U.PORTO
FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO

MSc

Synthetic data and re-identification risks

Diogo André França Fernandes

Dissertação de Mestrado apresentada à
Faculdade de Ciências da Universidade do Porto em
Data Science

2021

MSc

2.º CICLO

FCUP
ANO

U.PORTO

Insira título da dissertação, projeto ou estágio, letra Arial, tamanho 11, justificado à esquerda

Nome do Autor, letra Arial Bold tamanho 10, justificado à esquerda

FC

U.PORTO
FACULDADE DE CIÊNCIAS
UNIVERSIDADE DO PORTO

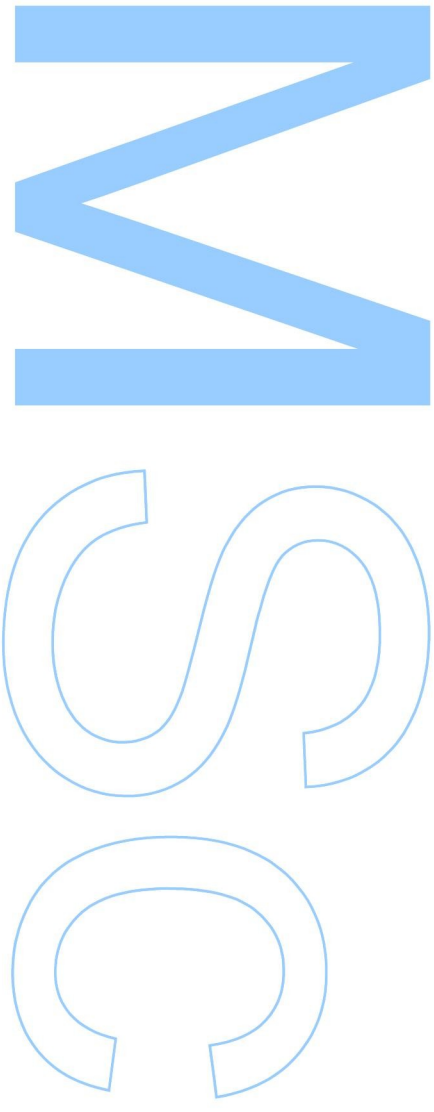
MSc

Synthetic data and re-identification risks

Diogo André França Fernandes
Mestrado Integrado em Engenharia de Redes e Sistemas Informáticos
Departamento de Ciência de Computadores
2021

Orientador
Ricardo João Cruz Correia, Professor Auxiliar, Faculdade de Medicina da Universida Porto

Coorientador
Luís Filipe Coelho Antunes, Professor Catedrático, Faculdade de Ciências da Univer do Porto



Aknowledgments

First of all, I would like to thank all the professors who have guided me during these last years, allowing me to complete my Bachelor's and Master's studies, providing the foundations and technical knowledge to reach this moment.

I would like to thank Professor Rui Prior, director of the Integrated Masters course in Network Engineering and Computer Systems.

I would then like to thank my supervisor, Professor Luis Filipe Antunes and co-supervisor Professor Ricardo Correia, who made this thesis possible.

I would like to thank everyone involved in the Department of Sciences and Computers, Faculty of Sciences, University of Porto, for all the help and kindness they showed in the years I attended the department. I would like to emphasize a special thanks to Professor Inês Dutra and Systems Administrator Paulo Ramos who provided me with a server to develop part of this thesis.

I want to thank João Almeida for his support on multiple themes and knowledge sharing.

I would also like to thank Alexandra Ferreira and Isabel Gonçalves for their amazing bureaucratic support during the course.

Last but not least I would like to thank my family and my girlfriend for their outstanding support and keeping me motivated through the most difficult times.

Abstract

Healthcare is becoming more integrated with technology, and data that would previously be transcripted and stored in the paper is now digitally stored. This transition was done progressively but nowadays the majority of facilities have a computer system to handle all of its needs. Data collection of this magnitude, of such sensitive data, in addition to regulations from governments, generates discussion on how should the data be handled. For these reasons and many more to come in the future, there is the need to understand how to possibly handle our clinical data safely, and still not put barriers on scientific research that could use that data for good use. Some of the previous methods to disclosure data are no longer viable, because of complex algorithms that compromise data powered by increasing computing power, also impacting the probability of data leaks. This is a concerning topic that can be mitigated with the solution we try to find on this thesis, that is the creation of synthetic healthcare records. In order to test if an approach like this is a viable in the future, for healthcare institutions to adopt. we investigated what models presented the option of generating tabular data with a good relation between quality and utility. Out of this analysis came out five promising models which were then fed with data from the MIMIC-III dataset, a renowned healthcare dataset with multiple types of tabular hospital data to choose from. After having both components we then generated new data that had to be further analyzed, first we tested how useful was the generated data since private data is useless if it has no quality and it bears itself useless. This lead to mixed results, by having models such as Synthpop that performed well across almost all categories, and others such as CTGAN which was a promising model but had disappointing results in the end. Finally, we had the re-identification risk analysis which is difficult at best, since quantification of this theme is still a wide open debate and there is no consensus across the area. Despite that, we performed what tests were available and came to the conclusion that despite still not being a solution without compromises, synthetic health data generation can still be used as a way to have less bureaucracy when the need to share with researchers arises.

Resumo

A saúde está cada vez mais integrada com a tecnologia e os dados que antes seriam transcritos e armazenados em papel, são agora armazenados digitalmente. Esta transição foi feita de forma progressiva, mas hoje em dia a maioria das instalações possui um sistema informático para lidar com todas as suas necessidades. A obtenção de dados sensíveis nesta magnitude, além de regulamentações dos governos, gera discussão sobre como os dados devem ser tratados. Por estas e muitas outras razões que virão, existe a necessidade de entender como é possivel lidar com segurança com os dados clínicos obtidos, e mesmo assim não impedir a pesquisa científica, que pode usar estes mesmos dados para o avanço da saúde. Alguns dos métodos anteriores para divulgação de dados não são os mais viáveis devido a algoritmos complexos que comprometem os dados e a ainda a possibilidade de fuga de informação. Este é um tópico preocupante que pode ser mitigado com a solução que tentamos encontrar nesta tese, a criação de registos sintéticos de saúde. Para testar se esta é uma solução viável para as instituições de saúde adotarem, investigamos quais os modelos que apresentavam a opção de gerar dados tabulares com o mínimo de configuração necessário. Desta análise surgiram cinco modelos promissores onde foram introduzidos dados do conjunto MIMIC-III, um conjunto de dados de saúde de remome com vários tipos de dados hospitalares tabulares para escolha. Depois de termos os dois componentes, geramos novos dados que precisaram ser analisados posteriormente. Primeiro testamos a utilidade dos dados gerados, uma vez que os dados privados são inúteis se não tiverem qualidade. Isso levou a resultados mistos, modelos como Synthpop tiveram um bom desempenho em quase todas as categorias, e outros como CTGAN, que era um modelo promissor, teve resultados dececionantes no final. Por fim, tivemos a análise de privacidade que é, na melhor das hipóteses, difícil, uma vez que a quantificação desse tema ainda é uma discussão em aberto e não há consenso em toda a área. Apesar disso, realizamos os testes que estavam disponíveis e chegamos à conclusão que apesar de ainda não ser uma solução perfeita, ainda pode ser utilizada como forma de reduzir a carga burocrática quando surgir a necessidade de compartilhar com investigadores.

Contents

List of Tables

List of Figures

Listings

Acronyms

AE Autoencoders

CAP Correct Attribution Probability

CART Classification and Regression Trees

CPU Central Processing Unit

CSV Comma Separeted Value

CS Chi-Squared test

CTGAN Conditional tabular GAN

CUDA Compute Unified Device Architecture

DCC Departamento de Ciência de Computadores

DP Differential Privacy

EHR Electronic health records

FCUP Faculdade de Ciências da Universidade do Porto

GAN Generative Adversarial Networks

GCAP Generalized Correct Attribution Probabilit

GPU Graphics Processing Unit

ICD International Statistical Classification of Diseases and Related Health Problems

IDC International Data Corporation

IDC International Data Corporation

ISO International Organization for Standardization

IoT Internet of Things

JSON JavaScript Object Notation

KS test Kolmogorov–Smirnov test

MIMIC Medical Information Mart for Intensive Care

ML Machine Learning

MSU Minimal Sample Unique

NaN Not a Number

PHI personal health information

RAM Random Access Memory

SDC Statistical disclosure control

SDV Synthetic Data Vault

SOC System on a Chip

SUDA Special Unique Detection Algorithm

SVR Support Vector Regression

TPU Tensor Processing Units

VAE Variational Autoencoders

kNN k Nearest Neighbour

Chapter 1

Introduction

Since 2000 it is possible to find that specialists have been debating over data privacy on the internet [1], and up to this day, there still is no clear answer on the subject. New laws keep being implemented [2] and companies try to get around those same laws.[3] This problem surges from the information age we are currently living in, and with more and more connected devices [4] as well as an abundant number of massive data breaches [5], for this matter, responsible entities should be looking for all possible solutions.

Countries have been trying to adapt to these emerging situations. Several data protection laws are already being implemented or are close to being passed.[6] This shows an effort to try and regulate possible future problems and mitigate the current ones.

The healthcare field is an area that is gathering unprecedented amounts of data of its users, although in contrast to other companies (this is by necessity). It is estimated that this field generated approximately 2314 exabytes by the end of 2020.[7] This number has increased by 1412.42% when comparing to 2013. As more Internet of things (IoT) devices are being added to hospitals and all healthcare-related facilities [8], and the increase in information systems used to collect patients' data, this number will only increase even further in these next few years.

These records are known as personal health information (PHI) and are extremely sensitive because if released in public, it would jeopardize individual patient rights and even their safety. For that precise manner, records of that nature are kept under close eye to prevent any information leaks. Although applying restrictions to access this data can improve the resilience to a data breach, they are hard to maintain and may halt the use of this data by researchers that would get the most out of the data.[9]

While there are many traditional anonymization techniques, these are not possible to use with the evolution of the datasets and its' complexity. That is why there is a need to find new ways to anonymize data.[10]

A new approach to anonymization has been the creation of synthetic datasets using machine learning. Since 2016 there is a noticeable increase in the amount of work done in this area and

saw a peak of interest in 2020 according to Google trends, as demonstrated in figure 1.1.[11]

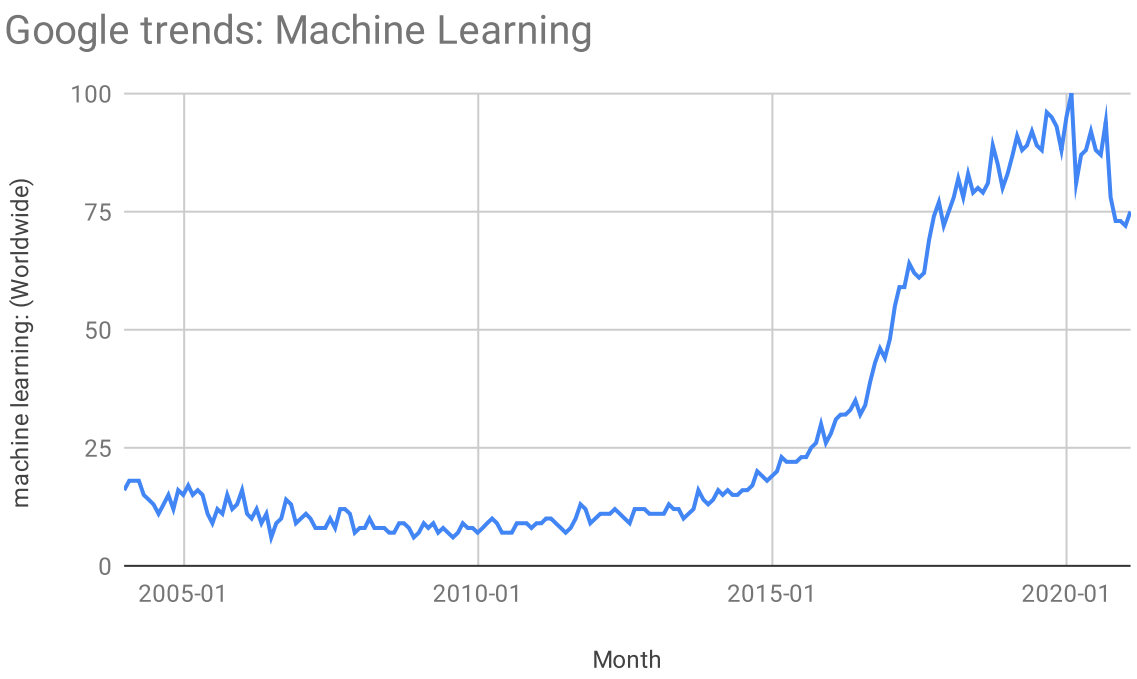


Figure 1.1: Google trends on graph on machine learning

All the scenario above leads us to what we aim to achive in this thesis, which is to assess the most prominent machine learning models of synthetic generation of healthcare data in terms of re-identification risks. In the first phase, to gather multiple datasets from hospitals. These datasets are provided by the company Healthy Systems, where I was integrated through the duration of the thesis.

The second step was to find state-of-the-art machine learning models that could create synthetic data from tabular data, giving preference to models already tested with clinical datasets. Following this step, we trained the models with the datasets and proceeded to analyze the output. And a way to remove some of the datasets was to assess the quality of the generated data.

The final step and the main contribution of this thesis was to verify what were the risks of re-identification on these datasets.

Chapter 2

State of the Art

2.1 Data

Data is formally described as a set of values of qualitative or quantitative variables about one or more persons or objects.[12] Structuring of data evolved from the information age as a result of the generation of copious amounts of data, since almost everything is connected to the internet and constantly collecting telemetry.

The healthcare field is not different, and with the COVID-19 world pandemic this increased the ammount of data even further than expected. According to International Data Corporations (IDC's) Global DataSphere, in 2020 64.2 Zetabytes of data were generated across multiple industries, and these numbers are expected to increase even further with upcoming years.[13]

Geting to know this reality it is important to stratify how data is structured the most relevant features to this thesis.

2.1.1 Microdata

The formal definition of microdata across the literature is that it "is data on the characteristics of units of a population, such as individuals, households, or establishments, collected by census, survey, or experiment".

According to Winkelmann e Boes [14] microdata:

- Is cross-sectional, which by definition, it is sourced from individual entities.
- Is observational, meaning, data that originates from surveys as well as administrative records.
- Often has a non-continuous measurement scale; for example, the distinction of continuous or categorical variables demands a non-continuous scale.

We then have the various types of microdata, which essentially can be segregated into quantitative and qualitative, also referred to as categorical, data. Each one of them has several distinguishing features:

Qualitative

- Variables are only discrete
- Can be binary, multinomial, and ordered.
- Variables may be censored, truncated.

Quantitative

- Can be discrete or continuous.
- Possible to distinguish data with restricted and unrestricted range.
- Variables may be non-negative, censored, truncated, or grouped.

2.1.2 Tabular data

Data is available in many forms, ranging from an unstructured state, which essentially is what microdata is, up to complex relational databases that retain a large quantity of information. Regardless of the format, there is a lot of information stored in copious forms of tabular data, for example, CSV and Excel files. [15] Kaggle, which is a well know platform used by data scientists for its datasets, shows that tabular data is the most used data type in business and the second most used in a more academic context.[16]

The basic structure of tabular data is essentially composed of, qualitative or quantitative microdata, which is then structured by aggregation.

2.1.3 Electronic health records

Electronic health records (EHR) date as far as the 90s, and since then, it has been in constant evolution. The International Organization for Standardization (ISO), defines EHR as "...a repository of information regarding the health of a subject of care in computer processable form, stored and transmitted securely, and accessible by multiple authorized users.[17] It has a commonly agreed logical information model which is independent of EHR systems. Its primary purpose is to support continuous, efficient and quality integrated healthcare, and it contains information which is retrospective, concurrent and prospective".[18]

According to Shortliffe and Cimino,[18] there are five functional characteristics that are fundamental to every EHR System, those are:

- Integrated view of patient data: This means capturing the most amount of data possible, taking into consideration complications such as old paper notes, which can not be transcribed to EHR, data that is isolated in external systems, and the different conceptualization of data between systems;
- Assist healthcare professionals when they must take actions and make decisions would reduce the number of errors as it provides the caregiver with opportunities to deliver decision support and reduce costs for every patient;
- Clinical decision support: This method has been proven to improve the care process.[19],[20] It plays a crucial role in providing follow-up to patients who do not attend routine check-ups and miss appointments overall. A second impactful situation is when it comes to prevention, as it can help pinpoint underlying problems that would not be detected otherwise and could not be related to the current appointment;
- Access to knowledge resources: EHR should be able to promptly present information from multiple information sources which are relevant to whichever situation healthcare workers are presented;
- Integrated communication and reporting support: The increasing number of healthcare professionals fragments the knowledge of the patient as each one has different insights over the patient, not only this but also the possibility to visit different hospitals and clinics pushes this knowledge splitting even further. In order to patients get the best treatment, professionals need to have access to the whole clinical situation. This final component of an EHR system is making the bridge between institutions and professionals alike.

