Neural Networks 153 (2022) 339–348

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Privacy preserving Generative Adversarial Networks to model Electronic Health Records

Rohit Venugopal, Noman Shafqat, Ishwar Venugopal, Benjamin Mark John Tillbury, Harry Demetrios Stafford, Aikaterini Bourazeri *

School of Computer Science and Electronic Engineering, University of Essex, Wivenhoe Park, Colchester CO4 3SQ, United Kingdom

ARTICLE INFO

Article history:
Received 18 October 2021
Received in revised form 13 May 2022
Accepted 16 June 2022
Available online 25 June 2022

Keywords:
AI
GAN
Machine learning
Privacy
Public health data

ABSTRACT

Hospitals and General Practitioner (GP) surgeries within National Health Services (NHS), collect patient information on a routine basis to create personal health records such as family medical history, chronic diseases, medications and dosing. The collected information could be used to build and model various machine learning algorithms, to simplify the task of those working within the NHS. However, such Electronic Health Records are not made publicly available due to privacy concerns. In our paper, we propose a privacy-preserving Generative Adversarial Network (pGAN), which can generate synthetic data of high quality, while preserving the privacy and statistical properties of the source data. pGAN is evaluated on two distinct datasets, one posing as a Classification task, and the other as a Regression task. Privacy score of generated data is calculated using the Nearest Neighbour Adversarial Accuracy. Cosine similarity scores of synthetic data from our proposed model indicate that the data generated is similar in nature, but not identical. Additionally, our proposed model was able to preserve privacy while maintaining high utility. Machine learning models trained on both synthetic data and original data have achieved accuracies of 74.3% and 74.5% respectively on the classification dataset; while they have attained an R2-Score of 0.84 and 0.85 on synthetic and original data of the regression task respectively. Our results, therefore, indicate that synthetic data from the proposed model could replace the use of original data for machine learning while preserving privacy.

1. Introduction

Hospitals and General Practitioner (GP) surgeries generally hold a large amount of patients' health data such as family medical history, chronic diseases, medications, dosing, vaccinations and so on. Because of the enormous amount of health data collected from the patients, it is quite challenging to manage and maintain it. However, the increasing amounts of public health data requires a secure and collaborative system that will improve data transparency and help the public health ministry to provide the best affordable access.

Hospitals and GP surgeries within a National Health Service (NHS) or private partnership collect patient information on a routine basis; this information is either discarded or sent to a central research centre; for example, a partnered University (Baker et al., 2009). This allows the researcher to create and distribute the data by specifying privacy and also help the public health centres for better data management. However, centrally storing such massive amounts of sensitive data as well as giving third-parties access to such data raises privacy concerns. Furthermore, with data breaches becoming more and more common in recent times, various nations have introduced new laws in order to regulate the transmission and storage of data. Some of these include the GDPR[1] in the European Union and the CCPA[2] in the United States of America.

While such laws help in regulating data usage and transmission to protect user privacy, it also hinders the scientific community as acquiring useful data becomes a complicated and long drawn out legal process. Therefore, in this paper, we present a novel approach, where a Generative Adversarial Network (GAN) is used to statistically model an input dataset, and generate synthetic data. The generated data will preserve the statistical properties of the original health records while compressing it, which reduces the risk of original patient information being compromised. Furthermore, since the generated data will not be as sensitive in nature, it can be stored and shared without additional privacy concerns.

The motivation behind using privacy-preserving Generative Adversarial Network (pGAN) for Electronic Health Records is to

* Corresponding author.
E-mail address: a.bourazeri@essex.ac.uk (A. Bourazeri).

[1] https://gdpr-info.eu.
[2] https://oag.ca.gov/privacy/ccpa.

https://doi.org/10.1016/j.neunet.2022.06.022

test the preposition that an appropriate GAN architecture is capable of generating synthetic data of high privacy and utility, while at the same time maintaining a similar distribution as the one of the original data.

Accordingly, this paper is structured as follows. Section 2 provides more details on the background, motivation and rationale for this work, focusing mainly on GANs and similar approaches that have been used in the past. Section 3 presents the proposed approach we followed to model our data and also the datasets we chose, while Section 4 describes our experimental results, which show that our approach preserves personal privacy, while managing to maintain the distribution and utility of the original data. We summarise and conclude in Section 5 with the argument that these results show significant improvement in performance for models trained on data generated using our approach, while some future research directions are also included in this section.

2. Background & motivation

Electronic Health Records have been widely adopted by hospitals and GP surgeries over the last years, and therefore new technologies are required to provide patient de-identification and data augmentation. GANs, specifically, can help with these issues as they can improve data de-identification ensuring data's privacy and security.

2.1. Generative Adversarial Networks

A Generative Adversarial Network (GAN) is composed of two neural network systems, which in turn 'compete' with each other for the generation of new synthetic instances of the real data. This architecture can be used to create synthetic data in domains like images (Karras, Aila, Laine, & Lehtinen, 2017; Radford, Metz, & Chintala, 2015), music (Briot, Hadjeres, & Pachet, 2017; Yang, Chou, & Yang, 2017), speech (Pascual, Bonafonte, & Serra, 2017) and so on, and hence, have been widely used in the fields of image, video and voice generation. Generating discrete data using GANs can be challenging in nature. Che et al. (2017), Kusner and Hernández-Lobato (2016) both address this problem by either modifying the loss function or by designing other special functions to build a differential model.

A GAN system comprises of a generator and a discriminator. Fig. 1 visualises the structure of the GAN. In this figure, **C** represents the concatenation operation. The generator takes as input a latent space vector, and then models it to produce synthetic data that preserves the distribution and correlation of the original dataset. The discriminator's task is to identify whether an input presented to it is real or fake. A discriminator is in effect, a binary classifier. Gradients from the discriminator backpropagate through the network in order to update the weights of both the generator and discriminator. In an ideal situation, Nash equilibrium will be achieved between the generator and the discriminator. Berthelot, Schumm, and Metz (2017), Gulrajani, Ahmed, Arjovsky, Dumoulin, and Courville (2017), Salimans et al. (2016) all discuss various techniques and methods to stabilise and speed up the training of GANs. Once synthetic data of high confidence is produced, it can then be applied to the same domain as the original data.

2.2. Related work

With data breaches becoming common in recent years, privacy concerns for data, especially sensitive data such as medical electronic health records (EHR), have gone up. As a result, data sharing and privacy have witnessed an increase in attention from

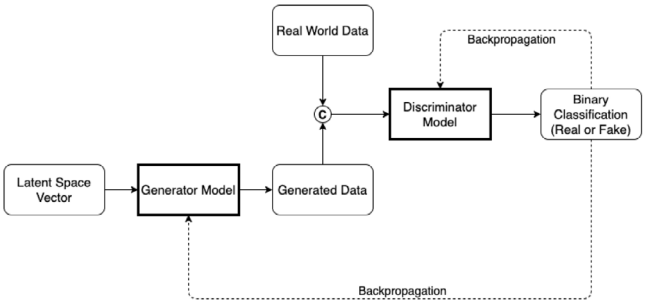

Fig. 1. Structure of GAN.

the research community. Recently, Federated Learning has garnered a lot of attention, as it proposes a system which enables secure data sharing as well as learning capabilities (Li et al., 2019). A federated learning system usually incorporates some type of Differential Privacy (Dwork, 2008) algorithm as a privacy mechanism. Similarly, the area of blockchain has also witnessed a lot of attention as a way to provide secure access and share data. Healthchain (Chenthara, Ahmed, Wang, Whittaker, & Chen, 2020) proposes a novel blockchain-based method for preserving the privacy of medical health records. However, these two areas are out of the scope of our paper. Henceforth, we shall limit our discussions to techniques and methods which try to preserve the privacy of sensitive information by anonymising the data; and increase the utility of data by modelling it.

