Tabular Synthetic Data Generation Methods

Mohammad Ahmed Basri[1] **Bing Hu**[2] **Abu Yousuf Md Abdullah**[3] **Shu-Feng Tsao**[4] **Zahid Butt**[4] **Helen Chen**[2,4,5]
[1]Department of Systems Design Engineering, [2]Cheriton School of Computer Science, [3]School of Planning
[4]School of Public Health Sciences, [5]Department of Statistics and Actuarial Science
University of Waterloo
{mabasri, bingxu.hu, aymabdullah, s7tsao, zahid.butt, helen.chen}@uwaterloo.ca

Abstract

This paper delves into the potential of synthetic tabular data as a viable alternative to real data, ensuring that essential information is used without compromising confidential information about individuals. Our experiments are centred around the deployment of the Conditional Tabular Generative Adversarial Networks (CTGAN) model for the synthetic data generation process. Recognizing the intricate nature of healthcare data and the precision it demands, the study emphasizes the importance of hyperparameter optimization of the synthetic generation process. By tuning these hyperparameters, our aim is to enhance the authenticity and relevance of the synthesized data, drawing it ever closer to real-world datasets. In an attempt to revolutionize data availability in healthcare research, we study different objective functions and their correlations for the most optimal combination of hyperparameters that results in the highest quality of synthetic data.

1 Introduction

Deep learning (DL) algorithms such as generative adversarial networks, transformers, and gradient boosting for synthetic data generation (SDG) involve a number of parameters to be set before training. Hyperparameter tuning strategies are second-level optimization procedures that try to minimize the expected generalization error of an algorithm over a hyperparameter search space using an objective function [1, 2]. In contrast to model parameters, which are learned during training, these tuning parameters (hyperparameters) have to be carefully selected to optimize model performance. Users have typically 3 choices for selecting an appropriate hyperparameter configuration for a specific dataset: (1) use default hyperparameter values as designed, (2) manually configure hyperparameter values based on recommendations from literature, experience, or trial-and-error, or (3) use hyperparameter tuning (HPT) strategies [1].

The main goal of hyperparameter optimizing is to automatically tune hyperparameters for users to apply machine learning models to practical problems effectively [3, 4]. Although hyperparameter tuning for classification and regression tasks often have a clear choice for objective functions such as any of the metrics computed from the confusion matrix, the choice of the objective function is not so clear for SDG models. As synthetic data is evaluated in a multitude of different ways such as machine learning efficacy (MLE), univariate distribution comparisons, discriminator measures, multivariate correlations, and privacy metrics [5–9], it is unclear how best to tune SDG hyperparameters.

Recent literature states the importance of hyperparameters on the performance of SDG models but there still lacks a clear framework for SDG hyperparameter tuning (HPT) [6, 7, 10]. In addition to a clear framework, it's equally important to have an SGD HPT framework that can be efficiently applied. Although machine learning efficacy is an important metric for SDG models, it can be expensive to compute as an objective function in a multi-objective HPT framework. In this paper, we aim to provide an efficient and clear framework for future work on HPT for SGD. To tackle the problem of inefficiencies of MLE as an HPT objective function, we propose to use differential pairwise correlation (DPC) as an alternative objective to using MLE given increased efficiency in computational costs.

2 Methods

2.1 Simulated Real Dataset

A real health dataset is simulated from MIMIC-IV to be used to generate synthetic data. Fields of ethnicity, gender, death, religion, marital status, insurance, and age are sampled from MIMIC-IV to create a profile for each patient. Additional binary flags for select diagnoses of sepsis, birth, chest pain, hypertension, and overdose are recorded for each patient over all their admissions. The simulated real dataset contains 58,977 rows of patients.

2.2 CTGAN Model

Like any deep learning model, the Conditional Tabular Generative Adversarial Network (CTGAN) [11] models performance is dependent on the hyperparameters. For the current study, three CTGAN hyperparameters are considered, which are batch size, generator learning rate, and discriminator learning rate. Some CTGAN hyperparameters that can be considered in future work on hyperparameter tuning can include generator weight decay, discriminator weight decay, number and size of layers, choice of optimizer, and choice of learning rate schedulers.

2.3 Hyperparameter Tuning for Synthetic Data

Hyperparameter tuning is an essential method to find out the combination of hyperparameters that gives the most optimal parameters. In our study, we use grid search as our HPT strategy.