A modern electronic health record serves multiple functions. Apart from the acquisition and storage of data previously mentioned, it is also used to summarize and display data to facilitate communication and sharing information and assist with decision support.[18]

EHR relies on two types of data capture: the system's interface and the direct manual entry, meaning no electronic entry. The first method improves the data quality and enables all benefits associated with Electronic Health Records, mainly providing structured data to perform further analysis and decision assistance.

2.2 Data Analysis

The sum of all previously mentioned topics give us what it we can call, real data collected by EHR systems. This data by itself is not useful as it first needs to go through data cleaning, data analysis, and interpretation of any results. Getting data is only the starting step, as it requires further steps before starting the analysis.

In the scope of this thesis, it is supposed to analyze if the quality of synthetic generated data is good enough to be shared not only from a privacy standpoint but also if it retains useful and

correct information. Albeit being difficult to quantify how useful generated data truly is, since it is arduous to compare if everything learned from the original data could potentially be learned from the synthetic one, it is important to empower what is described as global measures of data utility and try grasping what is truly left.[21]

It is essential to have good data, but it is meaningless if there is no good way of interpreting it. To tackle this problem we use data visualization, as it provides a way to perceive data, and understand what is behind everything as it always is easier to internalize something visible. Now more than ever, it is of utmost importance to understand how having good visuals when representing scientific data, it is crucial to pass a clear message of the results. As more tools appear and not all of them are put to good use, extra care is needed when approaching this topic and choosing what to use and when to use it. [22]

2.3 Machine Learning

Machine learning is on the rise in modern days. From the regular sense of more research-oriented applications to smartphone processors, companies are already starting to invest in machine learning cores on their systems on a chip (SoC) [23]. This shows that this once exclusive technology, is arriving at the hands of consumers, even if not obvious.

This will become a more predominant phenomenon as more models emerge from constant research, leading to programmers and companies finding more areas to apply them. Some more predominant areas where machine learning models are applied are Computer Vision, Natural Language Processing, Recommender Systems and Speech Recognition. All these can be divided into tasks, which are then solved by a specific model. From here, every field of study uses the one that is more convenient to the matter at hand, for example generating synthetic data.

Searching for articles shows that this is a relatively recent area, as most articles date to 2016 [24]. Since 2016 there have been many improvements as well as iterations of models. Furthermore, thanks to machine learning, a growing community of researchers, and the constant evolution of technology, it allows fast industry growth.

2.3.1 Generative Adversarial Networks

Most notably there is a regular appearance of Generative Adversarial Networks (GANs) , and several variations of this model applied to both healthcare data and more generic train data.

Generative Adversarial Networks are a powerful class of generative models that compete between two networks: a generator network produces synthetic data from the same distribution as the training data and some noise. The other is called a discriminator network, which tries to find if the data results are from the generator or true data. This form of training goes on until the discriminator cannot distinguish the true data from the synthetic one. Both of them train

each other by getting better at detecting each-others errors and forcing the other to improve. [25]

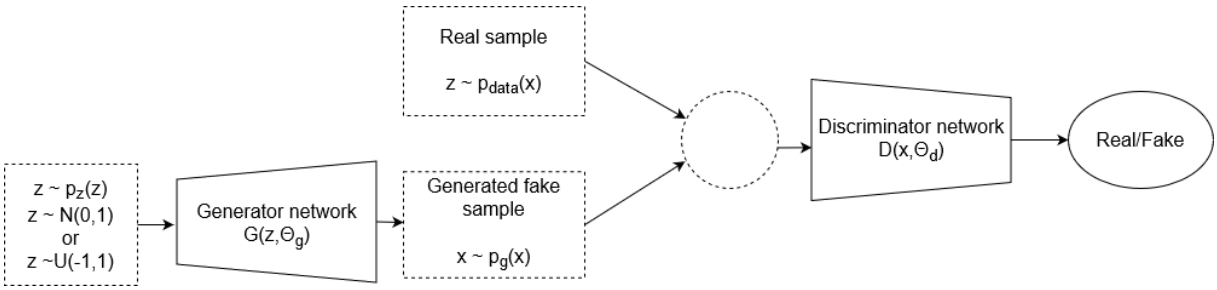


Figure 2.1: Generative Adversarial network framework

According to Vasilev et al, Figure 2.2 describes the framework on what lays the basis of a GAN and follows these notations. [26]

The generator $G(z, \theta_g)$ where θ_g are the network weights and z is the latent vector, meaning that it cannot be accessed throughout the training, this is what is fed to the generator. From there it will output a sample x with the probability of distribution of $p_g(x)$.

$D(x, \theta_d)$ denotes the discriminator, where once again the weights are presented as θ_g. It can get as input either real data, $x\ p_data(x)$ or the generated data $x\ p_g(x)$, proceeding to output if the data is generated or real.

Since GAN's are based on a competition between these two networks, each one will try to maximise its own results, translating to a sequential minmax zero-sum game represented by the following equation.[27]

$$min_g max_g V(G, D) = \frac{1}{2} E_{x\ p_data} log(D(x)) + \frac{1}{2} E_z (1 - log(D(x)))$$

This representation is the basis of what other GAN's are built upon, and some of models used on this thesis are variations themselves, where certain aspects are changed in order to accommodate for the difficulties related to generating synthetic tabular data.

2.3.2 Copulas

Reger B. Nelson describes copulas as "...functions that join or "copule", multivariate distribution functions to their one-dimensional marginal distribution functions." and "... copulas are multivariate distribution functions whose one-dimensional margins are uniform on the interval (0,1)."[28]

According to Gal Elidan, multivariate modeling is pivotal in areas like computational biology and computer vision and combining that with the previous statement it is understandable that copulas are useful for machine learning tasks.[29]

A more formal definition is given by Thorsten Schmidt[30]:

A d-dimensional copula $C : [0,1]^d :\rightarrow [0,1]$ is a function which is cumulative distribution function with uniform marginals.

Given that $C(u) = C(u_1, ..., u_d)$ is a copula, then C is a distribution function with the following properties

- As cumulative distribution functions only increase, so C(u) is increasing in each component u_i.
- The marginal in component i is obtained by setting $u_j = 1$ for all $j \neq i$ and as it must be uniformly distributed.
$$C(1, ..., 1, u_i, 1, ..., 1) = u_i$$
- For $a_i < b_i$, $P(U_1 \in [a_1, b_1], ..., P(U_d \in [a_d, b_d])$ must be greater that 0 leading to:
$$\sum_{i_1=1}^{2} ... \sum_{i_d=1}^{2} (-1)^{i_1+...+i_d} C(u_{1,i_1}, ..., u_{d,i_d}) \geq 0$$

2.3.3 Variational Autoencoder

This work is about Variational Autoencoders (VAE), which is a generative model and an iteration of regular autoencoders (AE).

This model used to not be so popular around the computer vision community. However, due to advances in training neural networks, by improving the back propagation-based function approximators, the trend shifted towards this model and became a fairly popular one.

According to Ian Goodfellow [27]

- The model draws a sample z from the code distribution $p_{model}(z)$
- z is then be fed to the differentiable generator network $g(z)$
- x is sampled from $p_{model}(x|z)$

Together with the previous steps, the encoder is used to obtain z and $p_{model}(x|z)$ which is the decoder as shown on figure 2.2.

Figure 2.2: Variational Autoencoder diagram

The architecture is composed of both an encoder and decoder, the change over the state of the art is, that it is possible to use the decoder for generative purposes. To do so the authors

had to have a regular latent space. Doing so involves encoding an input as a distribution over the latent space instead of encoding it as a single point.

In order to train the model, this solution had a loss function that consisted of two terms, one that consisted of the reconstruction cost similar to AE, and the second term represents the Kulback-Leibler divergence. This makes sure that the distribution that we are learning through the AE is not far from the normal distribution, forcing the latent distribution into being relatively close to an average of 0 and a standard deviation of 1 (N(0,1)).

As it is impossible to run back-propagation or push gradients through one sampling node, the authors present a crucial aspect of VAE, the reparameterization trick. To fully understand this concept, it is necessary to break down what is the sampled latent vector as:

$$z = \mu + \sigma\epsilon$$

- μ is the parameter the model is trying to learn
- σ is another learning parameter
- ϵ always N(1,0)

Instead of having a complete stochastic node that blocks all gradients, since it blocks back-propagation, this is split into parts where it is made possible to perform the back-propagation and another part that is stochastic, while not requiring training, as it is fixed.

2.3.4 Classification and regression trees

Classification and regression trees (CART) are, according to Wei-Yin Loh, "prediction models constructed by recursively partitioning a data set and fitting a simple model to each partition.".[31] It is essential to separate the model into classification trees and regression trees to fully understand the subject.

2.3.4.1 Classification trees

A classification tree can be summarized as a series of questions at each node which will eventually lead to a leaf. Each branch of a tree represents an attribute, and the leaves represent the final decision. The construction and growth of each tree depend on how many attributes are analyzed before achieving a leaf. Splitting nodes on classification trees is the most complicated part as it directly influences how tall the tree is, dictating the over-fitting. The most common methods for splitting are C4.5 which uses an entropy for impurity function, and CART uses the Gini Index.[32],[33]

2.3.4.2 Regression trees

Tree-based regression models are well-known simple models, yet efficient when dealing with a large number of variables and they can extend to multiple use cases.

According to Luis Torgo, [34] the approach to this model is to see it as an additive model [35],

$$m(x) = \sum_{i=1}^{l} k_i \times I(x \in D_i) \tag{2.1}$$

where k_i are constants; I() is and indicator function returning a boolean value; D_i are disjoint partitions of training data (D)

These trees are constructed using a recursive partitioning algorithm that consists of the following steps:

- A splitting rule.
- A way to determine if a node is terminal.
- A rule to assign a value to each terminal node.

This partition algorithm is then applied to every new branch created until each node becomes a terminal node, by the rule empowered in item two of the algorithm.

New ensemble techniques are also used to improve performance of regression trees, thee of the most common are, bagging (bootstrap aggregating), boosting, and random trees.

CART will use both of methodologies depending on what variable it has to predict. with classification it will attempt to predict a class label (finite values) and regression is used to predict a numerical label (infinite label).

2.4 Privacy

According to the Cambridge dictionary,[36] privacy is "someone's right to keep their personal matters and relationships secret".

With the increasing amount of data generated as technology evolves and every device is connected and generating data, privacy concern is paramount. The fast growth of technology and information technology in junction with the outdated and slow bureaucracy processes of creating and passing laws, as well as the lacking capability to analyze what is necessary to regulate the field, lead to companies taking advantage of the process and found what actual value all collected data from their platforms was truly worth.

Despite all negative sentiment towards these practices and all data hoarding that exists across the industry, instead of being scared and shutting everything down, legislation and people should adapt to it. Despite having several problems, this subject has much potential if used correctly. Even by today's standards, a good code of conduct is the code of fair information practices, as it still covers the fundamentals of what practices systems should follow.

2.4.1 Privacy in healthcare

Electronic health records have been massively adopted by now and are under constant development, one big development is the amount of data collected for each patient, leading to the creation of integrated data repositories impacting the observational clinical research, and 'big data' allowed analysis to progress to forms impossible prior to the technology.[37]

These EHRs need to follow some aspects to keep privacy: integrity, confidentiality, authenticity, accountability, audit, non-repudiation, anonymity, and unlinkability.[38] However, satisfying all these parameters is not easy, taking into consideration all variables at play.

The ramping creation of data and the need to access it across multiple facilities led to the point where data is stored in the cloud, with an increasing amount of hospitals that choose to do so.[39] This transition collides with the notions of privacy previously mentioned as it is of interest from the companies that store the data to sell it.

Despite all these advancements, these mechanisms still have a long way to go and obstacles to overcome, one of them being privacy concerns. Legal procedures are now more in line with what is expected regarding protecting patients' data. In European Union, the legislation is named General Data Protection Regulation. This regulation, despite tackling healthcare problems it does not cover all specific needs of this data as its nature is much more sensitive.[40]

While not disregarding privacy, it is important to note that medical data does help to improve manifold healthcare fields, and people greatly benefit from it. Such is why new techniques need to be developed while the wait for laws that can benefit both sides are approved, and every entity responsible figures how to make the most out of privacy and utility.

2.4.2 Differential Privacy

Differential privacy is a strong, mathematical definition of privacy in the context of statistical and machine learning analysis.[41]

This allows for data from all sources to be collected as well as processed while maintaining the privacy of the individuals [42]. All this has to be done while still ensuring the same conclusions of a query of the original dataset.

Differential privacy is a criteria that provides a mathematical model where it is possible to measure how information fares against privacy attacks such as re-identification, record-linkage

and differencing attacks.[41]

Understanding the formal definition requires an introduction to the concept of privacy loss parameter denoted as ϵ. This is what determines how much noise resides in the computation. So given a computation task, the value of ϵ will interfere directly on how private and accurate a dataset is in the end. [43]

"For a given computational task T and a given value of ϵ there will be many differentially private algorithms for achieving T in an ϵ-differentially private manner. Some will have better accuracy than others. When ϵ is small, finding a highly accurate ϵ-differentially private algorithm for T can be difficult, much as finding a numerically stable algorithm for a specific computational task can require effort." [43]

Definition of differential privacy: Let M be a randomized mechanism of adding noise to each individual query, it is considered to be ϵ-DP if for all of the dataset D, D' differs on one row and queries q $\forall$ sets T such that, $P[M(D,q) \in T] \leq e^{\epsilon} \cdot P[M(D',q) \in T]$. [44],[45],[46]

2.4.3 Disclosure

A common way that companies used to protect an individuals data was to simply remove data directly related to the identity of their users. This has been rendered essentially useless, as Sweeney [47] showed, it is possible to identify 87% of the United States population by only using three demographic attributes, being gender, date of birth and 5-digit zip code. Since these attributes are not considered identity ones, it is possible to identify a unique record related to the data, meaning these are a quasi-identifier [48].

2.4.3.1 Identity disclosure

Identity disclosure is one of the most dangerous acts regarding privacy concerns as it puts users' data in direct danger. Despite the advances in methods for data protection, for example, k-anonymity still has flaws, meaning that datasets created using these methods can still be cracked. This is because intruders can link data between released records and already public data, as well as the combination of rare values that form one-of-a-kind profiles easily identifiable. [49]

2.4.3.2 Attribute disclosure

When it comes to attribute disclosure, the intruder can determine a number of characteristics of an individual record based on the information of the whole dataset. This scenario happens when an entry is correctly re-identified, and the dataset contains variables containing previously unknown information to the intruder. [49]

2.4.3.3 Inferential disclosure

In inferential disclosure, "the intruder, though with some uncertainty, can predict the value of some characteristics of an individual more accurately with the released data." It is possible to accomplish this by using a regression model to predict an entry of sensitive information, through other data attributes. [49]

2.4.4 Statistical disclosure control (SDC)

It was previously stated that there is an increasing demand for data. Harvested data from individuals is extremely valuable as it can produce some good returns to whichever purpose it is used. The problem with making this data available to whoever requests it is that it breaks the confidentiality of the owners who generated the data.

Statistical disclosure control is a methodology that has been used as a method to release data while complying with some laws regarding data privacy protection laws.

Every method that tries to tackle the fundamental problem between privacy and utility has its limits. With an increasing number of attacks on this type of method and an increase in computational power, previously used methods of SDC are becoming obsolete.

Adding insult to injury, choosing what SDC method to use is trial and error. When we have larger datasets with sensitive and personal information, it becomes a risk that people and institutions are unwilling to take. [49]

Chapter 3

Methods and development

3.1 Dataset

Dataset selection was an arduous task, at first it was supposed to be non-public clinical data, but since time was short, and the impediments originated from the standard procedures which are necessary to access this type of sensitive data are too great, a different approach was taken.

Due to this setback, it was then decided to use a well-known semi-public clinical dataset named MIMIC-III.

According to "MIMIC (Medical Information Mart for Intensive Care) is an extensive, freely-available database comprising de-identified health-related data from patients who were admitted to the critical care units of the Beth Israel Deaconess Medical Center."[50]

In order to access this dataset, there are some procedures to take before being able to download and use it, hence the previous statement of being semi-public. The steps were as follows:

- Taking and passing a course in human subjects research;
- Sign in the data use agreement provided by the entity;
- Request recognition by this thesis tutor Prof. Dr. Ricardo Correia;

Only after approval from the responsible entity, PhysioNet, the access and use of the MIMIC-III database was granted.

3.1.1 Dataset in-depth

After the initial download (6Gb) and unpacking the compressed files, it totaled at 126 Gb of data, divided across 26 tables, from only three were chosen to perform experiments on Chapter 4.