Miotto, Li, Kidd, and Dudley (2016) proposed a deep learning method to extract a general purpose feature representation from patient Electronic Health Record (EHR) data. This representation was extracted by making use of a three stack layer of denoising auto-encoders; the feature representation was used for clinical modelling. Clinical modelling on these deep feature representations significantly outperformed the traditional approach of normal feature extraction. While this approach helped extract a feature representation which increased the utility of the dataset, the resulting privacy of the dataset was not addressed. Malekzadeh, Clegg, and Haddadi (2017) introduced the Replacement Auto-encoder, which given time-series data, transforms sensitive information into non-sensitive components to protect the user's privacy. This novel approach was able to preserve the privacy of sensitive information, while also being able to produce good results when fed into various machine learning models. The disadvantage of this approach was that, in the event of data leak, including non-sensitive data, a GAN could then be trained to potentially identify if a given data is real or fake. As such, in such scenarios, the privacy offered by this approach is being reduced.

Scardapane, Altilio, Ciccarelli, Uncini, and Panella (2018) proposed a technique where the dataset was distributed among multiple clinical parties, and was not stored in a centralised location due to privacy concerns. Any inference or data mining procedure applied to the dataset relied on the Euclidean distance among patterns in the data, spectral clustering, and Kernel methods. The experimental results showed that the proposed approach was efficient in performing both clustering and classification in distributed medical data. The approach presented in Scardapane et al. (2018) mainly addressed the privacy concern by distributing and storing the dataset in different locations and then accessing only small portions of it. Sadati, Nezhad, Chinnam, and Zhu (2019) did a comparative study of using different deep learning architectures to extract feature representation from EHR. They implemented and made use of methods such as stacked sparse auto-encoders, deep belief networks, adversarial and variational auto-encoders for feature representation, and obtained a

higher-level abstraction that can be used for predictive modelling. The study showed that for small datasets, stacked auto-encoders performed well, however for larger datasets, variational and adversarial auto-encoders outperformed the others due to their ability to learn feature representation as well as its distribution.

Choi et al. (2017) implemented a GAN that generated synthetic patient data from the original dataset which preserved the relationship and distribution amongst the features, and as such, could be used in the future for predictive modelling and other tasks, while maintaining the privacy of the original dataset. They further proposed and made use of a technique, which made sure that the synthetic data generated was as close as possible to the original data, while still being different. Another approach presented by Xu and Veeramachaneni (2018), caters to time-series data by making use of Recurrent Neural Networks (RNN) inside their Generator. Yale et al. (2019) presented Nearest Neighbour Adversarial Accuracy, a privacy estimation metric. The metric was tested on various GANs such as medGAN (Choi et al., 2017) and Wasserstein GANs (Arjovsky, Chintala, & Bottou, 2017; Gulrajani et al., 2017) to gauge its privacy score. Privacy results for medGAN were not as high as expected.

Torfi (2020) proposed a domain-agnostic metric which can be used to evaluate the quality of synthetic data produced. Furthermore, the paper also proposed a new framework, where auto-encoders are used to help the GAN produce non-continuous data; and which enforces Rényi differential privacy (Mironov, 2017) within the system (Torfi, 2020). Yale et al. (2020) extend their previous work (Yale et al., 2019) by detailing their methodology to produce synthetic data as well as their metric to evaluate the privacy quality of synthetic data.

2.3. Contributions

In this paper, we focus on three aspects; Distribution, Privacy and Utility. There have been approaches to model these aspects separately, with an importance being given to Distribution and Utility (Miotto et al., 2016; Sadati et al., 2019), or for Privacy (Xu & Veeramachaneni, 2018; Yale et al., 2020), however in our paper we present a novel GAN architecture which is capable of generating synthetic data of high Privacy and Utility, while maintaining a similar Distribution as that of the original data.

3. Methodology

In this paper, we model our data with the help of GANs, and then proceed to perform a 3-fold evaluation of the modelled data. To maintain the simplicity of our network architecture, we employ Multi-Layer Perceptrons (MLP) for our Generator and Discriminator. The general structure of GAN has already been explained in Section 2.

The generator consists of six fully connected layers with Batch Normalisation ($momentum = 0.8$) applied to each layer. The first two layers made use of Rectified Linear Unit (ReLu) as an activation function and the next three layers made use of Leaky ReLu ($\alpha = 0.2$). The activation function of the generator's final layer can be modified with respect to data and the task at hand. The output of the generator along with the real data, are fed in as the input to the discriminator. The discriminator follows a similar structure and consists of two fully connected layers with a Leaky ReLu activation ($\alpha = 0.2$) and dropout with a probability of 0.2. A deeper network for the generator is used, since it is tasked with modelling the data, which is considered to be complex. The system architecture schematic for both the generator and the discriminator can be seen in Fig. 2 (see Tables 1 and 2).

One of the drawbacks of GAN is the instability of the network while training, and the potential for mode collapse, where the discriminator performs really well which leads to the gradient of the generator to vanish, due to which the generator fails at learning. In order to deal with mode collapse, most GAN training methodologies train the generator for more steps than the discriminator. In our proposed approach, we have made use of dropout in the discriminator network to ensure that the model converges slower, while ensuring the robustness of the discriminator.

In recent years, Batch Normalisation and Dropout have been used with varying degrees of success to help build more robust and stable neural network models. The use of Batch Normalisation has been preferred over the years, due to its tendency to improve performance and reduce convergence time (Bjorck,

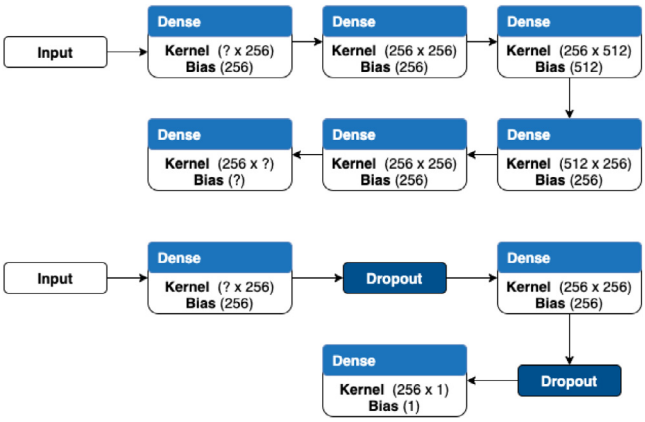

Fig. 2. The architecture of the Generator (top) and the Discriminator (bottom).

