

Article

GAN-Based Approaches for Generating Structured Data in the Medical Domain

Masoud Abedi ^{1,2,3,†}, Lars Hempel ^{1,2,3,†}, Sina Sadeghi ^{1,2,†} and Toralf Kirsten ^{1,2,3,*} 

¹ Department for Medical Data Science, Leipzig University Medical Center, 04107 Leipzig, Germany; masoud.abedi@medizin.uni-leipzig.de (M.A.); lars.hempel@medizin.uni-leipzig.de (L.H.); sina.sadeghi@medizin.uni-leipzig.de (S.S.)

² Institute for Medical Informatics, Statistics and Epidemiology, Leipzig University, 04107 Leipzig, Germany

³ Faculty Applied Computer and Bio Sciences, Mittweida University of Applied Sciences, 09648 Mittweida, Germany

* Correspondence: tkirsten@uni-leipzig.de

† These authors contributed equally to this work.

Abstract: Modern machine and deep learning methods require large datasets to achieve reliable and robust results. This requirement is often difficult to meet in the medical field, due to data sharing limitations imposed by privacy regulations or the presence of a small number of patients (e.g., rare diseases). To address this data scarcity and to improve the situation, novel generative models such as Generative Adversarial Networks (GANs) have been widely used to generate synthetic data that mimic real data by representing features that reflect health-related information without reference to real patients. In this paper, we consider several GAN models to generate synthetic data used for training binary (malignant/benign) classifiers, and compare their performances in terms of classification accuracy with cases where only real data are considered. We aim to investigate how synthetic data can improve classification accuracy, especially when a small amount of data is available. To this end, we have developed and implemented an evaluation framework where binary classifiers are trained on extended datasets containing both real and synthetic data. The results show improved accuracy for classifiers trained with generated data from more advanced GAN models, even when limited amounts of original data are available.

Keywords: synthetic data generation; data enlargement; data evaluation; GANs



Citation: Abedi, M.; Hempel, L.; Sadeghi, S.; Kirsten, T. GAN-Based Approaches for Generating Structured Data in the Medical Domain. *Appl. Sci.* **2022**, *12*, 7075. <https://doi.org/10.3390/app12147075>

Academic Editor: Pentti Nieminen

Received: 8 June 2022

Accepted: 9 July 2022

Published: 13 July 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Synthetic data generation has gained importance in healthcare applications in recent years [1–4]. It has become of particular interest in the medical field for two main reasons. First, real world personal data are usually captured and managed in accordance with general privacy regulations, such as the General Data Protection Regulation (GDPR) [5]. As a result, such data are hardly accessible to everyone. However, they may be shared with permission and only under strict conditions and regulations [6]. In most cases, for example, data sharing is subject to a contract between the data-providing and data-consuming institutions. This complicates the process of data acquisition and poses a major challenge for research groups, as they need access to such personal data for developing algorithms or testing their applicability to different datasets. For this reason, many data scientists and researchers rely on publicly available medical data to perform analyses and to develop innovative tools and methods [7–10]. Artificially generated synthetic data alternatively offer the opportunity to improve the current situation by providing health-related data with characteristics that are similar to those of the real world data, yet not associated with any real patients. Second, for diseases that affect very few patients, i.e., rare diseases, there exist insufficient data, even for diagnosis and treatment. According to the EU definition, a disease is considered rare when it occurs in less than 1 in 2000 people [11]. Novel artificial

intelligence algorithms and advanced deep learning methods, on the other hand, place high demands on the amount of data. The accuracy of such models indeed outperforms other conventional statistical and machine learning methods when a large dataset is available to them [12]. Synthetic data generation in the medical field can bridge this data scarcity by providing a large set of artificial data that represent the statistical properties of the original real data. Moreover, synthetic data can be utilized to facilitate the care of patients with rare diseases by referring them more efficiently to experienced medical experts. This is of great importance, because those experts are also rare, i.e., not available in every medical center.

In recent years, numerous methods and software have been developed for synthetic data generation [13–15]. The number of emerging tools being used in healthcare and medical science is also growing rapidly; examples include: SDV (synthetic data vault) [16], SynSys [1], Synthea [17], and Synthia [18]. Some methods leverage statistical methods, e.g., R packages such as synthpop [19] and simPop [20], while others rely on more advanced artificial neural networks. Two most important and popular families of such deep generative models are Variational Autoencoders (VAEs) [21,22] and Generative Adversarial Networks (GANs) [12,23]. VAEs are similar to autoencoders, which additionally employ variational inference to regularize the encoding distribution and ensure that the generation of new data is less prone to overfitting. VAE models, along with other generative models, have been widely used to bridge data scarcity and to improve machine learning performance in various application domains through data augmentation, a technique that increases the amount of data by adding slightly modified versions of existing data. For example, VAEs have been developed to generate synthetic eye-tracking data from a small dataset with a variety of technological applications [24]. They were also utilized to simulate the longitudinal clinical data of virtual patients while preserving privacy, which helps scientists to conduct counterfactual studies [25].

GANs are another class of deep generative models introduced by Goodfellow et al. that have gained wide popularity and interest in many different application areas [26,27]. A GAN model consists of two networks, namely a generator and a discriminator. The architecture of a GAN model in its original form is illustrated in Figure 1. The generator typically generates data from initially random patterns. These generated data are fed into the discriminator along with real data. The discriminator acts as a classifier. It is trained to validate the authenticity of the input data, i.e., to distinguish real data from generated data. The crucial point is that the generator only interacts with the discriminator and has no direct access to real data. The generator thus learns from its failure and improves its performance in generating realistic data through the training process (backpropagation). The two networks contest with each other in a zero-sum game; hence, their goals are adversarial. Provided that the generator is sufficiently trained, it generates artificial data that mimics the real ones, and can eventually fool the discriminator, i.e., the generated data are recognized as real by the discriminator.

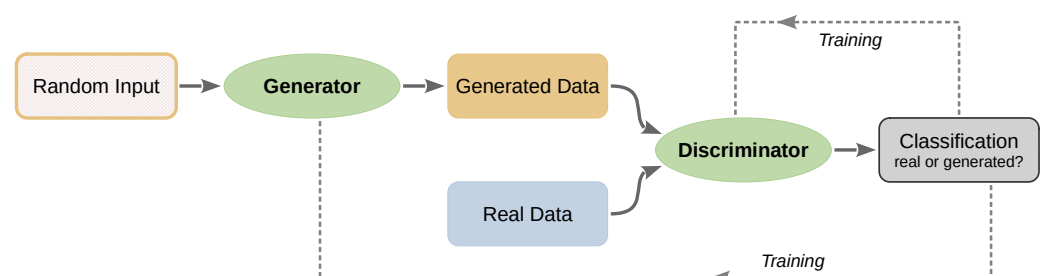


Figure 1. The architecture of a GAN model: Two adversarial networks are trained together. The generator is trained to generate new realistic data that are indistinguishable from real data, while the discriminator determines whether the data are real or generated.

GANs have also attracted a lot of interest in the medical field to generate synthetic data for various applications, such as data augmentation for patient data collected from

Internet of Medical Things devices, as the process of data collection can run into trouble for various reasons and cause problems for patient monitoring, and ultimately for clinical decision-making systems [28]. GAN-based models have also been used for the fast MRI (magnetic resonance imaging) reconstruction of blurry scans [29], as well as the segmentation of meibomian glands from infrared images, i.e., automatically identifying the area of meibomian glands [30], and style transfer from UBM (Ultrasound Biomicroscopy) to AS-OCT (Anterior Segment Optical Coherence Tomography) in ophthalmology image domains [31]. VAE-GAN models further combine the architecture of both methods, i.e., they use a GAN-like architecture where the discriminator is a variational autoencoder [32]. Variations of VAE-GAN models were developed, for example, for unsupervised anomaly detection and segmentation in MRI scans [33], and for generating three-dimensional brain MRIs from random patterns [34].