2.3.1 Grid Search

Grid search is the conventional method of hyperparameter optimization, where the model is trained across all combinations of all hyperparameters[12]. The method forms a grid of all the hyperparameters and their values and then creates unique combinations of these hyperparameters. For each trial of optimization, the aim is to find an optimal value of the objective function.

For the current study, three CTGAN hyperparameters are considered, which are batch size, generator learning rate, and discriminator learning rate. Three choices are provided for each of the three hyperparameters chosen: batch size can be one of 50, 100, or 200, generator learning rate can be one of 1e-3, 1e-4, or 1e-5, and discriminator learning rate can be one of 1e-3, 1e-4, or 1e-5. These hyperparameter values result in a grid of 3x3x3 with 27 unique combinations. As there are 27 unique combinations, under grid search, a total of 27 corresponding trials are run.

2.3.2 Objectives For SDG Models

Table 1 contains a list of possible evaluation metrics that can be used as objective functions for HPT of SDG models. In this paper, we use three of these evaluation metrics, machine learning efficacy, Hellinger distance, and differential pairwise correlations for HPT.

Machine Learning Efficacy (MLE) is a narrow measure that assesses the ability of the synthetic data to replicate a specific use case [13]. In MLE, a proxy classification task is defined and two models are trained on the real data training set and the generated synthetic training set. Both models are then evaluated on a real-data test set where a highly capable SDG model should be on par with its real-data counterpart. This whole process is shown in figure 1. MLE evaluates

Table 1: List of SDG model evaluation metrics gathered from literature that can be used as objective functions for HPT.

S.No.	Evaluation Metric	Description	
1	Machine Learning Efficacy (MLE)	Measures the ability of the synthetic data to train a model for a classification task, which is tested by real data	2.3.2, [13]
2	Hellinger Distance (HD)	Measures the similarity of the probability distribution of the synthetic and real data. Each variable will have one HD value.	2.3.2, [14, 15]
3	Differential Pairwise Correlations	Pairwise correlation measures the strength of correlation between the variables for the real data and synthetic data.	2.3.2, [16]
4	Kullback-Leibler (KL) Divergence	KL divergence measures the similarity of synthetic and real data probability mass functions for a given variable.	[17]
5	Log-Cluster	Log-cluster measures the similarity of the underlying dependency structure in terms of clustering	[17]
6	Propensity	Measures the distinguishability between the real and synthetic data.	[18]
7	Kolmogorov-Smirmov Type Statistic	Measures differences between the empirical cumulative distribution functions calculated on the synthetic and real data.	[19]
8	Distance to Closest Record (DCR)	Measures the Euclidean distance between a record r of synthetic data and the closest record r of the real data.	[20]
9	Model Size	Certain hyperparameters may alter the size of the model.	
10	Model Training and Sampling Speed	Certain hyperparameters may alter the time required for training and Sampling for the model.	

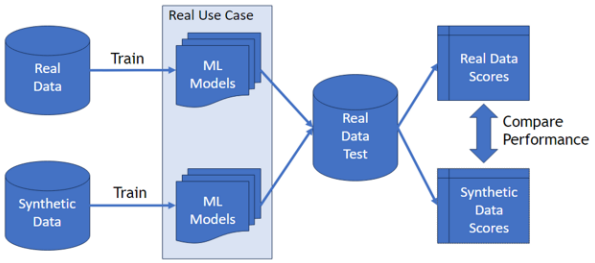

Fig. 1: Overview of the MLE evaluation framework.

the synthetic data on how it is expected to be used by researchers in the real world [8]. ML Models used to compute MLE can be subject to many hyperparameters such as early stopping, learning rates, and regularization.

In our paper, we chose the mortality prediction task as our MLE task. The ML model is a small 3-layer binary classifier neural network. The model is trained to predict mortality (expire flag variable in the dataset) from all other fields as input features. The trained model is then evaluated on a fixed real data test set using the accuracy metric.

Hellinger Distance (HD) quantifies the similarity between two probability distributions [15]. Given two discrete probability distributions $P = \{p_1, p_2, ..., p_n\}$ and $Q = \{q_1, q_2, ..., q_n\}$, the HD between P and Q is expressed in eq. 1.

$$HD^2(p,q) = \frac{1}{2}\sum_{i=1}^{n}(\sqrt{p_i} - \sqrt{q_i}) \quad (1)$$

HD provides a summary statistic of differences between each variable in the real and synthetic datasets. HD scores range between 0 to 1, where values closer to 0 are desired as they indicate lower differences in the distribution between real and synthetic datasets [14]. HPT can be applied to this evaluation metric given a desire to produce synthetic data that highly resembles the univariate distribution of the real data. Given a dataset with low HD (to the real data), the univariate distributions are very similar between the synthetic and real data, but correlations between variables may not be well preserved. To optimize the preservations of correlations between variables between the synthetic and real data, other additional evaluation metrics need to be used as HPT objectives.

