Synthetic Patient Data Generation and Evaluation in Disease Prediction Using Small and Imbalanced Datasets

Antonio J. Rodriguez-Almeida, Himar Fabelo, Samuel Ortega, Alejandro Deniz, Francisco J. Balea-Fernandez, Eduardo Quevedo, *Member, IEEE,* Cristina Soguero-Ruiz, Ana M. Wägner and Gustavo M. Callico, *Senior Member, IEEE*

***Abstract*—The increasing prevalence of chronic non-communicable diseases makes it a priority to develop tools for enhancing their management. On this matter, Artificial Intelligence algorithms have proven to be successful in early diagnosis, prediction and analysis in the medical field. Nonetheless, two main issues arise when dealing with medical data: lack of high-fidelity datasets and maintenance of patient's privacy. To face these problems, different techniques of synthetic data generation have emerged as a possible solution. In this work, a framework based on synthetic data generation algorithms was developed. Eight medical datasets containing tabular data were used to test this framework. Three different statistical metrics were used to analyze the preservation of synthetic data integrity and six different synthetic data generation sizes were tested. Besides, the generated synthetic datasets were used to train four different supervised Machine Learning classifiers alone, and also combined with the real data. F1-score was used to evaluate classification performance. The main goal of this work is to assess the feasibility of the use of synthetic data generation in medical data in two ways: preservation of data integrity and maintenance of classification performance.**

***Index Terms* — Synthetic Data, Artificial Intelligence, Machine Learning, Generative Adversarial Networks, Classification, Data Augmentation, Imbalance**

This work was supported in part by the Spanish Government and European Union (FEDER funds) as part of support program in the context of TALENT-HExPERIA (HypErsPEctRal Imaging for Artificial intelligence applications) project, under contract PID2020-116417RB-C42, and by the project PID2019-107768RA-I00 (AAVis-BMR) and by the found action by the Community of Madrid in the framework of the Multiannual Agreement with Rey Juan Carlos University in line of action 1, "Encouragement of Young Phd students investigation"' Project Mapping-UCI (Ref F661). Moreover, this work was completed while Antonio Rodríguez was beneficiary of a pre-doctoral grant given by the "Agencia Canaria de Investigacion, Innovaci+on y Sociedad de la Información (ACIISI)" of the "Consejería de Economía, Conocimiento y Empleo", which is part-financed by the European Social Fund (FSE) (POC 2014-2020, Eje 3 Tema Prioritario 74 (85%)) and, Himar Fabelo was beneficiary of the FJC2020-043474-I funded by MCIN/AEI/10.13039/501100011033 and by the European Union "NextGenerationEU/PRTR". *(Corresponding author: Himar Fabelo).*

A. J. R.-A., H. F., S. O., F. J. B.-F., E. Q. and G. M. C. are with the Research Institute for Applied Microelectronics, University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, Spain (e-mail: aralmeida@iuma.ulpgc.es; hfabelo@iuma.ulpgc.es; sortega@iuma.ulpgc.es; fbalea@cop.es; equevedo@iuma.ulpgc.es; gustavo@iuma.ulpgc.es).

S. O. is also with Norwegian Institute of Food, Fisheries and Aquaculture Research, Tromsø, Norway.

H.F. is also with Fundación Instituto de Investigación Sanitaria de Canarias, Las Palmas de Gran Canaria, Spain.

F. J. B.-F.is also with Dept. of Psychology, Sociology and Social Work, University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, Spain.

A. D. and A. M. W. are with the Endocrinology and Nutrition Department, Complejo Hospitalario Universitario Insular Materno-Infantil and with the Institute of Biomedical and Health Research. University of Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, Spain (e-mail: aledenizgarcia@gmail.com; ana.wagner@ulpgc.es).

C. S.-R. is with the Department of Signal Theory and Communications, Telematics and Computing Systems, Rey Juan Carlos University, Fuenlabrada, Madrid, Spain (e-mail: cristina.soguero@urjc.es).

The codes developed in the current study are available from the corresponding author on reasonable request. Code is also publicly available on the following GitHub repository: (github.com/antorguez95/synthetic_data_generation_framework).

I. Introduction

ARTIFICIAL Intelligence (AI) is arising as a potential truly helpful tool in clinical practice. Risk factor analysis, disease or disorder prediction, risk estimation or image segmentation are just a few examples of how AI could help physicians in their work [1],[2]. Besides, there are some diseases that specially need tools to enhance their management due to their continuously increase in prevalence all over the world [3].

Even though AI techniques are potentially beneficial in healthcare, they need to be fed with large amounts of data. When using small databases, these algorithms are not able to generalize to the overall population, and they overfit the data they are trained on [4]. In addition, there are two main issues that arise when dealing with medical data: (i) lack of available high-fidelity datasets and (ii) maintenance of patient's data privacy [5]. The former is due to the high cost of acquiring medical data and ethical bureaucracy associated to it, apart from the fact that some institutions might be suspicious to share their data. The latter refers to the potential, inappropriate re-identification of patients, even when using anonymized data, as patient's privacy must be kept [6].

To face these problems, different synthetic data generation techniques have emerged. Synthetic data generation can be defined as the technique of creating "fake" samples from a

real dataset that faithfully represent the data taken as reference [7]. Techniques such as rotation, blurring or re-sizing have been traditionally used as part of data augmentation methods in the image processing field [8]. However, these techniques are not useful when working with tabular data that contain numerical or categorical values (e.g., Electronic Health Records) containing sociodemographic, clinical and/or analytical variables).

There is some recent work evaluating different synthetic data generation techniques for dealing with medical data. Generative Adversarial Networks (GANs) are Deep Learning (DL) algorithms based on a discriminative model that learns to determine if a sample belongs to the real data distribution or to the generated distribution generated by the generative model. The generative model creates data that are evaluated by the discriminative model, so both improve their methods until generated data and real data are indistinguishable [9]. Several types of GANs have been designed based on their scope, such as MedGAN, dedicated to imaging translation [10], or EBGAN, that use a discriminator model that can be seen as an energy model [11], among others. Besides, GANs have improved classification results in liver lesion classification using images [12], classification tasks in diabetes mellitus using tabular data [13], or cancer classification based on gene expression data [14]. Bayesian Networks (BNs) have also been recently used in synthetic data generation. BNs are representations of probability distribution structures used to explicitly represent a group of variables and their conditional dependencies [15]. This technique has demonstrated its capability to reliably generate synthetic data. In different studies, minimal discrepancies were observed respect to the real data [16], keeping the classification performance similar [17]. Statistical approaches, like copulas, have also been evaluated in synthetic data generation. They can be understood as a mathematical function that allows describing the joint distribution of multiple random variables by analyzing the dependencies between their marginal distributions [18]. COPULA-SHIRLEY framework reported its robustness in privacy maintenance for synthetic data generation [19]. Besides, copulas have also been employed for Electronic Health Records synthetic generation with promising results in data fidelity and classification results [20]. Moreover, sequential trees have been also used in synthetic data generation with short datasets and taking into account the variables that are potentially susceptible to leak sensitive information [21]. Different works based on this technique have reported success replicating oncology [22] and COVID-19 [23] data, based on different classification and statistical metrics.

Therefore, the use of synthetic data generation techniques could enhance the development and evaluation of AI-based algorithms in medical research. Considering the scarcity of publicly available well-annotated medical data, synthetic data can help this process in several ways:

a) Developing AI models using synthetic data could avoid delays regarding data protection procedures when dealing with medical data, and also facilitate their public availability [24].
b) Augmenting small datasets with trustworthy medical data could help to develop more robust AI models [25].
c) Balancing datasets by generating synthetic samples of the minority class of a given dataset would avoid the development of severely biased models [26], (e.g., training a model to predict the existence of a disease when only few of the subjects of the subjects contained in such dataset suffer from it).
d) The use of synthetic data could avoid patient's re-identification, because models have been trained with fictional patients but keeping real clinical sense [20].
e) Using synthetic data could increase the AI models adaptability by increasing dataset diversity, although generated samples would be similar to the real ones [6].
f) Synthetic data have proven to be useful in AI models pre-training phase prior real data are used [27]. Furthermore, the combination of synthetic and real data in the training phase usually enhances the performance of the models [28].