GANs are typically known for their success in generating realistic images, and have been extensively studied for their applicability to image and video synthesis in a wide range of applications [35–37]. However, they can be extended to account for other forms of data such as structured (tabular) data, which are the most commonly used data type, particularly in healthcare and medical applications [38]. While GANs have shown impressive results in generating images and also text data in natural language [39,40], their performance on tabular data still remains a challenge [15]. This is more evident in the medical domain, for example, with a smaller amount of available data due to privacy issues [41,42].

In this paper, we are concerned with the generation of structured (tabular) synthetic data for applications in the medical field. Other data types such as (2D and 3D) images and genetic data are beyond the scope of this work, and will be considered in future work. We consider various GAN-based models that are most relevant to structured data, and investigate how they can efficiently work with structured data and generate high quality synthetic tabular data suitable for medical applications. The main goal is to use synthetic data from different GANs for training binary (malignant/benign) classifiers, and to compare their performance in terms of classification accuracy with cases where only real data are considered. We aim to investigate how synthetic data improve binary classification accuracy, especially when a small amount of data is available. This is of great importance for medical applications, such as rare diseases for which only limited data are provided. Therefore, we develop an evaluation framework that considers an extended dataset consisting of both real and synthetic data to train classifiers in a small sample setting with a limited amount of data. This is the novelty of the present evaluation in contrast to other works that consider either real or synthetic data for training the classifiers, and reserve the other (synthetic or real) dataset for testing [43–45].

The rest of this paper is structured as follows: Section 2 introduces several relevant GAN-based models, as well as publicly available medical data used for this study, the experimental setup, and the evaluation methodology. Sections 3 and 4 represent and discuss the results, and finally, Section 5 provides some conclusions and prospects for future work.

2. Methods

Since the introduction of the original GAN, several variations of GANs have been developed [46]. Each variant is designed for a specific purpose, and is usually used in a particular application domain [27,47–49]. In this study, we consider some major variations of GANs, mainly relevant to structured data, to investigate their applicability and advantages in the medical domain. We compare and summarize the performances of these generative models in generating tabular data using publicly available medical data. In what follows, we present selected GAN variants used in the present work.

2.1. Selected GAN Variants

2.1.1. Conditional Generative Adversarial Networks

Conditional Generative Adversarial Networks (CGANs) are an important extension of the original GAN [50]. CGANs contain two adversarial networks, a generator and a discriminator, both conditioned on some additional information. The extra information can be, for instance, class labels, feature correlations, or other auxiliary information. The goal is to improve the GAN performance, i.e., the accuracy of the generator and discriminator networks, and to generate targeted (labeled) data of a given type that are not present in the real data [50].

2.1.2. Specific GANs for Tabular Data

Another extension of GANs, developed specifically for tabular data, is the Tabular Generative Adversarial Networks (TGANs) model [51]. Tabular data are the most common form of data, widely used in a range of applications, including the medical field. They usually contain various features with continuous numerical and (discrete) categorical values. While continuous features may be distributed over some range, categorical features usually consist of highly imbalanced discrete values, which complicates the modeling. Therefore, Conditional Tabular Generative Adversarial Networks (CTGANs) were designed to account for categorical data in particular [45]. CTGANs use a conditional generator and the training-by-sampling method to overcome the complications of imbalanced data. They also introduce the mode-specific normalization technique to deal with more complicated non-Gaussian and multimodal distributions, for example [45]. Yet another interesting variation of GANs based upon the Copula theory is the CopulaGAN [52]. CopulaGANs consider a probabilistic model to generate samples with high fidelity. The model uses normalizing flows to learn nonlinear correlations between features, and models synthetic data with the distribution that resembles the original real data [52].

2.1.3. Wasserstein Generative Adversarial Networks

Wasserstein Generative Adversarial Networks (WGANs) were proposed to improve the the stability of learning in generative models [53]. As opposed to the original GANs, WGANs introduce a critic network instead of a discriminator. A discriminator in the original GANs uses the probability estimation to distinguish between the real and generated data. The critic, on the other hand, estimates the distribution of both the real and generated data, and then minimizes the Wasserstein distance between them as a metric [53]. This optimization increases the stability of the generative model and improves the quality of the generated data. However, WGANs sometimes output low-quality generated data or even fail to converge [54]. To tackle this problem, the Wasserstein Generative Adversarial Networks with Gradient Penalty (WGAN-GPs) have been developed [54]. Instead of the weight-clipping in WGANs, which imposes constraints on the critic, WGAN-GPs use a gradient penalty function to overcome the gradient vanishing problem. This way, they can also deal with the imbalanced class labels via an oversampling method that enables them to generate more synthetic samples for minority labels [55].

2.2. Data, Experimental Setup, and Evaluation Framework

2.2.1. Data

In this study, we used two well-known, publicly available datasets from the University of California Irvine (UCI) Machine Learning Repository: Breast Cancer Wisconsin (BCW) [56] and Breast Cancer Coimbra (BCC) [57]. Both datasets have been used extensively elsewhere, for example, for model-based classification [58–60].

The BCW consists of 569 patients with breast tumors, of which 212 are malignant and the remaining 357 cases are benign. Data have been computationally extracted from digitized medical images [61]. The tumors are then characterized by 32 features, including perimeter, radius, area, texture, smoothness, compactness, concavity, number of concave portions, symmetry, and fractal dimension. Each feature is represented by three properties:

mean, standard error, and worst value. The latter indicate the outlier in measurements, i.e., values that fall outside the medically specified range. In addition, each instance is identified by a pseudonym and is labeled as malignant or benign. The dataset has no missing values, and the features are specified by real values, except for the label, which is categorical.

The BCC was created by the Coimbra Hospital and University Center (CHUC) in Portugal, and published in 2018 [59]. It comprises 116 instances involving 64 patients with breast cancer and 52 healthy controls. Each instance is described by 10 features, which include age, body mass index (BMI), glucose, insulin, insulin resistance determined via Homeostasis Model Assessment (HOMA), leptin, adiponectin, resistin, MCP-1, and a label. The label divides the dataset into patients with breast cancer and healthy controls. There are no missing values in this dataset, and all features are represented by integer values.

2.2.2. Experiments and Evaluation Framework

The given dataset (here, BCW or BCC) is first divided into 70% Train and 30% Test datasets. The Train data are used to teach the generative models, i.e., to train the generator and discriminator networks in each GAN model to generate synthetic data. Once the data are generated by GAN variants, their quality needs to be evaluated to determine whether their statistics could reflect real data. To this end, we have developed an evaluation framework that measures the accuracy of a binary classification (malignant/benign) trained on a combination of synthetic and real data. A schematic overview of the evaluation method is illustrated in Figure 2. The Test dataset is kept separate and used only to verify the quality of the generated data via measuring the accuracy of the classifiers. The Train data, together with synthetic data, however, are used to train the classifiers.

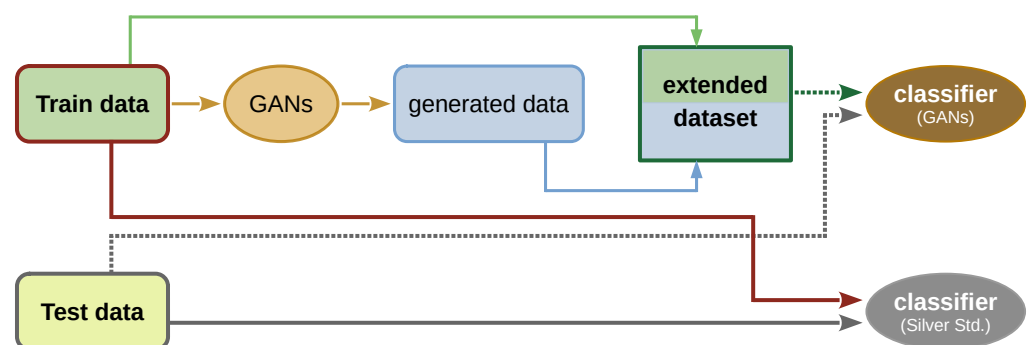


Figure 2. Schematic overview of the evaluation of data generated by GAN variants: Data are divided into Train and Test, the GAN models generate synthetic data based on the Train data; Train and generated data are combined into an extended dataset. The classifier is trained once with only the original Train data (Silver Standard), and then using the extended dataset (including generated data) corresponding to each GAN variants. The classifiers are in the end evaluated with Test data.