Table 1
Generator Architecture. Input shape depends on the dataset used. In the final layer, output shape of Dense has been denoted as n. Here n is the number of columns or attributes in the original dataset that has to be modelled.

Layer	Output shape	Number of parameters
Dense	(None, 256)	2048
Batch Normalization	(None, 256)	1024
ReLU	(None, 256)	0
Dense	(None, 256)	65792
Batch Normalization	(None, 256)	1024
ReLU	(None, 256)	0
Dense	(None, 512)	131584
LeakyReLU	(None, 512)	0
Batch Normalization	(None, 512)	2048
Dense	(None, 256)	131328
LeakyReLU	(None, 256)	0
Batch Normalization	(None, 256)	1024
Dense	(None, 256)	65792
LeakyReLU	(None, 256)	0
Batch Normalization	(None, 256)	1024
Dense	(None, n)	257xn
Total number of parameters:		402688+257xn

Table 2
Discriminator Architecture.

Layer	Output shape	Number of parameters
Dense	(None, 256)	2048
LeakyReLU	(None, 256)	0
Dropout	(None, 256)	0
Dense	(None, 256)	65792
LeakyReLU	(None, 256)	0
Dropout	(None, 256)	0
Dense	(None, 1)	257
Total number of parameters:		68097

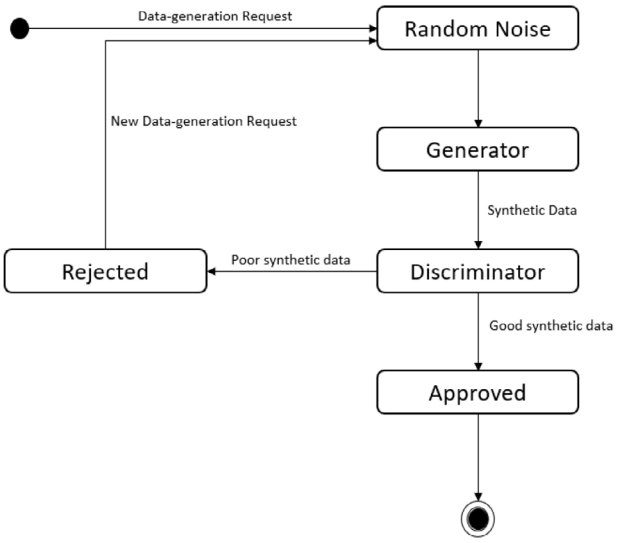

Fig. 3. Work flow of data generation in GAN.

Gomes, Selman, & Weinberger, 2018). On the other hand, while dropout helps prevent a network from overfitting, it can be noticed that it delays the convergence if a very small dropout rate is used (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). In our proposed approach, we utilise Batch Normalisation to stabilise and help our generator converge faster; meanwhile the dropout delays the learning process of the discriminator while ensures its robustness. This ensures that our generator learns faster while the discriminator slows down, thereby preventing mode collapse and increasing the stability of our network.

3.1. Data generation

Once the GAN has been fully trained, the generator learns the statistical distribution of the data, while the discriminator learns to distinguish between original data points and falsified/synthetic data. We use our trained generator to produce synthetic data, which is then fed into our discriminator. We then filter out and take all the synthetic data, which the discriminator classified as original. This ensures that the output synthetic data is highly similar to the original data points. This data generation process can be seen in Fig. 3.

3.2. Datasets

For the purpose of evaluating our model architecture as well as the privacy preserving ability, we select the following two tabular medical datasets, which have been widely used and are open source.

3.2.1. Medical cost personal dataset

This dataset was initially made available as part of the book titled "Machine Learning with R" by Lantz (2019). This particular dataset was compiled for the purpose of forecasting the insurance costs and is available on the Kaggle platform. It contains 1338 instances with the following features corresponding to each row:

- Age of the primary beneficiary (Numerical value)
- The gender of the insurance contractor (Categorical Value)
- Body-mass index (Numerical value)
- Number of dependants/children covered under the health insurance (Numerical value)
- Whether the person is a smoker or not (Categorical Value)
- The beneficiary's residential area in the US (Categorical Value)
- Individual Medical costs that are billed by the health insurance (Numerical value)

Pre-processing techniques like ordinal encoding and normalisation were applied to corresponding columns in the dataset.

3.2.2. PIMA Indian diabetes dataset

(Smith, Everhart, Dickson, Knowler, & Johannes, 1988)

This dataset was originally compiled by the National Institute of Diabetes and Digestive and Kidney Diseases. It was aimed for the task of predicting whether a given patient has diabetes or not, based on the different diagnostic features included in this dataset. This dataset is subject to the constraint that all patients are females of at least 21 years of age and Pima Indian heritage. There are 768 instances with the following independent variables/features:

- Number of pregnancies (Numerical value)
- Plasma Glucose concentration in an oral glucose tolerance test (Numerical value)
- Blood pressure in units of mm Hg (Numerical value)
- Triceps skin fold thickness in units of mm (Numerical value)
- Insulin content in units of mu U/ml (Numerical value)
- Body Mass Index (Numerical value)
- Diabetes Pedigree function (Numerical value)
- Age (Numerical value)

The target variable is 'Outcome' which is a categorical variable denoting whether the patient has diabetes or not.

4. Experiments & evaluation

4.1. Training

Prior to the training, the chosen datasets are split into 80% and 20% for training and testing sets respectively. The test set will be later used to evaluate the distribution, privacy and utility of the generated data. The 80% training set will be used to train the GAN model. As opposed to training schemes where a generator is trained more than the discriminator (Goodfellow, 2016), in our proposed approach, during a single step of training, both the generator and discriminator are trained only once. Even though they are trained for equal number of steps, since we use Batch Normalization and Dropout, the generator learns faster, while the discriminator converges slower.

As a benchmark, we also used the 80% training data set to train two different models: tGAN (Xu & Veeramachaneni, 2018) and HealthGAN (Yale et al., 2020). Our proposed model, pGAN, uses a batch size of 32 and uses Adam with a learning rate of $2e^{-4}$ as an optimiser. The model was trained for a total of 150 epochs and saved after every epoch. Synthetic data was generated by each of the saved models, and the best performing model was selected. A similar strategy was used to train tGAN and HealthGAN.

4.2. Data distribution testing

The quality and distribution properties of the synthetic data generated from respective models are evaluated in this section. Testing the distribution of the data essentially means whether the features of the data learned by the generator are the same as the actual data. For this purpose, we used various statistical techniques to visualise and evaluate the distribution of the generated data, such as Principal Component Analysis (PCA) and Uniform Manifold Approximation and Projection (UMAP), and Cosine Similarity.