However, physicians must feel confident about using synthetic data to develop AI models that will assist diagnosis or analysis of a given disease. Therefore, synthetic data generation techniques must be evaluated using reliable statistical metrics and data visualization to ensure that the synthetic patients preserve both statistical and medical meaning of real patients. Synthetic data should resemble the real data statistically and structurally (i.e., data analytic techniques applied on the synthetic data should achieve similar results respect to the use of the real data) [29].

Hence, the main goal of this work is to analyze the feasibility of different synthetic data generation algorithms in the medical field to generate trustworthy patient data, providing an in-depth analysis of the changes in the underlying structure of the data and the relations that this might have in a classification task. Particularly, this work is focused on testing synthetic data generation techniques targeting chronic diseases such as Diabetes Mellitus (DM) [30] and Alzheimer's disease [31], among others. These chronic diseases have become a major public health concern due to their increased prevalence, being necessary to develop tools for its prediction and early diagnosis [32].

II. Material and Methods

A. Medical Tabular Databases

In this work, the following eight databases were employed to evaluate the proposed synthetic data generation framework. Database were selected if they fulfilled the following conditions: a) they should contain only tabular data; b) they were collected approaching a binary classification problem (i.e., existence or nor existence of a given disease); and c) ideally, they contained information of a chronic disease. Table I summarizes the most relevant information of the selected databases: studied disease, original dimensions, existence of missing data, database balance or imbalance, and the amount of numerical, categorical and binary features contained on each database.

TABLE I
SUMMARY OF THE EIGHT MEDICAL DATABASES USED IN THIS WORK

Disease	Abbreviation	Subjects × features	Missing data	Controls/Cases (%)	Num./Cat./Bin. Features
Alzheimer [33]	*MNCD*	85 × 37	Yes	46/54	20/15/2
Alzheimer*	*MNCD-RED*	299 × 8	Yes	50/50	1/1/6
Diabetes Bangladesh** [34]	*BANG*	306 × 21	Yes	50/50	4/6/11
Early Diabetes [35]**	*EarlyDM*	520 × 16	No	38/62	1/1/14
Heart Disease [36]**	*HeartDis*	303 × 13	No	46/54	5/3/5
Kidney Chronic Disease [37]**	*Kidney*	400 × 24	Yes	37/63	10/6/8
Diabetes PIMA** [38]	*PIMA*	768 × 8	Yes	65/35	8/0/0
South Africa Cardio [39]**	*SACardio*	462 × 9	No	65/35	8/0/1

Num.: Numerical; Cat.: Number of Categories; Bin.: Number of Binary Features.
* This database was generated including more subjects in the MNCD, but less number of features.
** Databases publicly available in the portal *kaggle.com*

B. Synthetic Data Generation Techniques

In this work, two different synthetic data generation techniques were used. Firstly, algorithms designed to balance imbalanced datasets were applied. Once the proportion of controls and cases was well-adjusted, algorithms to augment data from the whole dataset were used. We expected that data balance would avoid introducing bias in the model due to the existence of a majority class, while data augmentation would improve the generalization of the model. In addition, training models with synthetic data would contribute to better maintain real patient's privacy.

1) Data Balancing Methods

To balance the datasets, two widely used methods were applied: SMOTE (Synthetic Minority Over-sampling TEchnique) [40] and ADASYN (ADAptive SYNthetic Sampling) [41].

The basis of the SMOTE algorithm is to oversample the minority class introducing random samples along the line segments joining any (or all) *k* minority sample neighbors. Apart from the original SMOTE implementations, four additional variants of the original algorithm were tested. The K-Means SMOTE applies a K-Means clustering before oversampling with SMOTE [42]. The SVM SMOTE detects samples to use as a reference through a SVM (Support Vector Machine) classifier prior to oversampling [43]. The Borderline SMOTE algorithm detects the borderline samples of each class and only the minority examples near the borderline are oversampled [44]. The SMOTE for Nominal and Continuous (Nominal-SMOTE) is a variant of SMOTE designed to work also with categorical data [40].

The ADASYN algorithm can be considered as an improvement of the SMOTE algorithm. Whereas the SMOTE algorithm generates arbitrary minority examples, this method uses weighted distributions for different minority class examples. The harder minority examples are to learn, the more likely they are to be generated [41].

2) Data Augmentation Methods

After data balance was performed, data augmentation was carried out. Aiming this, two different algorithms were employed: Gaussian Copulas [18] and Conditional Tabular Generative Adversarial Networks (CTGANs) [45]. The choice of these two methods was motivated by the fact that Gaussian Copula is an statistical approach and CTGAN a Deep Learning approach. Thus, a comparison between both approaches could provide a detailed insight of how different (or similar) those models behave after being trained with the same datasets which include synthetic data.

The Gaussian Copula is a copula constructed from a multivariate normal distribution, capable to reproduce a large variety of multivariate distributions.

CTGANs are GANs specifically designed to model tabular data, prepared to overcome the non-Gaussian and multimodal distributions and imbalanced datasets [45] Since these datasets, and in general Electronic Health Records, contain tabular data, the novelty of this algorithm, and the possibilities that it offers to generate samples under certain given conditions, CTGANs have also been employed in this work.

C. Machine Learning (ML) Techniques

For the classification task, four different ML supervised classifiers were used to test if the inclusion of synthetic data in the training process worsened, improved or kept constant their performances. The selected classifiers were SVM [46], Random Forest (RF) [47], K-Nearest Neighbors (KNN) [48] and XGBoost (XGB), which is a gradient boosting algorithm [49].

D. Synthetic Data Generation Evaluation Metrics

To determine if the different synthetic data generation techniques faithfully preserve (or not) the original underlying structure of the different datasets, three metrics were used based on their use in the literature of synthetic data generation and evaluation in medical data [15],[50]. When studying synthetic data generation metrics with different datasets, the obtained results must be carefully analyzed. Notice that all metrics are size-dependent. Hence, the larger the real dataset, the higher the value of each metric would be expected to be. Subsequently, values of the metrics were only evaluated within the same dataset, not compared with others. Conversely, the relations and changes between these metrics, classification performance and synthetic data size were compared within different datasets.

1) Pairwise Correlation Difference (PCD)

PCD measures if the synthetic data linear correlations correspond with the linear correlations in the real data. A value equal to zero means that all linear correlations have been replicated. The higher the value, the worse the linear correlation preservation. This parameter is measured in terms of Frobenius norm (F) of Pearson correlation ($Corr$) matrices [50] following (1), where X_R and X_S are the real and synthetic data matrices, respectively, and $\|\cdot\|_F$ is the Frobenius norm.

$$PCD(X_R, X_S) = ||Corr(X_R) - Corr(X_S)||_F \quad (1)$$

2) Maximum Mean Discrepancy (MMD)

MMD is a kernel based statistical test used to determine whether two distributions are the same. Lower MMD values indicate higher similarity distribution. It has been proven to be effective evaluating GANs [51]. This parameter is computed at the dataset level following (2), where $k(\cdot)$ is a linear kernel.

$$MMD(X_R, X_S) = k(X_R \,.\, X_R) - 2k(X_R \,.\, X_S) + k(X_S \,.\, X_S) \quad (2)$$

3) Kullback-Leibler Divergence (KLD)

KLD measures how different a probability distribution of a discrete variable V is from the reference one. This parameter is computed at feature level, not at dataset level. Hence, the KLD values of each feature are summed to obtain a single value, not measuring dependencies among variables [50]. KLD is computed following (3), where P_V and Q_V are the probability distributions of the real and synthetic data, respectively, and i represents the elements of the probability distributions. Notice that zero cannot be included in the values obtained by $\frac{P_V(i)}{Q_V(i)}$. For this reason, in those cases, zero values are substituted by 10^{-8}. This operation does not alter the meaning of the measure.

$$KLD(P_V, Q_V) = \sum_{i=1}^{|v|} P_V(i) log \frac{P_V(i)}{Q_V(i)} \quad (3)$$

E. Classification Evaluation Metrics

Accuracy, Area Under the Curve and F1-Score metrics were used in this work to measure the classification performance [52]. However, since one of the main objectives of this work is to deal with imbalanced data, and some of the datasets were tested with an imbalanced data subset, F1-Score was selected as the reference metric, which is the harmonic mean of precision and sensitivity [53]. It is computed following (4), where *TP* are the true positives, *FP* are the false positives and *TN* refers to true negatives.

$$F1 - Score = \frac{TP}{TP + \frac{1}{2}(FP + TN)} \quad (4)$$

F. Proposed Processing Framework

Fig. 1 shows the data processing framework proposed in this work for the evaluation, using ML algorithms, of the tabular data balance and augmentation methods. This framework was developed in Python programming language, using `sklearn` [54], `imblearn` [55] and `sdv` [56] libraries for the ML and synthetic data generation implementations. The datasets described on Section II.A were employed to evaluate this framework. Next, each step of the processing framework is explained in detail.