To investigate the impact of the size of the training data on the accuracy of the classifiers, the Train dataset is divided into several subsets for each given dataset. The BCW is divided into 10 subsets containing 5, 6, 7, 8, 9, 10, 20, 30, 40, or 50% of the Train data, respectively, while the BCC is split into nine subsets, each consisting of 10, 20, 30, 40, 50, 60, 70, 80, and 90% of the Train data. Moreover, two different sampling strategies are used for each dataset, random selection (RS) and random extension (RE). In the RS, data for each subset are randomly and independently selected from the Train dataset, while in the RE, the smallest subset is first randomly selected from the Train dataset and then extended by adding more random data to create a larger dataset with a desired size. In this way, the larger subsets contain all the subsets generated in previous steps of the RE. The goal of the RS strategy is to test a complete random selection where random effects on the results are expected. The RE strategy, on the other hand, is concerned with how much additional data should be added to an existing data subset to obtain reliable and robust results. The latter strategy is particularly important in the medical field to evaluate improved model

performance (in this case, classification accuracy) by incorporating newly available data when limited data are present, such as for rare diseases. The GAN models are trained using each data subset with RS or RE sampling to generate synthetic data. The same amount of generated data are then added to the original data subset, and the classifier is eventually trained using this larger extended dataset. For each setup, the experiments are performed on 10 random samples, and finally, the results of different samples are averaged, and the statistical mean accuracy and standard error are measured.

Data generated by all GAN variants were benchmarked using two different classification methods, namely Support Vector Machines (SVM) and classical neural networks, Multi-Layer Perceptron (MLP). Studies on the performance of different classification models for the BCW dataset have shown that the SVM and MLP methods outperform other classifiers in terms of accuracy [62–64]. While the MLP has the highest accuracy in most cases and is superior for large and complex datasets, SVM performs best in a small sample setting when there are only few data points with many features, which is the case in this study [62,65]. We note that to measure the quality of generated synthetic data, one should consider other statistical metrics to better compare the real and synthetic distributions. We calculated, for example, the pairwise correlation difference (PCD) between real and synthetic data to measure how much correlation among features in real data is captured by synthetic data [42,43,66]. However, since the main focus of this study is to improve the classification accuracy by incorporating synthetic data, particularly in a small sample setting, such comparisons are not present, and a comprehensive analysis of evaluation metrics is reserved for future work.

All experiments were conducted on a single Linux machine using the Python programming language (version 3.6.9). We designed and developed implementations for the standard GANs and CGANs using Keras (<https://github.com/fchollet/keras>, accessed on 7 June 2022). We also utilized the available implementations of CTGANs and CopulaGANs from the Synthetic Data Vault (SDV) [16], as well as an implementation of WGAN-GP available on GitHub (<https://github.com/justinengelmann/GANbasedOversampling>, accessed on 7 June 2022). Both SVM and MLP methods were implemented using the Python library SKLearn. In addition, we used other standard Python libraries such as Pandas, Numpy, Scipy, random, platyous, and csv throughout the benchmark. The visualizations, however, were performed in the R programming language (version 4.1.3) using the ggplot2, stringr, and tidymodels packages. The implementations of the GAN models presented in this paper are freely accessible in the GitLab repository (<https://gitlab.com/ul-mds/data-science/synthetic-data/gan-collection>, accessed on 7 June 2022).

3. Results

For each experiment, the GAN variants are trained independently on a subset of the Train data for a number of epochs to generate synthetic data. Table 1 summarizes the epoch numbers for the GAN models used for two BCW and BCC datasets, determined by optimizing the pairwise correlation difference between the real and synthetic data [42,43,66]. The same amount of generated data are then added to the subset of the Train data, and they form an extended dataset that is eventually used for training the classifiers. The relevant question here then is how these additional synthetic data can improve the performance of the classifiers. This is especially important when only a small amount of (real) Train data is available, e.g., for rare diseases in the medical field.

Table 1. Number of epochs for training GAN variants using two different datasets, BCW and BCC.

	GAN	CGAN	CTGAN	CopulaGAN	WGAN-GP
BCW	900	100	1000	800	300
BCC	300	300	300	900	100

The results are presented separately for two different datasets: First, using the BCW and then the BCC. For each dataset, we show the results of two sampling setups, namely RS and RE, whose outputs are then evaluated by two different classifiers, SVM and MLP. Figure 3 illustrates the mean accuracy of the classifiers trained on extended datasets of different sizes. The latter contain generated data from GAN variants as well as the Train data; see Figure 2. In fact, for each training data size, expressed as a percentage of the Train data, the same amount of synthetic data is generated and added to the subsets of the Train data. As a baseline for comparison, the classifiers are trained once using only the Train data, without generated data, which is indicated as the Silver Standard in the graphs. We note that the classifiers consider a larger dataset in the case of GAN variants, where they include synthetic data. In this way, the comparison of the GAN results with the Silver Standard shows how the inclusion of synthetic data can improve the performance of the classifier on (unseen) Test data. We also note that we use the terms Train and Test data to refer to data from the original real dataset and training data to refer to subsets of the Train data of different sizes.

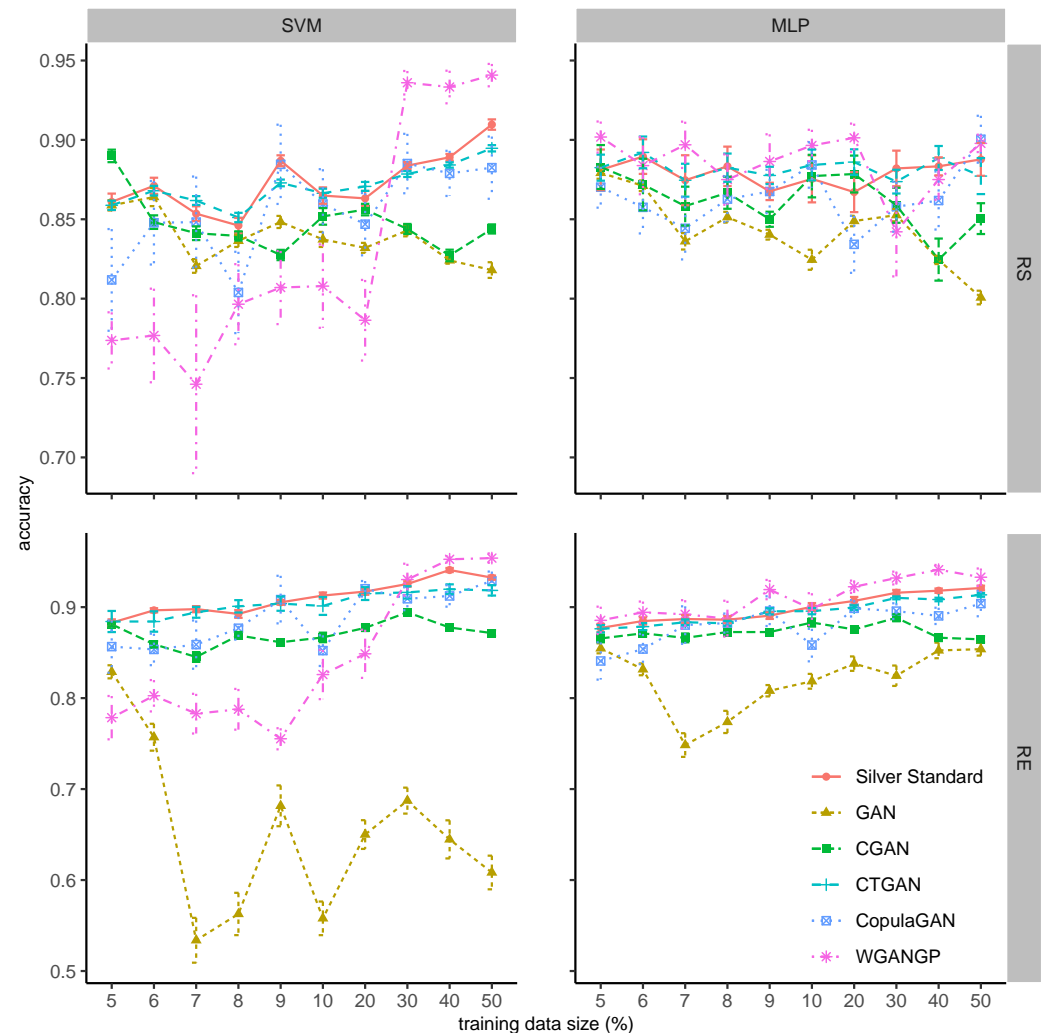


Figure 3. The mean accuracy of classifiers trained on data generated by GAN variants versus the size of the training data (as a percentage of the Train data) using the BCW dataset. Rows (**top** and **bottom**) show the RS and RE sampling, while columns (**left** and **right**) indicate the SVM and MLP classifiers. The Silver Standard only considers Train data (no generated data) for training the classifier. Error bars show the standard error of the mean over 10 samples for each point.