The previously mentioned tables are:

- **DIAGNOSES_ICD**, that represents the codes on the International Statistical Classification of Diseases and Related Health Problems (ICD) system. This table contained 651,047 entries and has the following variables on the table 3.1

Table 3.1: DIAGNOSES_ICD variables

ROW_ID	Numerical	Line counter of entries
SUBJECT_ID	Numerical	Patient identifier
HADM_ID	Numerical	Unique identifier to patient hospital stay
SEQ_NUM	Numerical	Order on what the diagnoses were performed. ICD diagnoses are ordered by priority.
ICD9_CODE	Categorical	ICD-9 code for the diagnose

- **PROCEDURES_ICD**, shows what procedure each patient (identified by the same numerical system) has undergone through. The procedure is coded using the International Statistical Classification of Diseases and Related Health Problems system. This table contained 240,095 entries and has the following variables in table 3.2

Table 3.2: PROCEDURES_ICD variables

ROW_ID	Numerical	Line counter of entries
SUBJECT_ID	Numerical	Patient identifier
HADM_ID	Numerical	Unique identifier to patient hospital stay
SEQ_NUM	Numerical	Order on what the procedures were performed
ICD9_CODE	Categorical	ICD-9 code for the procedure

- **PRESCRIPTIONS**, tell the ordered medications (not necessarily administrated) for a given patient (identified by the same numerical system). This table contained 4,156,450 entries and has the following variables in table 3.3

Table 3.3: PRESCRIPTIONS variables

ROW_ID	Numerical	Line counter of entries
SUBJECT$_ID$	Numerical	Unique patient identifier
HADM_ID	Numerical	Unique identifier of hospital stay
ICUSTAY_ID	Numerical	Unique to patient ICU stay
STARTDATE	Categorical	Start date of prescription validity
ENDDATE	Categorical	End date of prescription validity
DRUG_TYPE	Categorical	Type of drug prescribed
DRUG	Categorical	Name of the drug
DRUG_NAME_POE	Categorical	Name of the drug on the Provider Order Entry interface
DRUG_NAME_GENERIC	Categorical	Generic drug name
FORMULARY_DRUG_CD	Categorical	Formulary drug code
GSN	Categorical	Genetic Sequence Number
NDC	Categorical	National Drug Code
PROD_STRENGTH	Categorical	Strength of the drug
DOSE_VAL_RX	Categorical	Dose of the drug prescribed
DOSE_UNIT_RX	Categorical	Unit of measurement associated with dose
FORM_VAL_DISP	Categorical	Amount of the formulation dispensed
FORM_UNIT_DISP	Categorical	Unit of measurement associated with the formulation
ROUTE	Categorical	Route of administration

After an overview of the chosen tables, it is possible to explain why these were chosen. First of all, the simplicity of both the PROCEDURES and DIAGNOSES, which use ICD9 codes, makes it very easy for a model to create synthetic data. The PRESCRIPTIONS one, despite being more complex, the data it has is displayed in multiple manners. By reducing the number of redundant variables, it was possible to create good data. Another critical factor was the size of each one, ranging from a decent size of 240095 entries to 4156450, giving away to check if models would behave differently according to the amount of data.

During the development of the thesis, MIMIC-IV was released, and its use was initially considered but then discarded due to its marginal contribution to the work already done using the MIMIC-III.

3.2 Synthetic data creation

Creation of synthetic data is becoming a recurrent topic, its usages vary from creating fake images, video and audio to simple tabular data. The current most used application is regarding video and image, both with outstanding results. With continuously evolving machine learning methods it was expected that tabular data would evolve alongside these other topics. Despite

that, tabular data is complicated to generate for machine learning models, this is because it is a more sensitive type of data where errors in creation have a more noticeable impact over the final results.[51]

In chapter 2 it was introduced multiple machine learning models, which are the basis for the final models used in the experiments in chapter 5.

As a first state on choosing what models to use, we searched for either models which were conscious about privacy and models used on clinical data specifically, preferably both. These preferences lead the research to the models presented in table 3.4.

Table 3.4: Initial model research

Anonymization Data Synthesis Generative Adversarial Networks	Developed for medical data. Tested in four public medical datasets with privacy as a primary goal.
Suport Vector Machine	Generates a synthetic dataset while satisfying a level of differential privacy. Tested on a free range of datasets.
DP-VaeGM/DP-AuGM	Tested in 19 healthcare datasets of different medical specialities. Provides complex analysis on privacy for both models.
Conditional Generative Adversarial Network	Tested on 4 datasets, on of them being hospital data. Different kinds of privacy attacks were tested.

Unfortunately, despite these being the best models according to their papers, the authors did not provide the code and was not publicly available for free use. The only one providing any access to the code was the last one which was the Conditional Generative Adversarial Network. This was the one that led to finding the Synthetic Data Vault repository, which was ultimately what is being used to generate the synthetic data.

3.2.1 Synthetic Data Vault

According to the authors, the Synthetic Data Vault (SDV) "is a Synthetic Data Generation ecosystem of libraries that allows users to easily learn single-table, multi-table and timeseries datasets to later on generate new Synthetic Data that has the same format and statistical properties as the original dataset."[52]

This ecosystem is composed by four single table models as described bellow.

3.2.1.1 Gaussian Copula

It has already been introduced in chapter 2 what copulas are and how they work, but what has been set aside is that one of the most commonly used copula according to is the Gaussian Copula. The different qualities allow an easy algorithm for random variate generation which is a quality important in generating synthetic data.[53]

3.2.1.2 Conditional tabular GAN (CTGAN)

CTGAN is a Deep Learning data synthesizer based on the GAN model already explained in chapter 2. This model has the objective to fix some of the constraints and problems associated with GAN to create tabular synthetic data. These problems usually do not occur on other data types, namely on images, which is the primary use for GANs. The problems this model overcomes are, mixed data types, as observed the mimic dataset has discrete and continuous data, which makes it more difficult for a GAN to generate due to the activation function; non-gaussian distribution, which regularly occurs in continuous variables, is a problem because when applying the loss function it creates the vanishing gradient problem; according to experiments lead by Srivastava et al.[39] In Advances in Neural Information Processing Systems, vanilla GAN struggle with modeling the multimodal distribution of continuous data. While sparse one-hot-encoded vectors allow discriminators to check if the data is either real or fake by checking how sparse is the distribution of the vector, the imbalance nature of categorical data, which can lead to the generator over-optimizing for a specific discriminator. This phenomenon leads the discriminator not to identify that and will not get out of a cycle since the generators will cycle through a small set of output types, known as mode collapse.[54]

Overcoming these problems was possible by doing mode-specific normalization, a newly designed conditional generator, training-by-sampling, fully-connected networks, and other methods explained in chapter 4 on modeling tabular data using conditional GAN.

3.2.1.3 CopulaGAN

CopulaGAN is a variation of the CTGAN Model, which uses cumulative distribution function-based transformation that gaussian copulas (previously explained) apply, making learning data easier in CTGAN. [55]

3.2.1.4 Tabular VAE

This version of the VAE, mentioned in chapter 2, follows the same process as the regular VAE, but the loss function is changed to the lower-bound loss. [56]

These are the single table models provided by SDV. All of them were used to create synthetic data of the preceding referenced tables. Albeit it there was another model used to create synthetic data outside these libraries, namely *Synthpop*.

3.2.2 Synthpop

Synthpop is originally a R package to produce "synthetic versions of microdata containing confidential information so that they are safe to be released to users for exploratory analysis".

This system uses classification and regression trees (CART), previously explained in chapter 2 to generate synthetic data from all data types. [57]

Despite being an R package in order to facilitate data generation, it was used the python package of synthpop, created by Hazy.[58]

3.3 Hardware

Hardware is a crucial part of training machine learning models. Despite every consumer hardware capable of training a model, it takes a long time to train a single model, even with a modest dataset size. For these reasons, it is necessary to use specialized hardware. Two of the most known are Tensor processing units (TPUs) and Graphical Processing Units (GPUs) with Compute Unified Device Architecture (CUDA) capabilities. The one used throughout this thesis was the latter, as it is easier to get access without the limiting factor that Google imposes over the use of their TPUs.

By the end of development, three computers were used to run machine learning models, each with different configurations listed in Table 3.5.

Table 3.5: Hardware specifications

	Computer 1	Computer 2	Computer 3
CPU	Ryzen 5 3600	Intel Core i5 4460	AMD Opteron(tm) Processor 6344
RAM	16GB DDR4	16GB DDR3	64GB DDR3
GPU	None	Nvidia Tesla k20x	Nvidia Tesla k40c

- **Computer 1** is my personal computer at home. Despite having a GPU, it is not CUDA enabled. Therefore, it was used at an early stage, when multiple models were tested to see if they fit the purpose of this work with a small dataset. It was also analyzed the level of complexity of model implementation. This is the most simple configuration with little to no extra configuration apart from python packages.

- **Computer 2** is a home server with parts from older configurations except for the GPU, which was bought refurbished as it is server-grade hardware that has been discontinued from data centers. Despite having CUDA compatibility, there was a significant setback when setting up this configuration. The problem originated from the CUDA architecture version of this GPU, version 3.5. Both of the models that can use CUDA use PyTorch as the machine learning framework, but PyTorch standard installation only has the binaries compiled for CUDA versions above 3.5, giving the error *RuntimeError: CUDA error: no kernel image is available for execution on the device.* The solution provided by the community on the PyTorch GitHub was to compile the binaries locally, but after two consecutive tries to compile them, it was unsuccessful. The final solution to this problem

was provided by Nelson Liu. He builds new binaries for PyTorch that add back compatibility to older GPUs, such as the one on this build. [59]

- **Computer 3** is a server provided by *Departamento de Ciência de Computadores da Faculdade de Ciências da Universidade do Porto (DCC-FCUP)*, and it was used later in the development as it was difficult to access it. Despite having a more powerful and recent card, it shared the same problem as Computer 2 regarding the CUDA architecture compatibility with PyTorch. Having dealt with this problem previously, it was easy to fix it from the start. Since it is a more powerful GPU and with a lot more RAM, it was easier to train models even though there still were compromises to be made, which are stated later in Chapter 4.

3.3.1 Microsoft Azure

Another tool used to train some models with CUDA capability, when no other computer was available, was the Microsoft Azure service. This was possible because, it provides 100$ to students monthly, which proved to be not that much since training the models requires a GPU, and the price per hour of using a GPU is relatively high, fairing at 0.9$ per hour on the *pay as you go* model using the NC6 configuration which is presented in Table 3.6 . As a consequence the 100$ were used after only the first week of usage, making it extremely difficult to do everything on this platform.

Table 3.6: NC6 Microsoft Azure Configuration

CPU	6 core unspecified CPU
RAM	56GB RAM
GPU	NVIDIA Tesla K80

It has to be noted that the setup process is easy and it went with no setbacks as all the hardware is recent and all drivers come already installed, which allows for easy installation of all python packages and everything works seamlessly.

3.4 Data quality analysis

Getting good quality data is already complicated when comes to actual data. Adding the layer of generating tabular data, which itself is one of the most complicated types of data to generate using machine learning models, is not ideal. In this case, the dataset used is already processed, and it is known to have good quality, so that is where the fundamental analysis of the generated data will lay upon.

Testing if the data quality retains quality from the original dataset, is done by performing several metrics on synthetic data. These metrics were chosen after reading state of the art and

what regular metrics were used, as well as the provided tools by platforms such as SDV, which already has some useful recommended metrics.

3.4.1 Count of variables

A common starting point to analyze a dataset is to count the number of unique variables it presents. A good and simple metric to check if the dataset retains any utility from the original one, is the amount of unique values each one contains, this shows if there are any disparities from one another and if the model does not create any new fictional value.

$$Count(x) = \sum_{i=1}^{n} [s_i = x], \tag{3.1}$$

Where x is a variable from the dataset and s_i is the i-th entry of the same dataset, [...] are the Iverson brackets where $[s_i = x]$ has as an output 1 if true and 0 if false.

3.4.2 Mutual Information

Mutual information measures the amount of information obtainable from one random variable given another. [60]

This information regarding these two variables, namely X and Y can be formally obtained by the following

$$I(X;Y) \equiv H(X) - H(X|Y), \tag{3.2}$$

Where I(X;Y) is the mutual information between X and Y, H(X) is the amount of entropy of X and H(X|Y) is the conditional entropy for X given Y.

3.4.3 Goodness of fit

Goodness of fit describes how well a statistical model fits a set of observations, which determines any discrepancies between observed and expected values, fitting perfectly with the model of original and synthetic data we have. Despite existing many tests that can be used to test the goodness of fit, the preferred ones were Kolmogorov-Smirnov (KS) test and the Chi-Square (CS) test. Both these methods are provided by SDV package, making it easier to implement as it is expected to use the generated data from their models.

3.4.3.1 Kolmogorov–Smirnov test

Kolmogorov-Smirnov test, most specifically the two-sample test, is used to decide if a sample comes from a population with a specific distribution, using the empirical cumulative distribution function. This can be used with univariate and continuous data, which is the type of data we have from the original and synthetic datasets. The KS test is defined as,

$$D^+ = \max_i(i/n - F(x_{i:n})) \tag{3.3}$$

$$D^- = \min_i(F(x_{i:n}) - (i-1)/n) \tag{3.4}$$

$$D = \max(D^+, D^-) \tag{3.5}$$

Where $x_{i:n}$ denotes the i^{th} order statistic of the random sample and $F(x_{i:n}) = P\{X < i\}$ for the distribution that is being fit.

3.4.3.2 Chi-Squared test

Chi-Squared test, also known as χ^2, is used to test if a sample of data came from a population with a specific distribution.[61] In the context of our work, it is used to compare two discrete variable columns. A common for to represent is as follows,

$$\chi^2 = \sum_{k=1}^{n} \frac{(O_k - E_k)^2}{E_k} \tag{3.6}$$

3.4.4 Jaccard similarity coefficient

Jaccard similarity coefficient also known as Jaccard index is a metric commonly used to assess the similarity and diversity between datasets. The measuring is based on a percentage score, with 0% having no relation between the two and 100% being exactly the same.[62]

Calculating the index is defined as the size of the intersection then divided by the size of the union of both datasets as shown on formula 3.7. [63]

$$Jaccard(U, V) = \frac{|U \cap V|}{|U \cup V|} \tag{3.7}$$

3.4.5 Likelihood Metrics

As a last metric, we used likelihood metrics, which by definition tries to fit the real data using a probabilistic model and shows the joint probability between that result and the synthetic data. [64]

SDV provides three of these type of metrics (GMLogLikelihood, BNLikelihood, BNLogLikelihood), but only two were used, BNLikelihood and BNLogLikelihood.

Where in both cases the probabilistic model is a Baysean Network where it is used to extract posterior probability of each case, and this is the category BNLikelihood falls into in likelihood metrics . [65]

BNLogLikelihood is a logarithmic transformation of the original likelihood function, that in this case is the BNLikelihood, despite not adding new information, the literature assures that the log-likelihood functions is important to cement the results obtained from the likelihood functions, since it assures that each data point is used by being added to the total log-likelihood function. [66]

3.5 Privacy Assessment

Having the results from the data quality analysis, our final step is to understand how synthetic data, generated from the chosen models, fares regarding privacy concerns. Understanding what methods to use and what tools are available is crucial to approach this problem. In this section we present what methods have been used and the thought behind these metrics.

3.5.1 Duplicates

At first glance, the most obvious privacy metric to look at is to assess to what extent synthetic data is equal to real data. This is a major concern that even if most of the dataset is synthetic, it is possible to come across real entries, knowing how compressible that data is to the end-user is of utmost importance.

To understand how much of the real data was transferred to the synthetic data, it was not only a matter of understanding if there was a procedure or a diagnosis that was attributed to a single patient from that dataset. The threshold chosen to define if data was duplicate was if the patient, represented by the SUBJECT_ID column, had all of the same attributes from the original dataset. Only if that condition was met, the subject was considered a duplicate and therefore was at a greater risk of being disclosed.

Of course, this metric is not representative of the whole picture. It does not provide the context of how many identical subjects exist in the dataset, diminishing the risk of identification. It also does not consider if, despite not being exactly identical, it would only be missing one entry, making it also easy for an attacker to make an exemption of a possibility to identify that person.

3.5.2 Uniqueness

In addition to discovering the duplicate records, it was also necessary to understand how unique the records are, since even if there are records that are exactly the same, if they are not unique, it can impact how private they are. The Special Unique Detection Algorithm (SUDA) is based on the search for special uniques. This algorithm consists of two tasks; At first, it searches for all unique across all records. Secondly, the size and distribution of uniques within each record are used to make statistical inferences to determine the 'risk' for each individual record. The performed search uses only unique attribute sets without any unique subsets, also referred to as Minimal Sample Unique (MSUs).

Finding all minimal uniques up to a sample size (M) is what it is used to determine the score afterwards, and is denoted as, $\sum_{i=1}^{M} \binom{ATT}{i}$, where ATT stands for the total numbers of attributes in the dataset.

After detecting MSUs, the size of each one determines the 'risk, where the smaller the size in a record, the more 'risky' it is. Furthermore, a greater number of MSUs in a record presents a greater chance to be 'risky'.[67]

3.5.3 SDV metrics

SDV provides extensive metrics on privacy evaluation on synthetic data generated, and their website does not provide an extensive explanation over the provided metrics. To encounter any extra information is necessary to explore their source code and rely on comments made by the programmers. Only then it was possible to search for any published articles that would explain these metrics in depth.