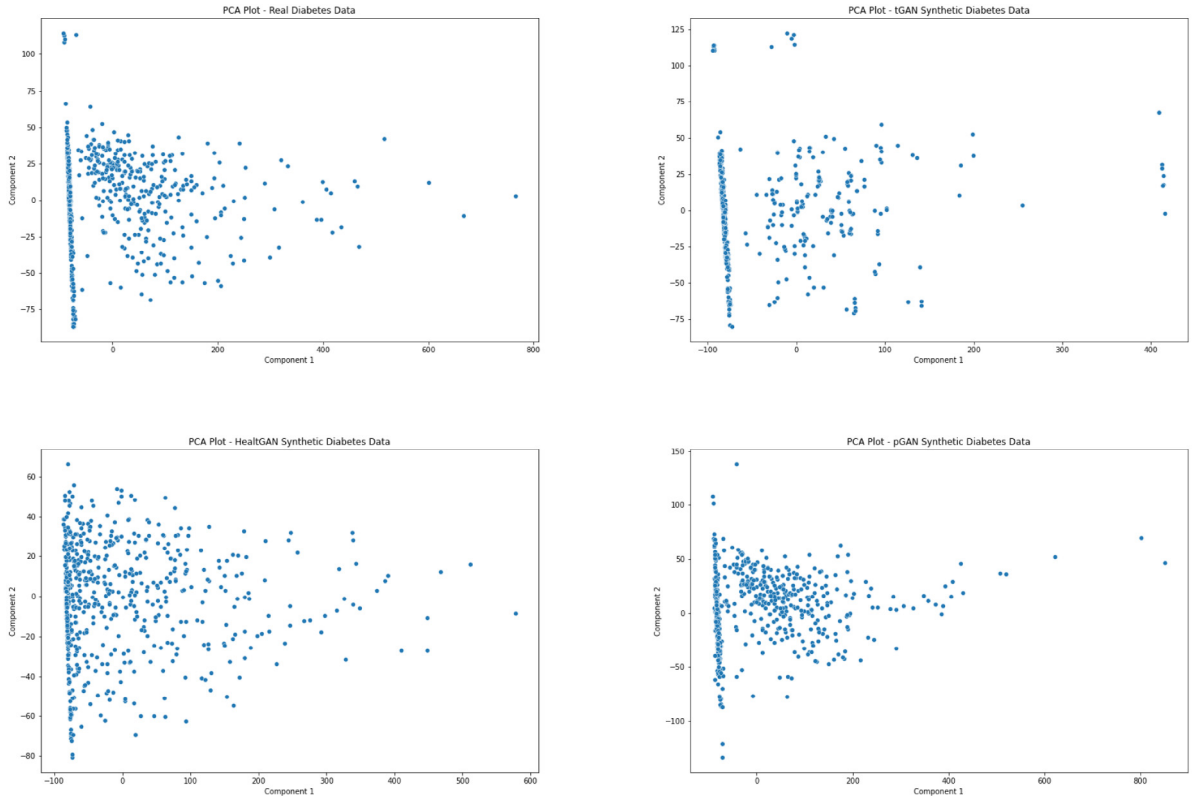

Fig. 4. PCA plots for Diabetes dataset: Real data (top left), tGAN synthetic data (top right), HealthGAN synthetic data (bottom left) and pGAN synthetic data (bottom right).

4.2.1. Diabetes dataset

Ideally, in PCA, the distribution of the synthetic data should be as close as possible to the original data, which would mean that the GAN has learnt the data distribution. UMAP is a dimensionality reduction technique which preserves global structure of the data. Figs. 4 and 5 visualise the PCA and UMAP distributions of the original data and the synthetic data from all three models. From the plots, we can observe that the synthetic data generated by pGAN is able to match the distribution of the original data relatively well.

4.2.2. Medical Cost Dataset

Figs. 6 and 7 show the PCA and UMAP plots for the Medical Cost Dataset. From the UMAP plots, we can see that all three models have come close to matching the original distribution of the data.

In addition to plotting PCA and UMAP graphs to visualise the distribution, Table 3 also shows the cosine similarities between the synthetic data and the real data points. Cosine similarity basically treats data points as vectors and calculates the angle between them. A cosine similarity score of 1 would mean that the data points are identical and pointing in the same direction, while a score of 0 would signify that the data points are orthogonal to each other (no similarity at all). When generating synthetic data, ideally we would like to obtain cosine similarities between 0.4–0.8 as this would mean that the generated data is close to the original, but is not identical. From Table 3 we can see that the cosine similarity scores of all three models are similar to each other and lie within the range of 0.4–0.8.

Table 3
Cosine similarities of synthetic data.

	Diabetes	Medical Cost
tGAN	0.668	0.515
HealthGAN	0.66	0.56
pGAN	0.679	0.503

4.3. Privacy risk testing

While training a GAN, the Discriminator checks the validity of the data generated, and this feedback helps the Generator to learn the distribution and statistical properties of the data. Since, the generated data has properties similar to the original input, it is necessary to evaluate the risk of predicting the original input using the synthetic data. To assess the privacy risk, we use Nearest Neighbour Adversarial Accuracy (NNAA) (Yale et al., 2020) between original data (S) and the generated data (T). NNAA uses Nearest Neighbours and Euclidean Distance to calculate the privacy, and is denoted by AA_{TS}.

$$AA_{TS} = \frac{1}{2}\left(A_T + A_S\right) \tag{1}$$

$$A_T = \frac{1}{n}\sum_{i=1}^{n} 1\left(d_{TS}(i) > d_{TT}(i)\right) \tag{2}$$

$$A_S = \frac{1}{n}\sum_{i=1}^{n} 1\left(d_{ST}(i) > d_{SS}(i)\right) \tag{3}$$

In the above equations, $d_{TS}(i) = min_j \|x_T^i - x_S^j\|$, is the Euclidean distance between $x_T^i \in S_T$ and the nearest neighbour S_S. Similarly, $d_{TT}(i) = min_{j,j\neq i}\|x_T^i - x_T^j\|$, is the 'leave-one-out' distance to the nearest neighbour (Yale et al., 2019). AA_{TS} gives the performance of an adversarial classifier, trying to distinguish between the real and synthetic data. An AA_{TS} score of 0.5 indicates that the two datasets are indistinguishable.