3.1. Breast Cancer Wisconsin—BCW

Figure 3 shows the accuracy of SVM and MLP classifiers trained on extended datasets of different sizes for two RS and RE samplings using the BCW dataset. For the Silver Standard, the classifier takes no generated data into account, and is trained only on subsets of the Train data of different sizes. In this case, the accuracy of the classifier improves as the size of the training data increases. This is more pronounced in the RE sampling, as the training size are gradually increased starting from the fixed initial size, while in the RS, the accuracy alternately increases and decreases as the data are randomly selected for each training data size. The randomness in RS leads in most cases to larger variations in the results, not only for the Silver Standard, but also for other models. The improved accuracy with the size of the training data, as observed for the Silver Standard, is to be expected because the more training data are fed into the classifiers, the higher the expected accuracy. The results of the GANs, however, exhibit a less accurate classification. Indeed, the SVM accuracy for GANs decreases clearly as more (extended) training data are provided to the classifier, and even the overall accuracy for the RE is lower compared to the same one in the RS. The latter is probably due to an overfitting of the GAN model to the (original) Train data when generating new data. This implies that the inclusion of the synthetically generated data from the original GANs might not readily improve the classifier performance. The reason would be that GANs learn only by recognizing patterns in the (original) Train data with no constraints on generating new data, which leads to a high variance in the generated data. To elucidate this, we performed a statistical measurement on the synthetic data for key features in a given dataset. Figure 4 shows the results for the mean perimeter of tumors in generated data from GAN variants over different training data sizes. It appears that GANs have a relatively high variance over the entire range of training sizes.

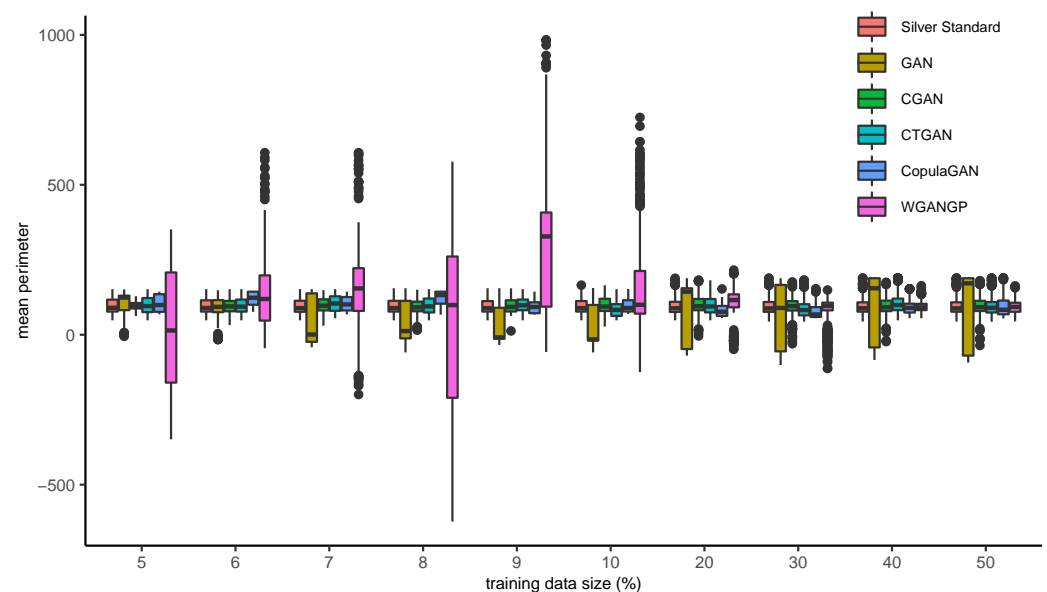


Figure 4. Mean perimeter of tumors, a key feature in synthetic data from GAN variants over 10 different samples, versus the size of the training data (in percent) using the BCW.

Compared to GANs, the CGAN exhibits a significantly improved accuracy, though it is still below the Silver Standard. This shows that the introduction of additional information into CGAN, in this case, class labels, can dramatically increase the accuracy of the classifier. In RE sampling, this becomes even more evident as the learning performance gradually improves, while in RS, the curves fluctuate more due to random selection of the training data. In fact, a comparison of the results for the two classifiers shows a relatively similar behavior, apart from the random variations in RS. The results of CTGAN show an even higher accuracy on average, because CTGANs are designed specifically to consider tabular data. The performance of the classifiers for CTGANs is almost identical to that of the Silver

Standard, i.e., the model generates relatively the same statistics as the original Train data. Hence, the accuracy of the classifier also improves as the size of the training data increases, as for the Silver Standard. A similar pattern is observed for the CopulaGAN, with only lower accuracy and larger fluctuations. In this study, CopulaGANs are not provided with the statistical information required for their optimal performance, as we trained GAN variants in their standard format. The generated data could increase the accuracy only if the generative models were provided with more informative statistics than those already included in the Train data. We emphasize that further investigations are to be pursued to explore the best performance of the GAN models when provided with some additional information to exploit their capabilities. We also note that these findings are based on the accuracy of the binary classifiers and for a more conclusive statistical comparison, a more thorough study is essential.

An interesting behavior is observed for the generated data from the WGAN-GP model. The MLP shows better accuracy than the Silver Standard over almost the entire range, in RS especially, and at some points in RE. The SVM, however, has poor accuracy for both RE and RS, when the classifier is fed in with a small amount of training data, 20% or less. This is due to the higher variance in the generated data by WGAN-GP at smaller training data sizes, as shown in Figure 4 for a key feature, the mean perimeter. In fact, the WGAN-GP becomes rather unstable when provided with smaller datasets for generating the synthetic data. However, the accuracy improves drastically when more than 30% of the Train data are used for training the classifier. This is also noticed in the standard error of the mean, which shrinks as the training data size increases. In general, we observe that the more advanced GAN models demonstrate a better performance in terms of the associated classification accuracy. The highest accuracy can be achieved with WGAN-GP data, while the CTGAN results are less volatile and more stable.

3.2. Breast Cancer Coimbra—BCC

To verify the performance of the GAN models, we conducted further experiments with the BCC dataset. The latter is a smaller dataset and has fewer features than the BCW. Figure 5 demonstrates the accuracy of binary classification (malignant/benign) trained on extended data generated from GAN variants using the BCC dataset. The accuracy of SVM and MLP classifiers at different sizes of Train data used with RS and RE sampling are shown. The Silver Standard exhibits an apparent increase in accuracy as the size of the training data increases, but the overall accuracy is lower compared to the BCW. This behavior is similarly observed for some GAN variants as well. The results for the BCC also show higher standard errors compared to the corresponding points in the previous dataset, possibly due to the smaller size of the Train dataset with a reduced number of features, making the GAN models more unstable. We note that the results for the BCC are shown over a wider range of training sizes.