The metrics are separated by categorical and numerical.

Categorical

- Generalized Correct Attribution Probability (GCAP)

  M. Elliot introduced the base concept of Correct Attribution Probability which measures the disclosure risk of the individual's real target value when the adversary has access to the synthetic data.[68] There is a CAP score for both original (fuction 3.8) and synthetic (function 3.10) datasets, represented by the following equations:

$$CAP_{o,j} := P_o(T_{o,j}|K_{o,j}) = \frac{\sum_{i=1}^{n}[T_{o,i} = T_{o,j} \wedge K_{o,i} = K_{o,j}]}{\sum_{i=1}^{n}[K_{o,i} = K_{o,j}]} \tag{3.8}$$

  Where $K_{o,j}$ is the representation of the original dataset values' key attributes, at a j-th position, and $T_{o,j}$ is the value to that attribute.

$$CAP_{s,j} := P_s(T_{o,j}|K_{o,j}) = \frac{\sum_{i=1}^{n}[T_{s,i} = T_{o,j} \wedge K_{s,i} = K_{o,j}]}{\sum_{i=1}^{n}[K_{s,i} = K_{o,j}]} \tag{3.9}$$

Where the previous representations remain the same and there are similar to the synthetic data set described as $T_{s,i}$ and $K_{s,i}$.

This representation shows that while using CAP, the attacker tries to search all synthetic records and match the key attribute values. The final value of this metric is defined by computing the mean over all the dataset records.[69]

Generalized Correct Attribution Probability improves upon this definition by treating the non-matches differently, by an approach similar to the Fixed-Radius Nearest Neighbor, tackling the problem referenced over at duplicate records.[70] Due to the nature of the data, the authors chose the Hamming Distance to use for this case in specific.

Hamming Distance definition: Given two vectors u,v $\in$F^n the distance between the vectors u and v, is defined by the number of places where u and v differ along the vector.

Now that we understand what changes are behind GCAP we can define it:

$$GCAP_{s,j} := \frac{\sum_{i=1}^{n}[T_{s,i} = T_{o,j} \wedge \Delta(K_{s,i}, K_{o,j} = \rho) = K_{o,j}]}{\sum_{i=1}^{n}[\Delta(K_{s,i}, K_{o,j} = \rho]}, \tag{3.10}$$

Where Δ represents the Hamming Distance and $\rho := min\{r \| \exists i \in \{1, ..., n\} : \Delta(K_{s,i}, K_{o,j} = r\}$ which tends to $\rho = 0$

- k Nearest Neighbour (kNN) attack, exploits the fact that similar users share the same rating on corresponding items that could potentially reveal user's private data.

  The attacker creates k synthetic users' data, and creates each users, with the m items which he knows to be present in the target user U 's previous records. He then inspects the list of items recommended by the system to any of the synthetic data. Any record which appears on the list and is not one of the m items from the synthetic users artificial history must be an record that U has. Any such record was not previously known to the attacker and learning about it constitutes a privacy breach. [71]

- Categorical Naive Baysian privacy metric uses a naive Bayesian classifier to calculate the score based on prediction accuracy.

  A Bayes classifier uses the Bayes probability model combined with a decision rule, commonly used is the maximum a posteriori decision rule. The function assigning a class label $\hat{y} = C_k$, where C_k is class, is the following:

$$\hat{y} = \underset{k \in \{1,...,K\}}{\arg\max}\ p(C_k) \prod_{i=1}^{n} p(x_i | C_k) \tag{3.11}$$

- Ensemble model is the last model used, this model retains three major variations, but the one used by SDV is the majority vote. This technique uses multiple models to make predictions on the results for each data entry of the synthetic dataset, which we will call a vote. Each prediction of each model is then taken into consideration and the one retaining the majority of votes is used as a final prediction.

After having all predictions the final score is calculated by once again using the prediction accuracy of the predictions.

Numerical

- Multilayer perceptrons regression model is based on at least three layers of nodes: an input layer, a hidden layer and the output layer an input vector. Each of these layers are composed of neurons (nodes) with each having a weight attributed to them. We then have an activation method (3.12) which is how the linear function mapping the weighted inputs to the output of the neuron; the specification of the learning method, which can be changed from the way neurons are arranged based on the weights, this is done by doing the following: $\epsilon(n) = \frac{1}{2}\sum_j e_j^2(n)$ and finally specification of events.

$$\frac{1}{1+e^{-x_i}} \tag{3.12}$$

  This fits our attacker problem, by fitting the synthetic data in order to try to predict the original records.

- Support vector regression (SVR) is a regression model, and its fundamentals are based on Support Vector Machines and Linear Regression. But unlike linear regression where it tries to minimize the l2-norm of the coefficient vector (w) as shown on 3.13, SVR does this and uses the support vectors technique behind SVMs, for a given maximum error defined as ϵ, as shown on 3.14. [72] This fits our attacker problem, by using the synthetic data in order to try to predict the original records.

$$MIN\frac{1}{2}||w||^2 \tag{3.13}$$

$$|y_i - w_i x_i| \leq \epsilon \tag{3.14}$$

- According to the SDV comments on the source code for Numerical Radius Nearest Neighbor Attacker: "The Radius Nearest Neighbor will predict the sensitive value to be a weighted mean of the entries in the synthetic table. Where this weight is given by a separate function, and typically describes the closeness between the given key and the corresponding entry in the table." The described function of the weights, is named as the *InverseCDFCutoff* (CDF standing for cumulative distribution function).

$$\frac{\sum_{i=1}^{n}(c_i(k_i) - c_i(k_{i'}))^p}{n} \leq f^p \tag{3.15}$$

  Where c_i are the cumulative distribution of each entry, k_i is the key (sensitive data argument passed on the function) and k_i' is the reference key (synthetic data of the respective key), f is the cutoff as a default value of 0.1 and p is a constant value of 2. If and only if this condition is met the weight is equal to 1.

3.5.4 Differential Privacy

Differential privacy was already explained on chapter 3, here we introduce what was done around this topic.

As a first approach there was no clear way to use this topic in a more practical way, as there was confusion on what differential privacy truly represented. After some analysis it was discovered that there were tools that could be used in order to find any kind of relation with the data we gathered.

From all of the discovered tools the ones which stood out the most were; OpenMinded project denominated PyDP (v1.1.1 at the time of writing) which in short stands for Python Differential Privacy;[73] IBM's diffprivlib (v0.4 at the time of writing) and finally OpenPD (v0.2.1 at the time of writing). All these had their pros and cons and ultimately the one chosen was the PyDP as it was the one with the best documentation, ease of use and most important it had the tools which were better fit for what we were trying to achieve.

Here we used the library, and an already existing example, provided at the library which did already fit the kind of result we were trying to achieve. This consisted in using the original dataset to create a scatter plot of the number of the count of each variable of the table using ϵ-differentially algorithms, this produces an aggregate statistic over this dataset.

Chapter 4

Experiments and testing

Chapter 4 will be used to explain the process behind the end decisions and some failed directions that were taken before finding the final solutions. Ultimately all solutions that end up being used to represent results are described in chapter 3.

4.1 Data processing

MIMIC-III is an well established dataset as previously stated in chapter 3, despite being a great quality dataset, there was the need to process it further, namely removing any NA values and dropping any columns that would not add significant value to the synthetic data generation as it would add complexity to the process. The criteria to remove these columns was based on the data they represented which was mostly related to temporal data. After preliminary tests, it showed that these models would not deal with the purged data very well. However, for each table, we will go through what the processing decisions were. For any and every data processing method it was used python 3.9 and the pandas package as it is the most common data analysis and manipulation tool.

- DIAGNOSES_ICD

  The first processing done was to remove any NA values from the table, as some models would have a tendency to learn from them, this was discovered at an early stage, as the first iterations were made with this table and it was assumed MIMIC III would not have any NA values. This was immediately fixed and all models behaved as expected from that point forward. The second step of data processing was to discard the columns that were thought to be not as necessary, as it would increase difficulty to train the models and the time to fine tune them was not enough in this context. The reasoning behind this purging was that the removed fields were associated with time, since these models are not prepared to deal with this type of data, hence came the decision to remove them.Here the purged options were HADM_ID and SEQ_NUM, these were common problematic in early

tests and in retrospective the information gained from them is fairly limited as it has no associated date, at least no on this table, so it retained no temporal table.

- PROCEDURES_ICD

  This table went through the same changes as DIAGNOSES_ICD, namely the removal of any NA values as well as dropping the HADM_ID and SEQ_NUM columns. Reasoning behind these decisions were the same as well as to keep consistency throughout the training.

- PRESCRIPTIONS

  PRESCRIPTIONS tables were different from the previous ones. As shown in chapter 3it retains substantially more variables than DIAGNOSES_ICD and PROCEDURES_ICD, therefore it needed a more extensive analysis on what columns should be dropped if any. After checking every column, it was observed that it had redundant tables, meaning that it was the same information but described differently.

  'STARTDATE', 'ENDDATE', 'HADM_ID' and 'ICUSTAY_ID' were removed for the reason of being all temporal variables. This would lead to bad generated data as the models chosen to generate data are not fit to take into consideration this data type. 'DRUG_NAME_POE' is either the same name as in column DRUG or an empty cell, 'DRUG_NAME_GENERIC' is also another iteration of the DRUG column or an NA value. 'FORMULARY_DRUG_CD', 'GSN', 'NDC', are all codes for the drug stated once again by the DRUG column, as it occurred with the previously mentioned columns, they also had occasional NA values. 'PROD_STRENGTH' is the last one removed. This column simply is a human understandable description of the combination of descriptors in columns DOSE_VAL_RX, DOSE_UNIT_RX, FORM_VAL_DISP, FORM_UNIT_DISP and ROUTE.

Our last step in data processing was a difficult decision to make since it directly impacts how models are trained. Since the hardware access was limited, as previously described in chapter 3, there was the need to make a trade-off on what data would be used. The first draft of possibilities was to train only using the amount of data that each model could handle of every table. This was the state of the work throughout a good portion of the thesis, as it was the most obvious take over this subject. After discussion with colleagues, there were a few consensuses over the matter and how it could be improved upon. As a first solution, it was found that the models were not retaining correct patients' profiles, so it was decided to sort the dataset by SUBJECT_ID. This would make sure that an user would have all its information on the dataset, not regarding the last subject on that table, where it could split information. Despite this method displaying more consistent results, it still was not optimal as it left out the majority of information in some cases. As an end result, there was an iteration over this method, the dataset would still be sorted by SUBJECT_ID but the whole table would be used. This was accomplished by splitting the dataset n times and training a model with all partitions iteratively. This was known that would influence the results in the end, but in the end it was the otimal solution given the circumnstances,

As a final method training models became extremely time consuming since we had to train a model from 4, up to 10 times for a single dataset depending on the partition number. PRESCRIPTIONS table was the most complicated one as it had a greater number of columns, thus consuming more resources per training session.

4.2 Synthetic data creation

For the synthetic data creation methods, all used models have been previously introduced on chapter 3. Each model had a dedicated python file, with three functions, one for each table, and in order to select the table it is needed to pass an hyperparameter, either diag for the DIAGNOSES_ICD, proc for the PROCEDURES_ICD and pres for PRESCRIPTIONS. This organization was crucial to improve the records of what has been done, and to save time in changing files if any modification was necessary.

Despite this, every model, apart from synthpop which comes from a different package, is organized the same. First, there is the initialization of the model class by calling the respective SDV function, namely CopulaGAN(), GaussianCopula(), CTGAN() and TVAE(), the only hyper-parameter used here was 'cuda=TRUE' in both CTGAN and TVAE, guaranteeing that it would force the use of any available GPU. This, in theory, should be unnecessary as it is the default value, but this way was easier to identify which ones used CUDA acceleration when training.

The lack of use of hyperparameters is backed by that the initial idea that for the future, this would be to implemented to get as input a wide spectrum of datasets, making it difficult to manually fine tune any hyper-parameter by hand. For that we wanted to test how well each model would fare, when every default value is not tampered with. In hindsight it would be preferable to fine tune each model for every table, and in the context of the work it would be possible, but ultimately was decided to leave it for any future improvements.

After the model instantiation, there are two steps that follow to create any data, which is fitting the data and sampling it. This is done by using the model class previously created and in order to fit the data, it is called the fit() method, which has as parameter a pandas dataframe, and sampling is done by calling the sample() method and has an integer as parameter, representing the number of data entries generated, which in our case is always the length of the dataset representing the training data. As an example of what has been described, we have listing 4.1

```
import pandas as pd
from sdv.tabular import ModelName

model = ModelName()
model.fit(dataframe)
synthetic_dataframe = model.sample(len(dataframe.index))
```

Listing 4.1: Example of a model instantiation where ModelName is referent to any model from the SDV library and dataframe is a pandas dataframe with the content of any previously mentioned dataset.

In line with what has been stated in section 4.1, not every time the whole dataset is not used at once, thus every dataframe passed as a parameter to the fit and its corresponding length is done under a loop cycle iterating over chunks of data which are represented at table 4.1, and at the end, appending the result to a final file. All resulting tables are saved under an output folder and the names are structured as a prefix 'mimic_output' followed by the name of what dataset has been used it is and finally what model created its results.

Table 4.1: Number of partitions of each dataset in regards to the corresponding model.

Model	Number of partitions		
	PRESCRIPTIONS	DIAGNOSES_ICD	PROCEDURES_ICD
CopulaGAN	10	1	1
CTGAN	10	1	1
TVAE	10	1	2
GaussianCopula	10	4	2
Synthpop	1	1	1

At the end of every iteration, there were still some problems at first, meaning that when training the second partition of data, it would run out of RAM where to put the new data. This problem was caused by the program not freeing up memory after each loop invariant. To solve it, we used Python's garbage collection interface and ran gc.collect() at the end of training. This frees the memory and makes it possible to allocate new data to it. These trained models take a considerable time to train, ranging from being capable to train a small dataset in under 1 hour to take more than a week. The most efficient one, in terms of time taken to train models, is Synthpop, as well as the resources used. This is observable in table 4.1 that it is the only model that does not need partitioning on any of the datasets and is the fastest one to train overall. Looking at the other side of the spectrum, there is the Gaussian Copula which is the least effective in terms of time, it does not have any CUDA acceleration and uses the CPU to train data, as well as being the worst at handling resources having the need to partition every dataset.

As previously explained, all final versions of the synthetic data were generated in an unreachable server. Due to the non-persistent connections of Secure Shell Connections (ssh), installing tmux is a way to keep a persistent terminal running the model in the background, even when the connection was dropped. Tmux is a terminal multiplexer with several functions, but the most important one to note here is the ability to keep any terminal running in the background even when the connection is closed with the host.

At the end of this whole process, we get synthetic data generated by selected models, and it

is possible to go to the next step, which is to evaluate how these models behaved and how good is data generated by them.

4.3 Data evaluation

Having synthetic data was only part of the work, this was the setup for the remaining of the thesis. Before assessing the privacy of the generated data, it was important to identify if any generated data was useful at all.

As the first alternative, since both PROCEDURES and DIAGNOSES could be seen as a network, where nodes are patients and the ICD code for the respective tables and each has the respective connection, it can be seen as a bipartite graph. The idea of a graph is also possible to adopt when it comes to the PRESCRIPTIONS table, but the graph becomes more complicated and is an undirected graph with multiple types of nodes. The sheer volume of data brings some difficulties, namely, constructing the graph itself is computationally intensive, and any metrics performed on these graphs suffer from the same problem. The idea behind using graphs was to try to create a metric that could both serve to measure data utility as well as privacy. This was eventually dropped, following a professor's advice, as it added a layer of complexity to the work that was not possible with the time constraints at hand. Despite being unfeasible, the graphs were constructed at an early stage and provided the first look of data visualization of the synthetic data. This could provide a different approach that the literature usually uses while giving a deeper understanding of the privacy outlook.

Now, there was the need to also use regular metrics found on literature. There are a plethora of options over the literature to determine how good quality is. After an extensive search, the most obvious options, as well as the most re-current, were using goodness of fit metrics, namely Kolmogorov–Smirnov test as well as the Chi-Squared test. These metrics require the dataset to be the same length adding to another motive to have the same generated data amount as the original data. This was discovered when doing preliminary tests, where the final implementation of the models split into partitions was still not in place.

SDV already provides python functions to these two goodness of fit metrics, and it is easy to adopt it to the generated data, as shown in listing 4.2.

```
from sdv.metrics.tabular import CSTest, KSTest

print(CSTest.compute(original_dataset, synthetic_dataset))
print(KSTest.compute(original_dataset, synthetic_dataset))
```

Listing 4.2: SDV functions of goodness of fit metrics

In the end, the KS test is not presented in the results as it gave NaN values for everything.

Overall there are a significant number of metrics, but only one more was chosen as it was seen

that it could provide greater context, the Jaccard index. This compares members for two tables and determines what members are shared and which are distinct as described on sub-chapter 3.4.4. This metric eventually dropped as it suffered from a common problem throughout this work which was that it did not work properly with our data.