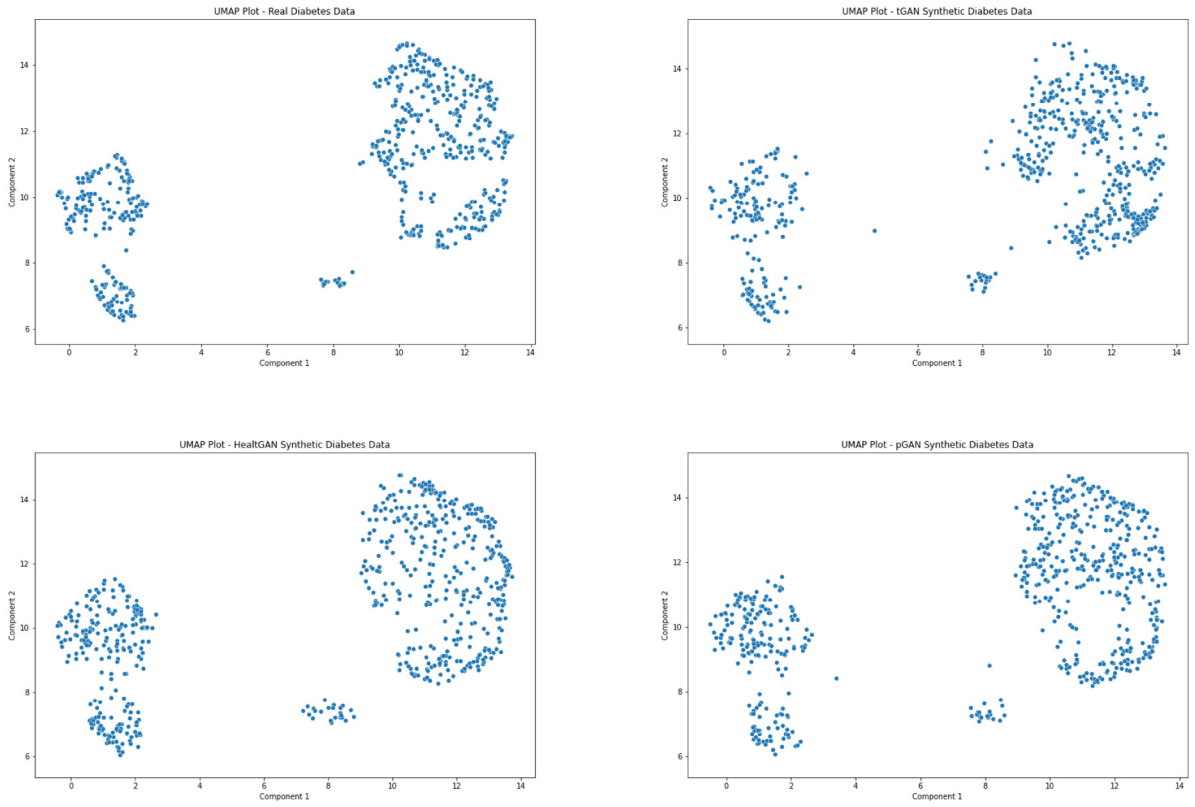

Fig. 5. UMAP plots for Diabetes dataset: Real data (top left), tGAN synthetic data (top right), HealthGAN synthetic data (bottom left) and pGAN synthetic data (bottom right).

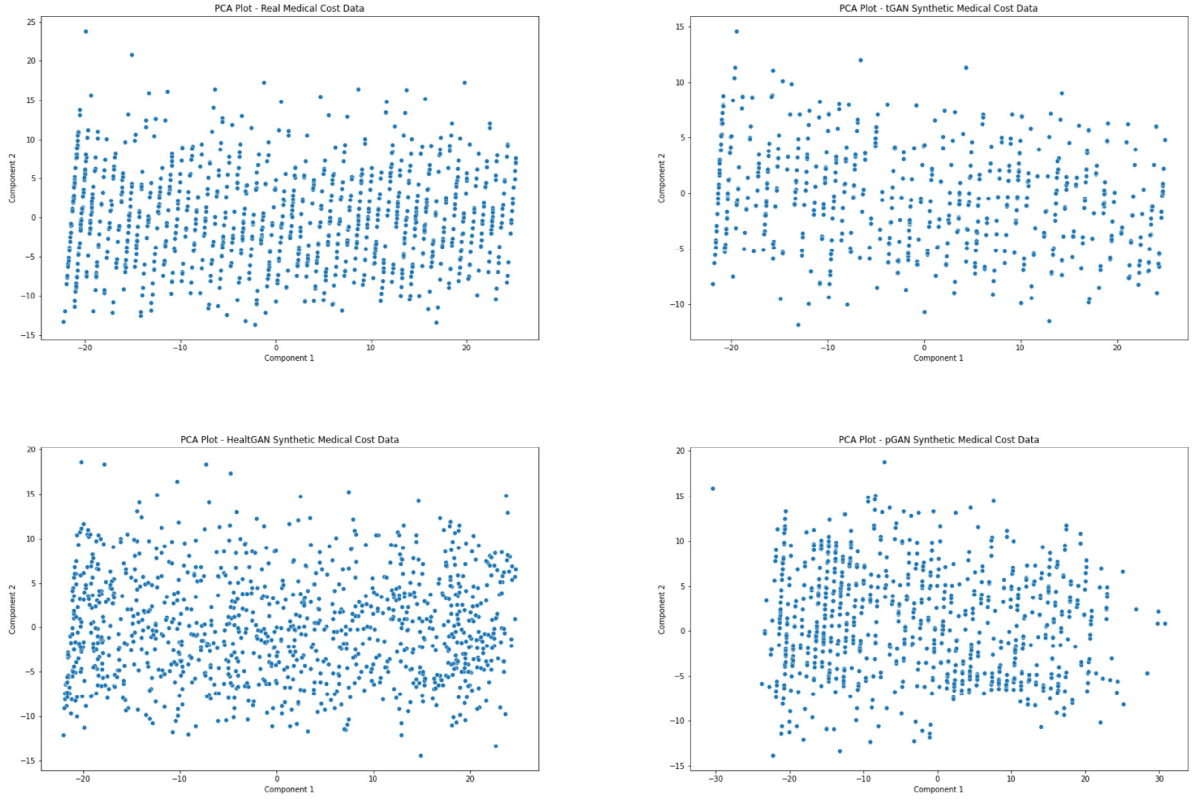

Fig. 6. PCA plots for Medical Cost dataset: Real data (top left), tGAN synthetic data (top right), HealthGAN synthetic data (bottom left) and pGAN synthetic data (bottom right).