Contrary to the results for the BCW, here, the GAN data yield some better results, especially for MLP, where the accuracy increases with increasing training size. However, the accuracy obtained for the GAN data remains the least accurate for SVM. Some improvements are noted in the CGAN results, mainly for the SVM classifier with RE sampling. This is also in contrast to the BCW, where the CGAN results are always below the Silver Standard. The accuracy curves for CTGANs and CopulaGANs are more unstable here, and they even display the worst results for the MLP classifier. It appears that more complicated models in their standard form have difficulty in generating synthetic data that are sufficient to improve classification accuracy. This is reflected in the larger standard errors as well, which means that the results for each point vary more. Interestingly, the WGAN-GP results illustrate quite different behaviors from the BCW. While they do not exceed the Silver Standard accuracy for MLP, their results for SVM have become more stable and show a more consistent improvement as the amount of training data grows. Additionally, they demonstrate improved accuracy over a wider range of sizes of training data, with better performance for smaller data subsets, likely due to less complicated data. Further

analysis is still required to examine the statistics of the generated data by GAN variants using different datasets, and to verify how corresponding synthetic data might improve the classification accuracy. This also allows us to determine how robust the results of the various GAN models are, even when only a small amount of data is available.

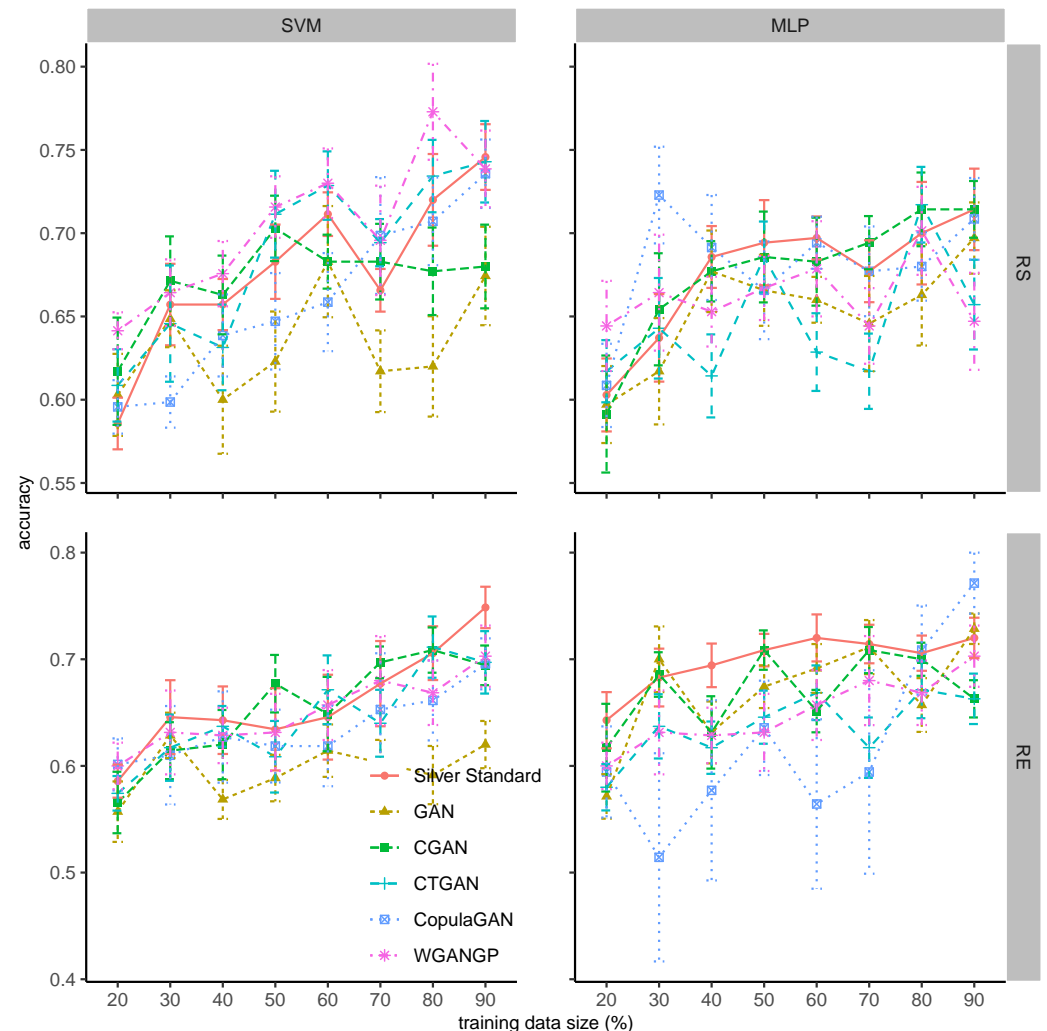


Figure 5. The mean accuracy of classifiers trained on data generated by GAN variants versus the size of the training data (as a percentage of the Train data) using the BCC dataset. Rows (**top** and **bottom**) show the RS and RE sampling, while columns (**left** and **right**) indicate the SVM and MLP classifiers. The Silver Standard only considers Train data (no generated data) for training the classifier. Error bars show the standard error of the mean over 10 samples for each point.

3.3. Time Usage of GAN Variants

Figure 6 compares the time consumed by different GAN models to generate synthetic data over a different number of epochs (see Table 1), using the BCW dataset. The boxplots illustrate the averaging over 10 different samples for each point. We see that the time increases linearly with the number of epochs, and that the overall time usage becomes higher for more advanced models. The only exception occurs for the WGANGP model, which utilizes specific optimization packages, resulting in less time usage. However, the statistical variance is higher for WGANGP, while it is negligible for GAN and CGAN, implying that they consume almost the same time for each sample, likely due to the simpler structure of the model.

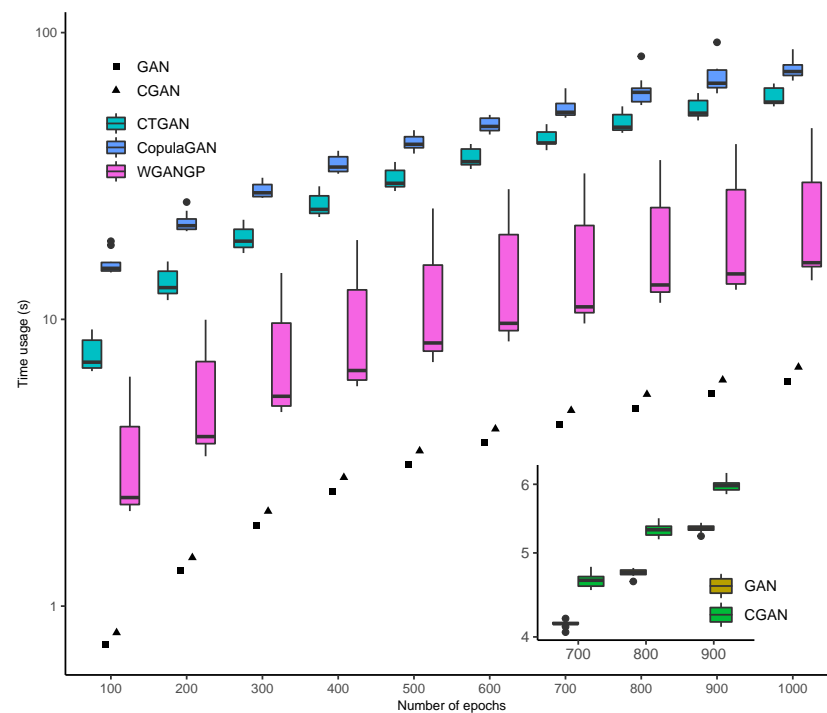


Figure 6. Time usage (seconds) of the GAN variants for the generation of synthetic data over different number of epochs using the BCW dataset. For illustrative purposes, the GAN and CGAN data points are represented by squares and triangles in the main panel. The variation in time usage of GAN and CGAN for epochs (700–900) are shown in the inset.

3.4. Memory Usage

The memory usage of the GAN variants for generating synthetic data using the BCW dataset is illustrated in Figure 7. The bars indicate the amount of memory used in megabytes, averaged over different samples, and the number of epochs for each GAN model. The memory consumption of GAN and CGAN differs only slightly, as they have almost the same structures, with only CGAN exploiting additional information. The CopulaGAN and WGANGP have the lowest memory usage, due to the specific packages they use, which limits memory consumption. We emphasize that all computations were performed by a CPU, limited by the system described in Section 2, and that we made no use of a GPU.