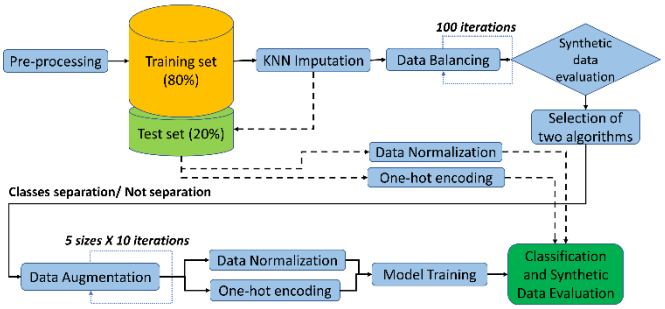

Fig. 1. Proposed processing framework for the evaluation of different tabular data balance and augmentation methods using ML algorithms.

1) Data pre-processing and partition

Raw data were analyzed and pre-processed, when necessary (e.g., zero values in the "Insulin" variable of the *PIMA* database were considered as missing data in this work). Then, the dataset was partitioned into training (80%) and test (20%) sets.

2) Data imputation

Missing data of both subsets were imputed using a KNN imputation method [57]. Considering one instance of the dataset having one missing value in a certain variable, this method will find K instances similar to that one, and will compute the weighted average in such variable to fill the missing one. Each instance of the test set was imputed independently using the training set [33]. From here, the processing framework was only applied on the training set, while the test set was used on the models' validation step.

3) Data balance

Data balancing methods were applied to avoid problems related to training the ML model using an imbalanced dataset. Since the generation of samples presents a certain level of randomness, this step was performed 100 times, computing the mean and standard deviation (std) of the abovementioned metrics. Based on this, the two methods that showed the best results (see Section III.A for a detailed analysis of these results) in terms of statistical data similarity were employed in the data augmentation algorithms.

4) Data augmentation

In this step, five different sizes of synthetic data generation were tested: (i) quarter (+25%), (ii) half (+50%), (iii) the same size (+100%), (iv) double (+200%) and (v) quadruple (+400%) of the original size of the dataset. Fig. 2 illustrates the proportion of real and synthetic data used to train the ML models.

These experiments will provide an insight of how data structure preservation and classification performance vary depending on the amount of synthetic data samples generated from a certain dataset. Data augmentation step was repeated 10 times for each synthetic data proportion, computing the mean and std of performance metrics. Thus, as in the balancing step, variability through different iterations was analyzed. Notice that data augmentation is more computationally demanding than data balance, since a larger number of samples is generated. Hence, instead of 100 (as in the balancing step), only 10 were executed in this step.

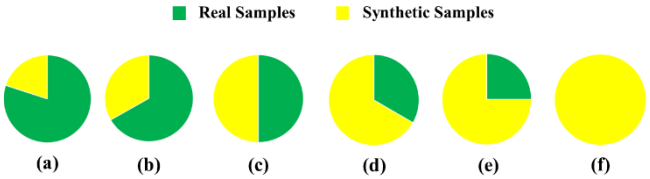

Fig. 2. Scheme of the proportion of real and synthetic training sets using to train ML with different data sizes. (a) +25%; (b) +50%; (c) +100%; (d) +200%; (e) +400% (f) only synthetic data.

Data were augmented following two different approaches: a) using the entire balanced dataset, and b) using the balanced cases and controls separately. Once augmented, synthetic data were again evaluated to check which methods better preserve the underlying structure of the balanced dataset. Since categorical data do not have numerical meaning and similarity metrics do, this evaluation was performed after coding those variables using *one-hot-encoding* [58]. Notice that the reference dataset in this evaluation varies depending on the balance method previously used.

5) Data standardization

After computing data augmentation metrics, numerical variables were standardized as described in (5) by centering and scaling the samples with the mean and std of each feature, where X' is the normalized matrix, X is the original matrix, μ is the mean of the original matrix and σ is the std of the original matrix. Each instance of the test set was standardized independently using the μ and σ obtained from the training set.

$$X' = \frac{X - \mu}{\sigma} \tag{5}$$

6) ML model training and optimization

This final step consists of training the ML models with the generated synthetic data combined with the real training set. A grid search method [59] was used to optimize the most relevant hyperparameters of each ML algorithm [60], [61]. This was performed following a 10-fold cross validation approach with the training set. Later, the models were tested using the test set. Hyperparameters for each model and their search space are detailed in Table II. The objective function for the optimization was F1-Score since this framework has been designed to properly work with highly imbalanced datasets. After this, the obtained results were evaluated, assessing the influence of synthetic data generation on the different classification tasks for each dataset. Reference performance results were obtained by using the real dataset without including synthetic data to train the AI models.

TABLE II
OPTIMIZED HYPERPARAMETERS AND SEARCH SPACE FOR EACH CLASSIFIER

Classifier	Hyperparameters	Search Space
SVM	Kernel	['rbf', 'linear']
	C	[0.1,1,2.5,5,10]
	Gamma	[0.01, 0.1, 1, 10]
RF	No. of estimators	[20, 50,100, 200]
	Maximum no. of features per tree	[2,3,5,7]
XGB	Learning rate	[0.01, 0.1, 0,5]
	No. of estimators	[20, 50, 100, 200]
KNN	No. of neighbors	[6, 8, 10, 12, 14, 16]
	Weights	['uniform', 'distance']

G. Statistical analysis

To try to elucidate if the proposed framework for synthetic data generation could be promising in the clinical practice, synthetic data generation twice the size of the real dataset with the best synthetic data generation combination for the PIMA database was performed. All variables of this dataset were numerical and continuous, so a Kolmogorov-Smirnov test assuming normality for all variables was performed with a significance level of 95% to explore if augmented and reference dataset came from the same distribution [62], (i.e., the null hypothesis H_0 was that the two distributions were identical. Python module `scipy.stats` was used for this step [63].

III. EXPERIMENTAL RESULTS AND DISCUSSION

Next, the experimental results obtained in each step of the processing framework will be presented.

A. Data Balancing

Fig. 3 illustrates two examples of the analysis performed to choose between the five balancing algorithms in Section II.B. This was carried out for the eight datasets described in Section II.A On The left, the triangular radar charts show the mean of the three metrics (PCD, KLD and MMD), in logarithmic scale. The best algorithm is the one that obtains the highest triangle area, where the metrics are optimal. On the right, boxplots corresponding to the different metrics are shown. The most robust algorithm through 100 iterations is the one presenting a tighter box, (i.e., a lower IQR (Interquartile Range)), meaning less variability. Notice that, as the variability in the synthetic data generation increases, the reliability of the algorithm decreases.

Regarding balancing algorithms particularities, ADASYN algorithm did not converge in *MNCD*, *MNCD-RED* and *HeartDis* databases since they were already balanced (see Table I). K-Means SMOTE did not find a minimum number of neighbors to produce samples for *MNCD* and *EarlyDM* databases. Finally, Nominal-SMOTE was not used in *PIMA* and *SACardio* databases, since they do not contain any categorical features.

For simplicity, only the results from two databases were shown. Fig. 3.a shows the results from *MNCD* database, which was almost perfectly balanced before the balancing step, and contains many categorical variables. On the other hand, Fig. 3.b illustrates the result after balancing *PIMA* database, a fully numerical database drastically imbalanced before this step.

From the comparison of both results, it is clear that, generating less samples to balance the datasets implies potentially better similarity metrics, (i.e., the more imbalanced the real dataset is, the more different will it be after the balancing step). Related to this, the boxplots show that all algorithms are more robust compared in the well-balanced database (*MNCD*) than in the imbalanced one (*PIMA*). These observations can be extrapolated to the results obtained in the six remaining databases.

Table III shows the two selected balancing algorithms for each database after performing the evaluation of such algorithms, and also details the proposed combined methods (data balance followed by data augmentation) to be studied in the next experiments. The algorithms choice was based the mean value of all metrics after 100 executions and its variability.