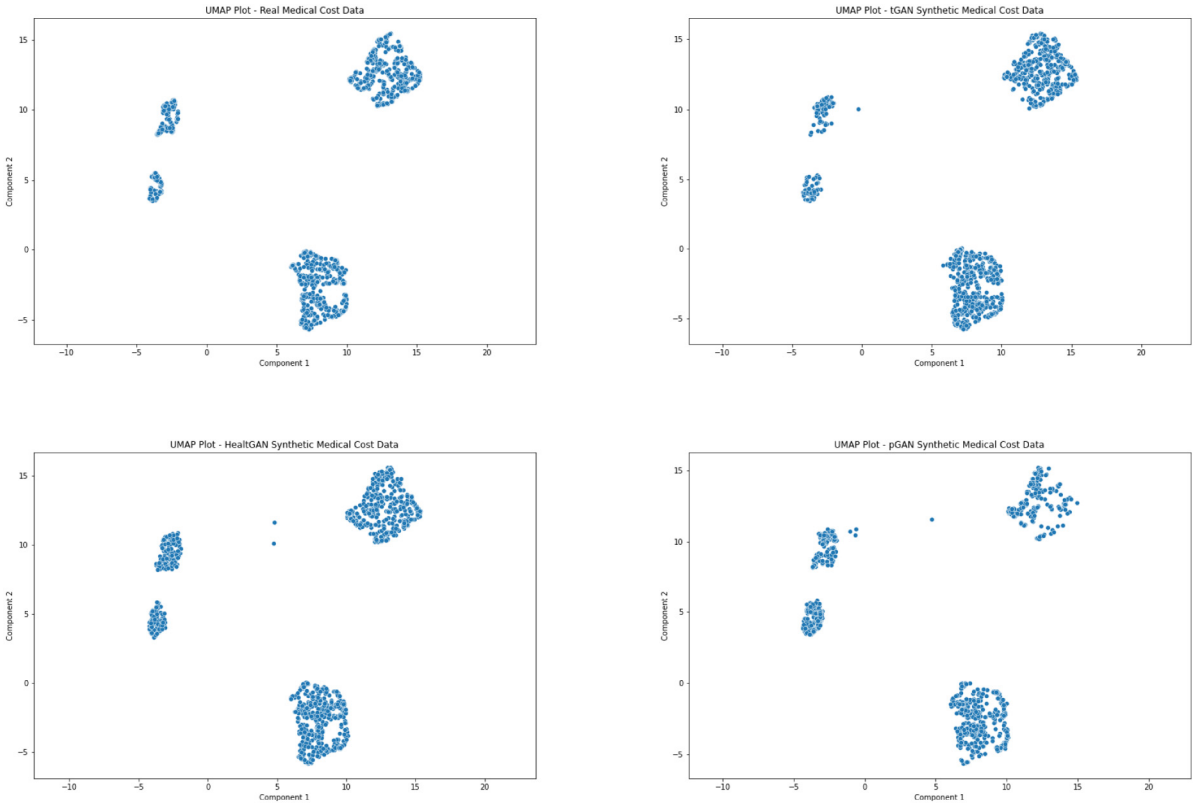

Fig. 7. UMAP plots for Medical Cost dataset: Real data (top left), tGAN synthetic data (top right), HealthGAN synthetic data (bottom left) and pGAN synthetic data (bottom right).

Table 4
Adversarial Accuracy and Privacy Loss.

	Diabetes Data			Medical Cost Data		
	Train AA	Test AA	Privacy Loss	Train AA	Test AA	Privacy Loss
tGAN	1	1	0	1	1	0
HealthGAN	0.54	0.54	0	0.65	0.6	−0.05
pGAN	1	1	0	1	1	0

AA_{TS} score is calculated between the synthetic data, and the original training input data, and is denoted by AA_{Train}, and similarly, the score is also calculated between synthetic data and original test data, and is denoted by AA_{Test}. Now, the privacy score is calculated using the following formula:

$$\textbf{Privacy Score} = AA_{Test} - AA_{Train} \tag{4}$$

Train and Test Adversarial Accuracy scores around 0.5, will result in a privacy loss of 0, and this indicates that the Generator was able to produce synthetic data that has good privacy, as well as good utility. However, if both Train and Test Adversarial Accuracy scores are much higher than 0.5 and privacy loss is still 0, this indicates that the Generator was able to produce synthetic data which preserved privacy, however utility may be low (Yale et al., 2019). Privacy is good when the difference between Train and Test Adversarial Accuracy is small.

Table 4 shows the Train and Test Adversarial Accuracy scores, and the Privacy Loss for synthetic data produced by all three models. As observed, all three models report privacy loss scores of 0, which signifies that every model is preserving privacy when generating data. However, for both tGAN and pGAN, the train and test adversarial accuracies are high and equal to 1. Based on the findings of Yale et al. (2019), this could mean that the utility of synthetic data of these 2 models might be low. Utility testing of synthetic data produced by all models and the respective results are being discussed in the following section.

4.4. Utility testing

Fig. 8 explains the process used to evaluate the performance/utility of the synthetic data. During this process, we randomly selected 20% of the data from the original dataset for testing. We trained a machine learning algorithm on the rest of the data and another model on the synthetic data. Both models were evaluated on the test set we separated from the original dataset.

The machine learning models used for the evaluation are the Dummy Classifier, Support Vector Machines (SVM), Random Forest (RF), k-Nearest Neighbours (kNN) and Multi Layer Perceptron (MLP) Classifier for the Diabetes dataset, and for the Medical Cost Data, Dummy Regressor, Support Vector Regressor (SVR), Linear and an MLP Regressor were used. Multiple Machine Learning models were used for each of the datasets to check the consistency of the results with various techniques.

4.4.1. Diabetes data

The results obtained after classification are presented in this section. Four different machine learning models were trained on the two chosen datasets (Diabetes data and Medical Cost data). One instance of each model was trained on 80% of the original dataset and the second, third and fourth instance of the model were trained on 100% of the synthetic data generated by tGAN, HealthGAN and pGAN respectively. All models were then tested

Table 5
Experimental results of various classifiers on data from different sources.

Classifier	Metric	Original data	tGAN data	HealthGAN	pGAN data
Dummy	Accuracy	0.518	0.531	0.540	0.517
	F1 Score	0.55	0.481	0.511	0.536
SVM	Accuracy	0.695	0.676	**0.740**	0.692
	F1 Score	0.701	0.684	**0.739**	0.704
RF	Accuracy	0.745	0.672	**0.713**	0.678
	F1 Score	0.745	0.672	0.715	**0.721**
MLP	Accuracy	0.683	0.615	0.648	**0.731**
	F1 Score	0.691	0.642	0.652	**0.743**

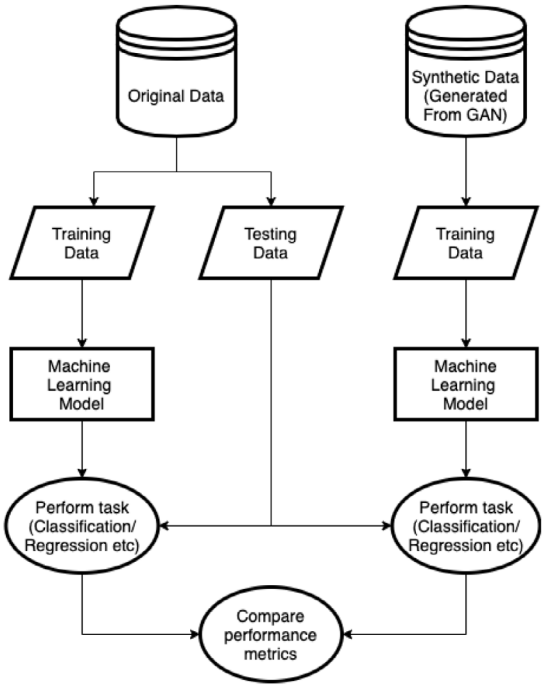

Fig. 8. Evaluation scheme to check the performance of the system.

on the 20% of the stratified test data separated before training. The results from the experiments are tabulated in Table 5.

From the models trained on the original data, Random Forest performed the best, achieving an F1-Score of 0.745; and with SVM and MLP scoring 0.701 and 0.691 respectively. On performing testing, after training the classifiers on synthetic data, MLP outperformed all other models with a score of 0.743 for pGAN data. Among the 3 models (tGAN, HealthGAN and pGAN), pGAN achieved better scores than the other two with MLP and Random Forest, while HealthGAN performed better when using an SVM classifier. The results obtained by all the classifiers can be visually seen in Figs. 9 and 10.