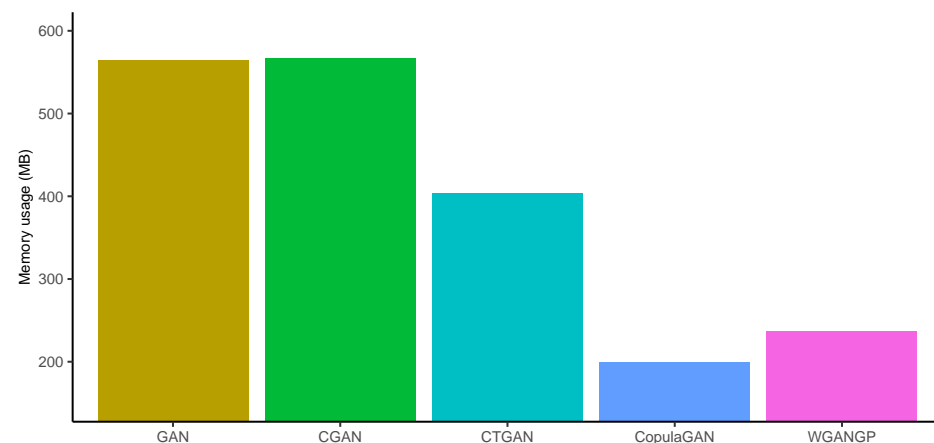


Figure 7. Memory usage (megabytes) of the GAN variants for the generation of synthetic data using the BCW dataset.

4. Discussion

We first briefly elaborate on the evaluation framework used in this study, and its significance. Then, we discuss the key observations regarding the impact of varying the size of the training data on the accuracy of the binary classification.

4.1. Evaluation of Synthetic Data

The key concern in generating synthetic data is whether they can represent the complex patterns present in the real data. This requires a rigorous evaluation method to examine the fidelity and quality of the synthetic data, and then a direct comparison between the synthetic data generated by different generative models. However, it turns out that different evaluation strategies and metrics lead to different results in favor of different generative models, especially for high-dimensional data, due to their greater heterogeneity and complexity [67]. Moreover, the evaluation results vary depending on the application domain, and they may not be readily transferable to other domains. Therefore, one needs to design and develop an appropriate evaluation framework specifically for the ongoing target problem.

Since we are mainly concerned with the medical applications where real data is limited, we have introduced an extended dataset comprising both generated and real data for training the classifier. This allows us to determine how the inclusion of synthetic data can improve the accuracy of the classifier, while at the same time exploiting the properties of the real data. This is in contrast to other evaluation methods where the classifier is trained on the real data only, and the synthetic data are used to test the classifier; see Refs. [43,44]. Conversely, others employ only synthetic data to train the classifier and then test it on the real test set [45]. We note that in the present evaluation framework, the same real data are used to train the generative models and partially in an extended dataset, to train the classifier. This way, however, we can address one of the major challenges in the medical domain, i.e., data scarcity where medical data become limited.

4.2. Data Scarcity

Synthetic data can overcome data limitations that exist for a variety of reasons and introduce a number of complications in the medical field. However, the quality of such data, usually verified by the accuracy of a supervised classification, is in most cases lower than that of the real data [68]. It is therefore critical to evaluate the amount of synthetic data that needs to be generated to achieve high accuracy based on the available (limited) real data. The present results show how the synthetic data generated by more advanced generative models such as the GAN variants in this study (i.e., CTGAN, CopulaGAN, and WGANGP) can improve the classification accuracy compared to the Silver Standard, trained on real data only. We note that the generative models encounter difficulties in situations where only a small amount of real data is considered and hence low quality data are generated. This concern is particularly important in the medical field, because of the limited data available, which makes it difficult to generate and test data quality, unlike other studies that use larger datasets [44].

5. Conclusions

The utilization of synthetic data is becoming more and more remarkable in various application domains. In the medical field, they are of particular interest for two primary reasons: Data privacy regulations and poorly available real data. Generative Adversarial Networks have shown a great ability to generate synthetic data that mimic real ones without referring to real personal data. In this study, we investigated five GAN variants, including GAN, CGAN, CTGAN, CopulaGAN, and WGANGP, using two different datasets, BCW and BCC. To evaluate the obtained synthetic data, we developed an evaluation framework considering an extended dataset with generated and real data. The results demonstrate that data generated with more advanced GAN models such as WGANGP can improve the binary classification accuracy, even with a smaller amount of data. Since the statistics of the

two given datasets are different, the corresponding results are also slightly different. In fact, the classification accuracy is more stable when using the BCC with smaller datasets and a reduced number of features, and it displays an obvious improvement as the size of the training data grows.

Future work needs to examine the statistics of the generated data in more detail, and to measure other metrics to allow for a more comprehensive comparison. The hyperparameter space of the generative models is to be explored in search of optimized settings that improve the quality of generated data and increase the accuracy. We have not considered all the capabilities of the GAN variants in this work. For instance, the CopulaGANs require additional information regarding feature correlations to reveal their optimal performance. Such model customization also needs to be considered and thoroughly explored.

6. Patents

We did not use any patents for this paper, and to the best of our knowledge, there is no patent available that touches upon the work provided throughout the paper.

Author Contributions: Conceptualization: M.A., L.H., S.S. and T.K.; Software: M.A., L.H. and S.S.; Validation: M.A., L.H. and S.S.; Data curation: L.H. and M.A.; Writing—original draft preparation: M.A., L.H., S.S. and T.K.; Writing—review and editing: M.A., L.H., S.S. and T.K.; Visualization: M.A. and L.H.; Supervision: S.S. and T.K.; Project administration: T.K.; Funding acquisition: T.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the German Ministry for Research and Education (BMBF) as part of the SMITH consortium (T.K., grant no. 01ZZ1803K), and by the German Ministry of Health as part of the LEUKO-Expert consortium (M.A. and L.H., grant no. ZMVI1-2520DAT94A). This work was conducted jointly by the Leipzig University Medical Center and the Mittweida University of Applied Sciences.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that have been used within the study is already publicly available. We also provide the source code of all methods included in the evaluation by the public accessible GitLab.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

GANs	Generative Adversarial Networks
CGANs	Conditional Generative Adversarial Networks
CTGANs	Conditional Tabular Generative Adversarial Networks
CopulaGANs	Copula Generative Adversarial Networks
WGANs	Wasserstein Generative Adversarial Networks
WGANGP	Wasserstein Generative Adversarial Networks with Gradient Penalty
SVM	Support Vector Machines
MLP	Multi-Layer Perceptron
BCW	Breast Cancer Wisconsin
BCC	Breast Cancer Coimbra

References

1. Dahmen, J.; Cook, D. SynSys: A Synthetic Data Generation System for Healthcare Applications. *Sensors* **2019**, *19*, 1181. [[CrossRef](#)] [[PubMed](#)]
2. Tucker, A.; Wang, Z.; Rotalinti, Y.; Myles, P. Generating high-fidelity synthetic patient data for assessing machine learning healthcare software. *NPJ Digit. Med.* **2020**, *3*, 147. [[CrossRef](#)] [[PubMed](#)]