The obtained results reveal that there is not a unique algorithm that has superior performance for all databases. This could be caused by the different nature of each database. However, as shown in Table III, Borderline-SMOTE was one of the chosen balan[cing algorithms in six out of eight databases, whereas ADASYN was one of the selected algorithms in four out of five databases for which it converged. Furthermore, Nominal-SMOTE was chosen in four out of six times when categorical variables were present. Conversely, K-Means SMOTE was the only algorithm that did not outperform the rest in any case, while SMOTE and SVM-SMOTE were selected just once. Thus, ADASYN and Borderline-SMOTE has presented the most robust performances among the eight medical tabular databases studied in this work.

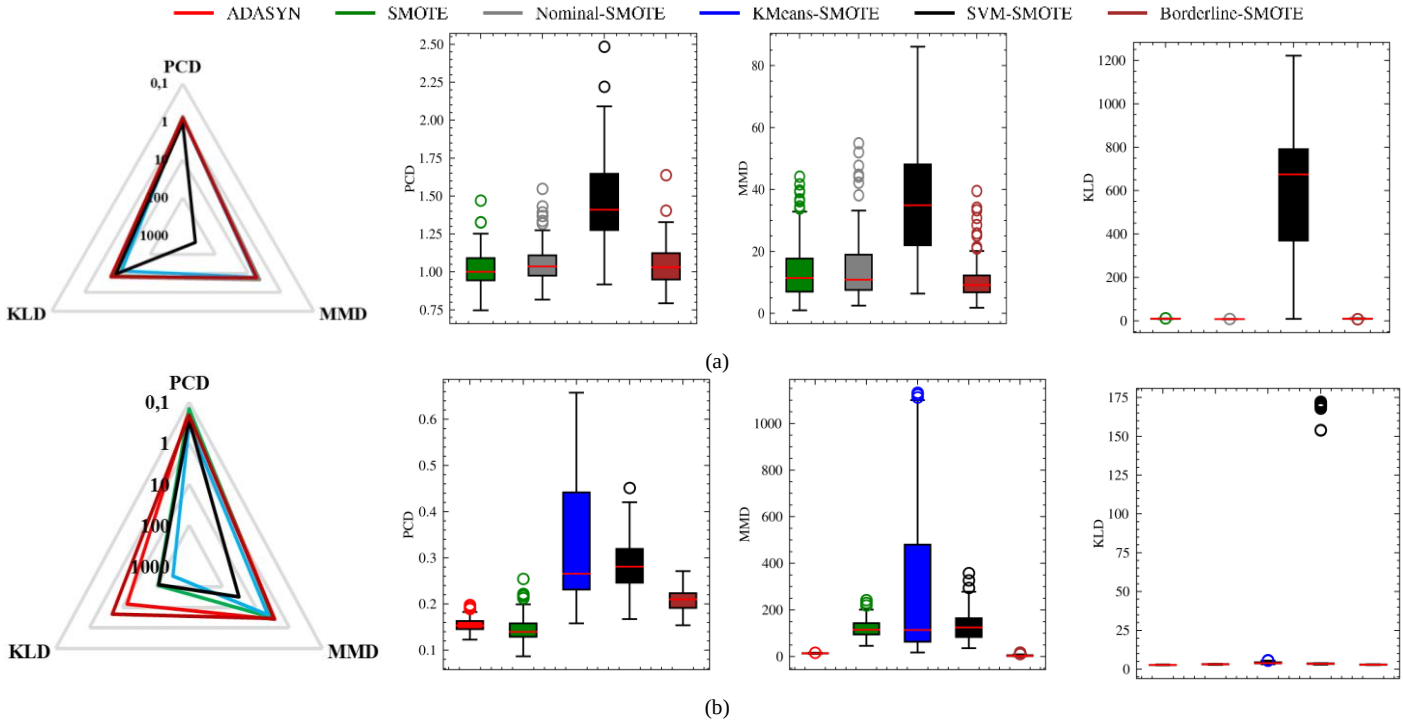

Fig. 3. Triangular radar charts placed on the left represent the mean of the studied metrics in logarithmic scale. On the right, boxplots of statistical metrics after 100 iterations (a) *MNCD*, (b) *PIMA*. Notice that in *PIMA* (a) there are no categorical variables (thus, Nominal-*SMOTE* was no computed), and in *MNCD* (b) ADASYN is not represented since it did not converge.

As an illustrative example, Fig. 4 compares the algorithms variability through 100 iterations in the *Glucose* variable histogram of the *PIMA* database. There is one red histogram per cell that represents the reference distribution, from which data balancing is performed. There are one hundred black histograms per cell, which represent the balancing executions performed by the corresponding algorithm. ADASYN and Borderline-SMOTE (Fig 5.a and Fig 5.d, respectively) are clearly the most robust algorithms, whereas K-SMOTE (Fig 5.c) presents more variability (high IQR) when balancing the dataset, which agrees with the results presented in Fig. 3.a.

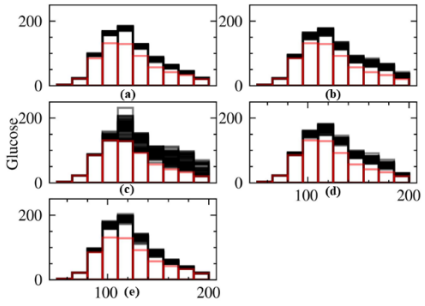

Fig. 4. Example of a comparison of the balancing algorithms in the generation of the *Glucose* variable for the PIMA database through 100 iterations. (a) ADASYN; (b) SMOTE; (c) K-SMOTE; (d) SVM-SMOTE; (e) Borderline-SMOTE.

A. Data Augmentation

Fig. 5 shows the influence of the number of synthetic samples generated in the statistical metrics (PCD, KLD and MMD) with the data augmentation algorithms for the eight databases. In these results, average values of the similarity metrics through 10 iterations are represented. Notice that, in these sample sizes, real samples are mixed with the synthetic samples in the proportions previously established in Fig. 2, except from the last point, where only synthetic data is evaluated. Moreover, these metrics are size-dependent. Their value is expected to be higher with a larger number of features.

In all databases, except for *BANG*, PCD and KLD increased nearly linearly when the number of synthetic data samples increase (i.e., correlations among variables are not perfectly preserved and the original distributions have changed), as shown in first and third rows of Fig. 5.c). In the case of the *BANG* (Fig. 5.c), this might be related due to high number of binary features.

TABLE III

SELECTED BALANCING ALGORITHM FOR EACH DATABASE AND EVALUATED COMBINED METHODS

Database	Selected BA	Combined Methods*
MNCD	Nominal-SMOTE (1) Borderline-SMOTE (2)	
MNCD-RED	Nominal-SMOTE (1) Borderline-SMOTE (2)	
BANG	ADASYN (1) Borderline-SMOTE (2)	
EarlyDM	Nominal-SMOTE (1) SVM-SMOTE (2)	BA+CTGAN BA+ Gaussian Copula BA+SEP+CTGAN BA+SEP+Gaussian Copula
HeartDis	SMOTE (1) Nominal-SMOTE (2)	
Kidney	ADASYN (1) Borderline-SMOTE (2)	
PIMA	ADASYN (1) Borderline-SMOTE (2)	
SACardio	ADASYN (1) Borderline-SMOTE (2)	

*There are eight combined methods in total for each database. BA: selected algorithms for the data balance in each database. SEP: partition between control and cases before data augmentation.