After further consideration, we felt that we needed a visualization method as it improves the perception of the results in a more convenient way. The first obvious form to do this was a count of individual variables and create a bar plot using matplotlib, this produced an impossible to understand plot.

To fix this visualization problem, the data had to be normalized, and the bar plot was changed to line plot. As a result, all data was more perceivable. These changes were still not good enough because the plot did not represent the data accurately. A definitive solution to represent the data was to use a scatter plot, which is used to visualize great amounts of data. The results following this change are shown on chapter 5.

While doing tests, it has been observed that the models did not retain all information because there were Nan values after doing the count of variables and having them compared. Following this discovery, it was decided to include how many values were left out and if the model created any new values.

Over at the privacy tests, it was possible to observe that these models did not create any non-existing values since some metrics required that the synthetic values were the same as the original one.

4.4 Privacy

Last but not least, we have our privacy concerns. This is the pivotal analysis of the data and the final indicator of whether this data can or not be used by companies or researchers for any kind of task.

Coming to a conclusion on what should be used here was an arduous task since the literature, despite existing, is obscure and not very well documented. Adding to that, most procedures are computationally intensive, and with the limitation in hardware that was experienced already on generating the datasets, it was translated when evaluating data.

At first, there was an inclination to use traditional attacks on the dataset, such as inference attacks, record linkage and using statistical disclosure methods. The main driver to at first use inference attacks and record linkage was because Stadeler et al [74] stated that they "demonstrate that synthetic data does not provide robust, trade-off free, protection against attribute inference and linkage attacks". Despite this bold claim, it is still uncertain that it can be proven only by their sample size of trained models as well as the reduced number of datasets used. It still makes important points regardless and it still should be taken into consideration by future work in this area.

Their method was not used in the final experiments, as the framework provided by the author had no documentation on how to use it and it required a specific treatment of data as well as creating JSON files to feed to the framework depending on what dataset is being used. This set-up was over-complicated to implement in our scenario, and it shows how complicated this matter can be to grasp and evaluate.

From SDV metrics, we had to prune all of the metrics explained in chapter 3, namely all numeric metrics, since all datasets have categorical values, it did not make it possible to use any numerical metric. Furthermore, due to computational constraints, all metrics except the Ensamble were taking too long to compute.

CategoricalCAP tried to compute for seven days for the smaller datasets and was still not finished. KNN was taking around a week to finish with the smaller datasets and the results were corrupted with NaN values, and since it was so unstable, we decided to cut them out. Rendering these metrics impractical for this thesis due to the time constraints as well as their reliability. CategoricalNB had no sufficient RAM to compute at all, and since these metrics needed to be computed using the whole dataset, or else they would accuse a warning of unexpected value. After troubleshooting it was discovered that it was because the synthetic dataset had values that the original did not. That is why the original strategy of dividing the datasets did not work when evaluating them. Leaving only the Ensamble metric in the end, that despite computing and giving no warnings or errors, it computed NaN values to all tables and models, leaving us without any options to use from the SDV.

Finally, when it comes to differential privacy, it was a smooth process after it was understood what could be done regarding this field as well as finding the most renowned tools. There was no extensive experimentation here since the choice of the library was done, with the criteria in mind of having good documentation and its ease of use.

4.4.1 Compute

SDV has a last metric that was tried out to wrap up everything, despite this it was not possible to get any significant results within the time constrains, since this metric is a combination of all previous mentioned metrics and others that were not specified in this work. Due to that it was impossible to include it as a way to have an overall view on the performance of the models and how good the datasets were.

Chapter 5

Results

In this chapter we will present and comment on the results. This section will follow the same structure of the previous chapters, showing the results incrementally and in the same defined order.

All results were obtained by using the code preset in my github repository GitHub Repository. [75]

5.1 Synthetic data creation

The process of creating synthetic data has been explained on chapter 3. This process generated, for each of the three tables, five new files, one for each used model, giving us a total of 15 synthetic datasets.

The output files consist of 651002 entries for the DIAGNOSES_ICD, 1472957 entries for PRESCRIPTIONS_ICD and finally 240096 entries for PROCEDURES_ICD.

Training these models varied greatly in terms of time taken. Training the PRESCRIPTIONS table was very time and resources consuming and this is why it had to be the one to be split into more partitions.

From all the trained models, the more efficient one, in terms of both resources and time, was Synthpop and the one who took the longest to train regardless of the table was Copula GAN. We have no precise metric, namely the time each model took to train, as this was an afterthought and it would take a significant amount of time to train every model again.

This is a brief section as the results of the models can be summarized as the created files, and it is only a briefing for what is going to be analyzed over at the next sections.

5.2 Data Evaluation

5.2.1 Metrics

Having all the resulting synthetic data, it came the time to perform analysis on that data.

The first step was using the SDV provided metrics, namely the CS Test, for each of the synthetic datasets as shown on table 5.1.

Table 5.1: CS Test on all synthetic datasets.

	CS Test		
	PRESCRIPTIONS	PROCEDURES	DIAGNOSES
TVAE	0.12487563285730306	nan	1
CopulaGAN	0.0	nan	2
CTGAN	0.28571428571428575	nan	1
Gaussian Copula	0.8572097246426187	nan	1
Synthpop	3.034739757978948e-85	nan	0

The CS test aims to try to understand if the two datasets were sampled from the same distribution. Looking at the results from both the PROCEDURES table and CopulaGAN, both tests give odd results. In the PROCEDURES, none of the values were possible to compute no matter what synthetic data it was used, and the data treatment that it went through always gave a NaN value as a result. In the CopulaGAN dataset, the CS test simply performed badly and could not compute plausible results. Despite being bad results, we decided to keep them in the results as it is useful to understand how unpredictable results from different models can be, and sometimes all the results can even be unusable.

Regardless of these outlying results, the remaining models were still able to provide data to analyze, with Synthpop having the best result by far with an extremely small p-value on PRESCRITPIONS and the DIAGNOSES being the only 0 across the other models. While Synthpop is the clear winner, both TVAE and CTGAN have comparable results and perform respectably by having a decent p-value, despite being far from what Synthpop accomplished, at least in comparison with Gaussian Copula that has a p-value close to 1, meaning that it is far from the distribution from the original tables.

Table 5.2: Likelihood metrics of all synthetic tables.

	BNLikelihood			BNLogLikelihood		
	PRESCRIPTIONS	PROCEDURES	DIAGNOSES	PRESCRIPTIONS	PROCEDURES	DIAGNOSES
TVAE	0,4792288562683456	nan	0.004190308510362174	-14,664978109347423	nan	-6.458216111743245
CopulaGAN	0,7966194092378945	nan	0.006965532275761499	-14,543736124097343	nan	-6.404823031178538
CTGAN	0.8590629451505191	nan	0.0075115366693330144	-14.474994160633274	nan	-6.374550389696791
Gaussian Copula	1	nan	0.015993815474715734	-10,047304098123823	nan	-4.424668158840339
Synthpop	0.00482387893771917 8	nan	0.004823878937719178	-15,463160091432692	nan	-6.809722311071295

Looking at the results from the likelihood metrics in table 5.2, the PROCEDURES table is still giving a NaN value no matter the data treatment PROCEDURES it went through. Other than that, PRESCRIPTIONS and DIAGNOSES behaved as expected, and when comparing both it is possible to see that when it comes to the likelihood metric the models did a decent job to mimic the distribution of the original. While the results are clearly better on the PRESCRIPTIONS table, as expected, since there is more data for all the models to learn from.

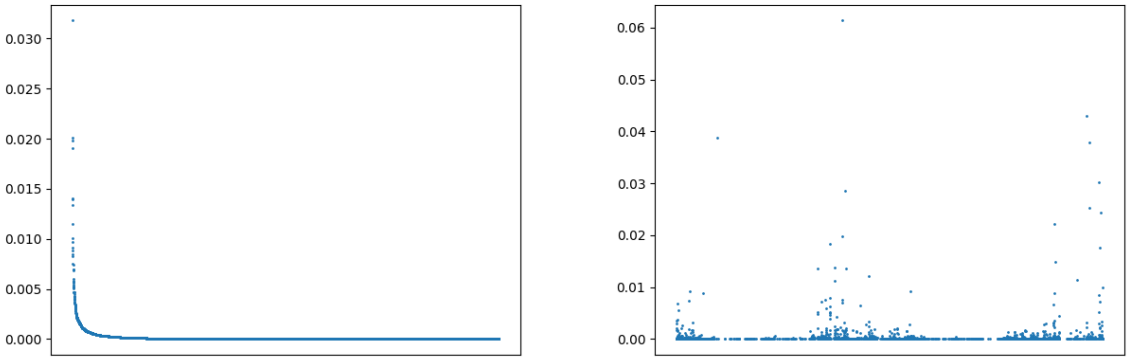
Individually, the Gaussian Copula keeps the trend of being the worst of the group, and here both the TVAE and Synthpop are closer together as the best performing models, TVAE actually being slightly better on the BNLikelihood. On BNLogLikelihood, Synthpop is once again back with the best result overall, while TVAE, CoulaGAN and CTGAN are close together.

Overall and considering the meaning of the likelihood metrics that here are used to understand the likelihood of the synthetic data belonging to the learned distribution. Synthpop is the one that overall is most likely to belong to the original learned distribution, all other models perform decently at this with the exception of Gaussian Copula which is consistently having bad results at every test.

5.2.2 Visual Representation

To help give the previously seen metrics, data visualization is a great tool to correlate information, for this we have scatter plots representing each of the original datasets, and as comparison the scatter plot from the synthetic data.

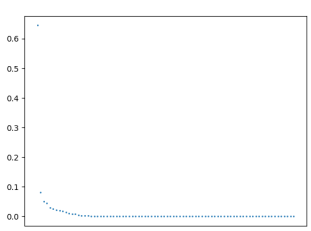
5.2.2.1 Original



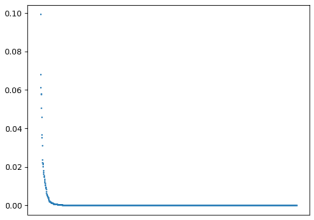
(a) Scatter plot of DIAGNOSES_icd

(b) Scatter plot of PROCEDURES_icd

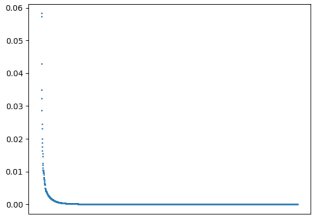
Figure 5.1: Plots are representing the normalized count of every variable of the respective table.



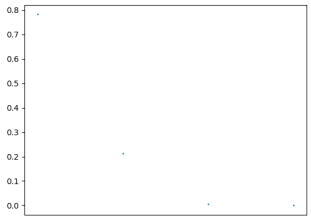
(a) Scatter plot of the PRESCRIPTIONS column DOSE_UNIT_RX



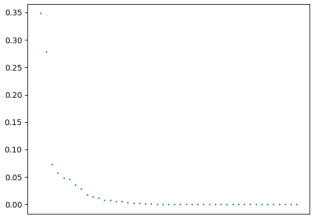
(b) Scatter plot of the PRESCRIPTIONS column DOSE_VAL_RX



(c) Scatter plot of the PRESCRIPTIONS column DRUG

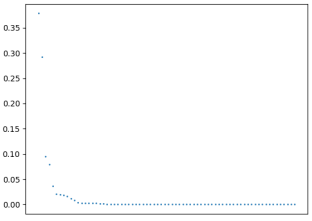


(d) Scatter plot of the PRESCRIPTIONS column DRUG_TYPE



(e) Scatter plot of the PRESCRIPTIONS column FORM_UNIT_DISP

(f) Scatter plot of the PRESCRIPTIONS column FORM_VAL_DISP



(g) Scatter plot of the PRESCRIPTIONS column ROUTE

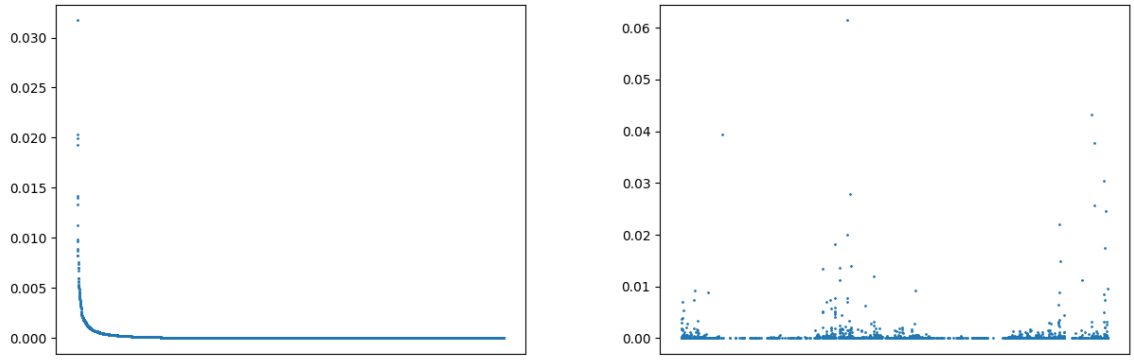
Figure 5.2: Plots are representing the normalized count of every variable of the PRESCRIPTIONS_ICD table.

Having these scatter plots from the original dataset on images 5.1 and 5.2 gives a baseline of what it should be expected when analyzing the synthetic data.

To facilitate the visualization, each model has the same structure as the original one, adding to all models are split int subsections and have a small description of what was observed, giving some context to the data.

5.2.2.2 Synthpop

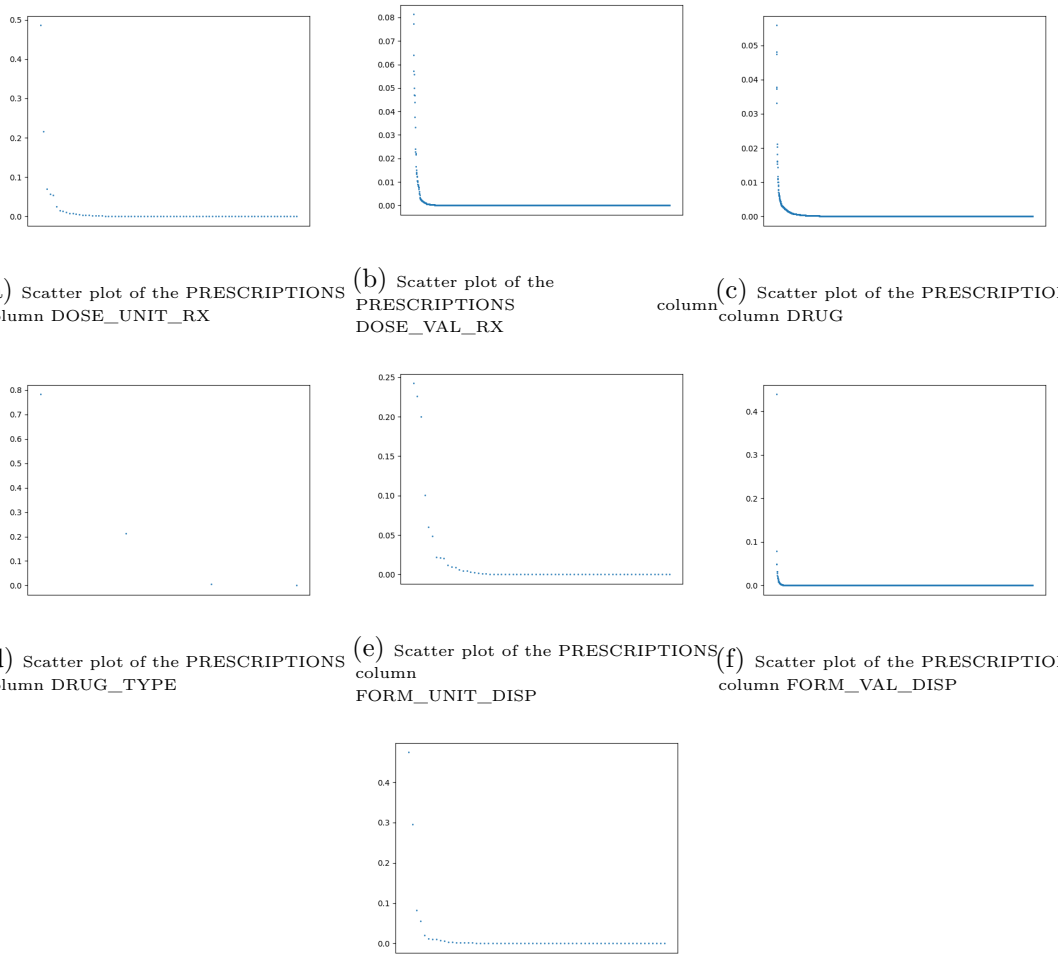
Here we have a visual representation of the scatter plot representing what synthetic data was generated from the Synthpop model. And we can clearly see a great resemblance when comparing the original plots, only slightly failing on the values that are in the curve of the distribution. It is the only model that did not have to use hardware acceleration, nor had to have the original dataset split to be able to train the model. It accomplished this while also being a reasonably fast model and consuming a few resources. Observing the plots, consolidates the knowledge, that Synthpop is a great model in terms of data quality.