Differential Pairwise Correlation is a bivariate measure of the correlation between the synthetic data and the real data [21]. Synthetic data that closely resembles real data should have similar bivariate pairwise correlations. In combination with the univariate HD metric, DPC provides a bivariate metric for open data policymakers to utilize as a standard to better compare synthetic datasets. If the real and synthetic datasets had high fidelity (i.e., the synthetic dataset closely resembled the real dataset), then the absolute difference would be close to 0 or very small.

For any fields containing continuous variables, the differential pairwise correlations in the real and synthetic data were evaluated to obtain fidelity in terms of bivariate statistics as shown in eq. 2.

$$\Delta CV_{continuous_{XY}} = |\rho_{XY_{real}} - \rho_{XY_{synthetic}}| \quad (2)$$

Here, X and Y denote the two continuous variables, whereas ρ_{XY} is the Pearson correlation coefficient for X and Y. The Pearson correlation coefficient can be defined over the real and synthetic data. In contrast, for categorical variables, the absolute differences for Chi-square statistics in the real and synthetic data are evaluated as shown in eq. 3

$$\Delta CV_{categorical_{XY}} = |\chi^2_{XY_{real}} - \chi^2_{XY_{synthetic}}| \quad (3)$$

Here, X and Y denote the two categorical variables, whereas χ^2_{XY} is the χ^2_{XY} statistic for X and Y. The χ^2 coefficient can be defined over the real and synthetic data. DPC can be used as an evaluation metric for hyperparameter optimization.

3 Results and Discussion

Table 2 shows the values of the different evaluation metrics for the different hyperparameter combinations. For the study, the Mortality Prediction Accuracy was considered as the objective function. Therefore, the best parameters are batch size: 200, generator learning rate: 1e-4, and discriminator learning rate: 1e-4. The corresponding accuracy for this combination is 89.6%. During every iteration of synthetic data generation with a unique combination of hyperparameters, the Hellinger distances for all the feature variables and the differential pairwise correlations were also noted. For simplicity, the average Hellinger distance is calculated by dividing the sum of the Hellinger distances by the number of variables.

Similarly, for the absolute differential pairwise correlations, the values are calculated for each variable pair, for the real and the synthetic data, then the average of the absolute difference of these values is noted. From the results, we observe, that the minimum average Hellinger distance is 0.052 for the hyperparameter combination (100, 1e-4, and 1e-3), and the corresponding Mortality prediction accuracy is 83.4%. Although average HD has a statistically significant correlation with mortality prediction accuracy (figure 3), the correlation is not as strong as compared to average DPC with mortality prediction

Table 2: Results of 27 trials of grid search for hyperparameter tuning a CTGAN synthetic generation model for objective functions of Mortality Prediction, Average Hellinger Distance, Median Hellinger Distance, and Average differential pairwise correlation. MLE accuracies above 80% are bolded.

Batch Size	Generator LR	Discriminator LR	Mort. Pred. Accuracy	Avg. HD	Median HD	Avg. ADC
200	1.0E-05	1.0E-05	59.4	0.0614	0.0315	0.0799
200	1.0E-05	1.0E-04	69.2	0.0612	0.0339	0.0415
200	1.0E-05	1.0E-03	69.7	0.0735	0.0423	0.043
200	1.0E-04	1.0E-05	63.1	0.119	0.0912	0.0575
200	1.0E-04	1.0E-04	**89.6**	0.061	0.0342	0.0222
200	1.0E-04	1.0E-03	**87.8**	0.0398	0.0207	0.0248
200	1.0E-03	1.0E-05	57.2	0.211	0.141	0.0645
200	1.0E-03	1.0E-04	68.4	0.125	0.131	0.0656
200	1.0E-03	1.0E-03	**86.3**	0.0615	0.0681	0.0317
100	1.0E-05	1.0E-05	69.4	0.0673	0.03	0.0589
100	1.0E-05	1.0E-04	68.7	0.0619	0.0199	0.0356
100	1.0E-05	1.0E-03	69.7	0.0762	0.035	0.0384
100	1.0E-04	1.0E-05	68.4	0.0821	0.0422	0.0435
100	1.0E-04	1.0E-04	**87.6**	0.0643	0.0506	0.0284
100	1.0E-04	1.0E-03	**83.4**	0.052	0.0369	0.0324
100	1.0E-03	1.0E-05	64.3	0.194	0.0956	0.0697
100	1.0E-03	1.0E-04	78.4	0.101	0.0609	0.0361
100	1.0E-03	1.0E-03	76.8	0.0792	0.0607	0.0409
50	1.0E-05	1.0E-05	70.2	0.0626	0.0302	0.0412
50	1.0E-05	1.0E-04	**81.6**	0.0678	0.039	0.0443
50	1.0E-05	1.0E-03	69.1	0.077	0.0269	0.0448
50	1.0E-04	1.0E-05	70.5	0.0973	0.0466	0.0356
50	1.0E-04	1.0E-04	**88.9**	0.0752	0.0557	0.0262
50	1.0E-04	1.0E-03	73.7	0.0648	0.0622	0.0403
50	1.0E-03	1.0E-05	52	0.167	0.0435	0.0581
50	1.0E-03	1.0E-04	**82.2**	0.0887	0.0982	0.0376
50	1.0E-03	1.0E-03	75	0.0611	0.0428	0.0492