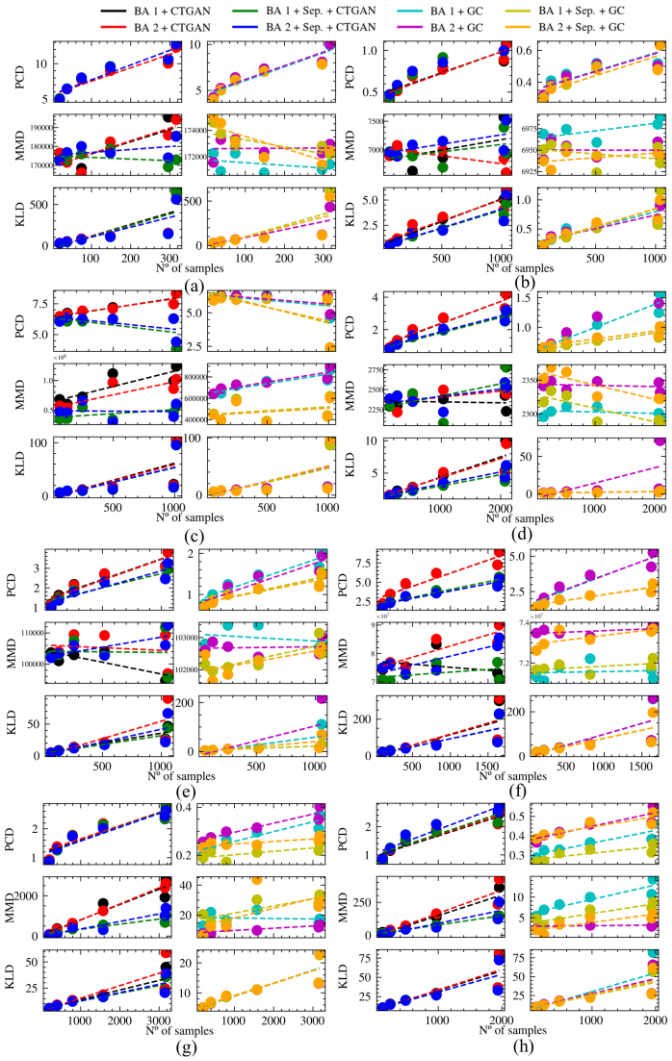

Fig. 5. Statistical metric results depending on the number of synthetic samples generated (according to Fig. 2) with different the different combinations of algorithms (Table II). (a) *MNCD*; (b) *MNCD-RED*; (c) *BANG*; (d) *EarlyDM*; (e) *HeartDis*; (f) *Kidney*; (g) *PIMA*; (h) *SACardio*. Dashed lines represent the linear approximations of the result trends. BA (1): Selected BA (1); BA (2): Selected BA (2) (see Table III).

Regarding MMD, only in *PIMA* (Fig. 5.g) and *SACardio* (Fig. 5.h) databases (mainly numerical variables) present a linear increment with synthetic data samples. In the rest of databases, these metrics in some cases decreased as the number of synthetic data samples increased (e.g., MMD in Fig. 5.d and Fig. 5.e), and in other cases the metrics increased (e.g., Fig. 5.f).

Furthermore, there was not observed any drastic worsening (i.e., increment) of any metrics when only synthetic data was used to evaluate the statistical similarity respected to the real data. In general, linear correlations and statistical distributions were fairly maintained in all synthetic databases.

Finally, Gaussian Copula-based combinations generally showed lower values in all metrics in each database. This means that, with the used configuration, Gaussian Copula fits better to the real data. This is not necessarily positive, since it could be related (or not) to an overfitting trend of the real dataset.

B. Classification Performance

Table IV summarizes the classification performance results obtained in the experiments carried out in this work (all the results generated can be found together with the codes in the GitHub repository). This table shows the best improvement achieved in the F1-score metric (*highest F1-score upgrade*) for each database, using the best combination of data balance and augmentation algorithms with a certain ML model and a percentage of synthetic data used for training the model (see Fig. 2). This value is computed compared with the reference result obtained using such ML model trained only with real data. Furthermore, in the same way, the highest decrease in F1-score is shown (*highest F1-score downgrade*). F1-score values are the average of 10 executions.

In five out of eight databases, the combination of synthetic and real data for training ML models achieved an improvement in the classification performance. On this regard, it is especially remarkable the improvement in *MNCD-RED* and *SACardio*, improving 0.22 and 0.18 the reference F1-score, respectively. Besides, *BANG* reference performance ($F1 - score = 1$) was the same as training the model with synthetic data. In the cases of *EarlyDM* and *HeartDis* the performances were lower than the reference. Nonetheless, the best case showed a decrement of only 0.01, which means that it is fairly near to the reference results.

It is worth noticing that some of the the best results were obtained when the amount of synthetic data was four times (+400%) the amount of real data (*MNCD-RED* and *PIMA*), and also when only synthetic data were used to train the model (*BANG*). Furthermore, training ML models only with synthetic data obtained improvements of 0.03 and 0.11 with *PIMA* and *SACardio* databases, respectively. Notice that these last two are the only databases with no categorical features, so this could be related with the challenges of modelling categorical variables by CTGAN and Gaussian Copula.

On the other hand, drastic decrements of F1-score were also found by using synthetic data, reaching -0.55 respect with the reference result in the worst scenario (*EarlyDM*). According to this result, it can be stated that the use of different synthetic data generation algorithms must be carefully analyzed depending on the use case. This idea is

reinforced by the unusual occurrence that, for *EarlyDM*, both the best and the worst classification performance are produced by the same combinations of synthetic data generation algorithms. In the case of the highest F1-score downgrades, five out of eight cases occurred when only synthetic data were used to train the model. This fact suggests that those combinations of algorithms produced low-quality data.

It is worth noticing that most of the worst results were produced by the CTGAN with no data splitting before data augmentation. This will be further analyzed later in the discussion, but it could be related by the fact that the CTGAN treats the target variable as an independent variable. Thus, this can produce a higher imbalance in the synthetic dataset compared to the real dataset, what may lead to biased models. In contrast, Gaussian Copula was present in five out of eight of the best classification results. Additionally, Gaussian Copula offered, in general, less variability in classification performance.

TABLE IV
SUMMARY OF CLASSIFICATION PERFORMANCE USING SYNTHETIC DATA IN ALL DATABASES

Database	Highest F1-score upgrade	Combined Method	ML model	Synthetic Data	Highest F1-score downgrade	Combined Method	ML model	Synthetic Data
MNCD	**+0.05**	**NC + GC**	**XGB**	**+25% & +200%**	**-0.5**	**NC + Sep. + CTGAN**	**RF**	**+200%**
MNCD-RED	**+0.22**	**NC + CTGAN**	**XGB**	**+25%**	**-0.19**	**BS + Sep. + GC**	**KNN**	**+100%**
BANG*	**0.00**	**ADASYN + GC**	**KNN**	**only-synth**	**-0.45**	**ADASYN + CTGAN**	**KNN**	**only-synth**
*EarlyDM***	-0.01	NC + CTGAN	SVM	**+25%**	-0.55	NC + CTGAN	XGB	only-synth
*HeartDis***	-0.01	NC + Sep. + GC	RF	**+400%**	-0.43	SMOTE + CTGAN	KNN	only-synth
Kidney	**+0.02**	**ADASYN + CTGAN**	**XGB**	**+100%**	**-0.27**	**ADASYN + CTGAN**	**KNN**	**only-synth**
PIMA	**+0.04**	**BS + GC**	**XGB**	**+400%**	**-0.16**	**BS + CTGAN**	**XGB**	**+200%**
SACardio	**+0.18**	**BS + GC**	**XGB**	**+400%**	**-0.32**	**BS + CTGAN**	**XGB**	**only-synth**

**** SVM, KNN and XGB reached F1-score=1.0, so an upgrade of 0 means that training with synthetic data achieved also F1-score=1.0.***
***** No F1-score upgrade is reached; the lowest downgrade is shown.***

On the one hand, Fig. 6.a shows the highest improvement of F1-score using the *MNCD-RED* and the XGB model. On the other hand, Fig. 6.b shows the worst-case scenario using the *EarlyDM*, where the XGB showed the lowest performance in the F1-score metric. In this figure, the reference classification values (using only real data for training) are represented with horizontal dashed lines for each classifier; bullets represent the mean value of the F1-Score through 10 iterations, and the error bars represent the F1-Score standard deviation. The combined method employed to obtain the results are detailed in Table IV.

In both results of Fig. 6 it is clear that generating more synthetic data does not necessarily imply achieve better or worse classification performance. Hence, there is not a unique solution for all databases, having to analyze each database independently. Classification Performance vs. synthetic data generation metrics

In order to reveal some insights on the extent to which the statistical similarity metrics and the classification performance are related, PCD, KLD and MMD were plotted versus F1-Score. One could expect that better statistical metrics (i.e., more similar to the real dataset) imply better classification. Nonetheless, no direct or inverse relations between metrics and classification performance were found among different classifiers and synthetic data generation methods. Sometimes, better statistical metrics implied better classification, but not always. Fig. 7 illustrates the abovementioned occurrence by showing PCD in *PIMA* (Fig. 7.a), MMD in *SACardio* (Fig. 7.b) and KLD in *Kidney* (Fig. 7.c), all versus F1-score.