(a) Scatter plot of DIAGNOSES

(b) Scatter plot of PROCEDURES

Figure 5.3: Plots are representing the normalized count of every variable of the respective table.



(a) Scatter plot of the PRESCRIPTIONS column DOSE_UNIT_RX

(b) Scatter plot of the PRESCRIPTIONS DOSE_VAL_RX column

(c) Scatter plot of the PRESCRIPTIONS column DRUG

(d) Scatter plot of the PRESCRIPTIONS column DRUG_TYPE

(e) Scatter plot of the PRESCRIPTIONS column FORM_UNIT_DISP

(f) Scatter plot of the PRESCRIPTIONS column FORM_VAL_DISP

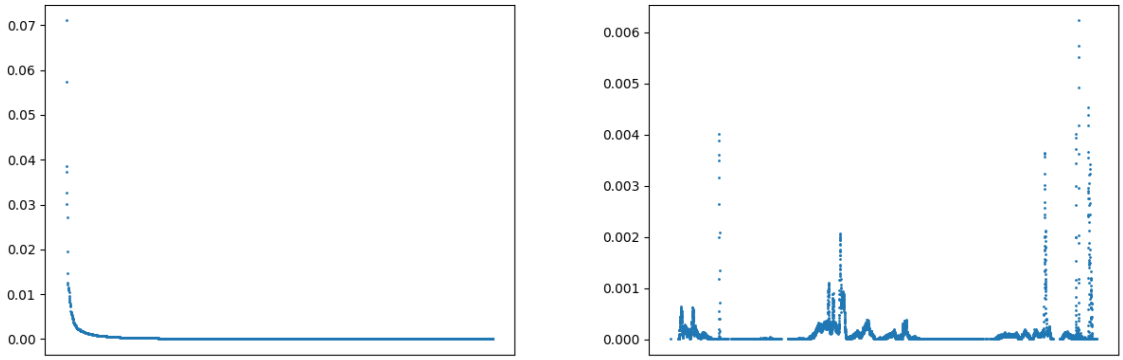
(g) Scatter plot of the PRESCRIPTIONS column ROUTE

Figure 5.4: Plots are representing the normalized count of every variable of the PRESCRIPTIONS_ICD table.

5.2.2.3 TVAE

Looking at TVAE plots we can observe mostly the same from Synthpop with an exception on the PRECDURES table, where there is a clear difference. Although the format of the scatter stays relatively the same most of the data had a tendency to cluster values, and it did not allow for some of the most loose points, although this does not depict the efficiency of the model, since what was observed on 5.2.1, this was a model that for the most part performed reasonably well and consistently.

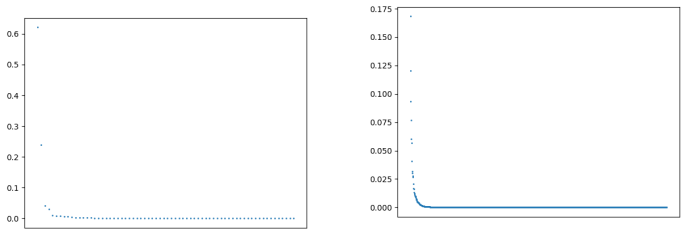
And when looking at results from the PRESCRIPTIONS it is a model that clearly takes advantage of greater number of data. And it adds up to the poor performance on the PROCEDURES data, since it was the dataset with the least amount of data to work with.



(a) Scatter plot of DIAGNOSES

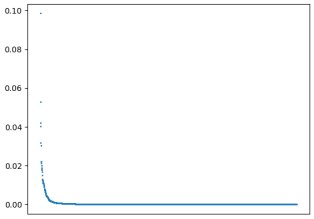
(b) Scatter plot of PROCEDURES

Figure 5.5: Plots are representing the normalized count of every variable of the respective table.

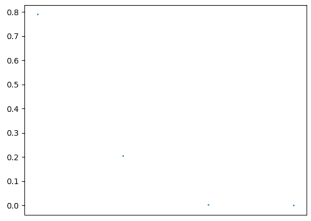


(a) Scatter plot of the PRESCRIPTIONS column DOSE_UNIT_RX

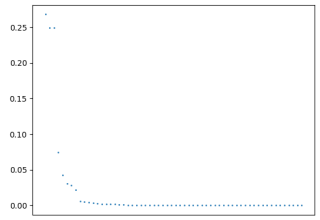
(b) Scatter plot of the PRESCRIPTIONS column DOSE_VAL_RX



(c) Scatter plot of the PRESCRIPTIONS column DRUG

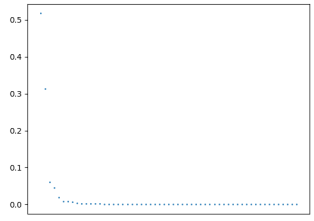


(d) Scatter plot of the PRESCRIPTIONS column DRUG_TYPE



(e) Scatter plot of the PRESCRIPTIONS column FORM_UNIT_DISP

(f) Scatter plot of the PRESCRIPTIONS column FORM_VAL_DISP

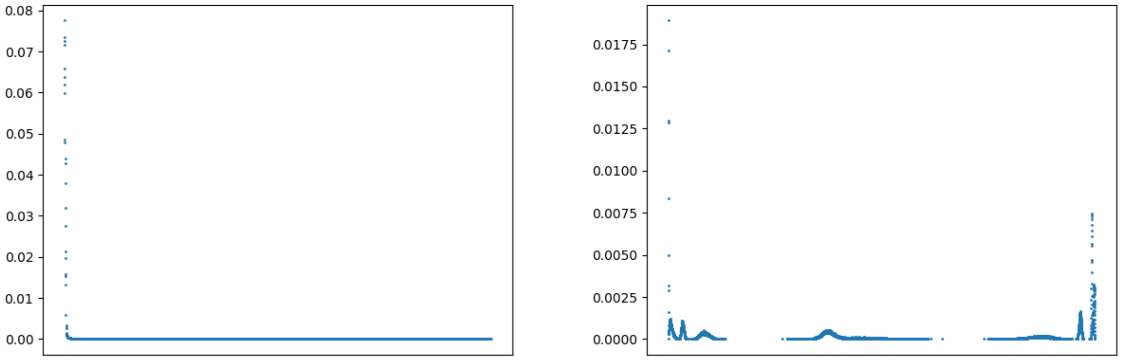


(g) Scatter plot of the PRESCRIPTIONS column ROUTE

Figure 5.6: Plots are representing the normalized count of every variable of the PRESCRIPTIONS_ICD table.

5.2.2.4 CopulaGAN

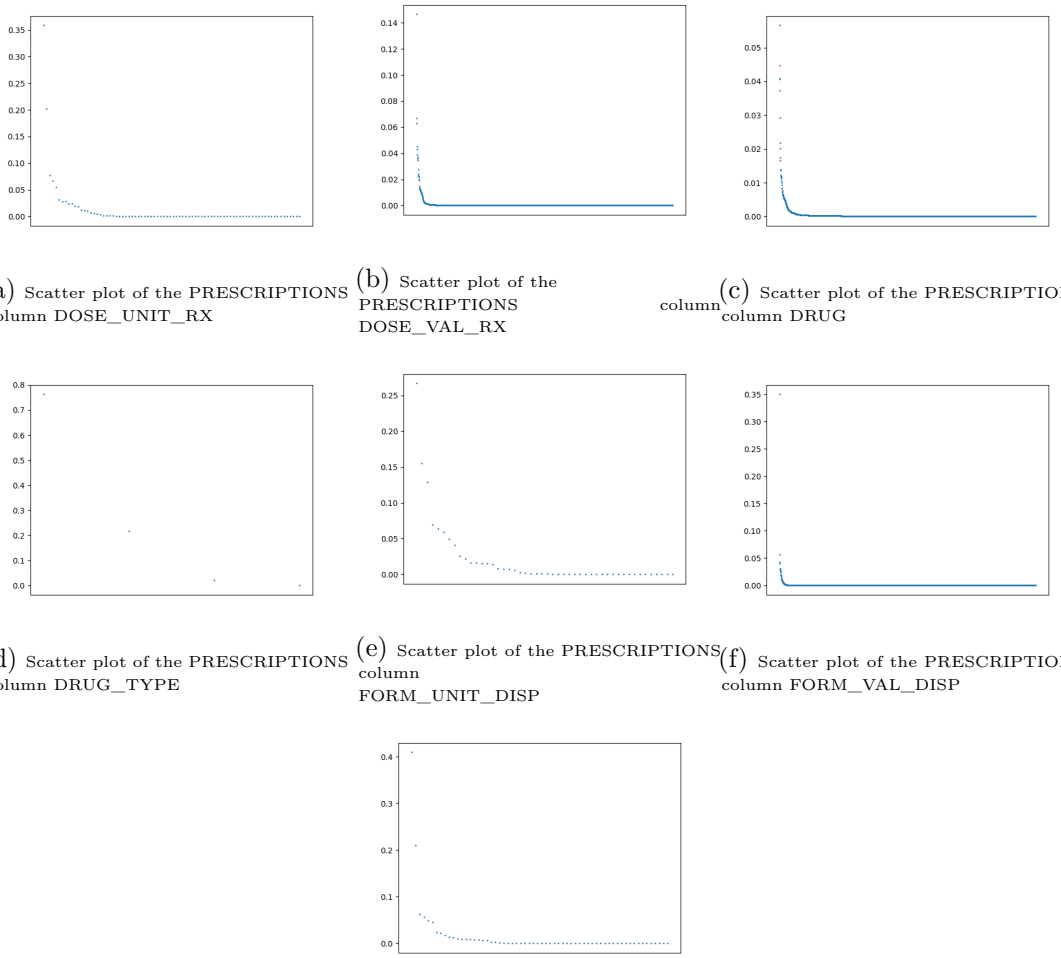
CopulaGAN keeps the bad results it had in 5.2.1, also struggles with less data, and these poor results might be because it actually needs even more data than the remaining values. This can be concluded because it is missing values from PROCEDURES which is a sign that it did not had enough data to fit the dataset. These missing values could also explain the odd values in the metrics, since it could not find all values from the original dataset and compare them to anything.



(a) Scatter plot of DIAGNOSES

(b) Scatter plot of PROCEDURES

Figure 5.7: Plots are representing the normalized count of every variable of the respective table.



(a) Scatter plot of the PRESCRIPTIONS column DOSE_UNIT_RX

(b) Scatter plot of the PRESCRIPTIONS column DOSE_VAL_RX

(c) Scatter plot of the PRESCRIPTIONS column DRUG

(d) Scatter plot of the PRESCRIPTIONS column DRUG_TYPE

(e) Scatter plot of the PRESCRIPTIONS column FORM_UNIT_DISP

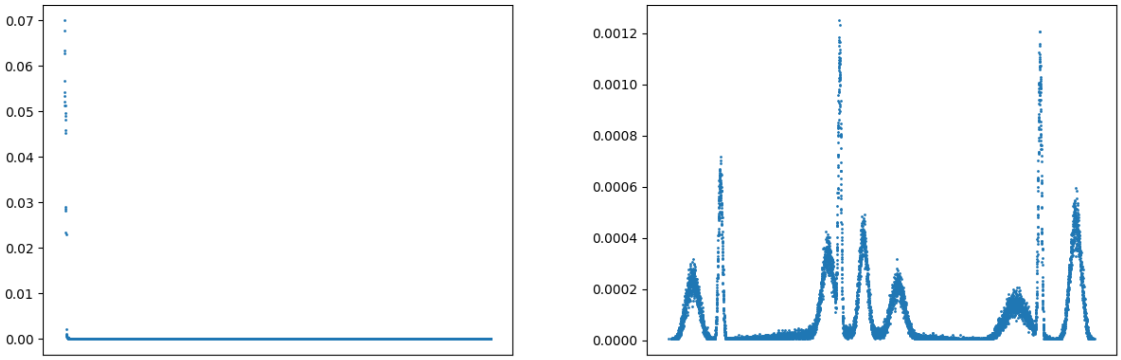
(f) Scatter plot of the PRESCRIPTIONS column FORM_VAL_DISP

(g) Scatter plot of the PRESCRIPTIONS column ROUTE

Figure 5.8: Plots are representing the normalized count of every variable of the PRESCRIPTIONS_ICD table.

5.2.2.5 CTGAN

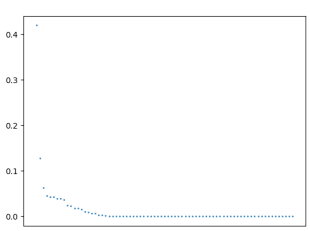
CTGAN shows a great difference across the board when comparing the plots to the metrics. It should be noted that those results can be inflated by the PRESCRIPTIONS results, since both the DIAGNOSES and PROCEDURES tables have tremendously different results from the original dataset. Even the PRESCRIPTIONS does not have outstanding results, missing two values on the DRUG_TYPE column, having only two of the 4 of the original dataset. Once again by the looks of the plots, it is also a model that could probably benefit from more data.



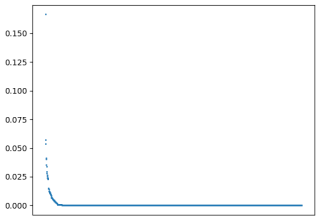
(a) Scatter plot of DIAGNOSES

(b) Scatter plot of PROCEDURES

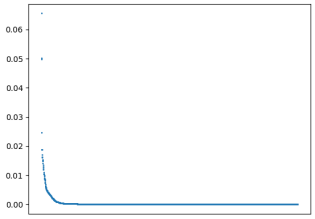
Figure 5.9: Plots are representing the normalized count of every variable of the respective table.



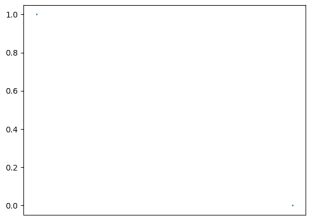
(a) Scatter plot of the PRESCRIPTIONS column DOSE_UNIT_RX



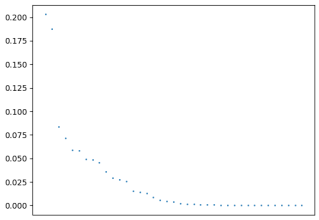
(b) Scatter plot of the PRESCRIPTIONS column DOSE_VAL_RX



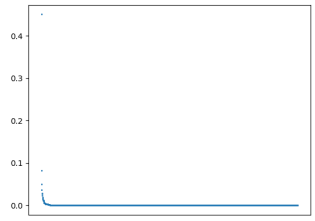
(c) Scatter plot of the PRESCRIPTIONS column DRUG



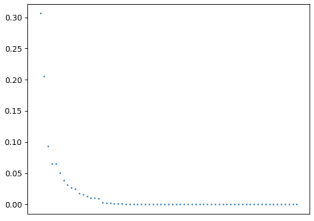
(d) Scatter plot of the PRESCRIPTIONS column DRUG_TYPE



(e) Scatter plot of the PRESCRIPTIONS column FORM_UNIT_DISP



(f) Scatter plot of the PRESCRIPTIONS column FORM_VAL_DISP

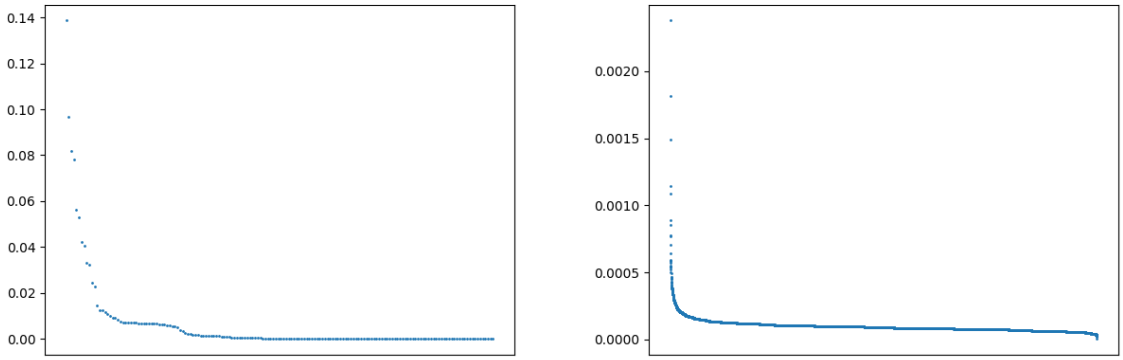


(g) Scatter plot of the PRESCRIPTIONS column ROUTE

Figure 5.10: Plots are representing the normalized count of every variable of the PRESCRIPTIONS_ICD table.