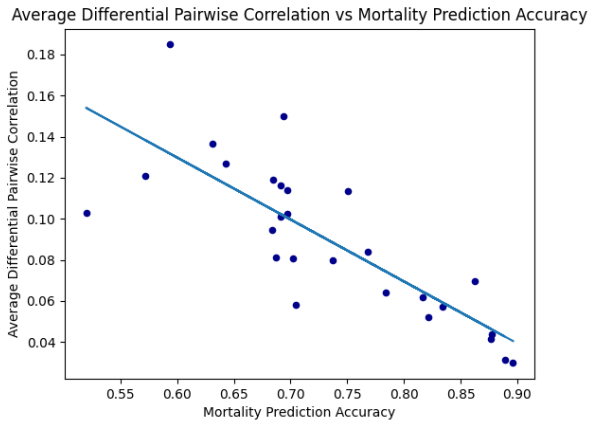

Fig. 2: Scatter plot of MLE of mortality prediction vs the average differential pairwise correlation. The line of best fit is plotted in blue. The correlation between mortality prediction MLE and the average DPC has a r-value of -0.807. The null hypothesis that there is not a significant linear relationship between mortality prediction MLE and average DPC is rejected ($p < 3.59e-7$)

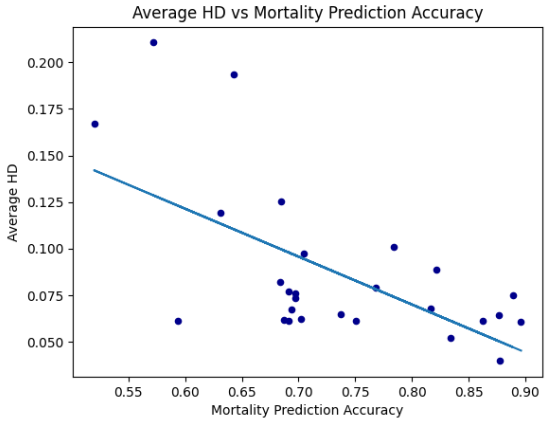

Fig. 3: Scatter plot of MLE of mortality prediction vs the average Hellinger distance. The line of best fit is plotted in blue. The correlation between mortality prediction MLE and the average HD has a r-value of -0.613. The null hypothesis that there is not a significant linear relationship between mortality prediction MLE and average HD is rejected ($p < 6.62e-4$)

accuracy (figure 2). From an optimization point of view, in this specific case, the optimal average HD alone does not maximize mortality prediction accuracy as the best average HD value of 0.0398 does not correspond to hyperparameters with the best MLE accuracy. On the other hand, in this case, DPC has a stronger correlation to the mortality prediction task (figure 2) when compared to average HD.

Utilizing MLE as part of the HPT objective requires running often computationally expensive training and testing during every trial. Given a large search space, MLE as part of HPT can increase runtime non-trivially over the course of hundreds or thousands of trials. As DPC and HD are shown to be strongly correlated to MLE and at a fraction of computational costs, they are possible alternative objectives to optimize over in lieu of using expensive MLE computations.

4 Conclusion

From the results of the experiments performed, we conclude that there is a strong correlation between the average differential pairwise correlation and the machine learning efficacy metrics. However, there is a weaker correlation between the average Hellinger distance and the machine learning efficacy. Depending on the use case, a decision can be made about the right metric to be used for the objective function for hyperparameter tuning. Future work will focus on developing multiple objective function, that finds the best combination, based on the most optimal values of all the different metrics under consideration. Another avenue of future work is to investigate MLE as a difference of metrics between real and synthetic data. This study will be extended

further by developing a use-case agnostic framework for hyperparameter tuning, that can generate more generalizable synthetic tabular data.