C. Statistical Analysis

Kolmogorov-Smirnov test reveals that, except for *Diabetes Pedigree Function* variable ($p < 0.05$, i.e., the null hypothesis is rejected, and synthetic distribution does not come from the same distribution that the reference one), all features belong to the same distribution as their analogous in the real dataset. As an example, Fig. 8 shows the original distribution (red), the distribution after data balance (green) and the distribution after the data augmentation (blue) for the *Glucose* (Fig. 8.a) and the *Diabetes Pedigree Function* (Fig. 8.b) features. The latter is the only feature that cannot be considered from the same distribution (after data augmentation) as the original feature according to the Kolmogorov-Smirnov test ($p < 0.05$ comparing real vs. augmented datasets).

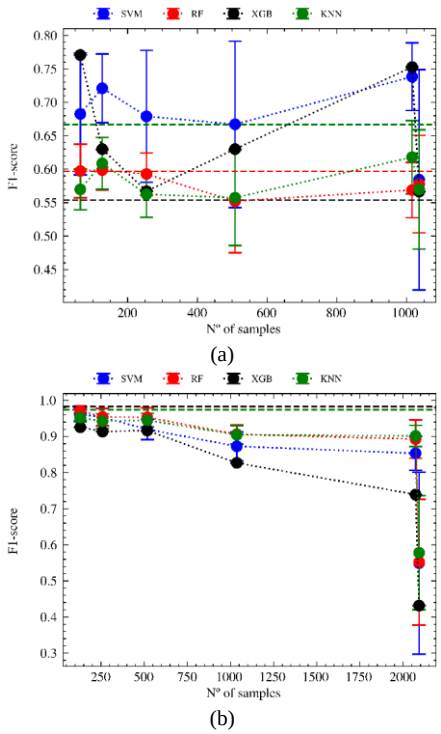

Fig. 6. F1-score versus training data size for (a) the highest upgrade (*MNCD-RED)* and (b) the highest downgrade (*EarlyDM*) of F1-score through the 8 databases. Dashed lines represent the reference results computed using the real data. SVM and RF references overlap in (a). In (b), XGB and RF overlaps, and so do SVM and KNN.

D. Discussion

As proven in this work, good classification performance is not necessarily related to perfectly keeping data underlying structure when generating synthetic data. Thus, synthetic data generation algorithms must ensure not just achieving good classification or prediction results, but also that statistical and clinical meaning of the data is kept. The application of AI-based algorithms in the real world within medical applications are conditioned by clinicians and physicians' needs. They do not want just achieving an automatic classification to determine if a subject does or does not have a certain disease. They want to know also why, when, and how the disease was developed [64]. For this reason, the advantages and limitations on the use of synthetic data for training AI-based algorithms have been studied in this work, analyzing the changes on linear correlations and distribution similarities between synthetic and real data using different synthetic data generation methods.

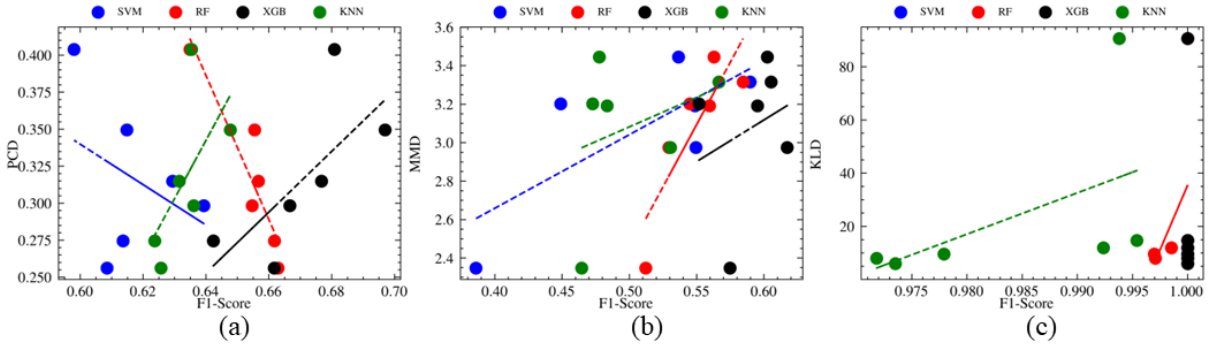

Fig. 7. Statistical metrics vs. F1-Score. (a) PCD vs. F1-score in *PIMA*, (b) MMD vs. F1-score in *SACardio*, and (c) KLD vs. F1-score in *BANG*. Dashed lines represent the linear approximations of the result trends.

Synthetic data similarity metrics employed to analyze numerical variables generated by synthetic data generation methods seem to be meaningful. In some cases, but not always, (*PIMA* and *SACardio*), better similarity metrics sometimes imply better classification performance. Nonetheless, in the case of categorical variables this assumption cannot be established. Further research is needed to evaluate why does this occur.

The observed correlation lost using synthetic data generation might (or might not) be reduced by using datasets larger than the reference. In this sense, the ideal case is, on the one hand, to compare the similarity metrics between different subsets of real data among them, and, on the other hand, to compare synthetic data with real data [17]. However, for this purpose, datasets with a large number of individuals are required.

It has been shown that Gaussian Copula works well generating synthetic data that preserve linear correlations. This is highly relevant when working with numerical data, as it has been demonstrated previously [18],[20]. Without the need for intensive optimization or tuning, these algorithms have outperformed reference classification results (without the presence of synthetic data) in some cases without compromising the underlying structure of the real data according to the synthetic data generation metrics studied. Therefore, Gaussian Copula has proven to be effective to generate synthetic patient data without the need of deep knowledge. The statistical analysis supports the fact that Gaussian Copula-generated data replicates real data. However, this good performance replicating the real data could be related with overfitting, so this must be studied cautiously

In the case of CTGANs, these algorithms require an optimization process to find the best hyperparameters configuration that will provide the best results for a certain application. In this work, only default values have been used for the CTGANs configurations. However, further work including optimization will be done in the future. CTGANs consider the independent variable of the database (i.e., the feature that indicates if a subject is a case or a control) as one more feature [65]. If data are generated without setting the condition to just generate control or cases, instances will be created randomly, not keeping the balance of the previous step. This partially explains the bad performance that this algorithm offers when increasing synthetic data samples. This can be overcome by setting this condition, significantly improving the CTGANs performance. Hence, this fact suggests that a detailed and well-studied optimization of this algorithm (tuning the learning rates, batch size, or number of epochs, among others) could potentially generate trustworthy synthetic data.

If synthetic data are not identical but very similar to the real data, patients' privacy could still be at risk. Although the approach of this work was not beyond the detailed analysis of the synthetic data itself, privacy preservation has been considered as a future work line. On this regard, Natural Language Processing has been employed to detect critical information from the Electronic Health Records, such as names or birth dates, and mask or remove them [66]. In a recently celebrated data privacy challenge, a noise-injection approach showed the best balance between data utility and privacy preservation [67]. Finally, only by using GANs, the work performed by Yale et. al [68] showed good performance in terms of privacy metrics and data utility. Nonetheless, both the development of robust de-identification methods and privacy-preservation metrics are emerging fields that need to be further studied [67].

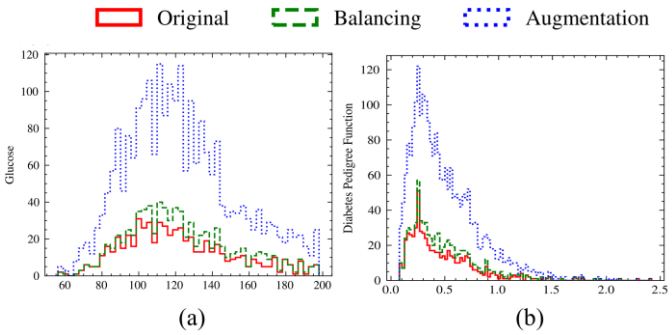

Fig. 8. Example of a comparison between the original distribution (red), the distribution after data balancing (green) and the distribution after data augmentation (blue) for (a) the *Glucose* ($p > 0.05$) and (b) Diabetes Pedigree Function ($p < 0.05$) from the PIMA database.

IV. .CONCLUSIONS

The proposed framework demonstrates the utility of different synthetic data generation algorithms for the generation of synthetic data in eight different medical tabular databases. It has been exhaustively tested with real data in small and imbalanced databases with different sizes, fairly keeping classification performance compared to the reference results using only real data and, in more than the half of the cases, improving that performance.