The models trained on synthetic data from pGAN performed similar to HealthGAN, which is one of the leading models currently used for synthetic data generation. From the previous section, even though pGAN synthetic data achieved higher than normal Adversarial Accuracy scores, from this utility testing, we can observe that the synthetic data generated by pGAN can be used in the place of the original data, without compromising the utility or privacy.

4.4.2. Medical Cost Data

The synthetic data generated from tGAN, HealthGAN and pGAN, along with the original dataset was used to train four different regressors. A similar testing strategy was followed as in the previous subsection, where, 20% of the data from the original dataset was used. The results of the experiment are tabulated in Table 6. Since this dataset poses a regression problem, we have used R2-Score as a metric to evaluate the performance of the regressors. For R2-Scores, a value closer to 1 signifies better performance, whereas a score of 0 would imply random fitting.

Table 6
Experimental results of various regressors on data from different sources.

Regressor	Original data R2 Score	tGAN data R2 Score	HealthGAN R2 Score	pGAN data R2 Score
Dummy	0.0	0.0	0.0	0.0
SVR	0.72	0.16	0.79	**0.84**
Linear	0.78	0.16	**0.78**	**0.78**
MLP	0.85	0.09	**0.85**	0.84

MLP Regressor performed the best on original data by achieving an R2-Score of 0.85, with the other models following closely behind (Linear=0.78 and SVR=0.72). Out of the synthetic data produced by all three GAN models, data from tGAN performed the worst, achieving scores of 0.16, 0.16 and 0.09 for SVR, Linear and MLP respectively. Both HealthGAN and pGAN performed similar to each other. The results can be visually seen in Fig. 11.

5. Conclusion

In this paper, we proposed a privacy-preserving GAN (pGAN) which is capable of producing synthetic data of high utility, while preserving the privacy and statistical properties of the source data. We evaluated our GAN architecture on 2 datasets. The Diabetes dataset posed a Classification problem, while the Medical Cost dataset posed a Regression problem. Various classifiers and regressors were used to evaluate the different sources of the data. In addition to this, the proposed model was benchmarked against tGAN (Xu & Veeramachaneni, 2018) and HealthGAN (Yale et al., 2020), which are one of the best performing models for synthetic data generation.

It can be observed from the results that all three GAN models were able to achieve a high degree of privacy, based on their Privacy Loss scores. When testing the performance of various models trained using synthetic data from different sources, we get to see different results. For example, tGAN performs relatively well on the Diabetes data (classification problem) but struggles to produce synthetic data of high quality with Medical Cost data, which is a regression problem. On the other hand, both HealthGAN and pGAN give consistent results across both datasets, and seem to be able to capture the properties of the data, while preserving privacy and maintaining high utility.

During the privacy testing stage, pGAN obtained a good privacy loss score, however, the train and test adversarial accuracy was high (equal to 1). According to Yale et al. (2019), this means that the synthetic data generated, preserved privacy but might be low in utility. However, upon further experiments in the Utility testing, we can observe that pGAN performs similar to HealthGAN, but better than tGAN. This implies that the proposed model did not suffer from low utility, but instead maintained high performance consistently during the utility testing.

HealthGAN makes use of Wasserstein GAN gradient penalty (Arjovsky et al., 2017; Gulrajani et al., 2017), while pGAN makes use of Min-Max loss that is usually used in vanilla GANs (Goodfellow et al., 2014). Even with a relatively straightforward architecture and loss function, pGAN was able to attain similar performance scores as that of HealthGAN for both datasets. Furthermore, Figs. 4 and 6 show that pGAN was able to produce synthetic data with better distribution as compared to HealthGAN.

As seen in the Section 4, the scores obtained by different machine learning models trained on synthetic data from pGAN

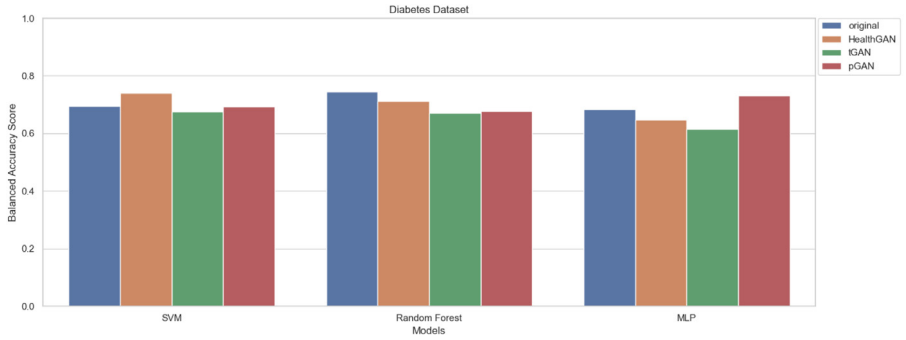

Fig. 9. Barplots visualising the balanced accuracy scores obtained by different classifiers on synthetic data from each GAN model.

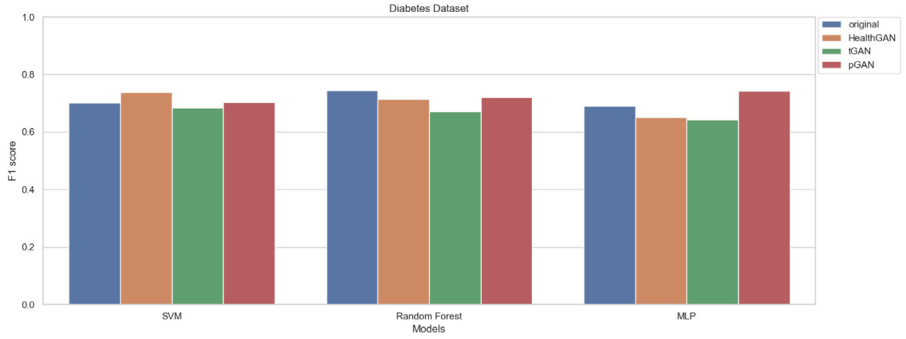

Fig. 10. Barplots visualising the F1-scores obtained by different classifiers on synthetic data from each GAN model.

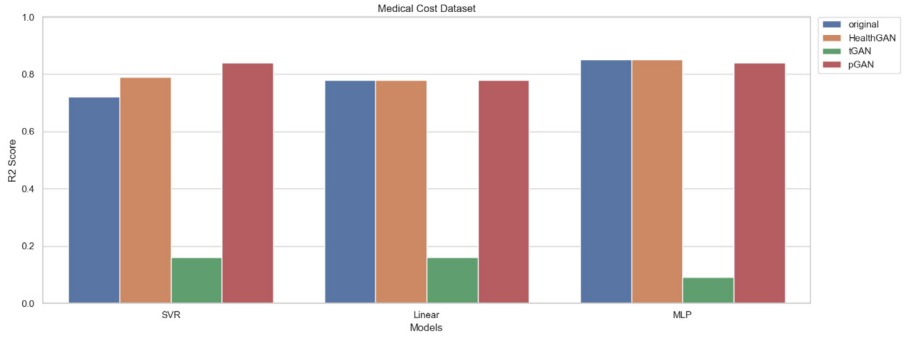

Fig. 11. Barplots visualising the R2 scores obtained by different regressors on synthetic data from each GAN model.

lie in the same range, which implies that the data produced is of high quality. The experiments conducted in our paper, show that, our approach preserves personal privacy, while managing to maintain the distribution and utility of the original data.