References

[1] P. Probst, A.-L. Boulesteix, and B. Bischl, "Tunability: Importance of hyperparameters of machine learning algorithms," *The Journal of Machine Learning Research*, vol. 20, no. 1, pp. 1934–1965, 2019.

[2] B. Bischl, O. Mersmann, H. Trautmann, and C. Weihs, "Resampling methods for meta-model validation with recommendations for evolutionary computation," *Evolutionary computation*, vol. 20, no. 2, pp. 249–275, 2012.

[3] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, 2020.

[4] R. Elshawi, M. Maher, and S. Sakr, "Automated machine learning: State-of-the-art and open challenges," *arXiv preprint arXiv:1906.02287*, 2019.

[5] A. Kotelnikov, D. Baranchuk, I. Rubachev, and A. Babenko, "Tabddpm: Modelling tabular data with diffusion models," *ArXiv*, vol. abs/2209.15421, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:252668788

[6] V. Borisov, T. Leemann, K. Seßler, J. Haug, M. Pawelczyk, and G. Kasneci, "Deep neural networks and tabular data: A survey," *IEEE transactions on neural networks and learning systems*, vol. PP, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:238353897

[7] V. Borisov, K. Seßler, T. Leemann, M. Pawelczyk, and G. Kasneci, "Language models are realistic tabular data generators," *ArXiv*, vol. abs/2210.06280, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:252846328

[8] A. Solatorio and O. Dupriez, "Realtabformer: Generating realistic relational and tabular data using transformers," *ArXiv*, vol. abs/2302.02041, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:256615552

[9] K. El Emam, L. Mosquera, and J. Bass, "Evaluating identity disclosure risk in fully synthetic health data: Model development and validation," *J Med Internet Res*, vol. 22, no. 11, p. e23139, Nov 2020. [Online]. Available: http://www.jmir.org/2020/11/e23139/

[10] J. Fonseca and F. Bacao, "Tabular and latent space synthetic data generation: a literature review," *Journal of Big Data*, vol. 10, no. 1, p. 115, 2023.

[11] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling tabular data using conditional gan," in *Neural Information Processing Systems*, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:195767064

[12] D. M. Belete and M. D. Huchaiah, "Grid search in hyperparameter optimization of machine learning models for prediction of hiv/aids test results," *International Journal of Computers and Applications*, vol. 44, no. 9, pp. 875–886, 2022.

[13] F. K. Dankar and M. Ibrahim, "Fake it till you make it: Guidelines for effective synthetic data generation," *Applied Sciences*, vol. 11, no. 5, p. 2158, 2021.

[14] K. El Emam, *Guide to the de-identification of personal health information*. CRC Press, 2013.

[15] R. Beran, "Minimum hellinger distance estimates for parametri models," *The Annals of Statistics*, vol. 5, 05 1977.

[16] F. K. Dankar, M. K. Ibrahim, and L. Ismail, "A multi-dimensional evaluation of synthetic data generators," *IEEE Access*, vol. 10, pp. 11 147–11 158, 2022.

[17] A. Goncalves, P. Ray, B. Soper, J. Stevens, L. Coyle, and A. P. Sales, "Generation and evaluation of synthetic patient data," *BMC medical research methodology*, vol. 20, no. 1, pp. 1–40, 2020.

[18] J. Snoke, G. M. Raab, B. Nowok, C. Dibben, and A. Slavkovic, "General and specific utility measures for synthetic data," *Journal of the Royal Statistical Society Series A: Statistics in Society*, vol. 181, no. 3, pp. 663–688, 2018.

[19] M.-J. Woo, J. P. Reiter, A. Oganian, and A. F. Karr, "Global measures of data utility for microdata masked for disclosure limitation," *Journal of Privacy and Confidentiality*, vol. 1, no. 1, 2009.

[20] N. Park, M. Mohammadi, K. Gorde, S. Jajodia, H. Park, and Y. Kim, "Data synthesis based on generative adversarial networks," *CoRR*, vol. abs/1806.03384, 2018. [Online]. Available: http://arxiv.org/abs/1806.03384

[21] M. L. McHugh, "The chi-square test of independence," *Biochemia medica*, vol. 23, no. 2, pp. 143–149, 2013.