3. Chen, R.J.; Lu, M.Y.; Chen, T.Y.; Williamson, D.F.K.; Mahmood, F. Synthetic data in machine learning for medicine and healthcare. *Nat. Biomed. Eng.* **2021**, *5*, 493–497. [\[CrossRef\]](#) [\[PubMed\]](#)
4. Hernandez, M.; Epelde, G.; Alberdi, A.; Cilla, R.; Rankin, D. Synthetic data generation for tabular health records: A systematic review. *Neurocomputing* **2022**, *493*, 28–45. [\[CrossRef\]](#)
5. Voigt, P.; von dem Bussche, A. *The EU General Data Protection Regulation (GDPR)*; Springer International Publishing: Berlin/Heidelberg, Germany, 2017. [\[CrossRef\]](#)
6. Gehring, S.; Eulenfeld, R. German Medical Informatics Initiative: Unlocking Data for Research and Health Care. *Methods Inf. Med.* **2018**, *57*, e46–e49. [\[CrossRef\]](#)
7. Bearnot, B.; Pearson, J.F.; Rodriguez, J.A. Using Publicly Available Data to Understand the Opioid Overdose Epidemic: Geospatial Distribution of Discarded Needles in Boston, Massachusetts. *Am. J. Public Health* **2018**, *108*, 1355–1357. [\[CrossRef\]](#)
8. Saldanha, I.J.; Smith, B.T.; Ntzani, E.; Jap, J.; Balk, E.M.; Lau, J. The Systematic Review Data Repository (SRDR): Descriptive characteristics of publicly available data and opportunities for research. *Syst. Rev.* **2019**, *8*, 334. [\[CrossRef\]](#)
9. Okeahalam, C.; Williams, V.; Otwombe, K. Factors associated with COVID-19 infections and mortality in Africa: a cross-sectional study using publicly available data. *BMJ Open* **2020**, *10*, e042750. [\[CrossRef\]](#)
10. Khan, S.M.; Liu, X.; Nath, S.; Korot, E.; Faes, L.; Wagner, S.K.; Keane, P.A.; Sebire, N.J.; Burton, M.J.; Denniston, A.K. A global review of publicly available datasets for ophthalmological imaging: Barriers to access, usability, and generalisability. *Lancet Digit. Health* **2021**, *3*, e51–e66. [\[CrossRef\]](#)
11. European Commission and Directorate-General for Research and Innovation. *Rare Diseases: A Major Unmet Medical Need*; Publications Office: Luxembourg, 2017. [\[CrossRef\]](#)
12. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016; Google-Books-ID: omivDQAAQBAJ.
13. Creswell, A.; White, T.; Dumoulin, V.; Arulkumaran, K.; Sengupta, B.; Bharath, A.A. Generative Adversarial Networks: An Overview. *IEEE Signal Process. Mag.* **2017**, *35*, 53–65. [\[CrossRef\]](#)
14. Bourou, S.; El Saer, A.; Velivassaki, T.H.; Voulkidis, A.; Zahariadis, T. A Review of Tabular Data Synthesis Using GANs on an IDS Dataset. *Information* **2021**, *12*, 375. [\[CrossRef\]](#)
15. Borisov, V.; Leemann, T.; Seßler, K.; Haug, J.; Pawelczyk, M.; Kasneci, G. Deep Neural Networks and Tabular Data: A Survey. *arXiv* **2021**, arXiv:2110.01889.
16. Patki, N.; Wedge, R.; Veeramachaneni, K. The Synthetic Data Vault. In Proceedings of the 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), Montreal, QC, Canada, 17–19 October 2016; pp. 399–410. [\[CrossRef\]](#)
17. Walonoski, J.; Kramer, M.; Nichols, J.; Quina, A.; Moesel, C.; Hall, D.; Duffett, C.; Dube, K.; Gallagher, T.; McLachlan, S. Synthea: An approach, method, and software mechanism for generating synthetic patients and the synthetic electronic health care record. *J. Am. Med. Inform. Assoc.* **2018**, *25*, 230–238. [\[CrossRef\]](#)
18. Meyer, D.; Nagler, T. Synthia: Multidimensional synthetic data generation in Python. *J. Open Source Softw.* **2021**, *6*, 2863. [\[CrossRef\]](#)
19. Nowok, B.; Raab, G.M.; Dibben, C. synthpop: Bespoke Creation of Synthetic Data in R. *J. Stat. Softw.* **2016**, *74*, 1–26. [\[CrossRef\]](#)
20. Templ, M.; Meindl, B.; Kowarik, A.; Dupriez, O. Simulation of Synthetic Complex Data: The R Package simPop. *J. Stat. Softw.* **2017**, *79*, 1–38. [\[CrossRef\]](#)
21. Kingma, D.P.; Welling, M. Auto-Encoding Variational Bayes. *arXiv* **2013**, arXiv:1312.6114.
22. Kingma, D.P.; Welling, M. An Introduction to Variational Autoencoders. *Found. Trends Mach. Learn.* **2019**, *12*, 307–392. [\[CrossRef\]](#)
23. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Networks. *arXiv* **2014**, arXiv:1406.2661.
24. Elbattah, M.; Loughnane, C.; Guérin, J.L.; Carette, R.; Cilia, F.; Dequen, G. Variational Autoencoder for Image-Based Augmentation of Eye-Tracking Data. *J. Imaging* **2021**, *7*, 83. [\[CrossRef\]](#)
25. Gootjes-Dreesbach, L.; Sood, M.; Sahay, A.; Hofmann-Apitius, M.; Fröhlich, H. Variational Autoencoder Modular Bayesian Networks for Simulation of Heterogeneous Clinical Study Data. *Front. Big Data* **2020**, *3*, 16. [\[CrossRef\]](#) [\[PubMed\]](#)
26. Alqahtani, H.; Kavakli-Thorne, M.; Kumar, G. Applications of Generative Adversarial Networks (GANs): An Updated Review. *Arch. Comput. Methods Eng.* **2021**, *28*, 525–552. [\[CrossRef\]](#)
27. Hameed, K.; Chai, D.; Rassau, A. Texture-based latent space disentanglement for enhancement of a training dataset for ANN-based classification of fruit and vegetables. *Inf. Process. Agric.* **2021**, in press. [\[CrossRef\]](#)
28. Vaccari, I.; Orani, V.; Paglialonga, A.; Cambiaso, E.; Mongelli, M. A Generative Adversarial Network (GAN) Technique for Internet of Medical Things Data. *Sensors* **2021**, *21*, 3726. [\[CrossRef\]](#)
29. Lv, J.; Zhu, J.; Yang, G. Which GAN? A comparative study of generative adversarial network-based fast MRI reconstruction. *Philos. Trans. R. Soc.* **2021**, *379*, 20200203. [\[CrossRef\]](#)
30. Khan, Z.K.; Umar, A.I.; Shirazi, S.H.; Rasheed, A.; Qadir, A.; Gul, S. Image based analysis of meibomian gland dysfunction using conditional generative adversarial neural network. *BMJ Open Ophthalmol.* **2021**, *6*, e000436. [\[CrossRef\]](#) [\[PubMed\]](#)
31. Wanichwecharungruang, B.; Kaothanthong, N.; Pattanapongpaiboon, W.; Chantangphol, P.; Seresirikachorn, K.; Srisuwanporn, C.; Parivisutt, N.; Grzybowski, A.; Theeramunkong, T.; Ruamviboonsuk, P. Deep Learning for Anterior Segment Optical Coherence Tomography to Predict the Presence of Plateau Iris. *Ranslational Vis. Sci. Technol.* **2021**, *10*, 7. [\[CrossRef\]](#) [\[PubMed\]](#)