On the one hand, Gaussian Copula-based methods do not require deep knowledge or hyperparameter tuning to offer good performance on tabular data. Linear correlations and distribution similarities are well-kept in many cases. On the other hand, CTGANs need deeper knowledge to optimize their performance, yet in some cases, they have shown promising results. The fact that Gaussian Copula showed better results than CTGANs does not mean that Gaussian Copula should unequivocally be selected over CTGANs.

As a conclusion, although synthetic data generation and its analysis should be further studied, the results shown in this, and other published works, are promising. In terms of classification tasks and real data similarity, positive results have been obtained. As future lines, a deep CTGAN study and tuning should be assess. Developing tools for the interpretability of this algorithm to "open the black box" would also clarify the synthetic data generation process for AI researchers and physicians. Finally, a robust clinical validation of synthetic data generation by physicians after detailed statistical and performance analysis of synthetic data generation could enhance the use of this technique in the clinical field, accelerating the development of AI-based algorithms that could assist during clinical practice.

REFERENCES

[1] T. Davenport and R. Kalakota, "The potential for artificial intelligence in healthcare," *Future Healthcare Journal*, vol. 6, no. 2, pp. 94–98, Jun. 2019, doi: 10.7861/futurehosp.6-2-94.

[2] K. H. Yu, A. L. Beam, and I. S. Kohane, "Artificial intelligence in healthcare," *Nature Biomedical Engineering 2018 2:10*, vol. 2, no. 10, pp. 719–731, Oct. 2018, doi: 10.1038/s41551-018-0305-z.

[3] M. Subramanian *et al.*, "Precision medicine in the era of artificial intelligence: implications in chronic disease management," *Journal of Translational Medicine 2020 18:1*, vol. 18, no. 1, pp. 1–12, Dec. 2020, doi: 10.1186/S12967-020-02658-5.

[4] D. M. Hawkins, "The Problem of Overfitting," *Journal of Chemical Information and Computer Sciences*, vol. 44, no. 1, pp. 1–12, Jan. 2004, doi: 10.1021/ci0342472.

[5] R. Coppen *et al.*, "Will the trilogue on the EU Data Protection Regulation recognise the importance of health research?," *The European Journal of Public Health*, vol. 25, no. 5, pp. 757–758, Oct. 2015, doi: 10.1093/eurpub/ckv149.

[6] R. J. Chen, M. Y. Lu, T. Y. Chen, D. F. K. Williamson, and F. Mahmood, "Synthetic data in machine learning for medicine and healthcare," *Nature Biomedical Engineering*, vol. 5, no. 6, pp. 493–497, Jun. 2021, doi: 10.1038/s41551-021-00751-8.

[7] T. E. Raghunathan, "Synthetic Data," *Annual Review of Statistics and Its Application*, vol. 8, no. 1, pp. 129–140, Mar. 2021, doi: 10.1146/annurev-statistics-040720-031848.

[8] H.-C. Shin *et al.*, "Medical Image Synthesis for Data Augmentation and Anonymization Using Generative Adversarial Networks," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11037 LNCS, Springer, Cham, 2018, pp. 1–11. doi: 10.1007/978-3-030-00536-8_1.

[9] I. J. Goodfellow *et al.*, "Generative Adversarial Nets," *Advances in Neural Information Processing Systems*, vol. 27, 2014, Accessed: Feb. 08, 2022. [Online]. Available: http://www.github.com/goodfeli/adversarial

[10] K. Armanious *et al.*, "MedGAN: Medical image translation using GANs," *Computerized Medical Imaging and Graphics*, vol. 79, p. 101684, Jan. 2020, doi: 10.1016/J.COMPMEDIMAG.2019.101684.

[11] L. Gonog and Y. Zhou, "A Review: Generative Adversarial Networks," in *2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, Jun. 2019, pp. 505–510. doi: 10.1109/ICIEA.2019.8833686.

[12] M. Frid-Adar, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "Synthetic data augmentation using GAN for improved liver lesion classification," in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, Apr. 2018, vol. 2018-April, pp. 289–293. doi: 10.1109/ISBI.2018.8363576.

[13] C. A. Hargreaves and W. L. E. Heng, "Simulation of Synthetic Diabetes Tabular Data Using Generative Adversarial Networks," *Elements*, 2021.

[14] P. Chaudhari, H. Agrawal, and K. Kotecha, "Data augmentation using MG-GAN for improved cancer classification on gene expression data," *Soft Computing*, vol. 24, no. 15, pp. 11381–11391, Aug. 2020, doi: 10.1007/s00500-019-04602-2.

[15] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian Network Classifiers," *Machine Learning 1997 29:2*, vol. 29, no. 2, pp. 131–163, 1997, doi: 10.1023/A:1007465528199.

[16] D. Kaur *et al.*, "Application of Bayesian networks to generate synthetic health data," *Journal of the American Medical Informatics Association*, vol. 28, no. 4, pp. 801–811, Mar. 2021, doi: 10.1093/jamia/ocaa303.

[17] A. Tucker, Z. Wang, Y. Rotalinti, and P. Myles, "Generating high-fidelity synthetic patient data for assessing machine learning healthcare software," *npj Digital Medicine 2020 3:1*, vol. 3, no. 1, pp. 1–13, Nov. 2020, doi: 10.1038/s41746-020-00353-9.

[18] P. Xue-Kun Song, "Multivariate Dispersion Models Generated From Gaussian Copula," *Scandinavian Journal of Statistics*, vol. 27, no. 2, pp. 305–320, Jun. 2000, doi: 10.1111/1467-9469.00191.

[19] S. Gambs, F. Ladouceur, A. Laurent, and A. Roy-Gaumond, "Growing synthetic data through differentially-private vine copulas," *Proceedings on Privacy Enhancing Technologies*, vol. 2021, no. 3, pp. 122–141, Jul. 2021, doi: 10.2478/popets-2021-0040.

[20] Z. Wang, P. Myles, and A. Tucker, "Generating and evaluating cross-sectional synthetic electronic healthcare data: Preserving data utility and patient privacy," *Computational Intelligence*, vol. 37, no. 2, pp. 819–851, May 2021, doi: 10.1111/coin.12427.

[21] J. P. Reiter, "Using CART to Generate Partially Synthetic Public Use Microdata." Accessed: Jun. 14, 2022. [Online]. Available: https://www.scb.se/contentassets/ca21efb41fee47d293bbee5bf7be7fb3/using-cart-to-generate-partially-synthetic-public-use-microdata.pdf

[22] K. el Emam, L. Mosquera, and C. Zheng, "Optimizing the synthesis of clinical trial data using sequential trees," *Journal of the American Medical Informatics Association*, vol. 28, no. 1, pp. 3–13, Jan. 2021, doi: 10.1093/jamia/ocaa249.

[23] K. el Emam, L. Mosquera, E. Jonker, and H. Sood, "Evaluating the utility of synthetic COVID-19 case data," *JAMIA Open*, vol. 4, no. 1, Mar. 2021, doi: 10.1093/jamiaopen/ooab012.

[24] B. Nowok, G. M. Raab, and C. Dibben, "Providing bespoke synthetic data for the UK Longitudinal Studies and other sensitive data with the synthpop package for R1," *Statistical Journal of the IAOS*, vol. 33, no. 3, pp. 785–796, Aug. 2017, doi: 10.3233/SJI-150153.

[25] Y. Park and J. Ghosh, "PeGS: Perturbed Gibbs Samplers that Generate Privacy-Compliant Synthetic Data," *TRANSACTIONS ON DATA PRIVACY*, vol. 7, pp. 253–282, 2014.

[26] M. J. Pencina, B. A. Goldstein, and R. B. D'Agostino, "Prediction Models — Development, Evaluation, and Clinical Application," *New England Journal of Medicine*, vol. 382, no. 17, pp. 1583–1586, Apr. 2020, doi: 10.1056/NEJMP2000589/SUPPL_FILE/NEJMP2000589_DISCLOSURES.PDF.

[27] M. Hernandez, G. Epelde, A. Alberdi, R. Cilla, and D. Rankin, "Synthetic data generation for tabular health records: A systematic review," *Neurocomputing*, vol. 493, pp. 28–45, Jul. 2022, doi: 10.1016/J.NEUCOM.2022.04.053.

[28] M. Hernandez *et al.*, "Incorporation of Synthetic Data Generation Techniques within a Controlled Data Processing Workflow in the Health and Wellbeing Domain," *Electronics 2022, Vol. 11, Page 812*, vol. 11, no. 5, p. 812, Mar. 2022, doi: 10.3390/ELECTRONICS11050812.