5.2.2.6 Gaussian Copula

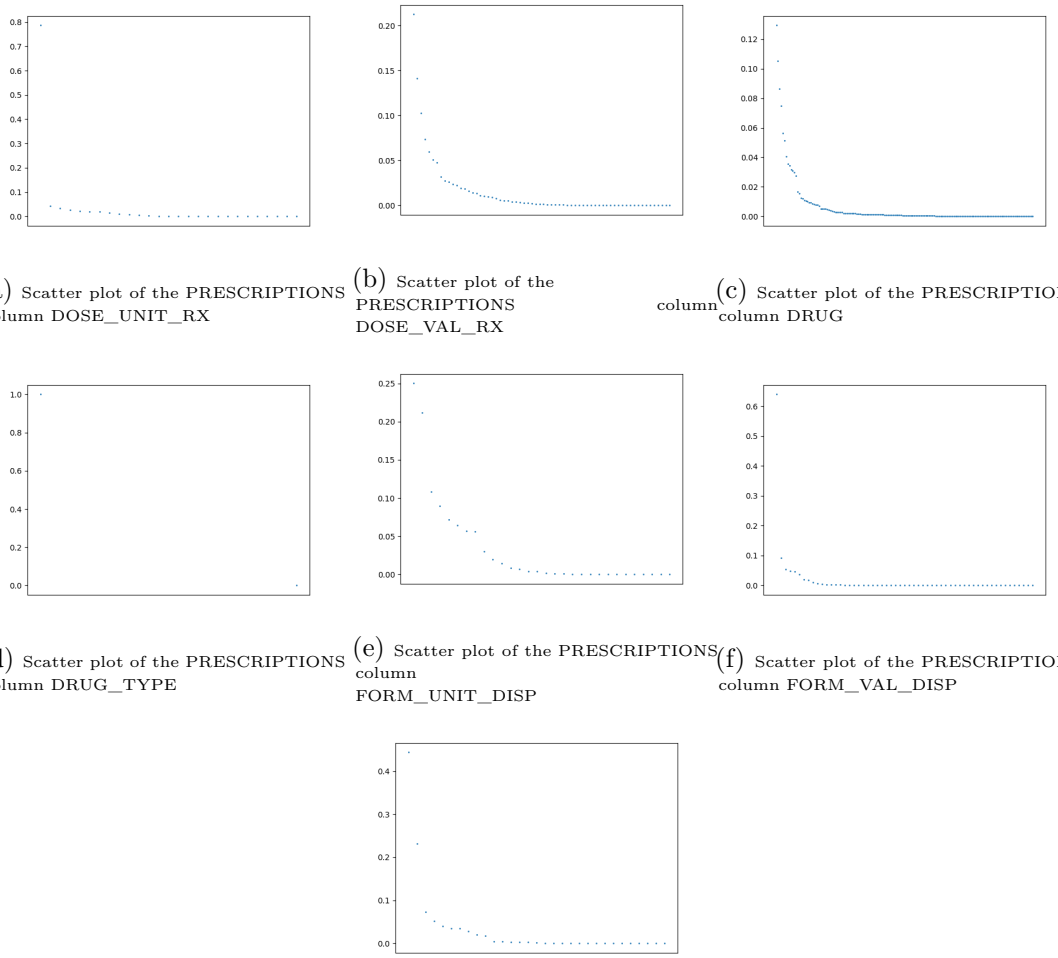
Gaussian Copula was the worst model overall in previous metrics, corroborated by the plots as it performs badly across every table. This could be due to the training method that did not favour this model, or that it needed more data to capture the features of the original dataset, or other different reason. From what we tried to get out of these plots, they do not add anything since data is sub-par with what was expected.



(a) Scatter plot of DIAGNOSES

(b) Scatter plot of PROCEDURES

Figure 5.11: Plots are representing the normalized count of every variable of the respective table.



(a) Scatter plot of the PRESCRIPTIONS column DOSE_UNIT_RX

(b) Scatter plot of the PRESCRIPTIONS DOSE_VAL_RX column

(c) Scatter plot of the PRESCRIPTIONS column DRUG

(d) Scatter plot of the PRESCRIPTIONS column DRUG_TYPE

(e) Scatter plot of the PRESCRIPTIONS column FORM_UNIT_DISP

(f) Scatter plot of the PRESCRIPTIONS column FORM_VAL_DISP

(g) Scatter plot of the PRESCRIPTIONS column ROUTE

Figure 5.12: Plots are representing the normalized count of every variable of the PRESCRIPTIONS_ICD table.

5.3 Privacy

Knowing how models performed in terms of data quality, in this section we investigate how private synthetic data truly is, and what difficulties are faced when trying to get this information.

5.3.1 Uniqueness

Searching uniqueness was achieved by measuring the SUDA score for each synthetic table, that calculation outputs two metrics the SUDA score and DIS-SUDA score.

The results were great in theory albeit suspicious on how useful and reliable they truly are. The function returned a value of 0 for all SUDA and DIS-SUDA, meaning that no value is unique to the point that becomes identifiable for being a such an unique value in the dataset. But when

looking at the original dataset we can also see that in all datasets we also have a SUDA and DIS-SUDA score of 0. This value is not to be trusted since it is an odd value to get throughout all experiments.

This results can also mean that this type of data is not well supported by the library, but after extensive checking over at the documentation, there were nothing that would insinuate that the data would not work.

5.3.2 Mutual Information

It has been explained in chapter 3 that the mutual information score could be used as a data quality metric, but here we can adapt it to understand privacy a bit better. As the more mutual information it can detect, it means that it is closer to the real data. The results shown in table 5.3 are normalized where when closer to 0 it means no mutual information and 1 is total correlation. To simplify the table, the PROCEDURES results are the average of all mutual information scores across all columns.

Table 5.3: Normalized mutual information values.

	PRESCRIPTIONS	PROCEDURES	DIAGNOSES
TVAE	0.0065325759257175	0.21247408141795304	0.11626879169244031
CopulaGAN	0.009804108862016	0.194109264944309	0.07232602039969128
CTGAN	0.010755034873551	0.2406026869643241	0.060790127945560504
Gaussian Copula	0.0053633559654087	0.299103606986669	0.07235952455075235
Synthpop	0.0068621918200562	0.09247803448977794	0.16341798700153004

Looking at the results it interesting to watch that no model had a lot of mutual information, the table with most mutual information is the PROCEDURES since it is the smallest dataset it is normal that more cases would translate over. PRESCRIPTIONS and DIAGNOSES had more similar values across the board. Since Synthpop was the best performing model on the data evaluation, we can start by observing that this also implies less mutual information in bigger datasets and in smaller it has more, bearing an interesting result.

The remaining models perform in an opposite manner to Synthpop and overall there is not much deviation from any of the models. CopulaGAN still has solid results here as well when comparing to the others, having an average of from all scores, close to what Synthpop provides.

5.3.3 SDV metrics

In chapter 4 we discussed that none of the SDV metrics were possible to use, but it is still an important result to discuss regardless, despite not having numbers to discuss. The difficulty that these metrics imposed on the work show the difficulty behind knowing how private synthetic

data is. It is important to note that these metrics did not work with the models of the same ecosystem, which in theory, the output is already optimized to be used afterwards.

The time and resources some models needed is also a good metric to take into consideration when thinking about adopting these kinds of synthetic data generation methods. Since data that is shared needs to be in line with a prerequisite, and if it is one of this metrics from SDV, it can be time consuming and expensive to match such standards.

5.3.4 Differential Privacy

The last results are related to differential privacy and how the generated data fares when compared to synthetic data.

Synthetic data was generated and had the results already shown, to more easily compare it to differential private data it was decided to have a scatter plot of the same nature having the count of each unique variable on the dataset for each column.

For this each dataset we used PyPD to do a calculation of each column, this was a straightforward procedure by following the example provided on the website with the default value ϵ of $\log 3$. The results of the outcome are visible at 5.13 and 5.14.

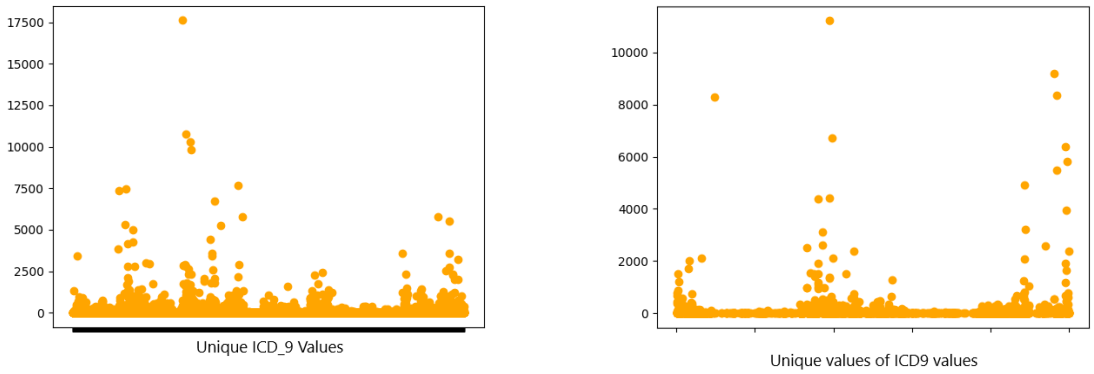
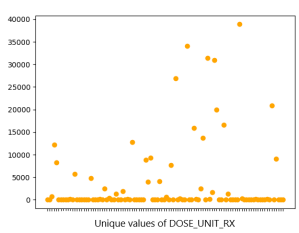


Figure 5.13: Plots are representing the normalized count of every variable of the respective table.

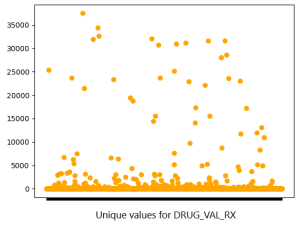
These scatter plots show significant difference from both the original data, alongside this difference it is possible to see that it keep a pattern across all columns as the spread of the values is uniform.

This is not ideal in terms of data quality when comparing to the original data, as it is clear from the scatter plots even to the naked eye.

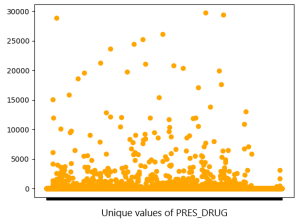
But the point of this comparison is to directly compare it to the synthetic data, and since most of synthetic data is similar to the original one. The only one that strikes a similarity is



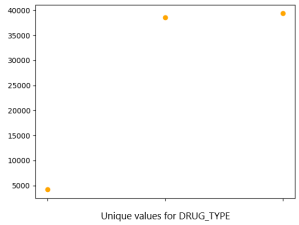
(a) Scatter plot of the PRESCRIPTIONS column DOSE_UNIT_RX



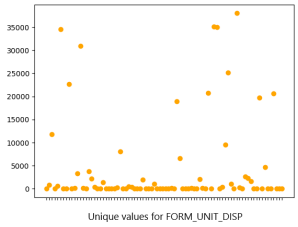
(b) Scatter plot of the PRESCRIPTIONS column DOSE_VAL_RX



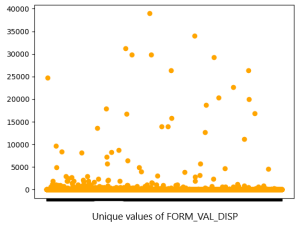
(c) Scatter plot of the PRESCRIPTIONS column DRUG



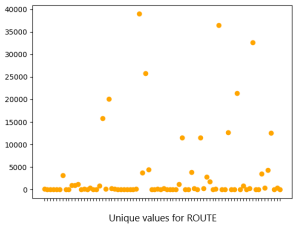
(d) Scatter plot of the PRESCRIPTIONS column DRUG_TYPE



(e) Scatter plot of the PRESCRIPTIONS column FORM_UNIT_DISP



(f) Scatter plot of the PRESCRIPTIONS column FORM_VAL_DISP



(g) Scatter plot of the PRESCRIPTIONS column ROUTE

Figure 5.14: Plots are representing the normalized count of every variable of the respective table.

the PROCEDURES one, as it retains most of the scatter properties from the synthetic datasets that performed better. It is assumed that it is mostly due to the fact that it fits the pattern the differential private algorithm is performing across the board.

As an overall view on the results, these were not the best in terms of what we were trying to achieve. Almost every model could use more data, and they also took an impact from the training methods. Despite this it still shows some promise to this field, as some models show some resemblance to the original dataset in terms of data quality. This of course shows on the trade-off when analysing the privacy.

Chapter 6

Conclusions

This thesis aimed at studying how synthetic data creation methods fared with health data types, and on top of trying to check the data utility, it was also attempted to measure how private it is. Based on the analysis, data creating still is complicated to automate, and without any tuning, it can become useless, which takes out the automated scenario of, intake any data and automatically retrieving synthetic data. Instead, we have an extra layer of work to analyze the data and then decide what hyper-parameters and even what model should be used accordingly. Despite this, we have Synthpop which was a great performer in achieving significant quality data when compared to the original datasets, this was done without a very powerful GPU, and it mainly needs RAM, depending on the amount of data and for faster processing, it only uses CPU power which is cheaper than a GPU. Synthpop has proven to be the most efficient model to create synthetic data overall and with the best results in terms of data quality. The other models did not perform as well as Synthpop, but most of them showed promise and probably just needed more data. At the same time, it can be unrealistic in a scenario of a small clinic or in a rural zone hospital to gather that much data, for either larger hospitals or even a collection of data across an entire country, it can be possible to use the other models to generate data with more outstanding quality as we have seen.

The more data that is used, the more computing power it will require to fit that data, and this could be a deal-breaker if the outcome is not profitable enough to cover the upfront cost of hardware, since most of the companies that would benefit from this technology, would be selling data to other companies.

We can see that the privacy metrics did not fare well in this context regarding the privacy evaluation. We still have the Suda score and the differential privacy data to compare it against. SUDA is not a good indicator since it is constant throughout all measurements, and it tells very little about how private the data is. Using a fairly tight privacy budget, the differential private data shows that some datasets can accidentally fall on the differential private category, such as the procedures table. We can not be counting on happening every time, because it does not, and the results prove it.

But looking outside of the metrics, we also have the matter, introduced in the differential private solution, we have the term plausible deniability, despite the context here not being exactly the same, we can still borrow the idea behind it. Even if the data generated is almost identical to the real data, it is possible for an individual to affirm that it is not his data, even if it is. Since not all data is exactly the same, it is possible to deny accusations or affirmations that that specific data belongs to anyone.

The fact that to test how private a data is, takes great computing power and time before it can be deemed worthy of being released could be another barrier to possible adopters. The process of generating and testing the data is lengthy by itself, and if something goes wrong, the whole process needs to be done from the ground up again, this is known because this thesis had these kinds of setbacks and endeavors, and the fact that the computing power was not much it took even longer. Transforming the idea of implementing a system like this "on a budget" is impractical at best.

Counterbalancing this issue is the time it usually takes for real data to be released. This work was supposed to be done with real and recent data from hospitals, but the bureaucracy and the time it took was so much, that it was dropped. So, if there was a system that allowed to release data, even if this data is neither truly private, or has all the information from the real one, in some cases, it can be worth the trade-off instead of having no data at all to work with.

Ultimately it is a matter of balancing what we are trying to achieve. If the data is going to be used in something that does not require great accuracy, it is possible to make it more private, if not, the data will always be less private the more close to the original it is. Furthermore, apart from that the technologies are still in infancy, and it still seems complicated to make a broad implementation of this in a fit all cases scenario, which would be ideal, but for now, it seems a complicated task made worst when looking at what other problems that the healthcare area is still facing regarding data collection and their systems.

6.1 Future work

For future work, there are several aspects that could be improved upon. Elaborate upon the state of the art and documentation of the methodologies making it easy for future work be easier to do. This kind of work also benefits from more powerful hardware, so it could be possible to train the datasets as a whole instead of breaking them into smaller partitions, this would also benefit the privacy metrics provided by SDV which would not take as much time as it did in this work.

The idea behind using a graph to create a metric that could both represent the relation between data quality and privacy is still a viable idea, this is because this type of data can be seen as a network, and using an already well known area such as network science it could expand

It would be also be of added value, to investigate how this technology fares on real data from

small and large healthcare facilities, so it would help to understand at what data volume this starts to have any significant effect.

Referências

[1] J. R. Reidenberg, "Resolving conflicting international data privacy rules in cyberspace," *Stanford Law Review*, vol. 52, no. 5, pp. 1315–1371, 2000. [Online]. Available: http://www.jstor.org/stable/1229516

[2] W. S. Blackmer, "EU general data protection regulation," *American Fuel and Petrochemical Manufacturers, AFPM - Labor Relations/Human Resources Conference 2018*, vol. 2014, no. April, pp. 45–62, 2018.

[3] T. Vega, "Code that tracks users' browsing prompts lawsuits," Sep 2010. [Online]. Available: https://www.nytimes.com/2010/09/21/technology/21cookie.html

[4] B. Rathonyi, A. Zaidi, and M. Hogan, "Cellular networks for Massive IoT," *White Paper*, no. January, p. 13, 2016. [Online]. Available: https://www.ericsson.com/res/docs/whitepapers/wp{_}iot.pdf

[5] B. Edwards, S. Hofmeyr, and S. Forrest, "Hype and heavy tails: A closer look at data breaches," *Journal of Cybersecurity*, vol. 2, no. 1, pp. 3–14, 2016.