32. Larsen, A.B.L.; Sønderby, S.K.; Larochelle, H.; Winther, O. Autoencoding beyond pixels using a learned similarity metric. In Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 20–22 June 2016; PMLR: London, UK, 2016; pp. 1558–1566.
33. Baur, C.; Wiestler, B.; Albarqouni, S.; Navab, N. Deep Autoencoding Models for Unsupervised Anomaly Segmentation in Brain MR Images. In Proceedings of the Conjunction with MICCAI 2018, Granada, Spain, 16 September 2018; Volume 11383, pp. 161–169.
34. Kwon, G.; Han, C.; Kim, D. Generation of 3D Brain MRI Using Auto-Encoding Generative Adversarial Networks. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Shenzhen, China, 13–17 October 2019.
35. Liu, M.Y.; Huang, X.; Yu, J.; Wang, T.C.; Mallya, A. Generative Adversarial Networks for Image and Video Synthesis: Algorithms and Applications. *arXiv* **2020**, arXiv:2008.02793. [[CrossRef](#)]
36. Karras, T.; Laine, S.; Aittala, M.; Hellsten, J.; Lehtinen, J.; Aila, T. Analyzing and Improving the Image Quality of StyleGAN. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020.
37. Shahriar, S. GAN Computers Generate Arts? A Survey on Visual Arts, Music, and Literary Text Generation using Generative Adversarial Network. *Displays* **2022**, *73*, 102237. [[CrossRef](#)]
38. Choi, E.; Biswal, S.; Malin, B.; Duke, J.; Stewart, W.F.; Sun, J. Generating Multi-label Discrete Patient Records using Generative Adversarial Networks. *arXiv* **2018**, arXiv:1703.06490.
39. Subramanian, S.; Rajeswar, S.; Dutil, F.; Pal, C.; Courville, A. Adversarial Generation of Natural Language. In Proceedings of the 2nd Workshop on Representation Learning for NLP, Vancouver, BC, Canada, 3 August 2017; Association for Computational Linguistics: Stroudsburg, PA, USA, 2017; pp. 241–251. [[CrossRef](#)]
40. Ren, Y.; Lin, J.; Tang, S.; Zhou, J.; Yang, S.; Qi, Y.; Ren, X. Generating Natural Language Adversarial Examples on a Large Scale with Generative Models. *arXiv* **2020**, arXiv:2003.10388. [[CrossRef](#)]
41. Baowaly, M.K.; Lin, C.C.; Liu, C.L.; Chen, K.T. Synthesizing electronic health records using improved generative adversarial networks. *J. Am. Med. Inform. Assoc.* **2019**, *26*, 228–241. [[CrossRef](#)] [[PubMed](#)]
42. Mendelevitch, O.; Lesh, M.D. Fidelity and Privacy of Synthetic Medical Data. *arXiv* **2021**, arXiv:2101.08658.
43. Goncalves, A.; Ray, P.; Soper, B.; Stevens, J.; Coyle, L.; Sales, A.P. Generation and evaluation of synthetic patient data. *BMC Med. Res. Methodol.* **2020**, *20*, 108. [[CrossRef](#)] [[PubMed](#)]
44. Zhao, Z.; Kunar, A.; Birke, R.; Chen, L.Y. CTAB-GAN: Effective Table Data Synthesizing. In Proceedings of the 13th Asian Conference on Machine Learning, PMLR, Virtual, 18–24 July 2021; pp. 97–112.
45. Xu, L.; Skoularidou, M.; Cuesta-Infante, A.; Veeramachaneni, K. Modeling Tabular data using Conditional GAN. *arXiv* **2019**, arXiv:1907.00503.
46. Gui, J.; Sun, Z.; Wen, Y.; Tao, D.; Ye, J. A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications. *arXiv* **2020**, arXiv:2001.06937. [[CrossRef](#)]
47. Wu, X.; Xu, K.; Hall, P. A survey of image synthesis and editing with generative adversarial networks. *Tsinghua Sci. Technol.* **2017**, *22*, 660–674. [[CrossRef](#)]
48. Pieters, M.; Wiering, M. Comparing Generative Adversarial Network Techniques for Image Creation and Modification. *arXiv* **2018**, arXiv:1803.09093. [[CrossRef](#)]
49. Torres-Reyes, N.; Latifi, S. Audio Enhancement and Synthesis using Generative Adversarial Networks: A Survey. *Int. J. Comput. Appl.* **2019**, *182*, 27–31. [[CrossRef](#)]
50. Mirza, M.; Osindero, S. Conditional Generative Adversarial Nets. *arXiv* **2014**, arXiv:1411.1784
51. Xu, L.; Veeramachaneni, K. Synthesizing Tabular Data using Generative Adversarial Networks. *arXiv* **2018**, arXiv:1811.11264.
52. Kamthe, S.; Assefa, S.; Deisenroth, M. Copula Flows for Synthetic Data Generation. *arXiv* **2021**, arXiv:2101.00598.
53. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein GAN. *arXiv* **2017**, arXiv:1701.07875.
54. Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; Courville, A. Improved Training of Wasserstein GANs. *arXiv* **2017**, arXiv:1704.00028.
55. Engelmann, J.; Lessmann, S. Conditional Wasserstein GAN-based Oversampling of Tabular Data for Imbalanced Learning. *Expert Syst. Appl.* **2021**, *174*, 114582. [[CrossRef](#)]
56. Wolberg, W.; Street, W.; Mangasarian, O. Breast Cancer Wisconsin (Diagnostic); UCI Machine Learning Repository. 1995. Available online: <https://archive-beta.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+diagnostic> (accessed on 10 May 2022).
57. Patrício, M.; Pereira, J.; Crisóstomo, J.; Matafome, P.; Gomes, M.; Seica, R.; Caramelo, F. Breast Cancer Coimbra; UCI Machine Learning Repository. 2018. Available online: <https://archive-beta.ics.uci.edu/ml/datasets/Breast+Cancer+Coimbra> (accessed on 10 May 2022).
58. Li, Y. Performance Evaluation of Machine Learning Methods for Breast Cancer Prediction. *Appl. Comput. Math.* **2018**, *7*, 212. [[CrossRef](#)]
59. Patrício, M.; Pereira, J.; Crisóstomo, J.; Matafome, P.; Gomes, M.; Seica, R.; Caramelo, F. Using Resistin, glucose, age and BMI to predict the presence of breast cancer. *BMC Cancer* **2018**, *18*, 29. [[CrossRef](#)]
60. Austria, Y.D.; Goh, M.L.; Sta Maria, L., Jr.; Lalata, J.A.; Goh, J.E.; Vicente, H. Comparison of Machine Learning Algorithms in Breast Cancer Prediction Using the Coimbra Dataset. *Int. J. Simul. Syst. Sci. Technol.* **2019**, *7*, 23.1–23.8. [[CrossRef](#)]

61. Wolberg, W.H.; Street, W.; Mangasarian, O. Machine learning techniques to diagnose breast cancer from image-processed nuclear features of fine needle aspirates. *Cancer Lett.* **1994**, *77*, 163–171. [[CrossRef](#)]
62. Shahnaz, C.; Hossain, J.; Fattah, S.A.; Ghosh, S.; Khan, A.I. Efficient approaches for accuracy improvement of breast cancer classification using wisconsin database. In Proceedings of the 2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC), Dhaka, Bangladesh, 21–23 December 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 792–797. [[CrossRef](#)]
63. Obaid, O.I.; Mohammed, M.A.; Ghani, M.K.A.; Mostafa, S.A.; AL-Dhief, F.T. Evaluating the Performance of Machine Learning Techniques in the Classification of Wisconsin Breast Cancer. *Int. J. Eng. Technol.* **2018**, *7*, 160–166. [[CrossRef](#)]
64. Agarap, A.F.M. On breast cancer detection: An application of machine learning algorithms on the wisconsin diagnostic dataset. In Proceedings of the 2nd International Conference on Machine Learning and Soft Computing—ICMLSC’18, Phu Quoc Island, Vietnam, 2–4 February 2018; ACM Press: New York, NY, USA, 2018; pp. 5–9. [[CrossRef](#)]
65. Anguita, D.; Ghio, A.; Greco, N.; Oneto, L.; Ridella, S. Model selection for support vector machines. *Advant. Disadvant. Mach. Learn. Theory* **2010**, *12*, 1–8. [[CrossRef](#)]
66. Dankar, F.K.; Ibrahim, M.K.; Ismail, L. A Multi-Dimensional Evaluation of Synthetic Data Generators. *IEEE Access* **2022**, *10*, 11147–11158. [[CrossRef](#)]
67. Theis, L.; Oord, A.v.d.; Bethge, M. A Note on the Evaluation of Generative Models. *arXiv* **2015**, arXiv:1511.01844.
68. Rankin, D.; Black, M.; Bond, R.; Wallace, J.; Mulvenna, M.; Epelde, G. Reliability of Supervised Machine Learning Using Synthetic Data in Health Care: Model to Preserve Privacy for Data Sharing. *JMIR Med. Inform.* **2020**, *8*, e18910. [[CrossRef](#)]