[29] Y. Yue, Y. Li, K. Yi, and Z. Wu, "Synthetic Data Approach for Classification and Regression," in *2018 IEEE 29th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*, Jul. 2018, vol. 2018-July, pp. 1–8. doi: 10.1109/ASAP.2018.8445094.

[30] "Diabetes," *World Health Organization*. https://www.who.int/news-room/fact-sheets/detail/diabetes (accessed Feb. 08, 2022).

[31] H. Niu, I. Álvarez-Álvarez, F. Guillén-Grima, and I. Aguinaga-Ontoso, "Prevalencia e incidencia de la enfermedad de Alzheimer en Europa: metaanálisis," *Neurología*, vol. 32, no. 8, pp. 523–532, Oct. 2017, doi: 10.1016/j.nrl.2016.02.016.

[32] I. Journal, A. Pawar, and S. Mary, "IRJET-Artificial Intelligence in Medicine and Healthcare Cite this paper Artificial Intelligence in Medicine and Healthcare," *Journal Use of Technology in Healt hCare IRJET Journal IRJET-Use of Technology in Healt hCare IRJET Journal International Research Journal of Engineering and Technology*, vol. 9001, 2008, Accessed: Feb. 08, 2022. [Online]. Available: www.irjet.net

[33] F. J. Balea-Fernandez *et al.*, "Analysis of Risk Factors in Dementia through Machine Learning," *Journal of Alzheimer's Disease*, vol. 79, no. 2, 2021, doi: 10.3233/JAD-200955.

[34] S. Asaduzzaman, F. al Masud, T. Bhuiyan, K. Ahmed, B. K. Paul, and S. A. M. M. Rahman, "Dataset on significant risk factors for Type 1 Diabetes: A Bangladeshi perspective," *Data in Brief*, vol. 21, pp. 700–708, Dec. 2018, doi: 10.1016/j.dib.2018.10.018.

[35] "Early Stage Diabetes Risk Prediction Dataset | Kaggle." https://www.kaggle.com/datasets/ishandutta/early-stage-diabetes-risk-prediction-dataset (accessed Jun. 21, 2022).

[36] "Heart Disease Cleveland UCI | Kaggle." https://www.kaggle.com/datasets/cherngs/heart-disease-cleveland-uci (accessed Jun. 21, 2022).

[37] "UCI Machine Learning Repository: Chronic_Kidney_Disease Data Set." https://archive.ics.uci.edu/ml/datasets/chronic_kidney_disease (accessed Jun. 21, 2022).

[38] P. Bennett, T. Burch, and M. Miller, "Diabetes Mellitus in American (PIMA) Indians," *The Lancet*, vol. 298, no. 7716, pp. 125–128, Jul. 1971, doi: 10.1016/S0140-6736(71)92303-8.

[39] "Cardiovascular Disease | Kaggle." https://www.kaggle.com/datasets/yassinehamdaoui1/cardiovascular-disease (accessed Jun. 21, 2022).

[40] N. v. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.

[41] Haibo He, Yang Bai, E. A. Garcia, and Shutao Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, Jun. 2008, pp. 1322–1328. doi: 10.1109/IJCNN.2008.4633969.

[42] G. Douzas, F. Bacao, and F. Last, "Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE," *Information Sciences*, vol. 465, pp. 1–20, Oct. 2018, doi: 10.1016/j.ins.2018.06.056.

[43] Q. Wang, Z. Luo, J. Huang, Y. Feng, and Z. Liu, "A Novel Ensemble Method for Imbalanced Data Learning: Bagging of Extrapolation-SMOTE SVM," *Computational Intelligence and Neuroscience*, vol. 2017, pp. 1–11, 2017, doi: 10.1155/2017/1827016.

[44] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3644 LNCS, Springer, Berlin, Heidelberg, 2005, pp. 878–887. doi: 10.1007/11538059_91.

[45] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling Tabular Data using Conditional GAN."

[46] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their Applications*, vol. 13, no. 4, pp. 18–28, Jul. 1998, doi: 10.1109/5254.708428.

[47] L. Breiman, "Random Forests," *Machine Learning 2001 45:1*, vol. 45, no. 1, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324.

[48] O. Kramer, "K-Nearest Neighbors," Springer, Berlin, Heidelberg, 2013, pp. 13–23. doi: 10.1007/978-3-642-38652-7_2.

[49] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, doi: 10.1145/2939672.

[50] 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, May 2020, doi: 10.1186/S12874-020-00977-1/TABLES/17.

[51] D. J. Sutherland *et al.*, "Generative Models and Model Criticism via Optimized Maximum Mean Discrepancy," *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*, Nov. 2016, Accessed: Feb. 08, 2022. [Online]. Available: http://arxiv.org/abs/1611.04488

[52] A. J. Larner, "The 2x2 Matrix," *The 2x2 Matrix*, 2021, doi: 10.1007/978-3-030-74920-0.

[53] H. Huang, J. Wang, and H. Abudureyimu, "Maximum F1-score discriminative training for automatic mispronunciation detection in computer-assisted language learning," *13th Annual Conference of the International Speech Communication Association 2012, INTERSPEECH 2012*, vol. 1, pp. 814–817, 2012, doi: 10.21437/INTERSPEECH.2012-248.

[54] "Scikit-learn: machine learning in Python — scikit-learn 1.0.2 documentation." https://scikit-learn.org/stable/ (accessed Feb. 08, 2022).

[55] "Imbalanced-learn documentation — Version 0.9.0." https://imbalanced-learn.org/stable/ (accessed Feb. 08, 2022).

[56] "SDV - The Synthetic Data Vault — SDV 0.13.1 documentation." https://sdv.dev/SDV/ (accessed Feb. 08, 2022).

[57] O. Troyanskaya *et al.*, "Missing value estimation methods for DNA microarrays," *Bioinformatics*, vol. 17, no. 6, pp. 520–525, Jun. 2001, doi: 10.1093/bioinformatics/17.6.520.

[58] M. Hardy, "Regression with Dummy Variables," *Regression with Dummy Variables*, May 2012, doi: 10.4135/9781412985628.

[59] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for Hyper-Parameter Optimization," *Advances in Neural Information Processing Systems*, vol. 24, 2011.

[60] N. Mohapatra, K. Shreya, and A. Chinmay, "Optimization of the Random Forest Algorithm," in *Lecture Notes on Data Engineering and Communications Technologies*, vol. 37, Springer, Singapore, 2020, pp. 201–208. doi: 10.1007/978-981-15-0978-0_19.

[61] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, Nov. 2020, doi: 10.1016/J.NEUCOM.2020.07.061.

[62] F. J. Massey, "The Kolmogorov-Smirnov Test for Goodness of Fit," *J Am Stat Assoc*, vol. 46, no. 253, pp. 68–78, 1951, doi: 10.1080/01621459.1951.10500769.

[63] "Statistical functions (scipy.stats) — SciPy v1.8.0 Manual." https://docs.scipy.org/doc/scipy/reference/stats.html (accessed Feb. 15, 2022).

[64] F. Wang, R. Kaushal, and D. Khullar, "Should health care demand interpretable artificial intelligence or accept 'black Box' Medicine?," *Annals of Internal Medicine*, vol. 172, no. 1, pp. 59–61, Jan. 2020, doi: 10.7326/M19-2548.

[65] B. Vega-Márquez, C. Rubio-Escudero, and I. Nepomuceno-Chamorro, "Generation of Synthetic Data with Conditional Generative Adversarial Networks," *Logic Journal of the IGPL*, Nov. 2020, doi: 10.1093/jigpal/jzaa059.

[66] C. A. Libbi, J. Trienes, D. Trieschnigg, and C. Seifert, "Generating Synthetic Training Data for Supervised De-Identification of Electronic Health Records," *Future Internet 2021, Vol. 13, Page 136*, vol. 13, no. 5, p. 136, May 2021, doi: 10.3390/FI13050136.

[67] "Hide-and-Seek Privacy Challenge: Synthetic Data Generation vs. Patient Re-identification." https://proceedings.mlr.press/v133/jordon21a (accessed Jun. 15, 2022).

[68] "Privacy Preserving Synthetic Health Data - Inria." https://hal.inria.fr/hal-02160496/ (accessed Jun. 15, 2022).