One of the primary objectives of the work undertaken in this paper was to investigate if synthetically generated health data could replace the use of actual health records in order to train machine learning models. From our experiments and results, it is clear that both simple and complex GAN architectures are capable of preserving privacy and maintaining a high level of utility even when dealing with sensitive health data. The use of high quality synthetic health data should have a huge impact in the coming years, since it will enable hospitals to generate synthetic data from their private medical records and share it with the research community without compromising the quality or privacy. The usage of synthetic data adds a layer of privacy in a simple manner, without needing to make use of new and upcoming technologies like Blockchain (Chenthara et al., 2020) or Federated Learning (Rieke et al., 2020).

Finally, the usage of Batch Normalization in the generator and Dropout with a low value of p in the discriminator helped our GAN to converge relatively fast. However, this may not always hold true, and as such, further study and research is required to conclusively identify the impact of Batch Normalization and Dropout in training GANs. One of the limitations of our proposed GAN model is its inability to deal with continuous or time-series data. For future work, our model could be improved by incorporating the ability to deal with datasets containing continuous variables. The model could be also adapted and improved to generate synthetic medical images such as Computed Tomography (CT) scans or X-rays, while preserving its privacy.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

The authors would like to acknowledge the support of the Business and Local Government Data Research Centre (ES/S007

156/1) funded by the Economic and Social Research Council (ESRC), United Kingdom for undertaking this work.

References

Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein GAN. ArXiv.

Baker, R., Robertson, N., Rogers, S., Davies, M., Brunskill, N., Khunti, K., et al. (2009). The national institute of health research (NIHR) collaboration for leadership in applied health research and care (CLAHRC) for leicestershire, northamptonshire and rutland (LNR): A programme protocol. *Implementation Science*, *4*(1), 72.

Berthelot, D., Schumm, T., & Metz, L. (2017). Began: Boundary equilibrium generative adversarial networks. arXiv preprint arXiv:1703.10717.

Bjorck, J., Gomes, C., Selman, B., & Weinberger, K. Q. (2018). Understanding batch normalization. arXiv preprint arXiv:1806.02375.

Briot, J. P., Hadjeres, G., & Pachet, F.-D. (2017). Deep learning techniques for music generation–A survey. arXiv preprint arXiv:1709.01620.

Che, T., Li, Y., Zhang, R., Hjelm, R. D., Li, W., Song, Y., et al. (2017). Maximum-likelihood augmented discrete generative adversarial networks. arXiv preprint arXiv:1702.07983.

Chenthara, S., Ahmed, K., Wang, H., Whittaker, F., & Chen, Z. (2020). Healthchain: A novel framework on privacy preservation of electronic health records using blockchain technology. *PLoS One*, *15*(12), Article e0243043.

Choi, E., Biswal, S., Malin, B., Duke, J., Stewart, W. F., & Sun, J. (2017). Generating multi-label discrete patient records using generative adversarial networks. arXiv preprint arXiv:1703.06490.

Dwork, C. (2008). Differential privacy: A survey of results. In *International conference on theory and applications of models of computation* (pp. 1–19). Springer.

Goodfellow, I. (2016). Nips 2016 tutorial: Generative adversarial networks. arXiv preprint arXiv:1701.00160.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). Generative adversarial nets. In *Advances in neural information processing systems* (pp. 2672–2680).

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A. C. (2017). Improved training of wasserstein gans. In *Advances in neural information processing systems* (pp. 5767–5777).

Karras, T., Aila, T., Laine, S., & Lehtinen, J. (2017). Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196.

Kusner, M. J., & Hernández-Lobato, J. M. (2016). Gans for sequences of discrete elements with the gumbel-softmax distribution. arXiv preprint arXiv:1611.04051.

Lantz, B. (2019). *Machine learning with R: Expert techniques for predictive modeling*. Packt publishing ltd.

Li, Q., Wen, Z., Wu, Z., Hu, S., Wang, N., Li, Y., et al. (2019). A survey on federated learning systems: Vision, hype and reality for data privacy and protection. arXiv preprint arXiv:1907.09693.

Malekzadeh, M., Clegg, R. G., & Haddadi, H. (2017). Replacement autoencoder: A privacy-preserving algorithm for sensory data analysis. arXiv preprint arXiv:1710.06564.

Miotto, R., Li, L., Kidd, B. A., & Dudley, J. T. (2016). Deep patient: An unsupervised representation to predict the future of patients from the electronic health records. *Scientific Reports*, *6*(1), 1–10.

Mironov, I. (2017). Rényi differential privacy. In *2017 IEEE 30th Computer security foundations symposium* (pp. 263–275). IEEE.

Pascual, S., Bonafonte, A., & Serra, J. (2017). SEGAN: Speech enhancement generative adversarial network. arXiv preprint arXiv:1703.09452.

Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434.

Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H. R., Albarqouni, S., et al. (2020). The future of digital health with federated learning. *NPJ Digital Medicine*, *3*(1), 1–7.

Sadati, N., Nezhad, M. Z., Chinnam, R. B., & Zhu, D. (2019). Representation learning with autoencoders for electronic health records: A comparative study. arXiv preprint arXiv:1908.09174.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., & Chen, X. (2016). Improved techniques for training gans. In *Advances in neural information processing systems* (pp. 2234–2242).

Scardapane, S., Altilio, R., Ciccarelli, V., Uncini, A., & Panella, M. (2018). Privacy-preserving data mining for distributed medical scenarios. In *Multidisciplinary approaches to neural computing* (pp. 119–128). Springer.

Smith, J. W., Everhart, J., Dickson, W., Knowler, W., & Johannes, R. (1988). Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. In *Proceedings of the annual symposium on computer application in medical care* (p. 261). American Medical Informatics Association.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*(1), 1929–1958.

Torfi, A. (2020). *Privacy-preserving synthetic medical data generation with deep learning* (Ph.D. thesis), Virginia Tech.

Xu, L., & Veeramachaneni, K. (2018). Synthesizing tabular data using generative adversarial networks. arXiv preprint arXiv:1811.11264.

Yale, A., Dash, S., Dutta, R., Guyon, I., Pavao, A., & Bennett, K. (2019). Privacy preserving synthetic health data.

Yale, A., Dash, S., Dutta, R., Guyon, I., Pavao, A., & Bennett, K. P. (2020). Generation and evaluation of privacy preserving synthetic health data. *Neurocomputing*, *416*, 244–255.

Yang, L.-C., Chou, S.-Y., & Yang, Y.-H. (2017). MidiNet: A convolutional generative adversarial network for symbolic-domain music generation. arXiv preprint arXiv:1703.10847.