[6] *Data protection regulations and international data flows: implications for trade and development.* UN, 2016.

[7] M. Mikulic, "Healthcare data volume globally 2020 forecast," Sep 2020. [Online]. Available: https://www.statista.com/statistics/1037970/global-healthcare-data-volume/

[8] Y. Bhatt and C. Bhatt, *Internet of Things in HealthCare.* Cham: Springer International Publishing, 2017, pp. 13–33. [Online]. Available: https://doi.org/10.1007/978-3-319-49736-5_2

[9] G. Szarvas, R. Farkas, and R. Busa-Fekete, "State-of-the-art anonymization of medical records using an iterative machine learning framework," *Journal of the American Medical Informatics Association*, vol. 14, no. 5, pp. 574–580, 2007.

[10] M. L. Maag, L. Denoyer, and P. Gallinari, "Graph anonymization using machine learning," in *2014 IEEE 28th International Conference on Advanced Information Networking and Applications.* IEEE, 2014, pp. 1111–1118.

[11] "Google trends on machine learning." [Online]. Available: https://trends.google.com/trends/explore?date=all&geo=US&q=Machine%20learning

[12] "Statistical language - what are data?" [Online]. Available: https://www.abs.gov.au/websitedbs/D3310114.nsf/Home/Statistical+Language+-+what+are+data

[13] "Data creation and replication will grow at a faster rate than installed storage capacity, according to the idc global datasphere and storagesphere forecasts," Mar 2021. [Online]. Available: https://www.businesswire.com/news/home/20210324005175/en/Data-Creation-and-Replication-Will-Grow-at-a-Faster-Rate-Than-Installed-Storage-Capacity-According-to

[14] R. Winkelmann and S. Boes, *Analysis of microdata*, 1st ed. New York, NY: Springer, 2006.

[15] I. Ermilov, S. Auer, and C. Stadler, "User-driven semantic mapping of tabular data," in *Proceedings of the 9th International Conference on Semantic Systems*, ser. I-SEMANTICS '13. New York, NY, USA: Association for Computing Machinery, 2013, p. 105–112. [Online]. Available: https://doi.org/10.1145/2506182.2506196

[16] L. Xu and K. Veeramachaneni, "Synthesizing tabular data using generative adversarial networks," 2018.

[17] J. Kent, "Big data to see explosive growth, challenging healthcare organizations," Dec 2018. [Online]. Available: https://healthitanalytics.com/news/big-data-to-see-explosive-growth-challenging-healthcare-organizations

[18] W. E. Hammond, C. Jaffe, J. J. Cimino, and S. M. Huff, "Standards in biomedical informatics," in *Biomedical informatics*. Springer, 2014, pp. 211–253.

[19] J.-D. Haynes, "Decoding and predicting intentions," *Annals of the New York Academy of Sciences*, vol. 1224, no. 1, pp. 9–21, 2011.

[20] A. Schedlbauer, V. Prasad, C. Mulvaney, S. Phansalkar, W. Stanton, D. W. Bates, and A. J. Avery, "What evidence supports the use of computerized alerts and prompts to improve clinicians' prescribing behavior?" *Journal of the American Medical Informatics Association*, vol. 16, no. 4, pp. 531–538, 2009.

[21] M.-J. Woo, J. P. Reiter, A. Oganian, and A. F. Karr, "Global Measures of Data Utility for Microdata Masked for Disclosure Limitation," *Journal of Privacy and Confidentiality*, vol. 1, no. 1, Apr. 2009. [Online]. Available: https://journalprivacyconfidentiality.org/index.php/jpc/article/view/568

[22] S. R. Midway, "Principles of Effective Data Visualization," *Patterns*, vol. 1, no. 9, p. 100141, Dec. 2020. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S2666389920301896

[23] "Snapdragon 8 series mobile platforms." [Online]. Available: https://www.qualcomm.com/products/snapdragon-8-series-mobile-platforms

[24] M. Yapıcı, A. Tekerek, and N. Topaloglu, "Literature review of deep learning research areas," vol. 5, pp. 188–215, 12 2019.

[25] I. Goodfellow, "NIPS 2016 Tutorial: Generative Adversarial Networks," *arXiv:1701.00160 [cs]*, Apr. 2017, arXiv: 1701.00160. [Online]. Available: http://arxiv.org/abs/1701.00160

[26] I. Vasilev, D. Slater, G. Spacagna, P. Roelants, and V. Zocca, *Python deep learning: exploring deep learning techniques and neural network architectures with PyTorch, Keras, and TensorFlow*, second edition ed. Birmingham Mumbai: Packt Publishing Limited, 2019.

[27] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[28] R. B. Nelsen, *An introduction to copulas*, 2nd ed., ser. Springer series in statistics. New York: Springer, 2006.

[29] G. Elidan, "Copulas in Machine Learning," in *Copulae in Mathematical and Quantitative Finance*, P. Jaworski, F. Durante, and W. K. Härdle, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, vol. 213, pp. 39–60, series Title: Lecture Notes in Statistics. [Online]. Available: http://link.springer.com/10.1007/978-3-642-35407-6_3

[30] T. Schmidt, "Coping with copulas," *Copulas-From theory to application in finance*, vol. 3, p. 34, 2007.

[31] W. Loh, "Classification and regression trees," *WIREs Data Mining and Knowledge Discovery*, vol. 1, no. 1, pp. 14–23, Jan. 2011. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/widm.8

[32] J. R. Quinlan, *C4. 5: programs for machine learning*. Elsevier, 2014.

[33] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and regression trees*. Routledge, 2017.

[34] L. F. R. A. Torgo, "Inductive learning of tree-based regression models," 1999.

[35] T. Hastie and R. Tibshirani, "Exploring the nature of covariate effects in the proportional hazards model," *Biometrics*, pp. 1005–1016, 1990.

[36] "Dicionário cambridge: Significados, definições e traduções." [Online]. Available: https://dictionary.cambridge.org/pt/

[37] V. Huser and J. J. Cimino, "Don't take your ehr to heaven, donate it to science: legal and research policies for ehr post mortem," *Journal of the American Medical Informatics Association*, vol. 21, no. 1, pp. 8–12, 2014.

[38] S. Haas, S. Wohlgemuth, I. Echizen, N. Sonehara, and G. Müller, "Aspects of privacy for electronic health records," *International journal of medical informatics*, vol. 80, no. 2, pp. e26–e31, 2011.

[39] A. Abbas and S. U. Khan, "A review on the state-of-the-art privacy-preserving approaches in the e-health clouds," *IEEE journal of Biomedical and health informatics*, vol. 18, no. 4, pp. 1431–1441, 2014.

[40] "European data protection supervisor on health." [Online]. Available: https://edps.europa.eu/data-protection/our-work/subjects/health_en

[41] A. Wood, M. Altman, A. Bembenek, M. Bun, M. Gaboardi, J. Honaker, K. Nissim, D. O'Brien, T. Steinke, and S. Vadhan, "Differential Privacy: A Primer for a Non-Technical Audience," *SSRN Electronic Journal*, 2018. [Online]. Available: https://www.ssrn.com/abstract=3338027

[42] A. Wood, M. Altman, A. Bembenek, M. Bun, M. Gaboardi, J. Honaker, K. Nissim, D. R. OBrien, T. Steinke, and S. Vadhan, "Differential privacy: A primer for a non-technical audience," *Vanderbilt Journal o1f Entertainment & Technology Law*, vol. 21, no. 1, pp. 209–275, 2018. [Online]. Available: http://www.jetlaw.org/journal-archives/volume-21/volume-21-issue-1/differential-privacy-a-primer-for-a-non-technical-audience/

[43] C. Dwork, A. Roth *et al.*, "The algorithmic foundations of differential privacy." *Foundations and Trends in Theoretical Computer Science*, vol. 9, no. 3-4, pp. 211–407, 2014.

[44] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating Noise to Sensitivity in Private Data Analysis," p. 20.

[45] I. Dinur and K. Nissim, "Revealing information while preserving privacy," 01 2003, pp. 202–210.

[46] A. Blum, C. Dwork, F. McSherry, and K. Nissim, "Practical privacy: The sulq framework," in *24th ACM SIGMOD International Conference on Management of Data / Principles of Database Systems, Baltimore (PODS 2005)*, June 2005. [Online]. Available: https://www.microsoft.com/en-us/research/publication/practical-privacy-the-sulq-framework/

[47] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[48] D. Zhu, X. B. Li, and S. Wu, "Identity disclosure protection: A data reconstruction approach for privacy-preserving data mining," *Decision Support Systems*, vol. 48, no. 1, pp. 133–140, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.dss.2009.07.003

[49] M. Templ, B. Meindl, and A. Kowarik, "Introduction to Statistical Disclosure Control (SDC)," p. 31.

[50] About MIMIC. [Online]. Available: https://mimic.mit.edu/docs/about/

[51] G. Andrews, "What is synthetic data?" Jun. 2021. [Online]. Available: https://blogs.nvidia.com/blog/2021/06/08/what-is-synthetic-data/

[52] "The Synthetic Data Vault. Put synthetic data to work!" [Online]. Available: https://sdv.dev/

[53] P. Embrechts, F. Lindskog, and A. Mcneil, "Modelling Dependence with Copulas and Applications to Risk Management," in *Handbook of Heavy Tailed Distributions in Finance*. Elsevier, 2003, pp. 329–384. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/B9780444508966500108

[54] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling Tabular data using Conditional GAN," p. 11.

[55] "Copulagan model from sdv." [Online]. Available: https://sdv.dev/SDV/user_guides/single_table/copulagan.html

[56] "TVAE Model — SDV 0.13.1 documentation." [Online]. Available: https://sdv.dev/SDV/user_guides/single_table/tvae.html

[57] B. Nowok, G. M. Raab, and C. Dibben, "**synthpop** : Bespoke Creation of Synthetic Data in *R*," *Journal of Statistical Software*, vol. 74, no. 11, 2016. [Online]. Available: http://www.jstatsoft.org/v74/i11/

[58] Hazy, "Hazy/synthpop: Python implementation of ther package synthpop." [Online]. Available: https://github.com/hazy/synthpop

[59] N. Liu, "Newer pytorch binaries for older gpus," Oct 2020. [Online]. Available: https://blog.nelsonliu.me/2020/10/13/newer-pytorch-binaries-for-older-gpus/

[60] I. H. Witten and I. H. Witten, Eds., *Data mining: practical machine learning tools and techniques*, fourth edition ed. Amsterdam: Elsevier, 2017.

[61] G. Snedecor and G. William, "Statistical methods/george w," *Snedecor and william g. Cochran*, 1989.

[62] P. Jaccard, "THE DISTRIBUTION OF THE FLORA IN THE ALPINE ZONE.1," *New Phytologist*, vol. 11, no. 2, pp. 37–50, Feb. 1912. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1111/j.1469-8137.1912.tb05611.x

[63] A. Agresti, *Categorical data analysis*. John Wiley & Sons, 2003, vol. 482.

[64] G. Casella and R. Berger, *Statistical Inference*, ser. Duxbury advanced series in statistics and decision sciences. Thomson Learning, 2002. [Online]. Available: https://books.google.pt/books?id=0x_vAAAAMAAJ

[65] E. Greenberg, *Introduction to Bayesian econometrics*. Cambridge University Press, 2012.

[66] A. Etz, "Introduction to the concept of likelihood and its applications," *Advances in Methods and Practices in Psychological Science*, vol. 1, 10 2017.

[67] M. J. Elliot, A. M. Manning, and R. W. Ford, "A computational algorithm for handling the special uniques problem," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 493–509, 2002.

[68] M. Elliot, "Final report on the disclosure risk associated with the synthetic data produced by the sylls team," *Report 2015*, vol. 2, 2015.

[69] J. Taub, M. Elliot, M. Pampaka, and D. Smith, "Differential correct attribution probability for synthetic data: an exploration," in *International Conference on Privacy in Statistical Databases*. Springer, 2018, pp. 122–137.

[70] M. T. Dickerson and R. Drysdale, "Fixed-radius near neighbors search algorithms for points and segments," *Information Processing Letters*, vol. 35, no. 5, pp. 269–273, Aug. 1990. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/0020019090900564

[71] J. A. Calandrino, A. Kilzer, A. Narayanan, E. W. Felten, and V. Shmatikov, "" you might also like:" privacy risks of collaborative filtering," in *2011 IEEE symposium on security and privacy*. IEEE, 2011, pp. 231–246.

[72] S. Hussain and V. Khamisani, "Using Support Vector Machines for Numerical Prediction," in *2007 IEEE International Multitopic Conference*. Lahore, Pakistan: IEEE, Dec. 2007, pp. 1–5. [Online]. Available: http://ieeexplore.ieee.org/document/4557695/

[73] "PyDP," Jan. 2022, original-date: 2019-09-15T12:58:45Z. [Online]. Available: https://github.com/OpenMined/PyDP

[74] T. Stadler, B. Oprisanu, and C. Troncoso, "Synthetic Data – A Privacy Mirage," *arXiv:2011.07018 [cs]*, Dec. 2020, arXiv: 2011.07018. [Online]. Available: http://arxiv.org/abs/2011.07018

[75] "Synthetic data and re-identification risks," Nov. 2021, original-date: 2021-11-29T14:02:05Z. [Online]. Available: https://github.com/Digas-2/Dissertation

COLLABORATIVE INSTITUTIONAL TRAINING INITIATIVE (CITI PROGRAM)

COMPLETION REPORT - PART 1 OF 2
COURSEWORK REQUIREMENTS*

* NOTE: Scores on this Requirements Report reflect quiz completions at the time all requirements for the course were met. See list below for details. See separate Transcript Report for more recent quiz scores, including those on optional (supplemental) course elements.

- **Name:** Diogo Fernandes (ID: 9940945)
- **Institution Affiliation:** Massachusetts Institute of Technology Affiliates (ID: 1912)
- **Institution Email:** up201503723@edu.fc.up.pt
- **Institution Unit:** Computer Science

- **Curriculum Group:** Human Research
- **Course Learner Group:** Data or Specimens Only Research
- **Stage:** Stage 1 - Basic Course

- **Record ID:** 41162439
- **Completion Date:** 02-Mar-2021
- **Expiration Date:** 01-Mar-2024
- **Minimum Passing:** 90
- **Reported Score*:** 100

REQUIRED AND ELECTIVE MODULES ONLY	DATE COMPLETED	SCORE
Belmont Report and Its Principles (ID: 1127)	01-Mar-2021	3/3 (100%)
History and Ethics of Human Subjects Research (ID: 498)	01-Mar-2021	5/5 (100%)
Basic Institutional Review Board (IRB) Regulations and Review Process (ID: 2)	01-Mar-2021	5/5 (100%)
Records-Based Research (ID: 5)	02-Mar-2021	3/3 (100%)
Genetic Research in Human Populations (ID: 6)	02-Mar-2021	5/5 (100%)
Populations in Research Requiring Additional Considerations and/or Protections (ID: 16680)	02-Mar-2021	5/5 (100%)
Research and HIPAA Privacy Protections (ID: 14)	02-Mar-2021	5/5 (100%)
Conflicts of Interest in Human Subjects Research (ID: 17464)	02-Mar-2021	5/5 (100%)
Massachusetts Institute of Technology (ID: 1290)	02-Mar-2021	No Quiz

For this Report to be valid, the learner identified above must have had a valid affiliation with the CITI Program subscribing institution identified above or have been a paid Independent Learner.

Verify at: www.citiprogram.org/verify/?k53aecf4c-f50f-4fbb-a127-1030d22dcc39-41162439

Collaborative Institutional Training Initiative (CITI Program)
Email: support@citiprogram.org
Phone: 888-529-5929
Web: https://www.citiprogram.org

PhysioNet Credentialed Health Data Use Agreement 1.5.0

Data Use Agreement for the MIMIC-III Clinical Database (v1.4)

If I am granted access to the database:

1. I will not attempt to identify any individual or institution referenced in PhysioNet restricte
2. I will exercise all reasonable and prudent care to avoid disclosure of the identity of any ind or institution referenced in PhysioNet restricted data in any publication or other communi
3. I will not share access to PhysioNet restricted data with anyone else.
4. I will exercise all reasonable and prudent care to maintain the physical and electronic secu PhysioNet restricted data.
5. If I find information within PhysioNet restricted data that I believe might permit identifica any individual or institution, I will report the location of this information promptly by ema PHI-report@physionet.org, citing the location of the specific information in question.
6. I have requested access to PhysioNet restricted data for the sole purpose of lawful use in sc research, and I will use my privilege of access, if it is granted, for this purpose and no othe
7. I have completed a training program in human research subject protections and HIPAA regulations, and I am submitting proof of having done so.
8. I will indicate the general purpose for which I intend to use the database in my application
9. If I openly disseminate my results, I will also contribute the code used to produce those res a repository that is open to the research community.
10. This agreement may be terminated by either party at any time, but my obligations with res PhysioNet data shall continue after termination.

SIGNED: **Diogo Fernandes**

DATED: **April 5, 2021**